

Lemax B2BAPI

V1.0.1

Code version: 5.18.5+

LEMAX

1. CONTENTS

1.	2	
2.	8	
3.	9	
	GetApiSettings	10
	Parameters	10
	Response	10
	Example	10
	GetAllCountries	10
	Parameters	10
	Response	10
	Example	11
	GetAllDestinations	11
	Parameters	11
	Response	11
	Example	11
	GetSearchFields	12
	Parameters	12
	Response	12
	Example	12
	GetSearchResults	13
	Parameters	13
	Response	13
	Example	13
	GetDetailedDescription	16
	Parameters	16
	Response	16
	Example	16
	GetTransportationSearchResults	20
	Parameters	20
	Response	20
	Example	20
	GetTransportationDetailedDescription	22
	Parameters	23
	Response	23
	Example	23
	GetReservation	31
	Parameters	31
	Response	31
	GetAllReservations	31
	Parameters	31

Response	31
Example	32
InsertReservation	32
Parameters	32
Response	32
Example	32
CancelReservationItem	33
Parameters	33
Response	34
Example	34
GetFlightDestinations	34
Parameters	34
Response	34
Example	35
GetFlightSearchResults	35
Parameters	35
Response	35
Example	35
GetPackageSearchResults	40
Parameters	40
Response	40
Example	40
GetPackageDetailedDescription	43
Parameters	43
Response	43
Example	44
4. 52	
ObjectTypeID	49
UnitTypeID	49
ObjectTypeGroupID	49
CategoryID	50
5. 54	
GetApiSettingsRQ	51
GetApiSettingsRS	51
GetSearchFieldsRQ	51
GetSearchFieldsRS	52
GetSearchResultsRQ	52
GetSearchResultsRS	55
GetAllCountriesRQ	56
GetAllCountriesRS	56

GetAllDestinationsRQ	57
GetAllDestinationsRS	57
GetDetailedDescriptionRQ	58
GetDetailedDescriptionRS	60
GetTransportationSearchResultsRQ	61
GetTransportationSearchResultsRS	64
GetTransportationDetailedDescriptionRQ	66
GetTransportationDetailedDescriptionRS	68
GetReservationRQ	69
GetReservationRS	69
GetAllReservationsRQ	70
GetAllReservationsRS	70
InsertReservationRQ	70
InsertReservationRS	71
CancelReservationItemRQ	71
CancelReservationItemRS	72
ReservationItemRQ	72
ReservationItemAdHocRQ	73
ReservationItemRS	73
ReservationItemDetailRS	73
ServiceRQ	74
ServiceAdHocRQ	74
ServiceRS	74
PassengerRQ	74
PassengerAdHocRQ	75
CustomFieldRQ	75
PriceCalculationStatus	75
Country	75
Region	75
Destination	76
CustomField	76
Passenger	77
DocumentItem	77
Document	77
ReservationItemTransferData	78
ReservationItemDetails	78
ReservationItem	79
Reservation	80

PaymentMethod	80
SortParameter	80
AttributeFilter	81
OutParameter	81
Attribute	82
AttributeGroup	83
Photo	84
AccommodationObject	85
Language	87
Currency	87
ObjectType	87
AccommodationUnit	87
UnitType	89
Service	89
SpecialOffer	91
Period	92
Date	93
Note	93
Status	94
Departure	94
CalculatedPriceInfo	94
Categories	95
Category	95
PriceFormat	95
PackagePeriod	95
PackageTour	96
PackageTourDetails	96
PackageUnit	97
PriceOnDayOfWeek	98
UnitMinimumPriceInfo	98
Translation	98
Transportation	98
TransportationUnit	100
PeriodGroup	102
DiscountRule	102
ObjectLocation	103
RoomOccupantCombination	103
RoomOccupantCase	103

RoomOccupantRule	103
ServiceGroup	104
ServiceGroupCalculatedPriceInfo	104
PassengerConfigurationFilter	104
Meal	105
ReservationNote	105
SortingParameters	105
GetFlightDestinationsRQ	105
GetFlightDestinationsRS	105
FlightDestinationsRSDestination	106
GetFlightSearchResultsRQ	106
FlightSearchRQSegment	107
GetFlightSearchResultsRS	107
FlightSearchRSFlight	108
FlightSearchRSDirection	108
FlightSearchRSOption	108
FlightSearchRSSegment	109
FlightSearchRSChange	109
GetPackageDetailedDescriptionRQ	110
GetPackageSearchResultsRQ	112
PackageTourDetailedDescriptionOutParameter	116
CategorySearchResultsRQ	116
GetPackageSearchResultsRS	116
AllPackageTourResultsAdditionalData	118
PackageTourObjectInfo	118
GetPackageDetailedDescriptionRS	118
ObjectCancellationData	119
ObjectCancellationRule	119
6. 127	
PriceType	121
Values	121
SortBy	121
Values	121
SortOrder	121
Values	121
ComparisonType	122
Values	122
ResponseDetail	122
Values	122

AttributeType	122
Values	122
UnitAvailabilityStatus	122
Values	123
ServiceType	123
Values	123
StatusCode	123
Values	123
ServicePriceType	123
Values	123
FlightDestinationType	124
Values	124
CancellationData	124
CancellationDataItem	124
CancellationFeeTypeEnum	125
Values	125

2. INTRO

Directions for B2B API methods usage:

- **URL:** <https://yourdomain.itravelsoftware.com/API/B2B/MethodName>
- **Authorization**
 - Basic with Username and Password needed
 - Credentials are in format **username:password** and **Base64** encoded
 - C# example of getting the Base64 encoded string
`Convert.ToBase64String(Encoding.UTF8.GetBytes("username:password"));`
 - Set "Authorization" header
 - Authorization: Basic dXNlcm5hbWU6cGFzc3dvcnQ=
- **Request**
 - Set request object to body of **POST** method
 - Use **JSON or XML**
 - Set "Content-Type" header to specify desired type of response
 - For JSON set "application/json"
 - For XML set "application/xml"

EXAMPLE

The example of POST request and response for GetApiSettings method.

REQUEST

```
POST https://yourdomain.itravelsoftware.com/API/B2B/GetApiSettings HTTP/1.1
Host: yourdomain.itravelsoftware.com
Accept: application/json
Accept-Language: en-US,en;q=0.5
Authorization: Basic dXNlcm5hbWU6cGFzc3dvcnQ=
Content-Type: application/json; charset=UTF-8
Content-Length: 36

{
  LanguageID:"en",
  CurrencyID: 978
}
```

RESPONSE

```
HTTP/1.1 200 OK
Content-Type: application/json; charset=utf-8
Content-Length: 135

{"Language":{"LanguageID":"en","LanguageName":"English"},"Currency":{"CurrencyID":978,"CurrencyName":"Euri","CurrencyShortName":"EUR"}}
```


3. METHODS

GetApiSettings

Returns API settings for given parameters

PARAMETERS

Name	Type	Default value	Description
GetApiSettingsRQ	GetApiSettingsRQ		Object containing the parameters

RESPONSE

Object of type [GetApiSettingsRS](#).

EXAMPLE

In this example request is made to get language that has id "en" and currency with id 978.

REQUEST

C# example

```
var request = new GetApiSettingsRQ();
request.LanguageID = "en";
request.CurrencyID = 978;
```

RESPONSE

```
<GetApiSettingsRS>
  <Language>
    <LanguageID>en</LanguageID>
    <LanguageName>English</LanguageName>
  </Language>
  <Currency>
    <CurrencyID>978</CurrencyID>
    <CurrencyName>Euri</CurrencyName>
    <CurrencyShortName>EUR</CurrencyShortName>
  </Currency>
</GetApiSettingsRS>
```

GetAllCountries

Returns list of countries for given parameters

PARAMETERS

Name	Type	Default value	Description
GetAllCountriesRQ	GetAllCountriesRQ		Object containing the parameters

RESPONSE

Object of type [GetAllCountriesRS](#).

EXAMPLE

In this example request is made to get list of countries whose name contains SearchQuery letters .

REQUEST

C# example

```
var request = new GetAllCountries();
request.LanguageID = "en";
request.SearchQuery = "us";
```

RESPONSE

```
<CountryList>
  <Country>
    <CountryID>23</CountryID>
    <CountryName>USA</CountryName>
    ...
  <Country>
  <Country>
    <CountryID>1618</CountryID>
    <CountryName>Virgin Islands (US)</CountryName>
    ...
  <Country>
</CountryList>
```

GetAllDestinations

Returns list of destinations for given parameters

PARAMETERS

Name	Type	Default value	Description
GetAllDestinationsRQ	GetAllDestinationsRQ		Object containing the parameters

RESPONSE

Object of type [GetAllDestinationsRS](#).

EXAMPLE

In this example request is made to get list of destinations for specific CountryID .

REQUEST

C# example

```
var request = new GetAllDestinations();
request.LanguageID = "en";
request.CountryID = 23;
```

RESPONSE

```
</CountryList>
</RegionList>
</DestinationList>
  <Destination>
```

```

    <RegionID>0</RegionID>
    <DestinationID>527</DestinationID>
    <DestinationName>New York, New York (122)</DestinationName>
    ...
  </Destination>
  <Destination>
    <RegionID>0</RegionID>
    <DestinationID>397</DestinationID>
    <DestinationName>Miami Beach, Florida (45)</DestinationName>
    ...
  </Destination>
</DestinationList>

```

GetSearchFields

Returns values for search form (categories, countries, regions and destinations)

PARAMETERS

Name	Type	Default value	Description
GetSearchFieldsRQ	GetSearchFieldsRQ		An object that contains parameters needed to get the correct set of search fields

RESPONSE

Object of type [GetSearchFieldsRS](#)

EXAMPLE

In this example request is made to get search fields. All parameters (except for the languageID) are null so all categories, countries, regions and destination are returned.

REQUEST

```

var request = new GetSearchFieldsRQ();
request.LanguageID = "en";

```

RESPONSE

```

<GetSearchFieldsRS>
  <Categories>
    <CategoryList>
      <Category>
        <CategoryName>Exclusive offer</CategoryName>
        <CategoryID>1</CategoryID>
      </Category>
      ...
    </CategoryList>
  </Categories>
  <CountryList>
    <Country>
      <CountryID>2</CountryID>
      <CountryName>Hrvatska</CountryName>
    </Country>
    ...

```

```

</CountryList>
<RegionList>
  <Region>
    <CountryID>2</CountryID>
    <RegionID>4</RegionID>
    <RegionName>Kvarner</RegionName>
  </Region>
</RegionList>
<DestinationList>
  <Destination>
    <RegionID>4</RegionID>
    <DestinationID>61</DestinationID>
    <DestinationName>Baška</DestinationName>
  </Destination>
  ...
</DestinationList>
</GetSearchFieldsRS>

```

GetSearchResults

Returns all units according to search parameters.

PARAMETERS

Name	Type	Default value	Description
GetSearchResultsRQ	GetSearchResultsRQ		Object that contains all search parameters for units

RESPONSE

Object of type GetSearchResultsRS.

EXAMPLE

In this example request is made to get search results for following parameters: startDate = 16th September 2010, endDate 23th September 2010, region = 5 (Nothr Dalmatia), destination and country are irrelevant when region is set, objectTypeIDList is set to 1 ("Hotel"), objectTypeGroupIDList is null, categoryIDListUnion is null, price range is set from 50 to 5000 Euros (priceFrom = 50, priceTo = 5000, currencyID = 978), price type is set to "per person per day", results will be sorted by price ascending, there will be returned 5 object's per page (pageSize = 5), only objects with 3 stars will be returned (objectAttributeFilterList contains AttributeFilter which defines AttributeID = 119 (attribute ID for "Stars"), AttributeValue = 3 and ComparisonType is set to Equals), unitAttributeFilterList is null, thumbnails max size is set to 200x150, large photo max size set to 800x600, 5 photos per object (outParameterList contains OutParameters that define how many photos per object and per unit should be returned).

Function returned one hotel with one room that matched search criteria.

REQUEST

```

var getSearchResultsParameters = new GetSearchResultsRQ();

getSearchResultsParameters.StartDate = DateTime.Now;
getSearchResultsParameters.EndDate = DateTime.Now.AddDays(7);
getSearchResultsParameters.RegionIDList = { 5 };

```

```

// Return only hotels
getSearchResultsParameters.ObjectTypeIDList = { 1 };
getSearchResultsParameters.PriceFrom = 50.0;
getSearchResultsParameters.PriceTo = 5000.0;
getSearchResultsParameters.InPriceType = PriceType.PerPersonPerDay;

// Sort by price ascending
SortParameter sortParameter = new SortParameter();
sortParameter.SortBy = SortBy.Price;
sortParameter.SortOrder = SortOrder.Ascending;
SortParameter[] sortParameterList = { sortParameter };

getSearchResultsParameters.SortParameterList = sortParameterList;
getSearchResultsParameters.PageSize = 5;

// Return only objects with 3 stars
AttributeFilter attributeFilter = new AttributeFilter();
attributeFilter.AttributeID = 119;
attributeFilter.ComparisonType = ComparisonType.Equals;
attributeFilter.AttributeValue = "3";
getSearchResultsParameters.ObjectAttributeFilterList = { attributeFilter };
getSearchResultsParameters.CurrencyID = 978;

// Return 5 object's photos
OutParameter outParameter = new OutParameter();
outParameter.NumberOfResults = 5;
outParameter.ResponseDetail = ResponseDetail.ObjectPhotos;
getSearchResultsParameters.OutParameterList = { outParameter };

getSearchResultsParameters.LanguageID = "en";
getSearchResultsParameters.IgnorePriceAndAvailability = true;
getSearchResultsParameters.DestinationName = "";

```

RESPONSE

```

<GetSearchResultsRS>
  <PriceFormat>
    <DecimalSeparator>.</DecimalSeparator>
    <GroupSeparator>,</GroupSeparator>
  </PriceFormat>
  <Status>
    <Code>OK</Code>
    <Description>OK</Description>
  </Status>
  <TotalNumberOfResults>5</TotalNumberOfResults>
  <AccommodationObjectList>
    <AccommodationObject>
      <ObjectID>904</ObjectID>
      <Name>Miran</Name>
      <DestinationID>80</DestinationID>
      <ObjectType>
        <ObjectTypeID>1</ObjectTypeID>
        <ObjectTypeName>Hotel</ObjectTypeName>
      </ObjectType>
      <AttributeGroupList>
        <AttributeGroup>
          <GroupID>1</GroupID>
          <GroupName>Hotel</GroupName>
          <AttributeList>
            <Attribute>
              <AttributeID>100</AttributeID>

```

```

        <AttributeName>Address</AttributeName>
        <AttributeValue>Vladimira Nazora 53, Šibenik</AttributeValue>
        <AttributeType>Textual</AttributeType>
        <Visible>>false</Visible>
    </Attribute>
    ...
</AttributeList>
</AttributeGroup>
</AttributeGroupList>
<PhotoList>
    <Photo>
        <PhotoID>540</PhotoID>

<PhotoUrl>https://yourdomain.itravelsoftware.com/fotografije_itravel/904/540_6341532390
62530000_800_600.jpg</PhotoUrl>

<ThumbnailUrl>https://yourdomain.itravelsoftware.com/fotografije_itravel/904/540_634153
239062530000_200_150.jpg</ThumbnailUrl>
    </Photo>
    ...
</PhotoList>
<UnitList>
    <AccommodationUnit>
        <UnitID>915</UnitID>
        <AvailabilityStatus>OnRequest</AvailabilityStatus>
        <Type>
            <UnitTypeID>11</UnitTypeID>
            <UnitTypeName>Double room</UnitTypeName>
        </Type>
        <AttributeGroupList>
            <AttributeGroup>
                <GroupID>11</GroupID>
                <GroupName>Double room</GroupName>
                <AttributeList>
                    <Attribute>
                        <AttributeID>69</AttributeID>
                        <AttributeName>View of the sea</AttributeName>
                        <AttributeValue>1</AttributeValue>
                        <AttributeType>Logical</AttributeType>
                        <Visible>>true</Visible>
                    </Attribute>
                    ...
                </AttributeList>
            </AttributeGroup>
        </AttributeGroupList>
        <PhotoList />
        <ServiceList>
            <Service>
                <ServiceID>3</ServiceID>
                <ServiceName>Half board</ServiceName>
                <ServiceType>Basic</ServiceType>
                <Price>50.557239130139351</Price>
                <PriceFormatted>50.56</PriceFormatted>
                <PriceType>PerPersonPerDay</PriceType>
            </Service>
        </ServiceList>

<BookingAddress>https://yourdomain.itravelsoftware.com/itravel/Accommodation/Cart.aspx?
sifraJezik=2&amp;unitsID=915&amp;startDate=634202411403701171&amp;endDate=6342084594037
01171&amp;amount=1&amp;persons=20</BookingAddress>
    </AccommodationUnit>
    ...
</UnitList>

```

```

    </AccommodationObject>
    ...
  </AccommodationObjectList>
  <Language>
    <LanguageID>en</LanguageID>
    <LanguageName>English</LanguageName>
  </Language>
  <Currency>
    <CurrencyID>978</CurrencyID>
    <CurrencyName>Euri</CurrencyName>
    <CurrencyShortName>EUR</CurrencyShortName>
  </Currency>
  <DestinationList>...</DestinationList>
  <RegionList>...</RegionList>
  <CountryList>...</CountryList>
  <CurrentPage>1</CurrentPage>
  <PageSize>5</PageSize>
  <StartDate>2010-09-16T13:39:00.3701171+02:00</StartDate>
  <EndDate>2010-09-23T13:39:00.3701171+02:00</EndDate>
  <NumberOfPersons>0</NumberOfPersons>
  <NumberOfDays>7</NumberOfDays>
</GetSearchResultsRS>

```

GetDetailedDescription

Returns [object's](#) (and its [unit's](#)) detailed description.

PARAMETERS

Name	Type	Default value	Description
GetDetailedDescriptionRQ	GetDetailedDescriptionRQ		Object that contains the parameters needed to get the unit details

RESPONSE

Object of type [GetDetailedDescriptionRS](#).

EXAMPLE

In this example request is made to get object with ID 127, prices to show are in period from 16th September 2010 to 23th September 2010, prices will be returned in Euros (currencyID = 978), thumbnails and large photos will be returned in default dimensions (null) and results will be in English (languageID = "en").

REQUEST

```

var request = new GetDetailedDescriptionRQ();

request.StartDate = DateTime.Now;
request.EndDate = DateTime.Now.AddDays(7);
request.NumberOfPersons = 1;
request.ObjectID = 127;
request.CurrencyID = 978;
request.LanguageID = "en";

```

RESPONSE

```

<GetDetailedDescriptionRQ>
  <Status>
    <Code>OK</Code>
    <Description>OK</Description>
  </Status>
  <AccommodationObject>
    <ObjectID>127</ObjectID>
    <Name>TN Centinera</Name>
    <DestinationID>51</DestinationID>
    <ObjectType>
      <ObjectTypeID>1</ObjectTypeID>
      <ObjectTypeName>Hotel</ObjectTypeName>
    </ObjectType>
    <AttributeGroupList>
      <AttributeGroup>
        <GroupID>1</GroupID>
        <GroupName>Hotel</GroupName>
        <AttributeList>
          <Attribute>
            <AttributeID>100</AttributeID>
            <AttributeName>Address</AttributeName>
            <AttributeValue>Indije 1</AttributeValue>
            <AttributeType>Textual</AttributeType>
            <Visible>>false</Visible>
          </Attribute>
          ...
        </AttributeList>
      </AttributeGroup>
    </AttributeGroupList>
    <PhotoList>
      <Photo>
        <PhotoID>52</PhotoID>

<PhotoUrl>https://yourdomain.itravelsoftware.com/fotografije_itravel/127/52.jpg</PhotoU
rl>

<ThumbnailUrl>https://yourdomain.itravelsoftware.com/fotografije_itravel/127/52.jpg</Th
umbnailUrl>
      </Photo>
      ...
    </PhotoList>
    <UnitList>
      <AccommodationUnit>
        <UnitID>128</UnitID>
        <AvailabilityStatus>OnRequest</AvailabilityStatus>
        <Type>
          <UnitTypeID>11</UnitTypeID>
          <UnitTypeName>Double room</UnitTypeName>
        </Type>
        <AttributeGroupList>
          <AttributeGroup>
            <GroupID>11</GroupID>
            <GroupName>Double room</GroupName>
            <AttributeList>
              <Attribute>
                <AttributeID>120</AttributeID>
                <AttributeName>Capacity</AttributeName>
                <AttributeValue>3</AttributeValue>
                <AttributeType>Textual</AttributeType>
                <Visible>>false</Visible>
              </Attribute>
              ...
            </AttributeList>
          </AttributeGroup>
        </AttributeGroupList>
      </AccommodationUnit>
    </UnitList>
  </AccommodationObject>
</GetDetailedDescriptionRQ>

```



```

    </AttributeList>
  </AttributeGroup>
</AttributeGroupList>
<PhotoList />
<ServiceList>
  <Service>
    <ServiceID>3</ServiceID>
    <ServiceName>Half board</ServiceName>
    <ServiceType>Basic</ServiceType>
    <Price>200.83999633789063</Price>
    <PriceFormatted>200.84</PriceFormatted>
    <PriceType>Total</PriceType>

  </Service>
  ...
</ServiceList>
<SpecialOfferList>
  <SpecialOffer>
    <ServiceID>294</ServiceID>
    <ServiceName />
    <ServiceType>SpecialOffer</ServiceType>
    <Price>172.14802932739258</Price>
    <PriceFormatted>172.15</PriceFormatted>
    <PriceType>Total</PriceType>
    <BillingType>
      <BillingTypeID>1</BillingTypeID>
    </BillingType>
    <PriceRowList>
      <PriceRow>
        <PriceRowID>1023</PriceRowID>
        <MinimumStay>1</MinimumStay>
        <MinimumPersons>1</MinimumPersons>
        <MaximumStay>0</MaximumStay>
        <PriceItemList>
          <PriceItem>
            <PeriodID>163</PeriodID>
            <StartDaysOfWeek>_____7</StartDaysOfWeek>
            <ListPriceOnDayOfWeek>
              <PriceOnDayOfWeek>
                <PriceOnDay>-0.1428571</PriceOnDay>
                <DayOfWeek>Sunday</DayOfWeek>
              </PriceOnDayOfWeek>
            </ListPriceOnDayOfWeek>
          </PriceItem>
        </PriceItemList>
      </PriceRow>
    </PriceRowList>
    <DateFrom>2010-09-18T00:00:00</DateFrom>
    <DateTo>2010-09-25T00:00:00</DateTo>
    <BasicPrice>200.83999633789063</BasicPrice>
    <BasicPriceFormatted>200.84</BasicPriceFormatted>
    <StartDatePrice>2010-09-18T00:00:00</StartDatePrice>
    <EndDatePrice>2010-09-25T00:00:00</EndDatePrice>
  </SpecialOffer>
</SpecialOfferList>

<BookingAddress>https://yourdomain.itravelsoftware.com/itravel/Accommodation/Cart.aspx?sifraJezik=2&amp;unitsID=128&amp;startDate=634202440080986328&amp;endDate=634208488080986328&amp;amount=1&amp;persons=20</BookingAddress>

<AvailabilityCalendarAddress>https://yourdomain.itravelsoftware.com/itravel/PregledZauzetosti.aspx?razina=2</AvailabilityCalendarAddress>
<NoteList />

```

```

    <AccommodationUnitPublicCode>H-21-1</AccommodationUnitPublicCode>
    <PriceListValidities>
      <PriceListValidity>
        <PriceListValidFrom>2006-01-01T00:00:00</PriceListValidFrom>
        <PriceListValidTo>2100-01-01T00:00:00</PriceListValidTo>
        <PriceListValidityID>3</PriceListValidityID>
      </PriceListValidity>
    </PriceListValidities>
    <CalculatedPriceInfo>
      <CalculatedPrice>200.83999633789063</CalculatedPrice>
      <CalculatedPriceFormatted>200.84</CalculatedPriceFormatted>
      <ServiceList>
        <Service>
          <ServiceID>3</ServiceID>
          <ServiceType>Basic</ServiceType>
          <Price>200.83999633789063</Price>
          <PriceFormatted>200.84</PriceFormatted>
          <PriceType>Total</PriceType>
        </Service>
        ...
      </ServiceList>
      <BasicCalculatedPrice>200.83999633789063</BasicCalculatedPrice>
      <BasicCalculatedPriceFormatted>200.84</BasicCalculatedPriceFormatted>
    </CalculatedPriceInfo>
  </AccommodationUnit>
  ...
</UnitList>
<NoteList />

<AvailabilityCalendarAddress>https://yourdomain.itravelsoftware.com/itravel/PregledZauz
etosti.aspx?razina=1</AvailabilityCalendarAddress>
  <AccommodationObjectPublicCode>H-21</AccommodationObjectPublicCode>
  <CategoryList>
    <Category>
      <CategoryName>Exclusive offer</CategoryName>
      <CategoryID>1</CategoryID>
    </Category>
    ...
  </CategoryList>
</AccommodationObject>
<Language>
  <LanguageID>en</LanguageID>
  <LanguageName>English</LanguageName>
</Language>
<PriceFormat>
  <DecimalSeparator>.</DecimalSeparator>
  <GroupSeparator>,</GroupSeparator>
</PriceFormat>
<Currency>
  <CurrencyID>978</CurrencyID>
  <CurrencyName>Euri</CurrencyName>
  <CurrencyShortName>EUR</CurrencyShortName>
</Currency>
<Destination>
  <RegionID>3</RegionID>
  <DestinationID>51</DestinationID>
  <DestinationName>Banjole</DestinationName>
</Destination>
<Region>
  <CountryID>2</CountryID>
  <RegionID>3</RegionID>
  <RegionName>Istria</RegionName>
</Region>

```

```

<Country>
  <CountryID>2</CountryID>
  <CountryName>Hrvatska</CountryName>
</Country>
<PeriodList>
  <Period>
    <PeriodID>164</PeriodID>
    <GroupID>29</GroupID>
    <DateList>
      <Date>
        <StartDate>2010-01-01T00:00:00</StartDate>
        <EndDate>2010-06-25T00:00:00</EndDate>
      </Date>
      <Date>
        <StartDate>2010-09-11T00:00:00</StartDate>
        <EndDate>2010-09-25T00:00:00</EndDate>
      </Date>
    </DateList>
  </Period>
  ...
</PeriodList>
<StartDate>2010-09-16T14:26:48.0986328+02:00</StartDate>
<EndDate>2010-09-23T14:26:48.0986328+02:00</EndDate>
<NumberOfDays>7</NumberOfDays>
<NumberOfPersons>1</NumberOfPersons>
<PriceType>Total</PriceType>
</GetDetailedDescriptionRS>

```

GetTransportationSearchResults

Returns all transportation according to search parameters.

