

# **RENT A CAR RESERVATION FUNCTIONS**

## **1. RentReservation (DataIn: TJsonValue): TJSONObject**

Remote method invocation:

<http://host:port/datasnap/rest/tservermethods/RentReservation/DataIn>

DataIn example:

```
{  
  "OverWrite": "Y",  
  "ClientID": "33497",  
  "DepartureDate": "01.01.2017 15:10",  
  "ArrivalDate": "21.01.2017 18:25",  
  "ReferenceNo": "123",  
  "FlightNo": "234",  
  "Email": "lis@winmentor.ro",  
  "PhoneNumber": "0748303288",  
  "Observation": "xxxx",  
  "DepartureLocationID": "GALATI",  
  "ArrivalLocationID": "ORADEA",  
  "PickUpAddress" : "aaa",  
  "DropOffAddress" : "bbb",  
  "Items": [  
    { "ID": "15082", "Discount": "3"},  
    { "ID": "15159"}  
  ]  
}
```

- Returns :

```
{"result":"ok","InternalNo":"98","ErrorList":[]}
```

If result value is "error" the ErrorList array contains the error list explanations.

Nu se poate modifica clientul pe o rezervare existenta. Daca se retrimite o rezervare cu acelasi nr de referinta dar alt client, se va returna mesajul "Reservation X does not exist in the database. Cannot overwrite it."

## 2. GetItemsList

Remote method invocation :

<http://host:port/datasnap/rest/tservermethods/GetItemsList>

Returns (example): {"result":"ok","ErrorList":[],"ItemsList":[{"Name":"INCHIRIERE SKODA RAPID","ID":"SKR"},{"Name":"FIAT 500","ID":"FIAT500"}]}

If result value is "error" the ErrorList array contains the error list explanations.

## 3. GetReservationStatus (ClientID, ReferenceNo : string ) : TJSONObject;-

Remote method invocation :

<http://host:port/datasnap/rest/tservermethods/GetReservationStatus/ClientID/ReferenceNo>

Returns (example) : {"result":"ok","status":"canceled","amount":"0.00","ErrorList":[]}

If result value is "error" the ErrorList array contains the error list explanations.

## 4. CancelReservation (ClientID, ReferenceNo : string ) : TJSONObject;

Remote method invocation :

<http://host:port/datasnap/rest/tservermethods/CancelReservation/ClientID/ReferenceNo>

Returns (example) : {"result":"ok","ErrorList":[]}

If result value is "error" the ErrorList array contains the error list explanations.

## 5. GetPriceList (ClientID : string) : TJSONObject

Remote method invocation:

<http://host:port/datasnap/rest/tservermethods/GetPriceList/ClientID>

Returns (example): {"result":"ok","PriceList": [{"ItemID":"22","Location":"Iasi","PriceOffer":"Broker 15.07.2015","PriceCategory":"Broker","Currency":"EUR","BookingStartDate":"15.07.2015 6:00:00 AM","BookingEndDate":"30.09.2017 7:00:00 PM","PickUpStartDate":"09.03.2017 12:00:00PM","Type":"days","Limit\_1":"1","Price\_1":"79","Limit\_2":"3","Price\_2":"78","Limit\_3":"5","Price\_3":"77","Limit\_4":"8","Price\_4":"76","Limit\_5":"15","Price\_5":"75","Limit\_6":"30","Price\_6":"74","FixValue":"4000","Warranty":"400"} ],"ErrorList":[]}

If result value is "error" the ErrorList array contains the error list explanations.



## 6. GetKmBord

Remote method invocation:

`http://host:port/datasnap/rest/tservermethods/GetKmBord/Serie`

Returns (example):

```
{"result":"ok","KmBord":"8677","DataBord":"21.03.207","ErrorList":[]}
```

\* serie reprezinta serie caroserie din Parc auto

## 7. UpdateKmBord

Remote method invocation:

`http://host:port/datasnap/rest/tservermethods/UpdateKmBord/DataIn`

DataIn (example):

```
{
  "ListaAuto": [{
    "Serie": "WF1019119000",
    "KmBord": "9100"
  }, {
    "Serie": "WF1019119000",
    "KmBord": "7528"
  }]
}
```

Returns (example):

```
{"result":"ok","ErrorList":[]}
```

## 8. InfoAuto

Remote method invocation:

`http://host:port/datasnap/rest/TServerMethods/UpdateInfoAuto/DataIn`

DataIn (exemplu) :

```
{
  "ListaAuto": [{
    "Serie": "WF1019119000",
    "NUMECOEXTENSIE":"valoare"
  }, {
    "Serie": "WF1019119622",
    "NUMECOEXTENSIE":"valoare"
  }]
}
```

Returns (example):

```
{"result":"ok","ErrorList":[]}
```

Functia este folosita pentru update extensii la parc auto. In loc de NUMECOEXTENSIE se va scrie (cu litere mari) numele campului din extensii pe care se doreste sa se faca update.