PARAMETERS

Name	Type	Default value	Description
GetTransportationSearchResultsRQ	GetTransportationSearchResultsRQ		Object that contains all search parameters for units

RESPONSE

Object of type [GetTransportationSearchResultsRS](#).

EXAMPLE

In this example request is made to get search results for following parameters: startDate = 12th July 2013, endDate 13th September 2013, pickup and dropoff destinations irrelevant, objectTypeGroupIDList has 7 and 9 which are both part of Transportation object type, objectTypeIDList is null, categoryIDListUnion is null, currencyID is 978, results will be sorted by priority descending, there will be returned 5 object's per page (pageSize = 5), only transportation units which can transport up to 2 people will be returned (objectAttributeFilterList contains AttributeFilter which defines AttributeID = 120 (attribute ID for "Stars"), AttributeValue = 2 and ComparisonType is set to GreaterOrEqualThan), unitAttributeFilterList is null.

Function returned one transportation with four transportation units that matched search criteria.

REQUEST

```
var getTransportationSearchResultsParameters = new GetTransportationSearchResultsRQ();

getTransportationSearchResultsParameters.StartDate = new DateTime(2013, 7, 12);
getTransportationSearchResultsParameters.EndDate = new DateTime(2013, 9, 13);

// Return only transportations
getTransportationSearchResultsParameters.ObjectTypeGroupIDList = { 7, 9 };

// Sort by price ascending
SortParameter sortParameter = new SortParameter();
sortParameter.SortBy = SortBy.Priority;
sortParameter.SortOrder = SortOrder.Descending;
SortParameter[] sortParameterList = { sortParameter };

getTransportationSearchResultsParameters.SortParameterList = sortParameterList;
getTransportationSearchResultsParameters.PageSize = 5;

// Return only transportation units which can transport up to 2 adults
AttributeFilter attributeFilter = new AttributeFilter();
attributeFilter.AttributeID = 120;
attributeFilter.ComparisonType = ComparisonType.GreaterOrEqualThan;
attributeFilter.AttributeValue = "2";
getTransportationSearchResultsParameters.UnitAttributeFilterList = { attributeFilter };
getTransportationSearchResultsParameters.CurrencyID = 978;

getTransportationSearchResultsParameters.LanguageID = "en";
getTransportationSearchResultsParameters.IgnorePriceAndAvailability = false;
getTransportationSearchResultsParameters.DestinationName = "";
```

RESPONSE

```
<GetTransportationSearchResultsRS>
  <PriceFormat xmlns="http://tempuri.org/">
    <DecimalSeparator>.</DecimalSeparator>
    <GroupSeparator>,</GroupSeparator>
  </PriceFormat>
  <Status xmlns="http://tempuri.org/">
    <Code>OK</Code>
    <Description>OK</Description>
    <ErrorCode>0</ErrorCode>
  </Status>
  <TotalNumberOfResults xmlns="http://tempuri.org/">1</TotalNumberOfResults>
  <TransportationList xmlns="http://tempuri.org/">
    <Transportation>
      <ObjectID>296</ObjectID>
      <Name>Oslo Airport Transfer</Name>
      <DestinationID>0</DestinationID>
      <ObjectType>
        <ObjectTypeID>131</ObjectTypeID>
        <ObjectTypeName>Transfer</ObjectTypeName>
        <UnitTypeList />
        <Level>0</Level>
      </ObjectType>
      <AttributeGroupList>
        ...
      </AttributeGroupList>
      <PhotoList>
        ...
      </PhotoList>
    </Transportation>
  </TransportationList>
</GetTransportationSearchResultsRS>
```

```

</PhotoList>
<TransportationUnitList>
  <TransportationUnit>
    <UnitID>297</UnitID>
    <Description>Individual transfers assume transfers in air conditioned minivans
that seat up to seven people plus driver.</Description>
    <AvailabilityStatus>Available</AvailabilityStatus>
    <Type>
      <UnitTypeID>132</UnitTypeID>
      <UnitTypeName>Transfer</UnitTypeName>
      <Level>0</Level>
    </Type>
    ...

    <CalculatedPriceInfo>
      <CalculatedPrice>240</CalculatedPrice>
      <CalculatedPriceFormatted>240.00</CalculatedPriceFormatted>
      <ServiceList>
        <Service>
          <ServiceID>208</ServiceID>
          <ServiceType>Basic</ServiceType>
          <Price>0</Price>
          <PriceFormatted>0.00</PriceFormatted>
          <PriceType>Total</PriceType>
          <GroupID>0</GroupID>
          <VATRate>0</VATRate>
          <ServicePriceType>Fixed</ServicePriceType>
          <PricelistValidFrom>0001-01-01T00:00:00</PricelistValidFrom>
          <PricelistValidTo>0001-01-01T00:00:00</PricelistValidTo>
          <EnabledByDefault>>false</EnabledByDefault>
        </Service>
      </ServiceList>
      <BasicCalculatedPrice>240</BasicCalculatedPrice>
      <BasicCalculatedPriceFormatted>240.00</BasicCalculatedPriceFormatted>
      <NumberOfPersons>2</NumberOfPersons>
      <CalculationStatus>
        <Code>OK</Code>
        <ErrorCode>0</ErrorCode>
      </CalculationStatus>
    </CalculatedPriceInfo>
  </TransportationUnit>
  ...
</TransportationUnitList>
</Transportation>
</TransportationList>
<Language xmlns="http://tempuri.org/">
  <LanguageID>en</LanguageID>
  <LanguageName>English</LanguageName>
</Language>
<Currency xmlns="http://tempuri.org/">
  <CurrencyID>978</CurrencyID>
  <CurrencyName>Euri</CurrencyName>
  <CurrencyShortName>EUR</CurrencyShortName>
</Currency>
<DestinationList xmlns="http://tempuri.org/" />
<RegionList xmlns="http://tempuri.org/" />
<CountryList xmlns="http://tempuri.org/" />
<CurrentPage xmlns="http://tempuri.org/">1</CurrentPage>
<PageSize xmlns="http://tempuri.org/">10</PageSize>
<StartDate xmlns="http://tempuri.org/">2013-07-12T00:00:00</StartDate>
<EndDate xmlns="http://tempuri.org/">2013-09-13T00:00:00</EndDate>
<NumberOfPersons xmlns="http://tempuri.org/">2</NumberOfPersons>
<NumberOfDays xmlns="http://tempuri.org/">63</NumberOfDays>

```

```
</GetTransportationSearchResultsRS>
```

GetTransportationDetailedDescription

Returns transportation's (and its [unit's](#)) detailed description.

PARAMETERS

Name	Type	Default value	Description
GetTransportationDetailedDescriptionRQ	GetTransportationDetailedDescriptionRQ		Object that contains the parameters needed to get the transportation details

RESPONSE

Object of type [GetTransportationDetailedDescriptionRS](#).

EXAMPLE

In this example request is made to get object with ID 296, prices to show are in period from 15th July 2013 to 22nd September 2010, prices will be returned in Euros (currencyID = 978) with the PriceType set to normal, thumbnails have dimensions of (142 x 191) and large photos will be returned in default dimensions (null) and results will be in English (languageID = "en").

REQUEST

```
var request = getDetailedDescriptionParameters = new
GetTransportationDetailedDescriptionRQ();

request.StartDate = DateTime.Now;
request.EndDate = DateTime.Now.AddDays(7);
request.NumberOfPersons = 1;
request.ObjectID = 296;
request.CurrencyID = 978;
request.LanguageID = "en";
request.InPriceType = PriceType.Total;
request.ThumbnailWidth = 142;
request.ThumbnailHeight = 191;
```

RESPONSE

```
<GetTransportationDetailedDescriptionRS>
  <Status>
    <Code>OK</Code>
    <Description>OK</Description>
    <ErrorCode>0</ErrorCode>
  </Status>
  <Transportation>
```

```

<ObjectID>296</ObjectID>
<Name>Oslo Airport Transfer</Name>
<DestinationID>72</DestinationID>
<ObjectType>
  <ObjectTypeID>131</ObjectTypeID>
  <ObjectName>Transfer</ObjectName>
  <UnitTypeList />
  <Level>0</Level>
</ObjectType>
<AttributeGroupList>
  <AttributeGroup>
    <GroupID>131</GroupID>
    <GroupName>Transfer</GroupName>
    <AttributeList>
      <Attribute>
        <AttributeID>121</AttributeID>
        <AttributeName>Name</AttributeName>
        <AttributeValue>Oslo Airport Transfer</AttributeValue>
        <AttributeType>Textual</AttributeType>
        <Visible>>false</Visible>
      </Attribute>
    </AttributeList>
  </AttributeGroup>
</AttributeGroupList>
<PhotoList>
  <Photo>
    <PhotoID>229</PhotoID>

    <PhotoUrl>https://yourdomain.itravelsoftware.com/fotografije\_itravel/296/229\_634814103234570000.jpg</PhotoUrl>

    <ThumbnailUrl>https://yourdomain.itravelsoftware.com/fotografije\_itravel/296/229\_634814103234570000\_191\_142.jpg</ThumbnailUrl>
    <Position>0</Position>
    <ObjectID>0</ObjectID>
    <UnitID>0</UnitID>
    <DestinationID>0</DestinationID>
    <PackageTourID>0</PackageTourID>
    <TransportationID>0</TransportationID>
    <Title />
    <AlternateText />
    <Caption />
    <Description />
    <GeoLocation />
  </Photo>

  <Photo>
    <PhotoID>214</PhotoID>

    <PhotoUrl>https://yourdomain.itravelsoftware.com/fotografije\_itravel/297/214\_634814097220210000.jpg</PhotoUrl>

    <ThumbnailUrl>https://yourdomain.itravelsoftware.com/fotografije\_itravel/297/214\_634814097220210000\_191\_142.jpg</ThumbnailUrl>
    <Position>0</Position>
    <ObjectID>0</ObjectID>
    <UnitID>0</UnitID>
    <DestinationID>0</DestinationID>
    <PackageTourID>0</PackageTourID>
    <TransportationID>0</TransportationID>
    <Title />
    <AlternateText />
    <Caption />

```

```

    <Description />
    <GeoLocation />
  </Photo>
  <Photo>
    <PhotoID>213</PhotoID>

    <PhotoUrl>https://yourdomain.itravelsoftware.com/fotografije_itravel/297/213_634814097217460000.jpg</PhotoUrl>

    <ThumbnailUrl>https://yourdomain.itravelsoftware.com/fotografije_itravel/297/213_634814097217460000_191_142.jpg</ThumbnailUrl>
    <Position>0</Position>
    <ObjectID>0</ObjectID>
    <UnitID>0</UnitID>
    <DestinationID>0</DestinationID>
    <PackageTourID>0</PackageTourID>
    <TransportationID>0</TransportationID>
    <Title />
    <AlternateText />
    <Caption />
    <Description />
    <GeoLocation />
  </Photo>
  ...
</PhotoList>
<ServiceList>
  <Service>
    <ServiceID>208</ServiceID>
    <ServiceName>Transfer (one way)</ServiceName>
    <ServiceType>Basic</ServiceType>
    <Price>120</Price>
    <PriceFormatted>120.00</PriceFormatted>
    <PriceType>Total</PriceType>
    <BillingType>
      <BillingTypeID>6</BillingTypeID>
      <BillingTypeName>Per person</BillingTypeName>
    </BillingType>
    <PriceRowList>
      <PriceRow>
        <PriceRowID>631</PriceRowID>
        <MinimumStay>1</MinimumStay>
        <MinimumPersons>1</MinimumPersons>
        <MaximumStay>0</MaximumStay>
        <PaidLocally>>false</PaidLocally>
        <PriceItemList>
          <PriceItem>
            <PeriodID>1</PeriodID>
            <StartDaysOfWeek>1234567</StartDaysOfWeek>
            <ListPriceOnDayOfWeek>
              <PriceOnDayOfWeek>
                <PriceOnDay>120</PriceOnDay>
                <NetPriceOnDay>0</NetPriceOnDay>
                <DayOfWeek>Monday</DayOfWeek>
              </PriceOnDayOfWeek>
              <PriceOnDayOfWeek>
                <PriceOnDay>120</PriceOnDay>
                <NetPriceOnDay>0</NetPriceOnDay>
                <DayOfWeek>Tuesday</DayOfWeek>
              </PriceOnDayOfWeek>
              <PriceOnDayOfWeek>
                <PriceOnDay>120</PriceOnDay>
                <NetPriceOnDay>0</NetPriceOnDay>
                <DayOfWeek>Wednesday</DayOfWeek>
              </PriceOnDayOfWeek>
            </ListPriceOnDayOfWeek>
          </PriceItem>
        </PriceItemList>
      </PriceRow>
    </PriceRowList>
  </Service>

```



```

        </PriceOnDayOfWeek>
        <PriceOnDayOfWeek>
          <PriceOnDay>120</PriceOnDay>
          <NetPriceOnDay>0</NetPriceOnDay>
          <DayOfWeek>Thursday</DayOfWeek>
        </PriceOnDayOfWeek>
        <PriceOnDayOfWeek>
          <PriceOnDay>120</PriceOnDay>
          <NetPriceOnDay>0</NetPriceOnDay>
          <DayOfWeek>Friday</DayOfWeek>
        </PriceOnDayOfWeek>
        <PriceOnDayOfWeek>
          <PriceOnDay>120</PriceOnDay>
          <NetPriceOnDay>0</NetPriceOnDay>
          <DayOfWeek>Saturday</DayOfWeek>
        </PriceOnDayOfWeek>
        <PriceOnDayOfWeek>
          <PriceOnDay>120</PriceOnDay>
          <NetPriceOnDay>0</NetPriceOnDay>
          <DayOfWeek>Sunday</DayOfWeek>
        </PriceOnDayOfWeek>
      </ListPriceOnDayOfWeek>
    </PriceItem>
  </PriceItemList>
</PriceRow>
</PriceRowList>
<GroupID>1</GroupID>
<VATRate>0</VATRate>
<ServicePriceType>Fixed</ServicePriceType>
<PricelistValidFrom>2006-01-01T00:00:00</PricelistValidFrom>
<PricelistValidTo>2100-01-01T00:00:00</PricelistValidTo>
<EnabledByDefault>false</EnabledByDefault>
</Service>
</ServiceList>
<SpecialOfferList />

```

```

<BookingAddress>https://yourdomain.itravelsoftware.com/itravel/Accommodation/Shopping
Cart.aspx?sifraJezik=2&amp;unitsID=297&amp;startDate=635094432000000000&amp;endDate=635
100480000000000&amp;amount=1&amp;persons=25&amp;sifraValuta=978</BookingAddress>

```

```

<AvailabilityCalendarAddress>https://yourdomain.itravelsoftware.com/itravel/PregledZa
uzetosti.aspx?razina=2</AvailabilityCalendarAddress>

```

```

<NoteList />
<AccommodationUnitPublicCode>TR-1-1</AccommodationUnitPublicCode>
<CalculatedPriceInfo>
  <CalculatedPrice>120</CalculatedPrice>
  <CalculatedPriceFormatted>120.00</CalculatedPriceFormatted>
</ServiceList>
  <Service>
    <ServiceID>208</ServiceID>
    <ServiceType>Basic</ServiceType>
    <Price>120</Price>
    <PriceFormatted>120.00</PriceFormatted>
    <PriceType>Total</PriceType>
    <GroupID>0</GroupID>
    <VATRate>0</VATRate>
    <ServicePriceType>Fixed</ServicePriceType>
    <PricelistValidFrom>0001-01-01T00:00:00</PricelistValidFrom>
    <PricelistValidTo>0001-01-01T00:00:00</PricelistValidTo>
    <EnabledByDefault>false</EnabledByDefault>
  </Service>
</ServiceList>
<BasicCalculatedPrice>120</BasicCalculatedPrice>

```

```

    <BasicCalculatedPriceFormatted>120.00</BasicCalculatedPriceFormatted>
    <NumberOfPersons>0</NumberOfPersons>
    <CalculationStatus>
      <Code>OK</Code>
      <ErrorCode>0</ErrorCode>
    </CalculationStatus>
  </CalculatedPriceInfo>
  <UnitMinimumPriceInfo>
    <Price>120</Price>
    <PriceFormatted>120.00</PriceFormatted>
    <BillingTypeName>Per person</BillingTypeName>
  </UnitMinimumPriceInfo>
  <OrdinalNumber>1</OrdinalNumber>
  <OccupantCombinationOK>true</OccupantCombinationOK>
  <ObjectID>0</ObjectID>
  <ActivityStatus>0</ActivityStatus>
  <DestinationList>
    <Destination>
      <RegionID>0</RegionID>
      <DestinationID>72</DestinationID>
      <DestinationName>Oslo Airport</DestinationName>
    </Destination>
    <Destination>
      <RegionID>0</RegionID>
      <DestinationID>62</DestinationID>
      <DestinationName>Oslo</DestinationName>
    </Destination>
  </DestinationList>
  <PickupDestinationID>72</PickupDestinationID>
  <DropoffDestinationID>62</DropoffDestinationID>
</TransportationUnit>
<TransportationUnit>
  <UnitID>299</UnitID>
  <Description>Individual transfers assume transfers in air conditioned minivans
that seat up to seven people plus driver.</Description>
  <AvailabilityStatus>Available</AvailabilityStatus>
  <Type>
    <UnitTypeID>132</UnitTypeID>
    <UnitTypeName>Transfer</UnitTypeName>
    <Level>0</Level>
  </Type>
  <AttributeGroupList>
    <AttributeGroup>
      <GroupID>132</GroupID>
      <GroupName>Transfer</GroupName>
      <AttributeList>
        <Attribute>
          <AttributeID>133</AttributeID>
          <AttributeName>Description</AttributeName>
          <AttributeValue>Individual</AttributeValue>
          <AttributeType>Textual</AttributeType>
          <Visible>>false</Visible>
        </Attribute>
        <Attribute>
          <AttributeID>120</AttributeID>
          <AttributeName>Capacity</AttributeName>
          <AttributeValue>7</AttributeValue>
          <AttributeType>Textual</AttributeType>
          <Visible>>false</Visible>
        </Attribute>
      </AttributeList>
    </AttributeGroup>
  </AttributeGroupList>

```

```
<PhotoList>
  <Photo>
    <PhotoID>217</PhotoID>

    <PhotoUrl>https://yourdomain.itravelsoftware.com/fotografije_itravel/299/217_634814098826050000.jpg</PhotoUrl>

    <ThumbnailUrl>https://yourdomain.itravelsoftware.com/fotografije_itravel/299/217_634814098826050000_191_142.jpg</ThumbnailUrl>
    <Position>0</Position>
    <ObjectID>0</ObjectID>
    <UnitID>0</UnitID>
    <DestinationID>0</DestinationID>
    <PackageTourID>0</PackageTourID>
    <TransportationID>0</TransportationID>
    <Title />
    <AlternateText />
    <Caption />
    <Description />
    <GeoLocation />
  </Photo>
  <Photo>
    <PhotoID>216</PhotoID>

    <PhotoUrl>https://yourdomain.itravelsoftware.com/fotografije_itravel/299/216_634814098823440000.jpg</PhotoUrl>

    <ThumbnailUrl>https://yourdomain.itravelsoftware.com/fotografije_itravel/299/216_634814098823440000_191_142.jpg</ThumbnailUrl>
    <Position>0</Position>
    <ObjectID>0</ObjectID>
    <UnitID>0</UnitID>
    <DestinationID>0</DestinationID>
    <PackageTourID>0</PackageTourID>
    <TransportationID>0</TransportationID>
    <Title />
    <AlternateText />
    <Caption />
    <Description />
    <GeoLocation />
  </Photo>
  <Photo>
    <PhotoID>218</PhotoID>

    <PhotoUrl>https://yourdomain.itravelsoftware.com/fotografije_itravel/299/218_634814098829680000.jpg</PhotoUrl>

    <ThumbnailUrl>https://yourdomain.itravelsoftware.com/fotografije_itravel/299/218_634814098829680000_191_142.jpg</ThumbnailUrl>
    <Position>0</Position>
    <ObjectID>0</ObjectID>
    <UnitID>0</UnitID>
    <DestinationID>0</DestinationID>
    <PackageTourID>0</PackageTourID>
    <TransportationID>0</TransportationID>
    <Title />
    <AlternateText />
    <Caption />
    <Description />
    <GeoLocation />
  </Photo>
</PhotoList>
<ServiceList>
```

```

<Service>
  <ServiceID>208</ServiceID>
  <ServiceName>Transfer (one way)</ServiceName>
  <ServiceType>Basic</ServiceType>
  <Price>120</Price>
  <PriceFormatted>120.00</PriceFormatted>
  <PriceType>Total</PriceType>
  <GroupID>1</GroupID>
  <VATRate>0</VATRate>
  <ServicePriceType>Fixed</ServicePriceType>
  <PricelistValidFrom>2006-01-01T00:00:00</PricelistValidFrom>
  <PricelistValidTo>2100-01-01T00:00:00</PricelistValidTo>
  <EnabledByDefault>>false</EnabledByDefault>
</Service>
</ServiceList>
<SpecialOfferList />

```

```

<BookingAddress>https://yourdomain.itravelsoftware.com/itravel/Accommodation/Shopping
Cart.aspx?sifraJezik=2&amp;unitsID=303&amp;startDate=635094432000000000&amp;endDate=635
100480000000000&amp;amount=1&amp;persons=25&amp;sifraValuta=978</BookingAddress>

```

```

<AvailabilityCalendarAddress>https://yourdomain.itravelsoftware.com/itravel/PregledZa
uzetosti.aspx?razina=2</AvailabilityCalendarAddress>

```

```

<NoteList />
<AccommodationUnitPublicCode>TR-1-4</AccommodationUnitPublicCode>
<CalculatedPriceInfo>
  <CalculatedPrice>66</CalculatedPrice>
  <CalculatedPriceFormatted>66.00</CalculatedPriceFormatted>
  <ServiceList>
    <Service>
      <ServiceID>208</ServiceID>
      <ServiceType>Basic</ServiceType>
      <Price>66</Price>
      <PriceFormatted>66.00</PriceFormatted>
      <PriceType>Total</PriceType>
      <GroupID>0</GroupID>
      <VATRate>0</VATRate>
      <ServicePriceType>Fixed</ServicePriceType>
      <PricelistValidFrom>0001-01-01T00:00:00</PricelistValidFrom>
      <PricelistValidTo>0001-01-01T00:00:00</PricelistValidTo>
      <EnabledByDefault>>false</EnabledByDefault>
    </Service>
  </ServiceList>
  <BasicCalculatedPrice>66</BasicCalculatedPrice>
  <BasicCalculatedPriceFormatted>66.00</BasicCalculatedPriceFormatted>
  <NumberOfPersons>0</NumberOfPersons>
  <CalculationStatus>
    <Code>OK</Code>
    <ErrorCode>0</ErrorCode>
  </CalculationStatus>
</CalculatedPriceInfo>
<UnitMinimumPriceInfo>
  <Price>66</Price>
  <PriceFormatted>66.00</PriceFormatted>
  <BillingTypeName>Per person</BillingTypeName>
</UnitMinimumPriceInfo>
<OrdinalNumber>4</OrdinalNumber>
<OccupantCombinationOK>>true</OccupantCombinationOK>
<ObjectID>0</ObjectID>
<ActivityStatus>0</ActivityStatus>
<DestinationList>
  <Destination>

```

```

    <RegionID>0</RegionID>
    <DestinationID>62</DestinationID>
    <DestinationName>Oslo</DestinationName>
  </Destination>
  <Destination>
    <RegionID>0</RegionID>
    <DestinationID>72</DestinationID>
    <DestinationName>Oslo Airport</DestinationName>
  </Destination>
</DestinationList>
  <PickupDestinationID>62</PickupDestinationID>
  <DropoffDestinationID>72</DropoffDestinationID>
</TransportationUnit>
</TransportationUnitList>
<DestinationIDList>
  <int>72</int>
  <int>62</int>
</DestinationIDList>
<PickupDestinationIDList>
  <int>72</int>
</PickupDestinationIDList>
<DropoffDestinationIDList>
  <int>62</int>
</DropoffDestinationIDList>
</Transportation>
<Language>
  <LanguageID>en</LanguageID>
  <LanguageName>English</LanguageName>
</Language>
<PriceFormat>
  <DecimalSeparator>.</DecimalSeparator>
  <GroupSeparator>,</GroupSeparator>
</PriceFormat>
<Currency>
  <CurrencyID>978</CurrencyID>
  <CurrencyName>Euri</CurrencyName>
  <CurrencyShortName>EUR</CurrencyShortName>
</Currency>
<DestinationList>
  <Destination>
    <RegionID>0</RegionID>
    <DestinationID>72</DestinationID>
    <DestinationName>Oslo Airport</DestinationName>
    <PhotoList />
  </Destination>
  ...
</DestinationList>
<Region>
  <CountryID>60</CountryID>
  <RegionID>61</RegionID>
  <RegionName>Oslo</RegionName>
  <PhotoList />
</Region>
<Country>
  <CountryID>60</CountryID>
  <CountryName>Norway</CountryName>
  <PhotoList />
</Country>
<PeriodList>
  <Period>
    <PeriodID>1</PeriodID>
    <GroupID>1</GroupID>
    <DateList>

```

```

    <Date>
      <StartDate>2006-01-01T00:00:00</StartDate>
      <EndDate>2007-01-01T00:00:00</EndDate>
      <NumberOfDays>365</NumberOfDays>
    </Date>
  </DateList>
  <PositionPeriod>1</PositionPeriod>
  <DistinctNumberOfDaysList>
    <int>365</int>
  </DistinctNumberOfDaysList>
</Period>
</PeriodList>
<StartDate>2013-07-15T00:00:00</StartDate>
<EndDate>2013-07-22T00:00:00</EndDate>
<NumberOfDays>7</NumberOfDays>
<NumberOfPersons>1</NumberOfPersons>
<PriceType>Total</PriceType>
</GetTransportationDetailedDescriptionRS>

```

GetReservation

Returns detailed info on the reservation from the parameters

PARAMETERS

Name	Type	Default value	Description
GetReservationRQ	GetReservationRQ		Object that contains parameters needed to get a reservation

RESPONSE

List of object of type [GetReservationRS](#).

REQUEST

```

GetReservationRQ resParam = new GetReservationRQ();
resParam.ReservationID = 34;
resParam.LanguageID = "en";

```

GetAllReservations

Returns the list of all reservations that matches filters from request.

PARAMETERS

Name	Type	Default value	Description
GetAllReservationsRQ	GetAllReservationsRQ		Object that contains parameter needed to get a reservation and contains optional filters

REQUIRED PROPERTIES

Name	Type	Description
LanguageID	Int	Language ID.

RESPONSE

List of objects of type [GetAllReservationsRS](#).

EXAMPLE

In this example a request is made to get all reservations. Results should be in English (languageID = "en"). Filters which are otherwise optional were added here – response will contain only reservations with Customer ID 10 and those who are created in the last week.

REQUEST

```
var request = new GetAllReservationsRQ();
request.LanguageID = "en";
request.ReservationCreationDateFrom = DateTime.Today.AddDays(-7);
request.ReservationCreationDateTo = null;
```

InsertReservation

Insert reservation based on sent parameters.

PARAMETERS

Name	Type	Default value	Description
InsertReservationRQ	InsertReservationRQ		Request object with parameters needed to get payment methods.

RESPONSE

Objects of type [InsertReservationRS](#).

EXAMPLE

In this example, basic reservation is calculated. As a response, the reservation calculation data are returned.

REQUEST

```
var request = new InsertReservationRQ
{
    ExecuteInsert = true,
    PaymentMethodID = 6,
    CurrencyID = 978,
    LanguageID = "en",
    ReservationItems = new List<ReservationItemRQ>
    {
        new ReservationItemRQ
        {
            UnitID = 4278,
```

```

StartDate = new DateTime(2015,5,15),
EndDate = new DateTime(2015,5,22),
ReservationItemOrder = 1,
Passengers = new List<PassengerRQ> { new PassengerRQ() },
SelectedServices = new List<ServiceRQ>
{
    new ServiceRQ { ServiceID = 221, Amount = 0 },
}
}
}
};

```

RESPONSE

```

<InsertReservationRS>
  <ReservationID>493</ReservationID>
  <Currency>
    <CurrencyID>978</CurrencyID>
    <CurrencyName>Euri</CurrencyName>
    <CurrencyShortName>EUR</CurrencyShortName>
  </Currency>
  <SellingPrice>700</SellingPrice>
  <ReservationItems>
    <ReservationItemRS>
      <ReservationItemOrder>1</ReservationItemOrder>
      <SellingPrice>700</SellingPrice>
      <ReservationItemDetailsList>
        <ReservationItemDetailRS>
          <Service>
            <ServiceID>221</ServiceID>
            <Name>Cancellation penalty fee</Name>
          </Service>
          <SellingPrice>700</SellingPrice>
        </ReservationItemDetailRS>
      </ReservationItemDetailsList>
      <PriceCalculationStatus>
        <Code>OK</Code>
      </PriceCalculationStatus>
    </ReservationItemRS>
  </ReservationItems>
  <EndReservationURL>/itravel/Accommodation/EndReservation.aspx?languageID=en&id=aDAgJiAF
bdNcwzTmam6whfoUuozsXRGVXx8yeyAuu%2bYzvxb2QSSU21YHpjJgdpPxLFQ39eDVrtSlOiu04%2fsTsiCXavz
DKlcheWzklH1RgYAfBoUavJxHmuAKvyWFW%2fJE4JfTSRok4zdB4PEcpSExSq2f3JjGJ0X%2bjzTuA83eyac%3d
</EndReservationURL>

  <Status>
    <Code>OK</Code>
    <ErrorCode>0</ErrorCode>
  </Status>
</InsertReservationRS>

```

CancelReservationItem

Cancel the reservation item. Only partner's reservation's items can be canceled.

PARAMETERS

Name	Type	Default value	Description
------	------	---------------	-------------

CancelReservationItemRQ	CancelReservationItemRQ	Request object with parameters needed to cancel the reservation item.
--------------------------------	-----------------------------------------	-----------------------------------------------------------------------

REQUIRED PROPERTIES

Name	Type	Description
ReservationItemID	int	ID of the reservation item that will be canceled.

RESPONSE

Objects of type [CancelReservationItemRS](#).

EXAMPLE

REQUEST

```
POST /API/B2B/CancelReservationItem HTTP/1.1
Host: yourdomain.itravelsoftware.com
Content-Type: application/json
Authorization: Basic dXNlcm5hbWU6cGFzZc3dvcmQ=
Accept-Encoding: application/gzip
Accept: application/json
```

```
{
  "ReservationItemID" : 12345,
  "CancellationReason" : "Some reason for this example",
  "CancelReservationIfAllItemsCancelled" : false
}
```

RESPONSE

```
{
  "Status": {
    "Code": 2,
    "Description": "Reservation item 12345 cannot be cancelled because it doesn't belong to your company.",
    "ErrorCode": 0
  }
}
```

```
{
  "Status": {
    "Code": 1,
    "Description": "Reservation item with ID=12345 has been successfully cancelled."
    "ErrorCode": 0
  }
}
```

GetFlightDestinations

Returns all flight destinations according to search parameters.

PARAMETERS

Name	Type	Default value	Description
GetFlightDestinationsRQ	GetFlightDestinationsRQ		Request object with parameters needed to search flight destinations.

REQUIRED PROPERTIES

Name	Type	Description

RESPONSE

Objects of type [GetFlightDestinationsRS](#).

EXAMPLE

REQUEST

```
POST /API/B2B/CancelReservationItem HTTP/1.1
Host: yourdomain.itravelsoftware.com
Content-Type: application/json
Authorization: Basic dXNlcm5hbWU6cGFzc3dvcmQ=
Accept-Encoding: application/gzip
Accept: application/json
```

```
{
  "NameOrIata" : "LHR"
}
```

RESPONSE

```
{
  "Status": {
    "Code": 2,
    "Description": "OK",
    "ErrorCode": 0
  },
  "FlightDestinationList": [{
    "FlightDestinationID": 7971,
    "TypeID": 3,
    "DisplayText": "Heathrow (LHR), London, United Kingdom",
    "AirportName": "Heathrow",
    "AirportCode": "LHR",
    "CityName": "London",
    "CityCode": "LON",
    "CountryName": "United Kingdom",
    "CountryCode": "GB"
  }]
}
```

GetFlightSearchResults

Searches for the flights.

PARAMETERS

Name	Type	Default value	Description
GetFlightSearchResultsRQ	GetFlightSearchResultsRQ		Object that contains all search parameters for flights

RESPONSE

Object of type [GetFlightSearchResultsRS](#).

EXAMPLE

In this example a request is made to get a flight destination based on IATA code "LHG".

Function returned one flight destination that matched search criteria.

REQUEST

```
POST /API/B2B/CancelReservationItem HTTP/1.1
Host: yourdomain.itravelsoftware.com
Content-Type: application/json
Authorization: Basic dXNlcm5hbWU6cGFzc3dvcmQ=
Accept-Encoding: application/gzip
Accept: application/json
```

```
{
  "AirlineCodes" = [{
    "BA"
  }],
  "AmountTypeID" = 1,
  ChildrenAges = [{
    1,
    5
  }],
  "CurrencyID" = 978,
  "DirectAccess" = false,
  "DirectFlightsOnly" = false,
  "FlightSearchSegments" = [{
    "DepartureEarliestDate" = new DateTime(2018, 3, 1, 00, 0, 0),
    "DepartureLatestDate" = new DateTime(2018, 3, 1, 23, 59, 59),
    "DestinationId" = 8480, // CDG
    "OriginId" = 7971, // LHR
    "ReturningEarliestDate" = new DateTime(2018, 3, 15, 00, 0, 0),
    "ReturningLatestDate" = new DateTime(2018, 3, 15, 23, 59, 59),
  }],
  "LanguageID" = "en",
  "MarketID" = 0,
  "NumberOfAdults" = 2,
  "OrderByID" = 1,
  "Page" = 1,
  "PriceFrom" = 10,
  "PriceTo" = 100000,
  "TicketClassID" = 1,
  "TripTypeID" = 2
};
```

RESPONSE

```
<FlightSearchResponse xmlns="http://tempuri.org/">
  <FlightSearchResult>
```

```

<Status>
  <Code>OK</Code>
  <Description>OK</Description>
  <ErrorCode>0</ErrorCode>
</Status>
<Flights>
  <FlightSearchRSFlight>
    <FlightDirections>
      <FlightSearchRSDirection>
        <DirectionTypeID>1</DirectionTypeID>
        <FlightOptions>
          <FlightSearchRSOption>
            <OptionCode>1-1</OptionCode>
            <FlightSegments>
              <FlightSearchRSSegment>
                <Key>83eKH87Q2BKARoP5CAAAAA==</Key>
                <FareBasisCode>OAP4V2R</FareBasisCode>
                <Arrival>2018-03-01T09:35:00</Arrival>
                <Departure>2018-03-01T07:20:00</Departure>
                <OriginCode>LHR</OriginCode>
                <OriginDescription>Heathrow, London</OriginDescription>
                <DestinationCode>CDG</DestinationCode>
                <DestinationDescription>Charles De Gaulle Intl Arpt,
Paris</DestinationDescription>
                <AirlineName>British Airways</AirlineName>
                <AirlineCode>BA</AirlineCode>
                <FlightNumber>304</FlightNumber>
                <AircraftCode>319</AircraftCode>
                <AircraftName>Airbus Industrie A319</AircraftName>
                <FlightTimeMinutes>75</FlightTimeMinutes>
                <Class>Economy</Class>
                <BookingCode>0</BookingCode>
                <IsConnection>>false</IsConnection>
                <Group>0</Group>
                <AvailabilitySource>A</AvailabilitySource>
            <AirlineLogoThumbnailUrl>http://server/itavel/images/airlines/BA.jpg</AirlineLogoThumbnailUrl>
            <AirlineLogoUrl>http://server/itavel/images/airlines/BA.jpg</AirlineLogoUrl>
          </FlightSearchRSSegment>
        </FlightSegments>
        <TravelTimeMinutes>75</TravelTimeMinutes>
        <OriginCode>LHR</OriginCode>
        <OriginDescription>Heathrow, London</OriginDescription>
        <DestinationCode>CDG</DestinationCode>
        <DestinationDescription>Charles De Gaulle Intl Arpt,
Paris</DestinationDescription>
        <Departure>2018-03-01T07:20:00</Departure>
        <Arrival>2018-03-01T09:35:00</Arrival>
        <AirlineLogoThumbnailUrls>
          <string>http://server/itavel/images/airlines/BA.jpg</string>
        </AirlineLogoThumbnailUrls>
        <AirlineLogoUrls>
          <string>http://server/itavel/images/airlines/BA.jpg</string>
        </AirlineLogoUrls>
      </FlightSearchRSOption>
    </FlightSearchRSOption>
    <OptionCode>1-2</OptionCode>
    <FlightSegments>
      <FlightSearchRSSegment>
        <Key>83eKH87Q2BKAToP5CAAAAA==</Key>
        <FareBasisCode>OAP4V2R</FareBasisCode>
        <Arrival>2018-03-01T17:20:00</Arrival>
        <Departure>2018-03-01T15:05:00</Departure>

```

```

        <OriginCode>LHR</OriginCode>
        <OriginDescription>Heathrow, London</OriginDescription>
        <DestinationCode>CDG</DestinationCode>
        <DestinationDescription>Charles De Gaulle Intl Arpt,
Paris</DestinationDescription>
        <AirlineName>British Airways</AirlineName>
        <AirlineCode>BA</AirlineCode>
        <FlightNumber>316</FlightNumber>
        <AircraftCode>320</AircraftCode>
        <AircraftName>Airbus Industrie A320-100/200</AircraftName>
        <FlightTimeMinutes>75</FlightTimeMinutes>
        <Class>Economy</Class>
        <BookingCode>O</BookingCode>
        <IsConnection>>false</IsConnection>
        <Group>0</Group>
        <AvailabilitySource>A</AvailabilitySource>
    <AirlineLogoThumbnailUrl>http://server/itravel/images/airlines/BA.jpg</AirlineLogoThumbnailUrl>
    <AirlineLogoUrl>http://server/itravel/images/airlines/BA.jpg</AirlineLogoUrl>
    </FlightSearchRSSegment>
</FlightSegments>
    <TravelTimeMinutes>75</TravelTimeMinutes>
    <OriginCode>LHR</OriginCode>
    <OriginDescription>Heathrow, London</OriginDescription>
    <DestinationCode>CDG</DestinationCode>
    <DestinationDescription>Charles De Gaulle Intl Arpt,
Paris</DestinationDescription>
    <Departure>2018-03-01T15:05:00</Departure>
    <Arrival>2018-03-01T17:20:00</Arrival>
    <AirlineLogoThumbnailUrls>
        <string>http://server/itravel/images/airlines/BA.jpg</string>
    </AirlineLogoThumbnailUrls>
    <AirlineLogoUrls>
        <string>http://server/itravel/images/airlines/BA.jpg</string>
    </AirlineLogoUrls>
</FlightSearchRSOption>
</FlightOptions>
</FlightSearchRSDirection>
<FlightSearchRSDirection>
    <DirectionTypeID>2</DirectionTypeID>
    <FlightOptions>
        <FlightSearchRSOption>
            <OptionCode>2-1</OptionCode>
            <FlightSegments>
                <FlightSearchRSSegment>
                    <Key>83eKH87Q2BKAfoP5CAAAAA==</Key>
                    <FareBasisCode>OAP4V2R</FareBasisCode>
                    <Arrival>2018-03-15T18:25:00</Arrival>
                    <Departure>2018-03-15T18:10:00</Departure>
                    <OriginCode>CDG</OriginCode>
                    <OriginDescription>Charles De Gaulle Intl Arpt,
Paris</OriginDescription>
                    <DestinationCode>LHR</DestinationCode>
                    <DestinationDescription>Heathrow, London</DestinationDescription>
                    <AirlineName>British Airways</AirlineName>
                    <AirlineCode>BA</AirlineCode>
                    <FlightNumber>323</FlightNumber>
                    <AircraftCode>320</AircraftCode>
                    <AircraftName>Airbus Industrie A320-100/200</AircraftName>
                    <FlightTimeMinutes>75</FlightTimeMinutes>
                    <Class>Economy</Class>
                    <BookingCode>O</BookingCode>
                    <IsConnection>>false</IsConnection>

```

```

        <Group>1</Group>
        <AvailabilitySource>A</AvailabilitySource>
    <AirlineLogoThumbnailUrl>http://server/ittravel/images/airlines/BA.jpg</AirlineLogoThumbnailUrl>
    <AirlineLogoUrl>http://server/ittravel/images/airlines/BA.jpg</AirlineLogoUrl>
    </FlightSearchRSSegment>
</FlightSegments>
    <TravelTimeMinutes>75</TravelTimeMinutes>
    <OriginCode>CDG</OriginCode>
    <OriginDescription>Charles De Gaulle Intl Arpt, Paris</OriginDescription>
    <DestinationCode>LHR</DestinationCode>
    <DestinationDescription>Heathrow, London</DestinationDescription>
    <Departure>2018-03-15T18:10:00</Departure>
    <Arrival>2018-03-15T18:25:00</Arrival>
    <AirlineLogoThumbnailUrls>
        <string>http://server/ittravel/images/airlines/BA.jpg</string>
    </AirlineLogoThumbnailUrls>
    <AirlineLogoUrls>
        <string>http://server/ittravel/images/airlines/BA.jpg</string>
    </AirlineLogoUrls>
</FlightSearchRSOption>
<FlightSearchRSOption>
    <OptionCode>2-2</OptionCode>
    <FlightSegments>
        <FlightSearchRSSegment>
            <Key>83eKH87Q2BKAhoP5CAAAAA==</Key>
            <FareBasisCode>OAP4V2R</FareBasisCode>
            <Arrival>2018-03-15T20:50:00</Arrival>
            <Departure>2018-03-15T20:35:00</Departure>
            <OriginCode>CDG</OriginCode>
            <OriginDescription>Charles De Gaulle Intl Arpt,
Paris</OriginDescription>
            <DestinationCode>LHR</DestinationCode>
            <DestinationDescription>Heathrow, London</DestinationDescription>
            <AirlineName>British Airways</AirlineName>
            <AirlineCode>BA</AirlineCode>
            <FlightNumber>327</FlightNumber>
            <AircraftCode>320</AircraftCode>
            <AircraftName>Airbus Industrie A320-100/200</AircraftName>
            <FlightTimeMinutes>75</FlightTimeMinutes>
            <Class>Economy</Class>
            <BookingCode>0</BookingCode>
            <IsConnection>false</IsConnection>
        <Group>1</Group>
        <AvailabilitySource>A</AvailabilitySource>
    <AirlineLogoThumbnailUrl>http://server/ittravel/images/airlines/BA.jpg</AirlineLogoThumbnailUrl>
    <AirlineLogoUrl>http://server/ittravel/images/airlines/BA.jpg</AirlineLogoUrl>
    </FlightSearchRSSegment>
</FlightSegments>
    <TravelTimeMinutes>75</TravelTimeMinutes>
    <OriginCode>CDG</OriginCode>
    <OriginDescription>Charles De Gaulle Intl Arpt, Paris</OriginDescription>
    <DestinationCode>LHR</DestinationCode>
    <DestinationDescription>Heathrow, London</DestinationDescription>
    <Departure>2018-03-15T20:35:00</Departure>
    <Arrival>2018-03-15T20:50:00</Arrival>
    <AirlineLogoThumbnailUrls>
        <string>http://server/ittravel/images/airlines/BA.jpg</string>
    </AirlineLogoThumbnailUrls>
    <AirlineLogoUrls>
        <string>http://server/ittravel/images/airlines/BA.jpg</string>
    </AirlineLogoUrls>

```

```

        </FlightSearchRSOption>
    </FlightOptions>
</FlightSearchRSDirection>
</FlightDirections>
<Price>385.30</Price>
<Currency>
    <CurrencyID>978</CurrencyID>
    <CurrencyName>Euri</CurrencyName>
    <CurrencyShortName>EUR</CurrencyShortName>
</Currency>
<BookingAddress>
    <!-- long data -->
</BookingAddress>
<FlightCode>
    <!-- long data -->
</FlightCode>
</FlightSearchRSFlight>
</Flights>
<PageSize>10</PageSize>
<CurrentPage>1</CurrentPage>
<TotalPages>4</TotalPages>
<TotalResults>34</TotalResults>
<Language>
    <LanguageID>en</LanguageID>
    <LanguageName>English</LanguageName>
</Language>
<AllFlightResultsAdditionalData>
    <AirlineList>
        <FlightSearchRSAirline>
            <Code>BA</Code>
            <Name>British Airways</Name>
        </FlightSearchRSAirline>
    </AirlineList>
</AllFlightResultsAdditionalData>
</FlightSearchResult>
</FlightSearchResponse>

```

GetPackageSearchResults

Returns all package tours according to search parameters.

PARAMETERS

Name	Type	Default value	Description
GetPackageSearchResultsRQ	GetPackageSearchResultsRQ		Object that contains parameters for the package tour search

RESPONSE

Object of type [GetPackageSearchResultsRS](#)

EXAMPLE

In this example request is made to get package tour search results for following parameters: startDate = today, endDate =today + 30 days, destination is irrelevant (country , region and destination are set to null) and price range is set from 50 to 5000 Euros per person (priceFrom = 50, priceTo = 5000, currencyID = 978).

REQUEST

```
LemaxAPI_3_0 api = LemaxWebsite.GetLemaxApi();

GetPackageSearchResultsParameters getPackageSearchResultsParameters
    = new GetPackageSearchResultsParameters();

getPackageSearchResultsParameters.StartDate = DateTime.Now;
getPackageSearchResultsParameters.EndDate = DateTime.Now.AddDays(30);
getPackageSearchResultsParameters.PriceFrom = 50.0;
getPackageSearchResultsParameters.PriceTo = 5000.0;
getPackageSearchResultsParameters.InPriceType = PriceType.PerPerson;
getPackageSearchResultsParameters.CurrencyID = 978;
getPackageSearchResultsParameters.LanguageID = "en";

api.GetPackageSearchResults(getPackageSearchResultsParameters);
```

RESPONSE

```
<?xmlversion="1.0"?>
<PackageSearchResultsxmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <PriceFormat>
    <DecimalSeparator>.</DecimalSeparator>
    <GroupSeparator>,</GroupSeparator>
  </PriceFormat>
  <Status>
    <Code>OK</Code>
    <Description>OK</Description>
  </Status>
  <TotalNumberOfResults>4</TotalNumberOfResults>
  <Language>
    <LanguageID>en</LanguageID>
    <LanguageName>English</LanguageName>
  </Language>
  <Currency>
    <CurrencyID>978</CurrencyID>
    <CurrencyName>Euri</CurrencyName>
    <CurrencyShortName>EUR</CurrencyShortName>
  </Currency>
  <DestinationList>
    ...
  </DestinationList>
  <RegionList>
    ...
  </RegionList>
  <CountryList>
    ...
  </CountryList>
  <CurrentPage>1</CurrentPage>
  <PageSize>5</PageSize>
  <StartDate>0001-01-01T00:00:00</StartDate>
  <EndDate>0001-01-01T00:00:00</EndDate>
  <NumberOfPersons>0</NumberOfPersons>
  <NumberOfDays>0</NumberOfDays>
  <PackageTourList>
    <PackageTour>
      <ObjectID>1983</ObjectID>
```



```
<Name>Prag</Name>
<DestinationID>116</DestinationID>
<ObjectType>
<ObjectTypeID>122</ObjectTypeID>
<ObjectName>Packet tour</ObjectName>
</ObjectType>
<AttributeGroupList>
<AttributeGroup>
<GroupID>122</GroupID>
<GroupName>Packet tour</GroupName>
<AttributeList>
<Attribute>
<AttributeID>121</AttributeID>
<AttributeName>Name</AttributeName>
<AttributeValue>Prag</AttributeValue>
<AttributeType>Textual</AttributeType>
<Visible>true</Visible>
</Attribute>
</AttributeList>
</AttributeGroup>
</AttributeGroupList>
<PhotoList />
<CategoryList>
<Category>
<CategoryName />
<CategoryID>8</CategoryID>
</Category>
</CategoryList>
<PackageUnitList>
<PackageUnit>
<UnitID>1984</UnitID>
<AvailabilityStatus>OnRequest</AvailabilityStatus>
<Type>
<UnitTypeID>11</UnitTypeID>
<UnitTypeName>Double room</UnitTypeName>
</Type>
<AttributeGroupList>
<AttributeGroup>
<GroupID>11</GroupID>
<GroupName>Double room</GroupName>
<AttributeList>
<Attribute>
<AttributeID>120</AttributeID>
<AttributeName>Capacity</AttributeName>
<AttributeValue>3</AttributeValue>
<AttributeType>Textual</AttributeType>
<Visible>>false</Visible>
</Attribute>
</AttributeList>
</AttributeGroup>
</AttributeGroupList>
<PhotoList />
<ServiceList>
<Service>
<ServiceID>205</ServiceID>
<ServiceName>Package tour</ServiceName>
<ServiceType>Basic</ServiceType>
<Price>4030.7278320193291</Price>
<PriceFormatted>4,030.73</PriceFormatted>
<PriceType>PerPerson</PriceType>
</Service>
...
</ServiceList>
```

```
<SpecialOfferList />
<PackagePeriodList>
<PackagePeriod>
<CalculatedPriceInfo>
<CalculatedPrice>173.13999938964844</CalculatedPrice>
<CalculatedPriceFormatted>173.14</CalculatedPriceFormatted>
<ServiceList>
<Service>
<ServiceID>205</ServiceID>
<ServiceType>Basic</ServiceType>
<Price>0</Price>
<PriceFormatted>0.00</PriceFormatted>
<PriceType>Total</PriceType>
</Service>
</ServiceList>
<BasicCalculatedPrice>173.13999938964844</BasicCalculatedPrice>
<BasicCalculatedPriceFormatted>173.14</BasicCalculatedPriceFormatted>
</CalculatedPriceInfo>
<StartDate>2010-10-07T00:00:00</StartDate>
<EndDate>2010-10-10T00:00:00</EndDate>
<NumberOfDays>4</NumberOfDays>
<BookingAddress>http://localhost/itravel/Accommodation/Cart.aspx?sifraJezik=2&amp;units
ID=1984&amp;startDate=634220064000000000&amp;endDate=634222656000000000&amp;amount=1&am
p;persons=20</BookingAddress>
<AvailabilityStatus>OnRequest</AvailabilityStatus>
<PriceNumberOfPersons>2</PriceNumberOfPersons>
<Visible>>true</Visible>
</PackagePeriod>
...
</PackagePeriodList>
<AccommodationUnitID>1984</AccommodationUnitID>
</PackageUnit>
</PackageUnitList>
<DaysList>
<int>4</int>
</DaysList>
<DistinctPackagePeriodList>
<PackagePeriod>
<CalculatedPriceInfo>
<CalculatedPrice>173.13999938964844</CalculatedPrice>
<CalculatedPriceFormatted>173.14</CalculatedPriceFormatted>
<ServiceList>
<Service>
<ServiceID>205</ServiceID>
<ServiceType>Basic</ServiceType>
<Price>0</Price>
<PriceFormatted>0.00</PriceFormatted>
<PriceType>Total</PriceType>
</Service>
</ServiceList>
<BasicCalculatedPrice>173.13999938964844</BasicCalculatedPrice>
<BasicCalculatedPriceFormatted>173.14</BasicCalculatedPriceFormatted>
</CalculatedPriceInfo>
<StartDate>2010-10-07T00:00:00</StartDate>
<EndDate>2010-10-10T00:00:00</EndDate>
<NumberOfDays>4</NumberOfDays>
<BookingAddress>http://localhost/itravel/Accommodation/Cart.aspx?sifraJezik=2&amp;units
ID=1984&amp;startDate=634220064000000000&amp;endDate=634222656000000000&amp;amount=1&am
p;persons=20</BookingAddress>
<AvailabilityStatus>OnRequest</AvailabilityStatus>
<PriceNumberOfPersons>2</PriceNumberOfPersons>
<Visible>>true</Visible>
</PackagePeriod>
```

```

    ...
</DistinctPackagePeriodList>
</PackageTour>
    ...
</PackageTourList>
</PackageSearchResults>

```

GetPackageDetailedDescription

Returns package tours detailed description.

PARAMETERS

Name	Type	Default value	Description
GetPackageDetailedDescriptionRQ	GetPackageDetailedDescriptionRQ		Object that contains parameters needed to get package tour info

RESPONSE

Object of type [GetPackageDetailedDescriptionRS](#)

EXAMPLE

In this example a request is made to get a package tour with ID 1983. Prices will be calculated for one person (numberOfPersons = 1) and returned in Euros (currencyID = 978), thumbnails and large photos will be returned in default dimensions (null) and results will be in English (languageID = "en").

REQUEST

```

LemaxAPI_3_0 api = LemaxWebsite.GetLemaxApi();
GetPackageDetailedDescriptionParameters getPackageDetailedDescriptionParameters
    = newGetPackageDetailedDescriptionParameters();

getPackageDetailedDescriptionParameters.NumberOfPersons = 1;
getPackageDetailedDescriptionParameters.PackageTourID = 1983;
getPackageDetailedDescriptionParameters.CurrencyID = 978;
getPackageDetailedDescriptionParameters.LanguageID = "en";
LemaxWebsite.ObjectToHTML(getPackageDetailedDescriptionParameters);

```

RESPONSE

```

<?xmlversion="1.0"?>
<PackageTourDetailsxmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
<Status>
<Code>OK</Code>
<Description>OK</Description>
</Status>
<PackageTour>
<ObjectID>1983</ObjectID>
<Name>Prag</Name>
<DestinationID>116</DestinationID>
<ObjectType>

```

```

<ObjectTypeID>122</ObjectTypeID>
<ObjectName>Package tour</ObjectName>
</ObjectType>
<AttributeGroupList>
<AttributeGroup>
<GroupID>122</GroupID>
<GroupName>Packet tour</GroupName>
<AttributeList>
<Attribute>
<AttributeID>121</AttributeID>
<AttributeName>Name</AttributeName>
<AttributeValue>Prag</AttributeValue>
<AttributeType>Textual</AttributeType>
<Visible>true</Visible>
</Attribute>
</AttributeList>
</AttributeGroup>
<AttributeGroup>
<GroupID>299</GroupID>
<GroupName>Transportation</GroupName>
<AttributeList>
<Attribute>
<AttributeID>840</AttributeID>
<AttributeName>Bus</AttributeName>
<AttributeValue>1</AttributeValue>
<AttributeType>Logical</AttributeType>
<Visible>true</Visible>
</Attribute>
</AttributeList>
</AttributeGroup>
</AttributeGroupList>
<PhotoList>
<Photo>
<PhotoID>1221</PhotoID>
<PhotoUrl>http://localhost/fotografije_itravel/1983/1221_634183295661800000.jpg</PhotoU
rl>
<ThumbnailUrl>http://localhost/fotografije_itravel/1983/1221_634183295661800000.jpg</Th
umbnailUrl>
</Photo>
...
</PhotoList>
<NoteList />
<AvailabilityCalendarAddress>http://localhost/itravel/PregledZauzetosti.aspx?razina=1</
AvailabilityCalendarAddress>
<AccommodationObjectPublicCode>PA-3</AccommodationObjectPublicCode>
<CategoryList />
<PackageUnitList>
<PackageUnit>
<UnitID>1984</UnitID>
<AvailabilityStatus>OnRequest</AvailabilityStatus>
<Type>
<UnitTypeID>11</UnitTypeID>
<UnitTypeName>Double room</UnitTypeName>
</Type>
<AttributeGroupList>
<AttributeGroup>
<GroupID>11</GroupID>
<GroupName>Double room</GroupName>
<AttributeList>
<Attribute>
<AttributeID>120</AttributeID>
<AttributeName>Capacity</AttributeName>
<AttributeValue>3</AttributeValue>

```

```

<AttributeType>Textual</AttributeType>
<Visible>>false</Visible>
</Attribute>
</AttributeList>
</AttributeGroup>
</AttributeGroupList>
<PhotoList />
<ServiceList>
<Service>
<ServiceID>205</ServiceID>
<ServiceName>Package tour</ServiceName>
<ServiceType>Basic</ServiceType>
<Price>0</Price>
<PriceFormatted>0.00</PriceFormatted>
<PriceType>PerPerson</PriceType>
<BillingType>
<BillingTypeID>6</BillingTypeID>
<BillingTypeName>per person/reservation</BillingTypeName>
</BillingType>
<PriceRowList>
<PriceRow>
<PriceRowID>12073</PriceRowID>
<MinimumStay>1</MinimumStay>
<MinimumPersons>1</MinimumPersons>
<MaximumStay>0</MaximumStay>
<PriceItemList>
<PriceItem>
<PeriodID>1564</PeriodID>
<StartDaysOfWeek>1234567</StartDaysOfWeek>
<ListPriceOnDayOfWeek>
<PriceOnDayOfWeek>
<PriceOnDay>1230</PriceOnDay>
<DayOfWeek>Sunday</DayOfWeek>
</PriceOnDayOfWeek>
...
</ListPriceOnDayOfWeek>
</PriceItem>
...
</PriceItemList>
</PriceRow>
...
</PriceRowList>
<GroupID>217</GroupID>
</Service>
...
</ServiceList>
<SpecialOfferList />
<AvailabilityCalendarAddress>http://localhost/itravel/PregledZauzetosti.aspx?razina=2</
AvailabilityCalendarAddress>
<NoteList />
<AccommodationUnitPublicCode>PA-3-1</AccommodationUnitPublicCode>
  <CategoryList>
    <Category>
      <CategoryName>Golf</CategoryName>
      <CategoryID>23</CategoryID>
    </Category>
  </CategoryList>
  <PriceListValidities>
    <PriceListValidity>
      <PriceListValidFrom>2006-01-01T00:00:00</PriceListValidFrom>
      <PriceListValidTo>2100-01-01T00:00:00</PriceListValidTo>
      <PriceListValidityID>156</PriceListValidityID>
    </PriceListValidity>

```

```

        </PriceListValidities>
<CalculatedPriceInfo>
<CalculatedPrice>0</CalculatedPrice>
<CalculatedPriceFormatted>0.00</CalculatedPriceFormatted>
<ServiceList>
<Service>
<ServiceID>205</ServiceID>
<ServiceType>Basic</ServiceType>
<Price>0</Price>
<PriceFormatted>0.00</PriceFormatted>
<PriceType>Total</PriceType>
<GroupID>0</GroupID>
</Service>
...
</ServiceList>
<BasicCalculatedPrice>0</BasicCalculatedPrice>
<BasicCalculatedPriceFormatted>0.00</BasicCalculatedPriceFormatted>
</CalculatedPriceInfo>
<PackagePeriodList>
<PackagePeriod>
<CalculatedPriceInfo>
<CalculatedPrice>173.13999938964844</CalculatedPrice>
<CalculatedPriceFormatted>173.14</CalculatedPriceFormatted>
<ServiceList>
<Service>
<ServiceID>205</ServiceID>
<ServiceName>Package tour</ServiceName>
<ServiceType>Basic</ServiceType>
<Price>0</Price>
<PriceFormatted>0.00</PriceFormatted>
<PriceType>Total</PriceType>
<GroupID>0</GroupID>
</Service>
<Service>
<ServiceID>177</ServiceID>
<ServiceType>Supplement</ServiceType>
<Price>13.850000381469727</Price>
<PriceFormatted>13.85</PriceFormatted>
<PriceType>PerPerson</PriceType>
<GroupID>0</GroupID>
</Service>
</ServiceList>
<BasicCalculatedPrice>173.13999938964844</BasicCalculatedPrice>
<BasicCalculatedPriceFormatted>173.14</BasicCalculatedPriceFormatted>
</CalculatedPriceInfo>
<StartDate>2010-10-07T00:00:00</StartDate>
<EndDate>2010-10-10T00:00:00</EndDate>
<NumberOfDays>4</NumberOfDays>
<BookingAddress>http://localhost/itravel/Accommodation/Cart.aspx?sifraJezik=2&units
ID=1984&startDate=634220064000000000&endDate=634222656000000000&amount=&
;persons=</BookingAddress>
<AvailabilityStatus>OnRequest</AvailabilityStatus>
<PriceNumberOfPersons>2</PriceNumberOfPersons>
<Visible>true</Visible>
</PackagePeriod>
...
</PackagePeriodList>
<AccommodationUnitID>1984</AccommodationUnitID>
</PackageUnit>
</PackageUnitList>
<DaysList>
<int>4</int>
</DaysList>

```

```

<DistinctPackagePeriodList>
<PackagePeriod>
<CalculatedPriceInfo>
<CalculatedPrice>173.13999938964844</CalculatedPrice>
<CalculatedPriceFormatted>173.14</CalculatedPriceFormatted>
<ServiceList>
<Service>
<ServiceID>205</ServiceID>
<ServiceName>Package tour</ServiceName>
<ServiceType>Basic</ServiceType>
<Price>0</Price>
<PriceFormatted>0.00</PriceFormatted>
<PriceType>Total</PriceType>
<GroupID>0</GroupID>
</Service>
<Service>
<ServiceID>177</ServiceID>
<ServiceType>Supplement</ServiceType>
<Price>13.850000381469727</Price>
<PriceFormatted>13.85</PriceFormatted>
<PriceType>PerPerson</PriceType>
<GroupID>0</GroupID>
</Service>
</ServiceList>
<BasicCalculatedPrice>173.13999938964844</BasicCalculatedPrice>
<BasicCalculatedPriceFormatted>173.14</BasicCalculatedPriceFormatted>
</CalculatedPriceInfo>
<StartDate>2010-10-07T00:00:00</StartDate>
<EndDate>2010-10-10T00:00:00</EndDate>
<NumberOfDays>4</NumberOfDays>
<BookingAddress>http://localhost/itravel/Accommodation/Cart.aspx?sifraJezik=2&amp;units
ID=1984&amp;startDate=634220064000000000&amp;endDate=634222656000000000&amp;amount=&amp
;persons=</BookingAddress>
<AvailabilityStatus>OnRequest</AvailabilityStatus>
<PriceNumberOfPersons>2</PriceNumberOfPersons>
<Visible>true</Visible>
</PackagePeriod>
...
</DistinctPackagePeriodList>
</PackageTour>
<AccommodationObjectList />
<DestinationList>
...
</DestinationList>
<RegionList>
...
</RegionList>
<CountryList>
...
</CountryList>
<Language>
<LanguageID>en</LanguageID>
<LanguageName>English</LanguageName>
</Language>
<Currency>
<CurrencyID>978</CurrencyID>
<CurrencyName>Euri</CurrencyName>
<CurrencyShortName>EUR</CurrencyShortName>
</Currency>
<PriceType>PerPerson</PriceType>
<PriceFormat>
<DecimalSeparator>.</DecimalSeparator>
<GroupSeparator>,</GroupSeparator>

```

```
</PriceFormat>
</PackageTourDetails>
```

GetReservationCancellation

Returns reservation cancellation data.

PARAMETERS

Name	Type	Default value	Description
GetReservationCancellationRQ	GetReservationCancellationRQ		Object that contains parameters needed to get reservation cancellation data

RESPONSE

Object of type [ReservationCancellationRS](#)

EXAMPLE

In this example request is made to cancel reservation item with Reservation Item ID 12272.

REQUEST

```
LemaxAPI_3_0 api = LemaxWebsite.GetLemaxApi();
GetReservationCancellationRQ getReservationCancellationParameters
    = new GetReservationCancellationRQ();

getReservationCancellationParameters.ReservationID = 2497;
LemaxWebsite.ObjectToHTML(getPackageDetailedDescriptionParameters);
```

RESPONSE

```
<?xmlversion="1.0"?>
<ReservationCancellationRS
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <Status>
    <Code>OK</Code>
    <ErrorCode>0</ErrorCode>
  </Status>
  <ReservationID>2497</ReservationID>
  <PaidAmount>0</PaidAmount>
  <RemainingAmount>5985.23</RemainingAmount>
  <Currency>
    <CurrencyID>784</CurrencyID>
    <CurrencyShortName>AED</CurrencyShortName>
  </Currency>
  <CancellationItems>
    <ReservationItemCancellationInfoDTO>
      <ReservationItemID>12278</ReservationItemID>
      <Name>Radisson Blue Resort</Name>
      <OriginalPrice>3230.8</OriginalPrice>
      <CancellationFee>3230.80</CancellationFee>
```



```

    <InputVAT>153.78</InputVAT>
    <OutputVAT>128.21</OutputVAT>
    <Commission>0</Commission>
  </ReservationItemCancellationInfoDTO>
  <ReservationItemCancellationInfoDTO>
    <ReservationItemID>12280</ReservationItemID>
    <Name>Apartment Studio Šestan</Name>
    <OriginalPrice>2754.43</OriginalPrice>
    <CancellationFee>2754.43</CancellationFee>
    <InputVAT>0</InputVAT>
    <OutputVAT>0</OutputVAT>
    <Commission>0</Commission>
  </ReservationItemCancellationInfoDTO>
</CancellationItems>
</ReservationCancellationRS>

```

CancelReservationItem

Cancels reservation item and returns canceled reservation item.

PARAMETERS

Name	Type	Default value	Description
CancelReservationItemRQ	CancelReservationItemRQ		Object that contains parameters needed to cancel reservation item

RESPONSE

Object of type [CancelReservationItemRS](#)

EXAMPLE

In this example a request is made to cancel a reservation item with Reservation Item ID 12272. We are also sending cancellation reasons and CancelReservationIfAllItemsCancelled parameter, which determines if reservation will be canceled, if all items were canceled.

REQUEST

```

LemaxAPI_3_0 api = LemaxWebsite.GetLemaxApi();
CancelReservationItemRQ cancelReservationItemParameters
    = new CancelReservationItemRQ();

cancelReservationItemParameters.ReservationItemID = 12272;
cancelReservationItemParameters.CancellationReason = "Testing cancellation";
cancelReservationItemParameters.CancelReservationIfAllItemsCancelled = true;

```

RESPONSE

```

<?xmlversion="1.0"?>
<CancelReservationItemRS
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema">

```

```

<Status>
  <Code>OK</Code>
  <Description>Reservation item with ID=12272 has been successfully
cancelled.</Description>
  <ErrorCode>0</ErrorCode>
</Status>
<ReservationItemID>12272</ReservationItemID>
</CancelReservationItemRS>

```

CancelReservationItems

Cancels reservation items and returns canceled reservation items.

PARAMETERS

Name	Type	Default value	Description
CancelReservationItemsRQ	CancelReservationItemsRQ		Object that contains parameters needed to cancel reservation items

RESPONSE

Object of type [CancelReservationItemsRS](#)

EXAMPLE

In this example request is made to cancel reservation items with Reservation Item ID 12201 and 12203 of reservation 2479. We are also sending cancellation reason for both items, CancelReservationIfAllItemsCancelled parameter which determines if reservation will be canceled if all items were canceled and GenerateRefundTransactionAutomatically parameter which determines if the refund will be generated automatically.

REQUEST

```

LemaxAPI_3_0 api = LemaxWebsite.GetLemaxApi();
CancelReservationItemsRQ cancelReservationItemsParameters
= new CancelReservationItemsRQ();

cancelReservationItemParameters.ReservationID = 2479;
cancelReservationItemParameters.CancelReservationItemBaseDTO = new
List<CancelReservationItemBaseDTO>{
  new CancelReservationItemBaseDTO{
    ReservationItemID = 12201,
    CancellationReason = ""
  },
  new CancelReservationItemBaseDTO{
    ReservationItemID = 12203,
    CancellationReason = ""
  }
};

cancelReservationItemParameters.GenerateRefundTransactionAutomatically = true;
cancelReservationItemParameters.CancelReservationIfAllItemsCancelled = true;

```

RESPONSE

```
<?xmlversion="1.0"?>
<CancelReservationItemsRS
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <ReservationItemCancellations>
    <CancelReservationItemRS>
      <Status>
        <Code>OK</Code>
        <Description>Reservation item with ID=12201 has been successfully
cancelled.</Description>
        <ErrorCode>0</ErrorCode>
      </Status>
      <ReservationItemID>12201</ReservationItemID>
    </CancelReservationItemRS>
    <CancelReservationItemRS>
      <Status>
        <Code>OK</Code>
        <Description>Reservation item with ID=12203 has been successfully
cancelled.</Description>
        <ErrorCode>0</ErrorCode>
      </Status>
      <ReservationItemID>12203</ReservationItemID>
    </CancelReservationItemRS>
  </ReservationItemCancellations>
  <Status>
    <Code>OK</Code>
    <Description>Reservation items cancelled successfully</Description>
    <ErrorCode>0</ErrorCode>
  </Status>
</CancelReservationItemsRS>
```

4. PARAMETERS DESCRIPTIONS

ObjectTypeID

Represents the ID of object type.

Every object has a type.

PREDEFINED VALUES

Object type numeric code	Object type name
1	Hotel
3	Apartment
14	Vacation house
70	Villa
72	Small hotel
74	Sailboat
76	Catamaran
78	Motor boat
83	Gulet
103	Arrangement
115	Excursion
118	Cruise

UnitTypeID

Represents the ID of unit type.

Every unit has a type.

PREDEFINED VALUES

Object type numeric code	Object type name
109	Package

ObjectTypeGroupID

Represents the ID of type group.

Every object type belongs to one group.

PREDEFINED VALUES

Object type group numeric code	Object type group name
3	Private accommodation
4	Hotel accommodation

5	Boats
6	Arrangements
10	Excursions

CategoryID

Represents the ID of object's or unit's category.

Every object or unit can belong to zero or more categories. CategoryID does not have predefined values and can be inserted and changed through Lemax.

Example of categories: "Recommended", "Special offer", etc.

5. CLASSES

GetApiSettingsRQ

This object contains the parameters for api settings (used in online search)

MEMBERS

Name	Type	Description
LanguageID	string	Id of a language to return in response
CurrencyID	int?	Id of a currency to return in response

GetApiSettingsRS

Contains information about API settings

MEMBERS

Name	Type	Description
Language	Language	Language used in API
Currency	Currency	Currency used in API

GetSearchFieldsRQ

This object contains the parameters used to get a list of desired search fields

MEMBERS

Name	Type	Description
LanguageID	string	Id of a language to return in the response
ObjectTypeIDList	byte[]	If objectTypeIDList is not empty, only categories, countries, regions and destinations that contain objects of those types will be returned. Details
ObjectTypeGroupIDList	byte[]	If objectTypeGroupIDList is not empty, only categories, countries, regions and destinations that contain objects whose type belongs to the group contained in that list will be returned. Details
CategoryIDList	int[]	If categoryIDList is not empty only countries, regions and destinations that contains objects in those categories will be returned. Details
CountryID	int?	Only regions and destinations that are in country with that id will be returned.

RegionID	int?	Only destinations that are in region with that id will be returned
-----------------	------	--------------------------------------------------------------------

GetSearchFieldsRS

Contains information necessary for search form (list of categories, countries, regions and destinations to choose from when searching for accommodation or package tour).

MEMBERS

Name	Type	Description
Categories	Categories	List of available categories to search from.
CountryList	Country []	List of available countries to search from.
DestinationList	Destination []	List of available destinations to search from.
RegionList	Region []	List of available regions to search from.

GetSearchResultsRQ

This object contains the parameters used to get the correct units in a search

MEMBERS

Name	Type	Description
StartDate	DateTime (nullable)	Start date for period in which unit should be available. If startDate is set to null or equals to endDate, availability is not checked. DateTime format
EndDate	DateTime (nullable)	End date for period in which unit should be available. If endDate is set to null or startDate, availability is not checked. DateTime format
DestinationIDList	Long []	[Obsolete] List of destinationIDs. If list is set, function returns only objects that belong to one of the destination with destinationID defined in the list.
RegionIDList	Long []	List of regionIDs. If list is set, function returns only objects that belong to region with regionID defined in the list.
CountryIDList	Long []	List of dountryIDs. If list is set, function returns only objects that belong to country with countryID defined in the list.

ObjectTypeIDList	Byte[]	If objectTypeIDList is not empty, function returns only objects whose types are contained in that list. Details
ObjectTypeGroupIDList	Byte[]	If objectTypeGroupIDList is not empty, function returns only objects whose type belongs to the group contained in that list. Details
CategoryIDListUnion	Int[]	If categoryIDListUnion is not empty, function returns only objects who belongs to at least one of the categories from list. Details
PriceFrom	Double (nullable)	If priceFrom is set, function returns only units with price greater or equal to priceFrom.
PriceTo	Double (nullable)	If priceTo is set, function returns only units with price less or equal to priceTo.
PriceType	PriceType	Price type for prices that would be returned in response. PriceType is also used when parameters priceFrom or priceTo are set. Then priceFrom and priceTo are of current priceType.
SortParameterList	SortParameter []	SortParameterList is used for sorting response. Response can be sorted by multiple values. Sort priority is equal to sort parameters order in sortParameterList (first element has top priority).
PageSize	Int (nullable)	Defines how many objects will be returned in response. Maximum allowed is 20. We recommend returning up to 10 objects, because returning more objects increases data transferred over internet and reduces speed.
CurrentPage	Int (nullable)	Defines what page of the response will be returned. If number that exceeds total number of pages is sent, last page will be returned.
ObjectAttributeFilterList	AttributeFilter []	When objectAttributeFilterList is not empty, objects in response will be filtered out by given attributes in objectAttributeFilterList. This parameter is used to filter objects by Name, Stars, etc.
UnitAttributeFilterList	AttributeFilter []	When unitAttributeFilterList is not empty, units in response will be filtered out by given attributes in unitAttributeFilterList. This

		parameter is used to filter units by Capacity, Air Condition, etc.
ThumbnailWidth	Int (nullable)	Maximum width of thumbnail photos. Thumbnail photos preserve original photo ratio. If thumbnailWidth and thumbnailHeight are set, photo will decrease to match these values preserving original photo ratio.
ThumbnailHeight	Int (nullable)	Maximum height of thumbnail photos. Thumbnail photos preserve original photo ratio. If thumbnailWidth and thumbnailHeight are set, photo will decrease to match these values preserving original photo ratio.
PhotoWidth	Int (nullable)	Maximum width of large photos. Large photos preserve original photo ratio. If photoWidth and photoHeight are set, photo will decrease to match these values preserving original photo ratio.
PhotoHeight	Int (nullable)	Maximum width of large photos. Large photos preserve original photo ratio. If photoWidth and photoHeight are set, photo will decrease to match these values preserving original photo ratio.
CurrencyID	Int (nullable)	Currency for prices that would be returned. Details
OutParameterList	OutParameter []	OutParameterList is used to limit information about objects and units in response. Only information given in the outParameterList will be returned.
UnitsAvailable	Byte (nullable)	Number of units (e.g. rooms) that have to be available in searched period. If UnitAvailabilityStatus is OnRequest, then unit will always be returned. If UnitAvailabilityStatus is Availabel, then specified number of units must be available in searched period.
LanguageID	String	ID of language on which data will be returned. Details
IgnorePriceAndAvailability	Bool	If ignorePriceAndAvailability is set to True, response will not contain any information about services, prices and will not filter out occupied units.
UnitTypeIDList	Byte[]	If unitTypeIDList is not empty, function returns only units of those types. Details

UnitCategoryIDList	Int[]	If unitCategoryIDList is not empty, function returns only units from those categories. Details
CategoryIDListIntersection	Int[]	If categoryIDListIntersection is not empty, function returns only objects who belongs to all categories from list. Details
OnlyOnSpecialOffer	Bool (nullable)	Search only objects that have special offer
ToDisplaySpecialOffers	Bool (nullable)	If true results will be grouped by special offer (so more than one instance of the same accommodation object can be returned but each will have only one special offer)
ChildrenAgeList	Int[]	A list of integers representing ages of the children present in the search
DestinationCodes	String[]	List of destinationIDs. If list is set, function returns only objects that belong to one of the destination with destinationID defined in the list.
IgnoreStartDay	Bool	If this parameter is set, start dows and end dows will be ignored in the search.
ObjectIDList	Int[]	List of Object IDs. The search will return only objects whose IDs are in the list. If passed NULL or the list is empty, the parameter is ignored.
ServiceIDList	Int[]	List of basic service IDs. The search will only return objects whose units have basic services from the list. On those objects, only units whose basic service is on the list will be returned. If the list is NULL or empty, the parameter is ignored.
PassengerConfigurationFilterList	PassengerConfigurationFilter []	Filter for passenger configurations (capacity cases) on search.

GetSearchResultsRS

Object that is returned in function [GetSearchResults](#). It contains information about returned objects, language, currency and object's destinations, regions and countries.

MEMBERS

Name	Type	Description
AccommodationObjectList	AccommodationObject []	List of objects returned in search results.

DestinationList	Destination[]	List of destination in which objects from AccommodationObjectList belong.
RegionList	Region[]	List of regions in which destinations from DestinationList belong.
CountryList	Country[]	List of countries in which regions from RegionList belong.
ObjectLocationList	ObjectLocation[]	List of coordinates for the objects returned in search results
Currency	Currency	Currency for prices returned by the function.
Language	Language	Language for data returned by the function.
Status	Status	Status of the response. Determines if the result data is valid or error has occurred on server.
TotalNumberOfResults	Int	Total number of objects that meets given parameters in GetSearchResults .
CurrentPage	Int	Defines what page of the response is returned. If number that exceeds total number of pages is sent, last page is returned
PageSize	int	Number of results per page
StartDate	DateTime	Search was performed between start date and end date
EndDate	DateTime	Search was performed between start date and end date
NumberOfDays	Int	Number of days between start and end date
NumberOfPersons	Int	Number of persons that search was performed with
PriceFormat	PriceFormat	This object contains information on how to properly display price on current language.

GetAllCountriesRQ

This object contains the parameters used to get list of countries

MEMBERS

Name	Type	Description
LanguageID	String	ID of language on which data will be returned. Details
SearchQuery	String	First two or more letters of the search query used to filter out countries by name

GetAllCountriesRS

Object that is returned in GetAllCountries function. It contains information about returned object, country list.

MEMBERS

Name	Type	Description
CountryList	Country[]	List of countries. List contains countries that match letters from SearchQuery

GetAllDestinationsRQ

This object contains the parameters used to get list of destinations

MEMBERS

Name	Type	Description
LanguageID	String	ID of language on which data will be returned. Details
TransferDestinations	bool(nullable)	True if we search for transfer destination
SearchQuery	String	First two or more letters of the search query used to filter out destinations by name within country
TransferPickupID	Int(nullable)	ID of a pickup destination used to filter the destinations when dropoff destination is searched, while pickup place is previously selected
SelectedCountryID	Int(nullable)	ID of the selected country

GetAllDestinationsRS

Object that is returned in GetAllDestinations function. It contains information about the returned object Country list, region list and destination list.

MEMBERS

Name	Type	Description
CountryList	Country[]	List of countries. List contains countries that match letters from SearchQuery
RegionList	Region[]	List of regions. List contains regions of found transportations whether they are pickup or dropoff destinations.

DestinationList	Destination []	List of destinations. List contains destinations of found transportations whether they are pickup or dropoff destinations.
------------------------	--------------------------------	----------------------------------------------------------------------------------------------------------------------------

GetDetailedDescriptionRQ

This object contains the parameters used to get the unit details

MEMBERS

Name	Type	Description
StartDate	DateTime (nullable)	Start date for period in which unit prices will be returned. If startDate is null, it will be set to current's date following day. DateTime format
EndDate	DateTime (nullable)	End date for period in which unit prices will be returned . If endDate is null, it will be set to seven days after startDate. DateTime format
NumberOfPersons	Int (nullable)	Number of persons for which price will be calculated. If numberOfPersons is null 1 person is assumed
ObjectID	Int	ID of object whose detailed description is returned. Required parameter.
ThumbnailWidth	Int (nullable)	Maximum width of thumbnail photos. Thumbnail photos preserve original photo ratio. If thumbnailWidth and thumbnailHeight are set, photo will decrease to match these values preserving original photo ratio.
ThumbnailHeight	Int (nullable)	Maximum height of thumbnail photos. Thumbnail photos preserve original photo ratio. If thumbnailWidth and thumbnailHeight are set, photo will decrease to match these values

		preserving original photo ratio.
PhotoWidth	Int (nullable)	Maximum width of large photos. Large photos preserve original photo ratio. If photoWidth and photoHeight are set, photo will decrease to match these values preserving original photo ratio.
PhotoHeight	Int (nullable)	Maximum width of large photos. Large photos preserve original photo ratio. If photoWidth and photoHeight are set, photo will decrease to match these values preserving original photo ratio.
CurrencyID	Int (nullable)	Currency for prices that would be returned. Details
LanguageID	String	ID of language on which data will be returned. Details
UnitsAvailable	Bool (nullable)	If unitsAvailable is set then returned price is calculated for this number of units.
IgnorePriceAndAvailability	Bool	If ignorePriceAndAvailability is set to True, response will not contain any information about service prices or unit's availability.
UnitID	Int (nullable)	If unitID is set, function returns only unit with that id form the object's unit list.
UnitTypeID	Byte (nullable)	If unitTypeID is set, function returns only units of that type. Details
UnitCategoryID	Int (nullable)	If unitCategoryID is set, function returns only units from that category. Details
MapWidth	Int (nullable)	If object's location is entered, mapWidth sets width of the map in pixels returned by the API. If mapWidth is not set, the default map width is returned.
MapHeight	Int (nullable)	If object's location is entered, mapHeight sets

		height of the map in pixels returned by the API. If mapHeight is not set, the default map height is returned.
InPriceType	PriceType (nullable)	Price type for prices that would be returned in response.
ObjectURL	String	SEO (search engine optimizations) object URL. If this parameter is set, function ignores objectID and languageID parameters and returns the object that the objectURL links to. The language is determined from the URL, too, because each language has a different URL for every object.
UnitIDList	Int[]	List of unit IDs for current object.
ChildrenAgeList	Int[]	A list of integer values representing ages of the children present in the search
IsRelativeUrl	bool(nullable)	Tells if object URL is relative or absolute
ObjectCode	String	
OutParameterList	AccommodationDetailedDescriptionOutparameter []	OutParameterList is used to limit information about objects and units in response or to fetch additional information. Only information given in the outParameterList will be returned.

GetDetailedDescriptionRS

Object that is returned in [GetDetailedDescription](#) function. It contains information about returned object, period list, language, currency and object's destination, region and country.

MEMBERS

Name	Type	Description
AccommodationObject	AccommodationObject	Object whose detailed information is returned by the function.
Destination	Destination	Destination in which returned AccommodationObject belongs.

Region	Region	Region in which returned Destination belongs.
Country	Country	Country in which returned Region belongs.
PeriodList	Period[]	Period list for prices of services in units contained by the returned object.
Currency	Currency	Currency for prices returned by the function.
Language	Language	Language for data returned by the function.
Status	Status	Status of the response. Determines if the result data is valid or error has occurred on server.
StartDate	DateTime	Start date used for price calculation
EndDate	DateTime	End date used for price calculation
NumberOfPersons	Int	Number of persons used in price calculation
PriceType	PriceType	Price type for prices returned by function
NumberOfDays	Int	Number of days between start and end date
PriceFormat	PriceFormat	This object contains information on how to properly display price on current language.

GetTransportationSearchResultsRQ

This object contains the parameters used to get the correct units in a search

MEMBERS

Name	Type	Description
StartDate	DateTime (nullable)	Start date for period in which unit should be available. If startDate is set to null or equals to endDate, availability is not checked. DateTime format
EndDate	DateTime (nullable)	End date for period in which unit should be available. If endDate is set to null or startDate, availability is not checked. DateTime format
PickupDestinationIDList	Long[]	List of pickup destinationIDs. If list is set, function returns only objects that belong to one of the pickup destinations with pickup destinationID defined in the list.
PickupRegionIDList	Long[]	List of pickup regionIDs. If list is set, function returns only objects that belong to regions with pickup regionID defined in the list.

PickupCountryIDList	Long[]	List of pickup countryIDs. If list is set, function returns only objects that belong to pickup countries with countryID defined in the list.
DropoffDestinationIDList	Long[]	List of dropoff destinationIDs. If list is set, function returns only objects that belong to one of the dropoff destinations with dropoff destinationID defined in the list.
DropoffRegionIDList	Long[]	List of dropoff regionIDs. If list is set, function returns only objects that belong to regions with dropoff regionID defined in the list.
DropoffCountryIDList	Long[]	List of dropoff countryIDs. If list is set, function returns only objects that belong to dropoff countries with countryID defined in the list.
ObjectTypeIDList	Byte[]	If objectTypeIDList is not empty, function returns only objects whose types are contained in that list. Details
ObjectTypeGroupIDList	Byte[]	If objectTypeGroupIDList is not empty, function returns only objects whose type belongs to the group contained in that list. Details
CategoryIDListUnion	Int[]	If categoryIDListUnion is not empty, function returns only objects who belongs to at least one of the categories from list. Details
PriceFrom	Double (nullable)	If priceFrom is set, function returns only units with price greater or equal to priceFrom.
PriceTo	Double (nullable)	If priceTo is set, function returns only units with price less or equal to priceTo.
InPriceType	PriceType	Price type for prices that would be returned in response. PriceType is also used when parameters priceFrom or priceTo are set. Then priceFrom and priceTo are of current priceType.
SortParameterList	SortParameter []	SortParameterList is used for sorting response. Response can be sorted by multiple values. Sort priority is equal to sort parameters order in sortParameterList (first element has top priority).
PageSize	Int (nullable)	Defines how many objects will be returned in response. Maximum allowed is 20. We recommend returning up to 10 objects, because returning more

		objects increases data transferred over internet and reduces speed.
CurrentPage	Int (nullable)	Defines what page of the response will be returned. If number that exceeds total number of pages is sent, last page will be returned.
ObjectAttributeFilterList	AttributeFilter[]	When objectAttributeFilterList is not empty, objects in response will be filtered out by given attributes in objectAttributeFilterList. This parameter is used to filter objects by Name, Stars, etc.
UnitAttributeFilterList	AttributeFilter[]	When unitAttributeFilterList is not empty, units in response will be filtered out by given attributes in unitAttributeFilterList. This parameter is used to filter units by Capacity, Air Condition, etc.
ThumbnailWidth	Int (nullable)	Maximum width of thumbnail photos. Thumbnail photos preserve original photo ratio. If thumbnailWidth and thumbnailHeight are set, photo will decrease to match these values preserving original photo ratio.
ThumbnailHeight	Int (nullable)	Maximum height of thumbnail photos. Thumbnail photos preserve original photo ratio. If thumbnailWidth and thumbnailHeight are set, photo will decrease to match these values preserving original photo ratio.
PhotoWidth	Int (nullable)	Maximum width of large photos. Large photos preserve original photo ratio. If photoWidth and photoHeight are set, photo will decrease to match these values preserving original photo ratio.
PhotoHeight	Int (nullable)	Maximum width of large photos. Large photos preserve original photo ratio. If photoWidth and photoHeight are set, photo will decrease to match these values preserving original photo ratio.
CurrencyID	Int (nullable)	Currency for prices that would be returned. Details
OutParameterList	OutParameter[]	OutParameterList is used to limit information about objects and units in response. Only information given in the outParameterList will be returned.

UnitsAvailable	Byte (nullable)	Number of units (e.g. rooms) that have to be available in searched period. If UnitAvailabilityStatus is OnRequest, then unit will always be returned. If UnitAvailabilityStatus is Availabel, then specified number of units must be available in searched period.
LanguageID	String	ID of language on which data will be returned. Details
IgnorePriceAndAvailability	Bool	If ignorePriceAndAvailability is set to True, response will not contain any information about services, prices and will not filter out occupied units.
UnitTypeIDList	Byte[]	If unitTypeIDList is not empty, function returns only units of those types. Details
UnitCategoryIDList	Int[]	If unitCategoryIDList is not empty, function returns only units from those categories. Details
CategoryIDListIntersection	Int[]	If categoryIDListIntersection is not empty, function returns only objects who belongs to all categories from list. Details
OnlyOnSpecialOffer	Bool (nullable)	Search only objects that have special offer
ToDisplaySpecialOffers	Bool (nullable)	If true results will be grouped by special offer (so more than one instance of the same accommodation object can be returned but each will have only one special offer)
ChildrenAgeList	Int[]	A list of integers representing ages of the children present in the search
PickupDestinationName	String	Name of pickup destination.
DropoffDestinationName	String	Name of dropoff destination.
DestinationCodes	String[]	List of destinationIDs. If list is set, function returns only objects that belong to one of the destination with destinationID defined in the list.
ObjectIDList	Int[]	List of Object IDs. The search will return only objects whose IDs are in the list. If passed NULL or the list is empty, the parameter is ignored.

GetTransportationSearchResultsRS

This object contains the parameters used to get the correct units in a search

MEMBERS

Name	Type	Description
StartDate	DateTime (nullable)	Start date for period in which transportation should be available. If startDate is set to null or equals to endDate, availability is not checked. DateTime format
EndDate	DateTime (nullable)	End date for period in which transportation should be available. If endDate is set to null or startDate, availability is not checked. DateTime format
NumberOfDays	Int	Difference between EndDate and StartDate shown as number of days of Transportation.
DestinationList	Destination []	List of destinations. List contains destinations of found transportations whether they are pickup or dropoff destinations.
RegionList	Region []	List of regions. List contains regions of found transportations whether they are pickup or dropoff destinations.
CountryList	Country []	List of countries. List contains countries of found transportations whether they are pickup or dropoff destinations.
TransportationList	Transportation []	List of found transportations. Result of TransportationSearch.
PriceFormat	PriceFormat (nullable)	PriceFormat for TransportationSearchResults.
PageSize	Int (nullable)	Defines how many objects will be returned in response. Maximum allowed is 20. We recommend returning up to 10 objects, because returning more objects increases data transferred over internet and reduces speed.
CurrentPage	Int (nullable)	Defines what page of the response will be returned. If number that exceeds total number of pages is sent, last page will be returned.
Currency	Currency	Currency for prices of TransportationSearchResults.
Status	Status	Status of search. Status code of 1 means the search finished successfully.
NumberOfPersons	Int	Number of persons designates for how many persons the transportation is needed. It is used as an upper bound in search for transfers.

ObjectLocationList	ObjectLocation []	OutParameterList is used to limit information about objects and units in response. Only information given in the outParameterList will be returned.
TotalNumberOfResults	Int	Number of returned results.
Language	Language	Language in which search was performed and the results are shown.
ServiceName	String	Name of the service with which to filter the results.

GetTransportationDetailedDescriptionRQ

This object contains the parameters used to get the transportation details

MEMBERS

Name	Type	Description
StartDate	DateTime (nullable)	Start date for period in which transportation prices will be returned. If startDate is null, it will be set to current's date following day. DateTime format
EndDate	DateTime (nullable)	End date for period in which transportation prices will be returned . If endDate is null, it will be set to seven days after startDate. DateTime format
NumberOfPersons	Int (nullable)	Number of persons for which price will be calculated. If numberOfPersons is null 1 person is assumed
ObjectID	Int	ID of object whose detailed description is returned. Required parameter.
ObjectCode	String	String representation of ObjectID. Used for GDS.
ThumbnailWidth	Int (nullable)	Maximum width of thumbnail photos. Thumbnail photos preserve original photo ratio. If thumbnailWidth and thumbnailHeight are set, photo will decrease to match these values preserving original photo ratio.

ThumbnailHeight	Int (nullable)	Maximum height of thumbnail photos. Thumbnail photos preserve original photo ratio. If thumbnailWidth and thumbnailHeight are set, photo will decrease to match these values preserving original photo ratio.
PhotoWidth	Int (nullable)	Maximum width of large photos. Large photos preserve original photo ratio. If photoWidth and photoHeight are set, photo will decrease to match these values preserving original photo ratio.
PhotoHeight	Int (nullable)	Maximum width of large photos. Large photos preserve original photo ratio. If photoWidth and photoHeight are set, photo will decrease to match these values preserving original photo ratio.
CurrencyID	Int (nullable)	Currency for prices that would be returned. Details
LanguageID	String	ID of language on which data will be returned. Details
UnitsAvailable	Bool (nullable)	If unitsAvailable is set then returned price is calculated for this number of units.
IgnorePriceAndAvailability	Bool	If ignorePriceAndAvailability is set to True, response will not contain any information about service prices or unit's availability.
UnitID	Int (nullable)	If unitID is set, function returns only unit with that id form the object's unit list.
UnitTypeID	Byte (nullable)	If unitTypeID is set, function returns only units of that type. Details
UnitCategoryID	Int (nullable)	If unitCategoryID is set, function returns only units from that category. Details
MapWidth	Int (nullable)	If object's location is entered, mapWidth sets

		width of the map in pixels returned by the API. If mapWidth is not set, the default map width is returned.
MapHeight	Int (nullable)	If object's location is entered, mapHeight sets height of the map in pixels returned by the API. If mapHeight is not set, the default map height is returned.
InPriceType	PriceType (nullable)	Price type for prices that would be returned in response.
ListUnitActivityStatus	UnitActivityStatus []	List of UnitActivityStatus enums. It is used to set the activity statuses of units.
ObjectURL	String	SEO (search engine optimizations) object URL. If this parameter is set, function ignores objectID and languageID parameters and returns the object that the objectURL links to. The language is determined from the URL, too, because each language has a different URL for every object.
ChildrenAgeList	Int[]	A list of integer values representing ages of the children present in the search
IsRealtiveUrl	Bool	Set in case url to transportation details is realtive
UnitIDList	Int[]	List of unit IDs present in transportation details.
OutParameterList	TransportationDetailedDescriptionOutParameter []	OutParameterList is used to limit information about objects and units in response or to fetch additional information. Only information given in the outParameterList will be returned.

GetTransportationDetailedDescriptionRS

Object that is returned in [GetTransportationDetailedDescription](#) function. It contains information about returned transportation, period list, language, currency and transportation's destination, region and country.

MEMBERS

Name	Type	Description
Transportation	Transportation	Transportation object whose detailed information is returned by the function.
DestinationList	Destination[]	Destinations in which returned Transportation belongs (pickup and dropoff places).
Region	Region	Region in which returned Destination belongs.
Country	Country	Country in which returned Region belongs.
PeriodList	Period[]	Period list for prices of services in units contained by the returned object.
Currency	Currency	Currency for prices returned by the function.
Language	Language	Language for data returned by the function.
Status	Status	Status of the response. Determines if the result data is valid or error has occurred on server.
StartDate	DateTime	Start date used for price calculation
EndDate	DateTime	End date used for price calculation
NumberOfPersons	Int	Number of persons used in price calculation
PriceType	PriceType	Price type for prices returned by function
NumberOfDays	Int	Number of days between start and end date
PriceFormat	PriceFormat	This object contains information on how to properly display price on current language.

GetReservationRQ

This object contains the parameters for getting detailed reservation data

MEMBERS

Name	Type	Description
LanguageID	string	Id of a language to return in response
ReservationID	long	The reservation we are trying to get info on

GetReservationRS

This object contains reservation details

MEMBERS

Name	Type	Description
Reservation	Reservation	Reservation with details
Status	Status	Status of the get action. OK if the reservation is found in the Database

GetAllReservationsRQ

This object contains parameters and filters for getting list of detailed reservation data

MEMBERS

Name	Type	Description
LanguageID	string	Id of a language to return in response
ReservationCreationDateFrom	DateTime (nullable)	Filters reservations newer than that date
ReservationCreationDateTo	DateTime (nullable)	Filters reservations older than that date
PageSize	Int (nullable)	Page size used for pagination
CurrentPage	Int (nullable)	Current page number
FetchDocuments	bool(nullable)	If this parameter is true this method will fetch documents, if false it won't.
SortParameters	SortingParameters	Sort parameters class used for sorting the response data.

GetAllReservationsRS

This object contains list of reservations with details

MEMBERS

Name	Type	Description
Reservations	Reservation[]	List of all reservations or filtered reservations if filter is applied
Status	Status	Status of the get action. OK if the reservation is found in the Database

InsertReservationRQ

This object represents parameters needed for calculation of the reservation without its insertion to database.

Used in method: [InsertReservation](#).

MEMBERS

Name	Type	Description
ExecuteInsert	Boolean	True: reservation from request will be inserted to system and ReservationID will be returned in response; False: Only reservation object will be returned in response, Reservation ID will be 0, and reservation won't be inserted to system.
CurrencyID	int	Reservation's currency ID.
PaymentMethodID	int	Payment method's ID used on current reservation.
LanguageID	string	Language ID. (2-letter culture ID)
ReservationCustomFields	CustomFieldRQ[]	List of reservation's custom fields
ReservationItems	ReservationItemRQ[]	List of reservation items
AdHocReservationItems	ReservationItemAdHocRQ[]	List of ad-hoc reservation items.

InsertReservationRS

This object represents response for insert reservation.

Used in method: [InsertReservation](#).

MEMBERS

Name	Type	Description
ReservationID	int	Reservation ID is greater than 0 only when ExecuteInsert is true. It's an ID of the reservation from system
Currency	Currency	Reservation's currency
SellingPrice	Decimal	Reservation's total selling price
ReservationItems	ReservationItemRS[]	List of reservation items.
EndReservationURL	string	URL that is generated after reservation is successfully created. It leads to the Thank you page from Lemax system (like on the Online booking process).
Status	Status	Status of insert reservation action

CancelReservationItemRQ

This object represents parameters needed for cancelling the reservation item.

Used in method: [CancelReservationItem](#).

MEMBERS

Name	Type	Description
ReservationItemID	int	ID of the reservation item that will be cancelled.
CancellationReason	string	Reason for cancelling the reservation item.
CancelReservationIfAllItemsCancelled	boolean	A flag that indicates if the reservation should be cancelled if after cancelling the current item all items are cancelled and the price of the reservation is zero.

CancelReservationItemRS

This object represents response for reservation item cancellation.

Used in method: [CancelReservationItem](#).

MEMBERS

Name	Type	Description
Status	Status	Status of reservation item cancellation action

ReservationItemRQ

This object represents reservation item in Insert reservation request.

MEMBERS

Name	Type	Description
ReservationItemOrder	int	Order of reservation item for which calculation is made. Same as order of reservation item parametar.
UnitID	int	ID of an unit that is reserved by item
UnitCode	String	Unit code. Additional data about the unit. Should be set if present in AccommodationUnit object.
StartDate	DateTime	Reservation item's start date
EndDate	DateTime	Reservation item's end date
Passengers	PassengerRQ []	List of passengers on reservation item
SelectedServices	ServiceRS []	List of selected services for reservation item
ClientComment	String	Reservation item client comment.
SelectedFlightOptions	String[]	Used only when booking flights. One flight has multiple directions.

		One direction has multiple options. One option has multiple segments. Each element in this list must correspond to one direction, meaning one option from each direction must be selected to book that flight.
--	--	----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

ReservationItemAdHocRQ

This object represents ad-hoc reservation item in Insert reservation request.

MEMBERS

Name	Type	Description
ReservationItemOrder	int	Order of reservation item for which calculation is made. Same as order of reservation item parametar.
AdHocItemName	String	Ad-hoc name of the reservation item.
StartDate	DateTime	Reservation item's start date
EndDate	DateTime	Reservation item's end date
Passengers	PassengerAdHocRQ []	List of passengers on reservation item
SelectedServices	ServiceAdHocRQ []	List of selected services for reservation item

ReservationItemRS

This object represents reservation item in Insert reservation response.

MEMBERS

Name	Type	Description
ReservationItemOrder	int	Order of reservation item for which calculation is made. Same as order of reservation item parametar.
SellingPrice	decimal	Selling price.
ReservationItemDetailsList	ReservationItemDetailRS []	List of reservation item details.
PriceCalculationStatus	PriceCalculationStatus	Status of reservation item.
CancellationData	CancellationData	Reservation item's cancellation information

ReservationItemDetailRS

This object represents reservation item detail in insert reservation in response.

MEMBERS

Name	Type	Description
Service	ServiceRS	Service of reservation item detail.
SellingPrice	decimal	Selling price.

ServiceRQ

This object represents service in request.

MEMBERS

Name	Type	Description
ServiceID	int	Service ID from database.
Amount	int	Service quantity.

ServiceAdHocRQ

This object represents ad-hoc service in request.

MEMBERS

Name	Type	Description
ServiceID	int	Service ID from database.
Amount	int	Service quantity.
AdHocPrice	decimal	When adding ad hoc item to reservation, ad hoc price will be used instead of retrieving price from pricelist.

ServiceRS

This object represents service in response.

MEMBERS

Name	Type	Description
ServiceID	int	Service ID from database.
Name	string	Service's name.

PassengerRQ

This object represents passenger on reservation item in insert reservation request.

MEMBERS

Name	Type	Description
Name	string	Passenger's first name.
Surname	string	Passenger's last name.
DateOfBirth	DateTime	Date of passenger's birth
SelectedServices	ServiceRS []	List of selected services for passenger on reservation item.

PassengerAdHocRQ

This object represents passenger on ad-hoc reservation item in insert reservation request.

MEMBERS

Name	Type	Description
Name	string	Passenger's first name.
Surname	string	Passenger's last name.
DateOfBirth	DateTime	Date of passenger's birth

CustomFieldRQ

Custom fields used in insert reservation in request.

MEMBERS

Name	Type	Description
ID	int	Identifier of custom field.
Value	string	Value for custom field ID.

PriceCalculationStatus

This object represents status of reservation's price calculation in insert reservation response.

MEMBERS

Name	Type	Description
Code	StatusCode	Price calculation status code.
Description	string	Description of price calculation status.

Country

Country object contains information about single country.

MEMBERS

Name	Type	Description	Importable
CountryID	Int	Country ID.	+
CountryName	String	Country name.	
CountryNameTranslationList	Translation[]	List of country name translations	+
Description	String	Destinations description	
DescriptionTranslationList	Translation[]	List of description translations	+
CountryCode	String	Country code	

Region

Region object contains information about region. Region belongs to one country.

MEMBERS

Name	Type	Description	Importable
RegionID	Int	Region ID.	+
RegionName	String	Region name.	
CountryID	Int	Id of Country current region belongs to.	+
RegionNameTranslationList	Translation []	List of region name translations	+
Description	String	Destinations description	
DescriptionTranslationList	Translation []	List of description translations	+
RegionCode	String	Region code	

Destination

Destination object contains information about single destination. Destination belongs to one region.

MEMBERS

Name	Type	Description	Importable
DestinationID	Int	Destination ID.	+
DestinationName	String	Destination name.	
DestinationNameTranslationList	Translation []	List of destination name translations	+
RegionID	Int	Id of Region current destination belongs to.	+
Description	String	Destinations description	
DescriptionTranslationList	Translation []	List of description translations	+
PhotoList	Photo []	List of destination's photos	

CustomField

The object contains info on custom fields (used on Reservation).

MEMBERS

Name	Type	Description
CustomFieldID	int	Custom field ID.
CustomFieldName	String	The name this custom field appears under.
CustomFieldValue	CustomFieldValue	Custom field value object.
CustomFieldType	int	Type of the custom field. Allowed values are: 0 – DropDownList 1 – TextBox 2 – TextArea 4 – CheckBox 5 – RadEditor

		6 – DatePicker
listCustomFieldValue	List< CustomFieldValue >	List of available Custom field values
AvailableValuesList	List<List< CustomFieldValue >>	List of Custom field values that are available for drop down list type of custom field.

Passenger

The object contains basic information about a passenger (used on Reservation)

MEMBERS

Name	Type	Description
PassengerID	long	Passenger ID.
Name	String	Name of the passenger.
Surname	String	Surname of the passenger
MiddleName	String	Middle name of the passenger (if available)
DateOfBirth	DateTime	Passenger's date of birth
Gender	String	M/F
FullName	string	Full passenger's name in format "Firstname Middlename Surname"
Telephone	string	Passenger's telephone number
MobilePhone	string	Passenger's mobile phone number.
Birthplace	string	Birth place of passenger.
PassportNumber	string	Passenger's passport number.
Address	string	Address where passenger lives.
PersonalID	string	Personal passenger's ID that is entered manually by user.

DocumentItem

Every priced item on a document is represented by this object

MEMBERS

Name	Type	Description
Price	Double	Item price
TaxPercentage	Double	The percentage that is added to the basic price as tax
PriceBeforeTax	Double	The basic price (before taxes are added)
Amount	Double	How many of these items are taken into account

Document

Object contains all information about a single Lemax document

MEMBERS

Name	Type	Description
LemaxDocumentID	String	ID of this document in the Lemax database
DocumentID	String	ID of this document in an external system

Number	String	Document identification number
Date	DateTime	The date when the document was issued
DueDate	DateTime	The date when the payment is due (used for invoices)
DocumentType	String	Type of the document (used as a description)
DocumentTypeID	Int	Type of the document. Allowed values are: 0 - Unknown 1 - Offer 2 - Pro forma invoice 3 - Invoice 4 - Voucher 5 - Contract 6 - Specification 7 - Advanced payment invoice 8 - Itinerary 9 - Waiting list 10 - Payment invoice (equivalent of 7) 11 - Payment confirmation 12 - Refund confirmation
DocumentItemList	DocumentItem []	List of items (with prices) on this document
Currency	Currency	The currency used on this document
CancelledDocument	Document	The document that is cancelled by this document
LinkedDocuments	Document []	A list of documents that are linked to this one
OfficeID	Int	The id of the office that issued the document
Price	Double	Price displayed on the document.

ReservationItemTransferData

The object contains all information about a transfer included in a reservation item

MEMBERS

Name	Type	Description
TransferID	long	The ID of the transfer data object in the database
TransferStart	DateTime	Time when the transfer starts
TransferEnd	DateTime	Time when the transfer ends
TransferNumber	String	An Identification string that is given manually
Location	String	The location of the pickup / dropoff for the passengers
Description	String	Any additional info goes here

ReservationItemDetails

The object contains all information about a single reservation item detail in the Lemax system

MEMBERS

Name	Type	Description
ReservationItemDetailsID	long	The ID of the reservation item details object
Service	Service	The service whose prices are shown in this detail

Price	Double	The selling price of this particular service on this reservation
PriceDefault	Double	The selling price of this particular service on this reservation on default market
Quantity	Int	Number of services for current detail
Description	String	Description of the reservation item detail

ReservationItem

The object contains all information about a single reservation in the Lemax system

MEMBERS

Name	Type	Description
ReservationItemID	long	The ID of the reservation item
StartDate	DateTime	Starting date of the travel reserved
EndDate	DateTime	Ending date of the travel reserved
PassengerList	Passenger []	List of passengers on this reservation item
ReservationItemDetailsList	ReservationItemDetails []	A list of details (services with prices) for this reservation item
AccommodationObject	AccommodationObject	AccommodationObject reserved with this item *
AccommodationUnit	AccommodationUnit	AccommodationUnit reserved with this item **
PackageTour	PackageTour	PackageTour reserved with this item *
PackageUnit	PackageUnit	PackageUnit reserved with this item **
Transportation	Transportation	Transportation reserved with this item *
TransportationUnit	TransportationUnit	TransportationUnit reserved with this item **
TransferDeparture	ReservationItemTransferData	Data on the departing transfer
TransferReturn	ReservationItemTransferData	Data on the returning transfer
DestinationID	long	Reservation destination ID
Name	string	If unit is adhoc, Name represents its name
FirstMeal	Meal	First meal on current reservation item.
LastMeal	Meal	Last meal on current reservation item.
SellingPrice	decimal	Total selling price
SellingPriceDefault	decimal	Total default selling price
CreatedByUserID	int	ID of the user who created this reservation item
ClientComment	String	Reservation item client comment.
SelectedFlightOptions	String []	Used only when booking flights. One flight has multiple directions. One direction has multiple options. One option has multiple segments. Each element in this list must correspond to one direction, meaning one option from each direction must be selected to book that flight.

* Only one of these objects may be present

** Only one of these units may be present

Reservation

The object contains all information about a single reservation in the Lemax system

MEMBERS

Name	Type	Description
ReservationID	long	The ID of the reservation
Currency	Currency	The currency that the prices are calculated in
CreationDate	DateTime	The date the reservation was made
OptionExpiryDate	DateTime?	The date this reservation's option status expires (payments must be paid before that date for the reservation to remain valid)
Status	Byte	Says which status this reservation is in (option, booking, cancelled...)
ReservationItemList	ReservationItem []	A list of items on this reservation
DocumentList	Document []	A list of documents made for this reservation
CustomFieldList	CustomField []	A list of custom fields (containing additional, user entered data about this reservation)
ReferenceNumber	string	Reservation's reference number
PaymentMethod	PaymentMethod	Payment method on reservation
SellingPrice	decimal	Total selling price
SellingPriceDefault	decimal	Default selling price
Commission	decimal	Partner commission
RemainingAmount	decimal	Remaining amount to be payed
PaidAmount	decimal	Paid amount
CreatedByUserID	int	ID of a user who created reservation
ListReservationNotes	ReservationNote []	A list of notes on this reservation
BookingConfirmationDate	DateTime?	Date when reservation is confirmed. If reservation isn't confirmed at any point, this field will be null

PaymentMethod

This class contains ID and name of payment method

MEMBERS

Name	Type	Description
PaymentMethodID	int	ID of payment method
PaymentMethodName	string	Name of payment method

SortParameter

This class is used in [GetSearchResults](#) function. It is used to define sorting parameters.

MEMBERS

Name	Type	Description
------	------	-------------

SortBy	SortBy	Defines by which parameter search should be sorted.
SortOrder	SortOrder	Defines in which direction search should be sorted

AttributeFilter

This class is used in [GetSearchResults](#) function. It is used to define by which [attribute](#) search results should be filtered.

MEMBERS

Name	Type	Description
AttributeID	Int	Attribute ID by which search results is filtered. Some attribute IDs are listed in the Table 1 - Attributes predefined values , but other values can be used too (you need to know attribute id from Lemax).
AttributeValue	String	Attribute value by which search results are filtered.
AttributeValue2	String	Attribute value by which search results are filtered. It's used only if ComparisonType is "Between"
ComparisonType	ComparisonType	Defines how is attribute value compared to object's or unit's attribute (equals, greater or equals, etc.)

EXAMPLE

In following example search results are filtered by object's stars. Object has 3 stars. AttributeID for stars is 119, AttributeValue is 3, ComparisonType is Equals. Because search is filtered by object's attributes, AttributeFilter is in objectAttributeFilterList.

```
...
AttributeFilter attributeFilter = new AttributeFilter();
attributeFilter.AttributeID = 119;
attributeFilter.AttributeValue = "3";
attributeFilter.ComparisonType = ComparisonType.Equals;

AttributeFilter[] objectAttributeFilterList = { attributeFilter };
...
```

OutParameter

This class is used in [GetSearchResults](#) function. It is used to define how detailed data should be returned and number of returned items. Limiting returned data will increase performance of [GetSearchResults](#) function.

MEMBERS

Name	Type	Description
------	------	-------------

ResponseDetail	ResponseDetail	Defines which response detail is current OutParameter referring to (e.g. object's photos).
NumberOfResults	Int	Number of results of current ResponseDetail (e.g. two object's photos).

EXAMPLE

In following example is defined that GetSearchResults function returns only one object's photo. ResponseDetail is set to ObjectPhotos and NumberOfResults is set to 1.

```
...
OutParameter outParameter = new OutParameter();
outParameter.ResponseDetail = ResponseDetail.ObjectPhotos;
outParameter.NumberOfResults = 1;

OutParameter[] outParameterList = { outParameter };
...
```

Attribute

Attribute object contains information about single [object's](#) or [unit's](#) attribute.

MEMBERS

Name	Type	Description
AttributeID	Int	Attribute ID.
AttributeName	String	Attribute name.
AttributeValue	String	Attribute value.
AttributeType	AttributeType	Defines attribute type. If attribute is of type Textual, then attribute should be written as: AttributeName + AttributeValue (e.g. "Distance from the sea: 100m", where "Distance from the sea" is AttributeName, and "100m" is AttributeValue). If attribute is of type Logical, then attribute should be written as: AttributeName (e.g. "Air conditioner", where "Air conditioner" is AttributeName and "1" is AttributeValue).
Visible	Bool	Defines attributes visibility. If Visibility is set to false, then attribute should not be shown to user. Attributes that are not visible are most often used to pass data with special meaning between Lemax and web.

EXAMPLE

On the **Picture 1 - Attribute example** Attributes are marked with red border. AttributeNames are marked with green color and AttributeValues are marked with blue color. Only two attributes on picture are marked: first one has Textual AttributeType and second one has Logical AttributeType.



Picture 1 - Attribute example

PREDEFINED ID VALUES

In **Table 1 - Attributes predefined values** are listed predefined ID values of most common used attributes. These IDs can be used in [GetSearchResults](#) function to filter out search results.

Attribute ID	Attribute name
119	Stars
121	Name
120	Number of persons

Table 1 - Attributes predefined values

AttributeGroup

AttributeGroup is used to group [object's](#) or [unit's attributes](#).

MEMBERS

Name	Type	Description
GroupID	Int	Attribute group ID.
GroupName	String	Attribute group name.
AttributeList	Attribute []	List of attributes that belongs to current group.

EXAMPLE

On the **Picture 2 - AttributeGroup example** AttributeGroups are marked with red border and GroupNames are marked with green border.



Picture 2 - AttributeGroup example

Photo

The Photo object contains information about an [object's](#) or [unit's](#) photo.

MEMBERS

Name	Type	Description	Importable
PhotoID	Int	Photo ID.	
PhotoUrl	String	Url to large photo.	+
ThumbnailUrl	String	Url to thumbnail photo.	+
Position	Int	Position within the owner's photo list (object, unit or destination)	+
PhotoBytesString	String	The photo is represented by an array of bytes in the system. This array is then encoded as a Base64 string.	+
PhotoDownloadUrl	String	Url that contains the picture we wish to import	+
ObjectID	Int	ID of the owner accommodation object	+
UnitID	Int	ID of the owner unit (all types of 2 nd level units)	+
DestinationID	Int	ID of the owner destination	+
PackageTourID	Int	ID of the owner package tour	+
TransportationID	Int	ID of the owner transportation	+

PHOTO BYTES STRING (C# EXAMPLE)

```
// transforming an image to a base64 string
public string ImageToBase64(System.Drawing.Image image,
System.Drawing.Imaging.ImageFormat format)
{
    using (MemoryStream ms = new MemoryStream())
    {
        // Convert Image to byte[]
    }
}
```

```

    image.Save(ms, format);
    byte[] imageBytes = ms.ToArray();

    // Convert byte[] to Base64 String
    string base64String = Convert.ToBase64String(imageBytes);
    return base64String;
}

// using the code above to get the photo bytes string
System.Drawing.Imaging.ImageFormat format = System.Drawing.Imaging.ImageFormat.Jpeg;
System.Drawing.Image img = System.Drawing.Image.FromFile("C:/testing.png");

string photoString = ImageToBase64(img, format);

```

AccommodationObject

AccommodationObject contains information about a single object (e.g. Hotel Dubrovnik, Apartment object Dubrovnik,, etc.).

MEMBERS

Name	Type	Description	Importable
ObjectID	Int	Object ID.	+
DestinationID	Int	ID of the Destination to which object belongs.	+
ObjectCode	String		
ObjectType	ObjectType	Object type.	+
AttributeGroupList	AttributeGroup[]	List of attribute groups contained by current object.	+
PhotoList	Photo[]	List of object's photos.	
Description	String	Object's description. Important: Description is in HTML format, but in response it can be HTML encoded (depends on Lemax settings) and if it is it must be HTML decoded before displaying to the user.	
DescriptionTranslationList	Translation[]	List of description translations to different languages	+
UnitList	AccommodationUnit[]	List of units contained by current object.	
ShortDescription	String	Shorter version of the description	
ShortDescriptionTranslationList	Translation[]	List of short description translations	+

Title	String	Title of the accommodation object	
TitleTranslationList	Translation []	List of title translations	+
SEODescription	String	Search Engine Optimization description of the accommodation object	
SEODescriptionTranslationList	Translation []	List of SEODescription translations	+
SEOKeywords	String	Keywords	
SEOKeywordsTranslationList	Translation []	List of keywords in multiple languages	+
NoteList	Note []	List of object's notes.	+
CancellationPolicy	String	Objects cancellation policy. Important: Description is in HTML format, but in response it can be HTML encoded (depends on Lemax settings) and if it is it must be HTML decoded before displaying to the user.	
CancellationPolicyTranslationList	Translation []	List of cancellation policy translations	+
MapLocationUrl	String	Url that shows object's location on map if object's location is entered through Lemax.	
AvailabilityCalendarAddress	String	Url where availability of current unit can be checked	
AccomodationObjectPublicCode	String	Accomodation's object public code	
CategoryList	Category []	List of categories this accommodation object belongs to	+
Name	String	Accommodation object's name	Imports as a mandatory attribute
ObjectURL	String	Accommodation object's SEO URL	+
ObjectUrlTranslationList	Translation []	List of SEO URLs for different languages	+
CancellationData	ObjectCancellationData	Cancellation policy and rules data related to dates of travel and market from request.	

Language

Language object contains information about single language.

MEMBERS

Name	Type	Description
LanguageID	String	LanguageID is ISO 639-1 - two letter representation of language. List of language codes can be found here .
LanguageName	String	Language name.

Currency

Currency object contains information about single currency.

MEMBERS

Name	Type	Description
CurrencyID	Int	Currency ID. CurrencyID is ISO 4217 three digit number representation of currency. List of currency codes can be found here .
CurrencyName	String	Currency name.
CurrencyShortName	String	Currency short name (e.g. EUR for currency Euro, or USD for US Dollar).

ObjectType

ObjectType contains information about single object type. Every object is of one object type.

MEMBERS

Name	Type	Description
ObjectTypeID	Int	Object type ID.
ObjectTypeName	String	Object type name.
UnitTypeList	UnitType []	List of subordinate unit types

AccommodationUnit

AccommodationUnit contains information about single unit. An accommodation object can have multiple accommodation units (the object representing Hotel Excelsior, while the accommodation units represent a Double room, a single room, a three bed room etc. within Hotel Excelsior).

MEMBERS

Name	Type	Description	Importable
UnitID	Int	Unit ID.	+
UnitCode	String	Unit code. Additional data about the unit. Should be	

		sent in pair with UnitID to all methods that accept it.	
Description	String	Units description. Important: Description is in HTML format, but in response it can be HTML encoded (depends on Lemax settings) and if it is it must be HTML decoded before displaying to the user.	
DescriptionTranslationList	Translation[]	List of description translations to different languages	+
AvailabilityStatus	UnitAvailabilityStatus	Unit's availability status. If AvailabilityStatus is OnRequest, unit cannot be directly booked, and inquiry must be made. If AvailabilityStatus is Available, unit can be directly booked.	
Type	UnitType	Unit type.	+
AttributeGroupList	AttributeGroup[]	List of attribute groups contained by current unit.	+
PhotoList	Photo[]	List of object's photos.	
ServiceList	Service[]	List of unit's services.	
SpecialOfferList	SpecialOffer[]	List of unit's special offers.	
AvailabilityCalendarAddress	String	Url where availability of current unit can be checked	
BookingAddress	String	Url where booking of current unit can be made.	
NoteList	Note[]	List of unit's notes.	+
ShortDescription	String	Units short description. Important: Short description is in HTML format, but in response it can be HTML encoded (depends on Lemax settings) and if it is it must be HTML decoded before displaying to the user.	
ShortDescriptionTranslationList	Translation[]	List of short description translations	+
Name	String	Units name.	Imports as an attribute
DepartureList	Departure[]	List of available departures for current unit. Available if ObjectType is Cruise	
AccommodationUnitPublicCode	String	Accommodation unit public code	

CalculatedPriceInfo		Contains information about calculated accommodation unit prices	
UnitMinimumPriceInfo	UnitMinimumPriceInfo	Contains information on the minimum price calculated for the accommodation unit	
OrdinalNumber	Int	Ordinal number of the unit. It's used in function GetDetailedDescription .	
OccupantCombinationOK	Bool	If it is set to true, the guest list corresponds to at least one of the assigned occupant combinations.	
ObjectID	int	ID of the AccommodationObject that contains this unit	+
ActivityStatus	int	Unit ActivityStatus (0 = inactive, 1= published on web, 2 = active)	+
CategoryList	Category []	List of categories this accommodation unit belongs to	+
RoomOccupantCombination	RoomOccupantCombination	RoomOccupantCombination holds the case-s and rule-s for an occupant combination on a specific accommodation unit.	
ServiceGroupCalculatedPriceInfo List	ServiceGroupCalculatedPriceInfo []	List of ServiceGroupCalculatedPriceInfo objects that holds information about ServiceGroup and information about the calculated price.	
UnavailableDates	DateTime []	List of unavailable dates for the unit for the next two years.	
CancellationData	CancellationData	AccommodationUnit cancellation data.	

UnitType

UnitType contains information about single unit type. Every unit is of one unit type.

MEMBERS

Name	Type	Description
UnitTypeID	Int	Unit type ID.
UnitTypeName	String	Unit type name.

Service

Service contains information about single service.

MEMBERS

Name	Type	Description	Importable
ServiceID	Int	Service ID.	+
ServiceName	String	Service name.	
NameTranslationList	Translation []	List of service name translations	+
ServiceType	ServiceType	Type of service. Service can be basic, supplement, discount, etc.	+
Price	Double	For given parameters startDate and endDate in functions GetSearchResults and GetDetailedDescription approximately calculated price.	
PriceType	PriceType	Price type for returned Price.	
GroupID	Int	GroupID is used to group periods that belong to the same group.	
PriceFormatted	String	Properly formatted price in selected language	
ServicePriceType	ServicePriceType	Enumerator that specifies if price value is absolute or percentage amount	+
VATRate	Double	VAT rate paid for the service	+
RuleList	DiscountRule []	List of age rules that refer to discount services	+

EXAMPLE

On the **Picture 3 - Service example** service is marked with red border. ServiceName is marked with green border. BillingType is marked with yellow border, PriceRowList is marked with blue border.

Overnight accommodation - (EUR) - per accommodation/day				
Person	Min. days	1.1 - 5.7 29.8 - 31.12	5.7 - 29.8	
1	1	184.00	165.00	
2	1	184.00	165.00	
3	1	239.00	220.00	

Picture 3 - Service example

SpecialOffer

SpecialOffer inherits [Service](#). It contains information about a single special offer.

MEMBERS

SpecialOffer contains all members from Service. Additional members are listed in the table below.

Name	Type	Description	Importable
AccommodationObject	AccommodationObject	Object whose units are on special offer.	
DateFrom	DateTime	Date from the special offer is valid.	+
DateTo	DateTime	Date until the special offer is valid.	+
BasicPrice	Double	Unit's price without special offer	
BasicPriceFormatted	String	Properly formatted price in selected language	
StartDatePrice	DateTime	Start date for which BasicPrice and Price is calculated.	
EndDatePrice	DateTime	End date for which BasicPrice and Price is calculated.	
NumberOfPersons	Int	Number of persons which the price is calculated for	
NumberOfDays	Int	Number of days between StartDatePrice and EndDatePrice	
BookingAddress	String	Booking address string (contains all the necessary info to open the online booking page)	
DaysBeforeTravel	Int	Booking has to take place this many days before the actual travel starts in order for this special offer to apply	+
CaclulationType	Byte	1=PerStay, 2= PerArrival	+
TravelDateFrom	DateTime	Starting date of the travel	+
TravelDateTo	DateTime	Ending date of the travel	+
Comment	String	Additional info on the special offer	+
StartDaysOfWeek *	String	Days which can be the start of a special offer	+
IsRecurringPeriod *	Bool	Determines wether the special offer has a recurring period	+
RecurringPeriod *	Int	Duration of the recurring period	+

MinimumStay *	Int	Minimum duration of the travel in order for the special offer to apply	+
SellingPrice *	Double	Special offer selling price	+
NumberOfFreeNights *	Byte	Number of nights considered as free of charge (out of the whole duration of the trip)	+
UnitIDList *	Int[]	List of Unit IDs that have this special offer	+
PeriodGroup **	PeriodGroup	The special offer's period group	+

- * applies to percentage special offers (ServicePriceType = Percentage, FreeNights) only.
- ** applies to fixed special offers (ServicePriceType = Fixed, FixedDiscount) only.

Period

Period contains information about a single period. Each period can contain multiple start and end days. In that period service had the same price.

MEMBERS

Name	Type	Description
PeriodID	Int	Period ID.
GroupID	Int	GroupID is used to group periods that belong to the same PeriodGroup .
DateList	Date []	Period's date list.
PositionPeriod	Int	Period's position within the pricelist

EXAMPLE

In the **Picture 4 - Period example** periods are marked with red border. First period has two dates in DateList.

Overnight accommodation - (EUR) - per accommodation/day

Person	Min. days	1.1 - 5.7 29.8 - 31.12	5.7 - 29.8
1	1	184.00	165.00
2	1	184.00	165.00
3	1	239.00	220.00

Picture 4 - Period example

Date

Date contains information about a single date.

MEMBERS

Name	Type	Description
StartDate	DateTime	Start date. DateTime format
EndDate	DateTime	End date. DateTime format
NumberOfDays	int	Gets the number of days between StartDate and EndDate

EXAMPLE

In the **Picture 5 - Date example** dates are marked with red border StartDate is marked with green border and EndDate is marked with blue border.

Overnight accommodation - (EUR) - per accommodation/day

Person	Min. days	29.6 - 31.12	5.7 - 29.8
1	1	184.00	165.00
2	1	184.00	165.00
3	1	239.00	220.00

Picture 5 - Date example

Note

Contains information about unit's or object's notes.

MEMBERS

Name	Type	Description
NoteID	Int	Note ID.
NoteText	String	Note text.
NoteTitle	string	Note title.
ValidDateFrom	DateTime (nullable)	Date from which the note is valid.
ValidDateTo	DateTime (nullable)	Date until the note is valid.
TravelDateFrom	DateTime (nullable)	Travel date from which the note is valid.
TravelDateTo	DateTime (nullable)	Ending travel date until the note is valid.

Status

Contains information about reponse's status.

MEMBERS

Name	Type	Description
Code	StatusCode	Status code.
Description	String	Status description.
ErrorCode	Int	Unique error code (provides further information about the error that occurred)

Departure

Contains information about departure

MEMBERS

Name	Type	Description
DateFrom	DateTime	Departure's start date
DateTo	DateTime	Departure's end date
Price	Double	Departure's price for given period
PriceFormatted	String	Properly formatted price in selected language
PriceType	PriceType	Price type for departure's price
BookingAddress	String	Url where booking of current departure can be made

CalculatedPriceInfo

Contains information about calculated price

MEMBERS

Name	Type	Description
BasicCalculatedPrice	Double	Calculated price for given period with no special offers included
BasicCalculatedPriceFormatted	String	Properly formatted price in selected language
CalculatedPrice	Double	Calculated price for given period with included special offers (special offers that are not active in given period are not used in price calculation)
CalculatedPriceFormatted	String	Properly formatted price in selected language
ServiceList	Service[]	List of services and their prices for given period

Categories

Contains list of Categories. It's used in the SearchFields object which is returned from [GetSearchFields](#) function.

MEMBERS

Name	Type	Description
CategoryList	Category []	List of categories

Category

Contains information about a single category.

MEMBERS

Name	Type	Description
CategoryID	Int	Category ID
CategoryName	String	Name of the category

PriceFormat

Contains necessary information to properly display price.

MEMBERS

Name	Type	Description
DecimalSeparator	String	Specifies the decimal point character
GroupSeparator	String	Specifies the thousands separator character

PackagePeriod

Contains information about package tours departures. Each package period describes one departure.

MEMBERS

Name	Type	Description
AvailabilityStatus	UnitAvailabilityStatus	Package tours availability status. If AvailabilityStatus is OnRequest, package tour cannot be directly booked in that period, and inquiry must be made. If AvailabilityStatus is Available, package tour can be directly booked in that period
BookingAddress	String	Url where booking of current package tour can be made.
CalculatedPriceInfo	CalculatedPriceInfo	Contains information about calculated prices for current package tour
StartDate	DateTime	Package tours departure date
EndDate	DateTime	Package tours return date
NumberOfDays	Int	Number of days between start and end date
PriceNumberOfPersons	Int	Number of persons for which price is calculated
Visible	Bool	Specifies if package period is shown to user in package period list

ServiceGroupCalculatedPriceInfoList	ServiceGroupCalculatedPriceInfo[]	List of ServiceGroupCalculatedPriceInfo objects that holds information about ServiceGroup and information about the calculated price.
TotalNumberOfSeats	int	Total number of available seats. <i>Only available if the PackageTourDetailedDescriptionOutParameter is passed to the GetPackageDetailedDescriptionParameters request object.</i>

PackageTour

PackageTour inherits [AccommodationObject](#) and contains information about single package tour.

MEMBERS

PackageTour contains all members from [AccommodationObject](#). Additional members are listed in table below.

Name	Type	Description
PackageUnitList	PackageUnit	List of package units in package tour
DaysList	Int[]	Distinct list of days from package period list
DistinctPackagePeriodList	PackagePeriod	Distinct list of package periods from package unit list
PackageTourName	string	Name of the package tour.

PackageTourDetails

Object that is returned in [GetPackageDetailedDescription](#) function. It contains information about returned object, period list, language, currency and object's destination, region and country.

MEMBERS

Name	Type	Description
PackageTour	PackageTour	Object whose detailed information is returned by the function.
AccommodationObjectList	AccommodationObject[]	List of accommodation objects that appear in the package tour
DestinationList	Destination[]	Destination of every object in AccommodationObjectList is contained in this list
RegionList	Region[]	Region of every object in AccommodationObjectList is contained in this list

CountryList	Country[]	Country of every object in AccommodationObjectList is contained in this list
Currency	Currency	Currency for prices returned by the function.
Language	Language	Language for data returned by the function.
Status	Status	Status of the response. Determines if the result data is valid or error has occurred on server.
PriceType	PriceType	Price type for prices returned by function
PriceFormat	PriceFormat	This object contains information on how to properly display price on current language.
PeriodList	Period[]	Period list for prices of services in units contained by the returned object.

PackageUnit

PackageUnit inherits [AccommodationUnit](#) and contains information about single package tour.

MEMBERS

PackageUnit contains all members from [AccommodationUnit](#). Additional members are listed in table below.

Name	Type	Description
PackagePeriodList	PackagePeriod[]	List of package periods in package unit
PackageUnitName	String	Name of the package unit
AccommodationUnitID	Int	ID of the accommodation unit that package unit is related to

PriceOnDayOfWeek

Contains information about price on single day of week

MEMBERS

Name	Type	Description
DayOfWeek	DayOfWeek	Defines a single day of week
PriceOnDay	Float	Defines a price for the single day of week

UnitMinimumPriceInfo

This object contains detailed info on the minimum price calculated for a given AccommodationUnit.

MEMBERS

Name	Type	Description
Price	Double	Minimum price calculated from the price list
PriceFormatted	String	Formatted string output of the Price property
BillingTypeName	String	Description of the payment conditions

Translation

This object is usually a member of a list. It defines a piece of information as well as the language it is in.

MEMBERS

Name	Type	Description
LanguageID	String	Language ID in which data will be returned. Details
TranslationText	String	The text of translation on language

Transportation

Transportation contains information about single transfer, flight or rent a car.

MEMBERS

Name	Type	Description	Importable
ObjectID	Int	Object ID.	+
DestinationIDList	Int[]	List of the Destination IDs which are available for the rent a car (only used for the rent a car type)	+ (only for rent a car. It is ignored for Flight and Transfer types)
ObjectCode	String		
ObjectType	ObjectType	Object type.	+
AttributeGroupList	AttributeGroup []	List of attribute groups contained by current object.	+
PhotoList	Photo []	List of object's photos.	
Description	String	Object's description. Important: Description is in HTML format, but in response it can be HTML encoded (depends on Lemax settings) and if it is it must be HTML decoded before displaying to the user.	
DescriptionTranslationList	Translation []	List of description translations to different languages	+

TransportationUnitList	TransportationUnit[]	List of units contained by the current object.	
ShortDescription	String	Shorter version of the description	
ShortDescriptionTranslationList	Translation[]	List of short description translations	+
Title	String	Title of the accommodation object	
TitleTranslationList	Translation[]	List of title translations	+
SEODescription	String	Search Engine Optimization description of the accommodation object	
SEODescriptionTranslationList	Translation[]	List of SEODescription translations	+
SEOKeywords	String	Key words	
SEOKeywordsTranslationList	Translation[]	List of keywords in multiple languages	+
NoteList	Note[]	List of object's notes.	+
CancellationPolicy	String	Objects cancellation policy. Important: Description is in HTML format, but in response it can be HTML encoded (depends on Lemax settings) and if it is it must be HTML decoded before displaying to the user.	
CancellationPolicyTranslationList	Translation[]	List of cancellation policy translations	+
MapLocationUrl	String	Url that shows object's location on map if object's location is entered through Lemax.	
AvailabilityCalendarAddress	String	Url where availability of current unit can be checked	
AccomodationObjectPublicCode	String	Accomodation's object public code	
CategoryList	Category[]	List of categories this accommodation object belongs to	+
Name	String	Accommodation object's name	Imported as a mandatory attribute
ObjectURL	String	Accommodation object's SEO URL	
ObjectUrlTranslationList	Translation[]	List of SEO URLs for different languages	+

CancellationData	ObjectCancellationData	Cancellation policy and rules data related to dates of travel and market from request.	
-------------------------	----------------------------------------	----------------------------------------------------------------------------------------	--

TransportationUnit

TransportationUnit contains information about single unit of transportation (e.g. Audi A3 for rent a car).

This is a sub-object of the Transportation (e.g. when defining a rent a car service, the transportation object represents the rent a car service that can be used in New York, Washington and Boston, while the transportation units represent types of cars that can be rented (Ford Fiesta, VW Passat, etc.)

MEMBERS

Name	Type	Description	Importable
UnitID	Int	Unit ID.	+
DestinationList	Destination []	List of destinations (only used for flight and transfer). Must contain pickup and drop-off destination (2) or a single destination if both are the same.	+ (only for Flight and Transfer types. It is ignored for rent a car)
Description	String	Units description. Important: Description is in HTML format, but in response it can be HTML encoded (depends on Lemax settings) and if it is it must be HTML decoded before displaying to the user.	
DescriptionTranslationList	Translation []	List of description translations to different languages	+
AvailabilityStatus	UnitAvailabilityStatus	Unit's availability status. If AvailabilityStatus is OnRequest, unit cannot be directly booked, and inquiry must be made. If AvailabilityStatus is Available, unit can be directly booked.	
Type	UnitType	Unit type.	+
AttributeGroupList	AttributeGroup []	List of attribute groups contained by current unit.	+
PhotoList	Photo []	List of object's photos.	
ServiceList	Service []	List of unit's services.	
SpecialOfferList	SpecialOffer []	List of unit's special offers.	

AvailabilityCalendarAddress	String	Url where availability of current unit can be checked	
BookingAddress	String	Url where booking of current unit can be made.	
NoteList	Note []	List of unit's notes.	+
ShortDescription	String	Units short description. Important: Short description is in HTML format, but in response it can be HTML encoded (depends on Lemax settings) and if it is it must be HTML decoded before displaying to the user.	
ShortDescriptionTranslationList	Translation []	List of short description translations	+
Name	String	Units name.	Imports as an attribute
DepartureList	Departure []	List of available departures for current unit. Available if ObjectType is Cruise	
AccommodationUnitPublicCode	String	Accommodation unit public code	
CalculatedPriceInfo		Contains information about calculated accommodation unit prices	
UnitMinimumPriceInfo	UnitMinimumPriceInfo	Contains information on the minimum price calculated for the accommodation unit	
OrdinalNumber	Int	Ordinal number of the unit. It's used in function GetDetailedDescription .	
OccupantCombinationOK	Bool	If it is set to true, the guest list corresponds to at least one of the assigned occupant combinations.	
ObjectID	int	ID of the AccommodationObject that contains this unit	+
ActivityStatus	int	UnitActivityStatus (0 = inactive, 1= published on web, 2 = active)	+
CategoryList	Category []	List of categories this accommodation unit belongs to	+

PeriodGroup

This object contains info about a group of periods that a price list is defined for.

When defining a price list (e.g. for an accommodation object), different prices are defined for different periods in time (e.g. 23.1 – 23.2.). Those periods are defined in this object. More than one period can have the same price.

One can choose between more period groups when defining prices for an object since more than one period group can be assigned to one object.

NOTE: The first day of the period is included in the period and the last day is not. For example, if the period is from 7.10 till 15.11 the price on 7.10 will be taken from that period and on 15.11 you will get the price from another period.

MEMBERS

Name	Type	Description	Importable
GroupID	Int	The ID of the pricelist schedule in the database	+
ObjectID	Int	The ID of the owner object (or transportation) in the database (1 st level unit)	+
PeriodGroupName	String	The name that will be assigned to the group of periods	+
PeriodList	Period []	A list of periods in this group	+

DiscountRule

Rules used on discount services. They determine the ages of persons required on the reservation in order for the discount to apply. Every rule represents one person on the reservation. If the persons' ages match the rule set on a discount service, the discount applies to the reservation.

MEMBERS

Name	Type	Description	Importable
AgeFrom	Int	Minimum person's age	+
AgeTo	Int	Maximum person's age	+
IsDefault	Bool	True means that this is the person that gets the discount False means that this is one of the persons required to be on the reservation in order for the default person to get the discount	+

ObjectLocation

This object contains info about an accommodation object's location in the world (geographic longitude and latitude)

MEMBERS

Name	Type	Description
------	------	-------------

ObjectID	Int	The ID of the owner object (or transportation) in the database (1 st level unit)
Latitude	Double	Geographic latitude value (X coordinate)
Longitude	Double	Geographic longitude value (Y coordinate)

RoomOccupantCombination

Room occupant combination object that holds a list of RoomOccupantCases.

MEMBERS

Name	Type	Description
RoomOccupantCaseList	RoomOccupantCase []	A list of RoomOccupantCase objects.

RoomOccupantCase

Room occupant case object that holds a list of RoomOccupantRules for a specific case.

MEMBERS

Name	Type	Description
RoomOccupantRuleList	RoomOccupantRule []	A list of RoomOccupantRule objects for a specific case.

RoomOccupantRule

Room occupant rule object that holds properties for a specific rule. It has properties NumberOfPersons, AgeFrom and AgeTo.

MEMBERS

Name	Type	Description
NumberOfPersons	Int	This property holds the number of persons for this rule.
AgeFrom	Int	This property holds the ageFrom value for a rule.
AgeTo	int	This property holds the ageTo value for a rule.

ServiceGroup

Service group object that contains two properties, ID of the service group, and name of the service group.

MEMBERS

Name	Type	Description
ID	Int	This property holds the ID of the service group.
Name	String	This property holds the name of the service group.

ServiceGroupCalculatedPriceInfo

ServiceGroupCalculatedPriceInfo object holds as properties a ServiceGroup and the calculated price info for that service group.

MEMBERS

Name	Type	Description
ServiceGroup	ServiceGroup	This property holds the service group object.
CalculatedPriceInfo	CalculatedPriceInfo	This property holds the calculated price info object.

PassengerConfigurationFilter

Filter for passenger configurations (capacity cases) on search.

MEMBERS

Name	Type	Description
RoomsNumber	int	Number of rooms for searching in single configuration.
AdultsNumber	int	Number of adults in single configuration.
ChildrenAges	int[]	List of ages for every child in search.

Meal

Meal object holds basic information about the meal.

MEMBERS

Name	Type	Description
MealID	int	Identifier of the meal type.
Name	string	Specific meal name.

ReservationNote

ReservationNote object is representing a note on reservation which contains some text and is of certain type.

MEMBERS

Name	Type	Description
ReservationNoteID	int	Identifier of the reservation note.
Text	string	Text of the note.
Type	int	These numbers represent different types of reservation note: 1 – Internal 2 – Voucher 3 – Contract 4 – Rooming List 5 – Flight List

SortingParameters

This class is used to define sorting parameters.

MEMBERS

Name	Type	Description
SortBy	Int (nullable)	Integer value of the sort by enum value.
OrderBy	Int (nullable)	Integer value of the sort order enum value.

GetFlightDestinationsRQ

This object contains the parameters for getting flight destinations data.

MEMBERS

Name	Type	Description
NameOrlata	String	Name or lata code.

GetFlightDestinationsRS

This object contains the flight destinations in response.

MEMBERS

Name	Type	Description
Status	Status	Status of get flight destinations
FlightDestinationList	FlightDestinationsRSDestination []	List of flight destinations

FlightDestinationsRSDestination

This object contains the flight destinations in response.

MEMBERS

Name	Type	Description
------	------	-------------

FlightDestinationID	Int	ID of the flight destination
TypeID	Int	Type of the destination (FlightDestinationType).
DisplayText	String	Formatted text to display in as the result text.
AirportName	String	Name of the airport, if the destination is an airport.
AirportCode	String	Airport IATA code, if the destination is an airport.
CityName	String	Name of the city in which the airport is located, or city name.
CityCode	String	City IATA code of the city in which the airport is located, or city IATA code.
CountryName	String	Name of the country in which the city is located.
CountryCode	String	Country code.

GetFlightSearchResultsRQ

This object contains parameters for getting list of flights

MEMBERS

Name	Type	Description
FlightSearchSegments	FlightSearchRQSegment[]	List of search segments.
NumberOfAdults	Int	Number of adults.
ChildrenAges	Int[]	List of children ages.
TripTypeID	Int	Filter by trip type: 1 – OneWay, 2 – Return, 3 – MultiCity.
TicketClassID	Int	Filter flights by ticket class: 1 – NotRelevant, 2 – Economy, 3 – PremiumEconomy, 4 – Business, 5 – First, 6 – PremiumFirst.
DirectFlightsOnly	Boolean	Show only flights without stops.
DirectAccess	Boolean	Show only direct access flights.
CustomerID	Int	Consumer.
MarketID	Int	Market identifier.
CurrencyID	Int	Currency identifier. e.g. 978 – EUR
UserID	Int	User identifier.
LanguageID	String	Language identifier. e.g. "en" – English
AmountTypeID	Int	Price display: 1 – Total, 2 – Per person.

PriceFrom	Decimal	Filter flights by lowest price.
PriceTo	Decimal	Shows only flights cheaper than this price.
AirlineCodes	String[]	Shows only flights more expensive than this price.
OrderByID	Int	Order flight results by: 1 – Cheapest, 2 – Quickest, 3 – Best.
Page	int	Shows desired result page.

FlightSearchRQSegment

This object contains parameters for getting list of flight segments

MEMBERS

Name	Type	Description
OriginId	Int	Origin identifier.
DestinationId	Int	Destination identifier.
DepartureEarliestDate	DateTime	Earliest date and time of the flight.
DepartureLatestDate	DateTime	Latest date and time of the flight.
ReturningEarliestDate	DateTime	Earliest date and time of returning flight. Used only if TripType – Return.
ReturningLatestDate	DateTime	Latest date and time of returning flight. Used only if TripType – Return.

GetFlightSearchResultsRS

This object contains list of available flights

MEMBERS

Name	Type	Description
Status	Status	Status of the response. Determines if the result data is valid or error has occurred on server.
Flights	FlightSearchRSFlight[]	List of available flights.
PageSize	Int	Returned page size.
CurrentPage	Int	Number of current page.
TotalPages	Int	Total number of pages.
TotalResults	Int	Total number of results.
Language	Language	Language for data returned by the function.

FlightSearchRSFlight

This object contains the flight details

MEMBERS

Name	Type	Description
FlightDirections	FlightSearchRSDirection[]	List of the flight segments.
Price	Decimal	Flight price.
Currency	Currency	Currency for prices returned by the function.
BookingAddress	String	Address of the form used to book this flight.
FlightCode	String	Provider data for this flight, containing all direction options.

FlightSearchRSDirection

This object contains the flight direction options

MEMBERS

Name	Type	Description
FlightOptions	FlightSearchRSOption[]	List of the flight segment options.
DirectionTypeID	Int	Direction type: 1 – Departure, 2 – Return.

FlightSearchRSOption

This object contains the flight direction option segments

MEMBERS

Name	Type	Description
FlightOptions	FlightSearchRSSegment[]	List of flight direction option segments.
OptionCode	String	Option code.
TravelTimeMinutes	Int	Travel time in minutes including stop time.
OriginCode	String	Origin code.
OriginDescription	String	Origin description.
DestinationCode	String	Destination code.
DestinationDescription	String	Destination description.
Departure	DateTime	Departure date and time.
Arrival	DateTime	Arrival date and time.
AirlineLogoThumbnailUrls	String[]	List of airline logo thumbnail URLs.
AirlineLogoUrls	String[]	List of airline logo URLs.

FlightSearchRSSegment

This object contains the flight direction option segment details

MEMBERS

Name	Type	Description
Key	String	Flight direction option segment detail key.
FareBasisCode	String	Fare basis code.
Arrival	DateTime	Arrival date and time.
Departure	DateTime	Departure date and time.
OriginCode	String	Origin code.
OriginDescription	String	Origin description.
DestinationCode	String	Destination code.
DestinationDescription	String	Destination description.
AirlineName	String	Airline name.
AirlineCode	String	Airline code.
FlightNumber	String	Flight number.
AircraftCode	String	Aircraft code.
AircraftName	String	Aircraft name.
FlightTimeMinutes	Int	Flight time in minutes.
Class	String	Flight class.
BookingCode	String	Booking code.
IsConnection	String	Indicates if this flight segment has connection.
NextFlightChange	FlightSearchRSChange	If flight has connection, this object contains information about that change.
Group	Int	Flight segment group.
AvailabilitySource	String	Availability source.
AirlineLogoThumbnailUrl	String	List of airline logo thumbnail URL.
AirlineLogoUrl	String	List of airline logo URL.

FlightSearchRSChange

This object contains the flight segment connection information.

MEMBERS

Name	Type	Description
FlightChangeTypeID	Int	Identifier.
PlaneChangeAirportIata	String	If connection requires plane change, this field contains airport IATA.
PlaneChangeCityName	String	If connection requires plane change, this field contains city name.
AirportChangeFromCode	String	If connection requires airport change, this field contains origin airport code.
AirportChangeToCode	String	If connection requires airport change, this field contains destination airport code.
ConnectionTimeMinutes	Int	Duration of stop time in minutes between flights.

GetPackageDetailedDescriptionRQ

This object contains the parameters used to get the package tour details

MEMBERS

Name	Type	Description
NumberOfPersons	Int (nullable)	Number of persons for which price will be calculated. If numberOfPersons is null 1 person is assumed
PackageTourID	Int	ID of package tour whose detailed description is returned. Required parameter.
ObjectIDList	Int[]	List of object's ids whose detailed information will be returned. For objects whose id is not contained in this list only basic information will be returned.
Periods	Date []	List of periods for which detailed information will be returned. For other periods that are not in this list only basic information will be returned.
ThumbnailWidth	Int (nullable)	Maximum width of thumbnail photos. Thumbnail photos preserve original photo ratio. If thumbnailWidth and thumbnailHeight are set, photo will decrease to match these values preserving original photo ratio.
ThumbnailHeight	Int (nullable)	Maximum height of thumbnail photos. Thumbnail photos preserve original photo ratio. If thumbnailWidth and thumbnailHeight are set, photo will decrease to match these values preserving original photo ratio.
PhotoWidth	Int (nullable)	Maximum width of large photos. Large photos preserve original photo ratio. If photoWidth and

		photoHeight are set, photo will decrease to match these values preserving original photo ratio.
PhotoHeight	Int (nullable)	Maximum width of large photos. Large photos preserve original photo ratio. If photoWidth and photoHeight are set, photo will decrease to match these values preserving original photo ratio.
CurrencyID	Int (nullable)	Currency for prices that would be returned. Details
LanguageID	String	ID of language on which data will be returned. Details
UnitsAvailable	Bool (nullable)	If unitsAvailable is set then returned price is calculated for this number of units.
IgnorePriceAndAvailability	Bool	If ignorePriceAndAvailability is set to True, response will not contain any information about service prices or unit's availability.
AffiliateID	String	ID of affiliate partner.
MapWidth	Int (nullable)	If object's location is entered, mapWidth sets width of the map in pixels returned by the API. If mapWidth is not set, the default map width is returned.
MapHeight	Int (nullable)	If object's location is entered, mapHeight sets height of the map in pixels returned by the API. If mapHeight is not set, the default map height is returned.
ObjectURL	String	SEO (search engine optimizations) object URL. If this parameter is set, function ignores objectID and languageID parameters and returns

		the object that the objectURL links to. The language is determined from the URL, too, because each language has a different URL for every object.
DoNotCountMandatoryServices	Bool	If this is true the system will not count mandatory services on price calculation in api search results.
OutParameterList	PackageTourDetailedDescriptionOutParameter[]	OutParameterList is used to limit information about objects and units in response or to fetch additional information. Only information given in the outParameterList will be returned.
UserID	Int (nullable)	ID of the "logged in" user. Used on custom B2B portals to correctly generated the booking link.
InPriceType	PriceType (nullable)	The returned price will be calculated and displayed in requested price type (Per person, total, ...).
MarketIDList	Int[]	List of market IDs used to get the prices. Market IDs are ordered by priority.

GetPackageSearchResultsRQ

This object contains the parameters used to get correct package tours in a search

MEMBERS

Name	Type	Description
StartDate	DateTime (nullable)	Start date for period in which unit should be available. If startDate is set to null or equals to endDate, availability is not checked. DateTime format
EndDate	DateTime (nullable)	End date for period in which unit should be available. If endDate is set to null or startDate, availability is not checked. DateTime format
DestinationIDList	Long[]	[Obsolete] List of destinationIDs. If list is set, function returns only objects that

		belong to one of the destination with destinationID defined in the list.
RegionIDList	Long[]	List of regionIDs. If list is set, function returns only objects that belong to region with regionID defined in the list.
CountryIDList	Long[]	List of dountryIDs. If list is set, function returns only objects that belong to country with countryID defined in the list.
CategoryIDListUnion	Int[]	If categoryIDListUnion is not empty, function returns only objects who belongs to at least one of the categories from list. Details
PriceFrom	Double (nullable)	If priceFrom is set, function returns only units with price greater or equal to priceFrom.
PriceTo	Double (nullable)	If priceTo is set, function returns only units with price less or equal to priceTo.
PriceType	PriceType	Price type for prices that would be returned in response. PriceType is also used when parameters priceFrom or priceTo are set. Then priceFrom and priceTo are of current priceType.
SortParameterList	SortParameter []	SortParameterList is used for sorting response. Response can be sorted by multiple values. Sort priority is equal to sort parameters order in sortParameterList (first element has top priority).
PageSize	Int (nullable)	Defines how many objects will be returned in response. Maximum allowed is 20. We recommend returning up to 10 objects, because returning more objects increases data transferred over internet and reduces speed.
CurrentPage	Int (nullable)	Defines what page of the response will be returned. If number that exceeds total number of pages is sent, last page will be returned.
ObjectAttributeFilterList	AttributeFilter []	When objectAttributeFilterList is not empty, objects in response will be filtered out by given attributes in

		objectAttributeFilterList. This parameter is used to filter objects by Name, Stars, etc.
UnitAttributeFilterList	AttributeFilter []	When unitAttributeFilterList is not empty, units in response will be filtered out by given attributes in unitAttributeFilterList. This parameter is used to filter units by Capacity, Air Condition, etc.
ThumbnailWidth	Int (nullable)	Maximum width of thumbnail photos. Thumbnail photos preserve original photo ratio. If thumbnailWidth and thumbnailHeight are set, photo will decrease to match these values preserving original photo ratio.
ThumbnailHeight	Int (nullable)	Maximum height of thumbnail photos. Thumbnail photos preserve original photo ratio. If thumbnailWidth and thumbnailHeight are set, photo will decrease to match these values preserving original photo ratio.
PhotoWidth	Int (nullable)	Maximum width of large photos. Large photos preserve original photo ratio. If photoWidth and photoHeight are set, photo will decrease to match these values preserving original photo ratio.
PhotoHeight	Int (nullable)	Maximum width of large photos. Large photos preserve original photo ratio. If photoWidth and photoHeight are set, photo will decrease to match these values preserving original photo ratio.
CurrencyID	Int (nullable)	Currency for prices that would be returned. Details
UnitsAvailable	Byte (nullable)	Number of units (e.g. rooms) that have to be available in searched period. If UnitAvailabilityStatus is OnRequest, then unit will always be returned. If UnitAvailabilityStatus is Availabel, then specified number of units must be available in searched period.
OutParameterList	OutParameter []	OutParameterList is used to limit information about objects and units in response. Only

		information given in the outParameterList will be returned.
LanguageID	String	ID of language on which data will be returned. Details
AffiliateID	String	ID of affiliate partner.
CategoryIDListIntersection	Int[]	If categoryIDListIntersection is not empty, function returns only objects who belongs to all categories from list. Details
OnlyOnSpecialOffer	Bool (nullable)	Search only package tours that have special offer
ToDisplaySpecialOffers	Bool (nullable)	If true results will be grouped by special offer (so more than one instance of the same accommodation object can be returned but each will have only one special offer)
ChildrenAgeList	Int[]	A list of integer values representing ages of the children present in the search
DestinationCodes	String[]	List of destinationIDs. If list is set, function returns only objects that belong to one of the destination with destinationID defined in the list.
DoNotCountMandatoryServices	Bool	If this is true the system will not count mandatory services on price calculation in api search results.
ObjectIDList	Int[]	List of Object IDs. The search will return only objects whose IDs are in the list. If passed NULL or the list is empty, the parameter is ignored.
UnitCategoryIDList	Int[]	If UnitCategoryIDList is not empty, function returns only units from those categories. Details
UnitID	Int (nullable)	ID of the "logged in" user. Used on custom B2B portals to correctly generated the booking link.
ListCategorySearchResults	CategorySearchResultsRQ[]	List of parameters used to manage how many search results by category are

PackageTourDetailedDescriptionOutParameter

Out parameter for package tours while fetching detailed description.

MEMBERS

Name	Type	Description
------	------	-------------

ResponseDetail	Int	Defines which response detail is current OutParameter referring to. 1 = OccupiedSeats (currently only for Simple tours) 2 = ApplyNotesValidityOnPeriods (Fill the price item's note list with valid notes based on the validity dates defined in the note) 3 = GetUnavailableDatesForObject (Refers to getting all unavailable dates for the object (intersection of all unavailable dates on each unit)) 4 = GetPaymentPlan (Fill the package tour object payment plan if it is defined)
-----------------------	-----	-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

CategorySearchResultsRQ

Filter that restricts search results to specified number of results that match the category IDs.

MEMBERS

Name	Type	Description
CategoryID	Int	ID of a category that is searched for.
MaxNumberOfResults	Int	Maximum number of results that match the category ID.

GetPackageSearchResultsRS

It contains information about returned objects, language, currency and object's destinations, regions and countries.

MEMBERS

Name	Type	Description
PackageTourList	PackageTour []	List of package tours returned in search results
AccommodationObjectList	AccommodationObject []	Important: Returned only if OutParameter's ResponseDetail is set to value GetHotels! In that case only the following properties are filled in a response: - ObjectID - DestinationID - AttributeGroupList - PhotoList - PhotoUrl - ThumbnailUrl - Description - ShortDescription - Name

		- ObjectURL All other properties are empty!
DestinationList	Destination []	List of destination in which objects from AccommodationObjectList belong.
RegionList	Region []	List of regions in which destinations from DestinationList belong.
CountryList	Country []	List of countries in which regions from RegionList belong.
ObjectLocationList	ObjectLocation []	List of coordinates for the objects returned in search results
Currency	Currency	Currency for prices returned by the function.
Language	Language	Language for data returned by the function.
Status	Status	Status of the response. Determines if the result data is valid or error has occurred on server.
TotalNumberOfResults	Int	Total number of objects that meets given parameters in search results.
CurrentPage	Int	Defines what page of the response is returned. If number that exceeds total number of pages is sent, last page is returned
PageSize	int	Number of results per page
StartDate	DateTime	Search was performed between start date and end date
EndDate	DateTime	Search was performed between start date and end date
NumberOfDays	Int	Number of days between start and end date
NumberOfPersons	Int	Number of persons that search was performed with
AllResultsAdditionalData	AllPackageTourResultsAdditionalData	Container class that holds additional data for the entire result set (not only for the current page).

AllPackageTourResultsAdditionalData

Container class that holds additional data for the entire result set (not only for the current page). Used on package tour search results.

MEMBERS

Name	Type	Description
------	------	-------------

ObjectInfoList	PackageTourObjectInfo []	List of object info object containing additional data for the entire results set (not only from current page). Additional info include object names and seo urls.
-----------------------	------------------------------------------	-------------------------------------------------------------------------------------------------------------------------------------------------------------------

PackageTourObjectInfo

DTO object used to store additional info for the package tour object.

MEMBERS

Name	Type	Description
LemaxObjectID	Int	ID of the object whose name is stored here.
OtherSystemObjectID	String	ID used for 3PS purposes.
Name	String	Name of the object.
SeoURL	String	Object's SEO url.

GetPackageDetailedDescriptionRS

It contains information about returned object, period list, language, currency and object's destination, region and country.

MEMBERS

Name	Type	Description
PackageTour	PackageTour	Object whose detailed information is returned by the function.
AccommodationObjectList	AccommodationObject []	List of accommodation objects that appear in the package tour
DestinationList	Destination []	Destination of every object in AccommodationObjectList is contained in this list
RegionList	Region []	Region of every object in AccommodationObjectList is contained in this list
CountryList	Country []	Country of every object in AccommodationObjectList is contained in this list
Currency	Currency	Currency for prices returned by the function.
Language	Language	Language for data returned by the function.
Status	Status	Status of the response. Determines if the result data is valid or error has occurred on server.
PriceType	PriceType	Price type for prices returned by function

PriceFormat	PriceFormat	This object contains information on how to properly display price on current language.
PeriodList	Period []	Period list for prices of services in units contained by the returned object.

ObjectCancellationData

Hold object cancellation policy and rules data.

MEMBERS

Name	Type	Description
Name	String	Cancellation policy name.
Vat	Decimal	Vat applied to cancellation fee.
ServiceTypes	Int []	Service type IDs cancellation applies to.
CancellationTimeLimit	Int	Time (in seconds) until reservation can be cancelled on the date of travel. By default reservation cannot be cancelled on the date of travel.
Rules	ObjectCancellationRule []	List of cancellation rules.

ObjectCancellationRule

Defines single rule of object cancellation policy data.

MEMBERS

Name	Type	Description
DaysBeforeTravel	Int	Days before travel date that are applicable for this rule
CancellationFeeType	Int	Cancellation fee type ID.
Percentage	Decimal	Percentage amount (if the type is percentage).
FixedNetAmount	Decimal	Fixed net amount (if the type is fixed amount).

GetReservationCancellationRQ

Defines get reservation cancellation request.

MEMBERS

Name	Type	Description
ReservationID	Int	ID of reservation for which we want to access cancellation data

ReservationCancellationRS

Defines get reservation cancellation response.

MEMBERS

Name	Type	Description
ReservationID	Int	ID of reservation for which we want to access cancellation data
Status	Status	Operation Status
PaidAmount	Decimal	The amount paid for this reservation (NET)
RemainingAmount	Decimal	The amount of remaining amount to be paid for this reservation
Currency	Currency	Reservation currency
CancellationItems	List< ReservationItemCancellationInfoDTO >	List of available cancellation items

CancelReservationItemRQ

Defines cancel reservation item request.

MEMBERS

Name	Type	Description
ReservationItemID	Int	ID of reservation item which we want to cancel
CancellationReason	String	Reason for canceling the reservation item
CancelReservationIfAllItemsCancelled	Bool	Defines if reservation will be canceled if all reservation items were canceled

CancelReservationItemRS

Defines cancel reservation item response.

MEMBERS

Name	Type	Description
ReservationItemID	Int	ID of reservation item which we canceled
Status	Status	Status of cancellation of reservation item ID

CancelReservationItemsRQ

Defines cancel reservation items request.

MEMBERS

Name	Type	Description
ReservationItemID	Int	ID of reservation item which we canceled
CancelledItems	List< CancelReservationItemBaseDTO >	List of cancelled reservation items of type
CancelReservationIfAllItemsCancelled	Bool	Determines if reservation will be canceled if all reservation items were canceled
GenerateRefundTransactionAutomatically	Bool	Determines if refund will be generated for the canceled items

CancelReservationItemsRS

Defines cancel reservation items response.

MEMBERS

Name	Type	Description
ReservationItemCancellations	List< CancelReservationItemRS >	Reservation items that were canceled
Status	Status	Cancel reservation items response status

ReservationItemCancellationInfoDTO

Defines reservation item cancellation info.

MEMBERS

Name	Type	Description
ReservationItemID	Int	ID of reservation item
Name	String	Name/ Description of reservation item
OriginalPrice	Decimal	Price of reservation item in original currency
CancellationFee	Decimal	Cancellation fee in original reservation currency
InputVAT	Double	Input VAT in original reservation currency
OutputVAT	Double	Output VAT in original reservation currency
Commission	Double	Commission

CancelReservationItemBaseDTO

Defines cancel reservation items response.

MEMBERS

Name	Type	Description
ReservationItemID	Int	ID of reservation item
CancellationReason	String	Reason for canceling the reservation item

6. ENUMERATORS

PriceType

Defines price type for service price.

VALUES

Value	Description
PerPerson	Price per person. Total price is calculated by multiplying price with number of persons. $totalPrice = price * numberOfPersons$
PerDay	Price per day. Total price is calculated by multiplying price with number of days. $totalPrice = price * numberOfDays$
PerPersonPerDay	Price per person per day. Total price is calculated by multiplying price with number of persons multiplied by number of days. $totalPrice = price * numberOfPersons * numberOfDays$
Total	Total price. Total price is equal to price. $totalPrice = price$
PerWeek	Price per week. Total price is calculated by multiplying price with number of weeks. $totalPrice = price * numberOfWeeks$
PerPersonPerWeek	Price per person per week. Total price is calculated by multiplying price with number of persons multiplied by number of weeks. $totalPrice = price * numberOfPersons * numberOfWeeks$

SortBy

Defines parameters that are used to sort responses.

VALUES

Value	Description
Price	Response is sorted by price.
Destination	Response is sorted by destination.
Region	Response is sorted by region.
Country	Response is sorted by country.
Stars	Response is sorted by object stars count.
Priority	Response is sorted by objects priority.

SortOrder

Defines response's sorting direction.

VALUES

Value	Description
-------	-------------

Ascending	Response is sorted ascending.
Descending	Response is sorted descending.

ComparisonType

Defines comparison type when search results are filtered by object's or unit's [attributes](#).

VALUES

Value	Description
Equals	Attribute value must be equal to object's or unit's attribute value.
GreaterOrEqualThan	Attribute value must be greater or equal to object's or unit's attribute value.
LessOrEqualThan	Attribute value must be less or equal to object's or unit's attribute value.
Between	Attribute value must be between object's or unit's attribute value and attribute value 2

ResponseDetail

Defines what should be returned in GetSearchResults function.

VALUES

Value	Description
ObjectPhotos	Refers to object's photos .
UnitPhotos	Refers to unit's photos .
ObjectDetailedAttributes	Refers to object's detailed attributes . Detailed attributes are attributes that are not in object's basic group .
UnitDetailedAttributes	Refers to unit's detailed attributes . Detailed attributes are attributes that are not in unit's basic group .
ObjectDescription	Refers to object's html description.
UnitDescription	Refers to unit's html description.
CalculatedPriceInfo	Refers to unit's prices
MapCoordinates	Refers to object's location on the world map
GetHotels	Refers to getting hotels for package tours on search.

AttributeType

Defines [attribute](#) type.

VALUES

Value	Description
Logical	Attribute is logical type.
Textual	Attribute is textual type.

UnitAvailabilityStatus

Defines [unit's](#) availability status.

VALUES

Value	Description
OnRequest	Unit cannot be directly booked, and inquiry must be made.
Available	Unit can be directly booked.
NotAvailable	Unit is not available

ServiceType

Defines [service](#) type

VALUES

Value	Description
Basic	Service is a basic service.
Supplement	Service is a supplement.
Discount	Service is a discount.
SpecialOffer	Service is a special offer.
AdditionalService	Service is an additional service
MandatoryService	Service is a mandatory service

StatusCode

Defines status codes of response.

VALUES

Value	Description
OK	Response is valid.
Error	Response is not valid. Error has occurred on the server.
PartialSuccess	Response is valid, but some parts of responses are not valid (for example if results from GDS are returned, it is possible that some GDS returned error, and other returned Success)

ServicePriceType

Defines if service price amount is absolute amount or percentage.

VALUES

Value	Description
Fixed	Price is defined as an absolute amount
Percentage	Price is defined as a percentage of another amount
FixedDiscount	Price is defined as a fixed discount on another amount
FreeNights	Price is defined as number of nights (out of the whole interval) that are considered free of charge.

FlightDestinationType

Flight destination type

VALUES

Value	Description
Country	1
City	2
Airport	3

CancellationData

This class contains reservation item's cancellation information.

MEMBERS

Name	Type	Description
CancellationDateWithoutFee	DateTime?	First cancellation date. Until this date cancellation is free of charge. If this is null this mean that cancellation deadline has passed.
IsNonRefundable	bool	A flag that indicates if the reservation item is non-refundable.
CancellationNote	string	A text note related to cancellation.
ListCancellationItems	CancellationDataItem[]	List of cancellation dates. Each reservation item can have multiple cancellation dates with different amount. Example 15.1. -> 200 EUR, 16.1. -> 400 EUR 17.1. 800EUR...
CancellationBufferMinutes	Int	Offsets every cancellation date in the list of cancellation items by this many minutes to the past.
VAT	Decimal?	VAT for the cancellation fee price calculation.
DisplayText	string	Preformatted HTML string which is displayed to users in back office

CancellationDataItem

This class contains cancellation information for a particular deadline.

MEMBERS

Name	Type	Description
Date	DateTime	Cancellation date.
NetAmount	decimal	Net amount of the cancellation fee.
SellingAmount	decimal	Selling amount of the cancellation fee.

CurrencyShortName	string	A symbol or abbreviation of the cancellation fee currency.
--------------------------	--------	------------------------------------------------------------

CancellationFeeTypeEnum

Defines the type of a cancellation fee.

VALUES

Value	Description
Percentage	1 - Cancellation fee is calculated as a percentage of the reservation's price.
FixedAmount	2 - Cancellation fee is defined with fixed net and selling amounts.
ChargeableNights	3 - Cancellation fee is calculated by charging a set amount of nights.