IOP Ass 5c

Paul Bartl

May 12, 2022

Address of the WSDL: http://wwwlab.cs.univie.ac.at/~bartlp20/iop/implementation/airplanemanufacturing.wsdl

0.1 getTeamsByEmployees(x)

0.1.1 Purpose

Returns all teams in 1_data.xml with less than or equal x amount of employees.

0.1.2 Return data

```
<Team teamid="609" employees="3">
    <ISO_Code>IT</ISO_Code>
    <HourlyRate>199.61</HourlyRate>
    <Units>
        <Unit unitid="1ff5cb55-ff0e-47a2-8e44-c9a40bdb90dc" registration="WEQ5188">
            <Wingtips>true</Wingtips>
            <Windows>57</Windows>
            <seatconfig>
                <seat compartment="eco" cover="fabric">
                    <amount>61</amount>
...
```

0.1.3 Covered Requirements

Operation	R1	R2	R3	R4	R5	R6	R7
getTeamsByEmployees		X					X

0.2 getTeamsByCountries(x)

0.2.1 Purpose

Returns all teams in 1_data.xml which are based in countries provided by array x.

0.2.2 Return data

```
<Team teamid="36" employees="3">
    <ISO_Code>IS</ISO_Code>
    <HourlyRate>68</HourlyRate>
    <Units>
        <Unit unitid="c7dfc8da-2097-48aa-aca4-5caa470b4f60" registration="LSX8073">
            <Wingtips>true</Wingtips>
            <Windows>50</Windows>
            <seatconfig>
                <seat compartment="eco" cover="fabric">
                    <amount>44</amount>
                    <tv>false</tv>
                </seat>
...
```

0.2.3 Covered Requirements

Operation	R1	R2	R3	R4	R5	R6	R7
getTeamsByCountries	X				5.1		X

0.3 getSmallestTeamID()

0.3.1 Purpose

Returns the team with the smallest teamId in 1_data.xml.

0.3.2 Return data

13

0.3.3 Covered Requirements

Operation	R1	R2	R3	R4	R5	R6	R7
getSmallestTeamID							

0.4 getCustomerByID(x)

0.4.1 Purpose

Returns customer where customerId = x in 2_data.xml, Return data customer has id = 119

0.4.2 Return data


```
{
    "Firstname": "Joy",
    "Lastname": "Robles"
}
```


0.4.3 Covered Requirements

Operation	R1	R2	R3	R4	R5	R6	R7
getCustomerByID		X				X	

0.5 getCustomersByZip(x)

0.5.1 Purpose

Returns all customers in 2_data.xml whose address has a zip of less than x. Example zip is 10000.

0.5.2 Return data


```
{
    "Firstname": "Jeremy",
    "Lastname": "Claes",
    "Address": {
        "@attributes": {
            "zip": "2708",
            "city": "Michaelland"
        },
        "Street": "Melendez Locks",
        "Housenumber": "381"
    }
}
```


0.5.3 Covered Requirements

Operation	R1	R2	R3	R4	R5	R6	R7
getCustomersByZip		X				X	

0.6 getStreetsByZip(x)

0.6.1 Purpose

Returns all streets in 3_data.xml which are based city whose zip code is less than x. Example zip is 10000.

0.6.2 Return data


```
{
    "Street": [
        "Melendez Locks",
        "Cantu Road",
        "William Course",
        "Cantu Road",
        "Hughes Ford",
        "Hughes Ford",
        "William Course"
    ]
}
```


0.6.3 Covered Requirements

Operation	R1	R2	R3	R4	R5	R6	R7
getStreetsByZip		X				X	

0.7 getBarsOfPlane(x)

0.7.1 Purpose

Returns all bars of plane with registration x. Example registration is RTM7433.

0.7.2 Return data


```
{
    "Bar": {
        "@attributes": {
            "unitid": "5ed2e601-1236-4a61-ba4b-6bb605d73831",
            "teamid": "39"
        },
        "Minifridges": {
            "Amount": "8"
        },
        "Glasses": {
            "Type": "high ball",
            "Amount": "43"
        },
        "Beverages": {}
    }
}
```


0.7.3 Covered Requirements

Operation	R1	R2	R3	R4	R5	R6	R7
getBarsOfPlane			X			X	

0.8 getBarMinifridgeAndColor(x)

0.8.1 Purpose

Returns all registrations of planes which match the input criteria of array x. Array x consists of dict fridgesInterval and array colors. Every plane with at least one bar that has the amount of fridges between fridgesInterval min/max or matches one color in the colors array are included in the returned data.

0.8.2 Return data

```
<?xml version="1.0"?>
<Registrations>
        <Registration>CKN7593</Registration>
        <Registration>TEX8779</Registration>
        <Registration>WPA4428</Registration>
        <Registration>MPF3551</Registration>
        <Registration>YWK5335</Registration>
        <Registration>FMW2950</Registration>
        <Registration>BVK3060</Registration>
        <Registration>ZIM6990</Registration>
        <Registration>FAJ8688</Registration>
        <Registration>YKF6723</Registration>
...
```

0.8.3 Covered Requirements

Operation	R1	R2	R3	R4	R5	R6	R7
getBarMinifridgeAndColor					5.3		X

0.9 getPlanesByColor(x)

0.9.1 Purpose

Returns all planes which match one of the colors supplied by array x. Searches in 4_data.xml

0.9.2 Return data

```
<Plane registration="TEX8779">
        <Color>white</Color>
        <Livery>eurowhite</Livery>
        <Bars>
                <Bar unitid="8882156c-f456-40e3-a564-2c55e1d28b68" teamid="767">
                        <Minifridges>
                                <Amount>5</Amount>
                        </Minifridges>
                        <Glasses>
                                <Type>high ball</Type>
                                <Amount>32</Amount>
                        </Glasses>
                        <Beverages>
...
```

0.9.3 Covered Requirements

Operation	R1	R2	R3	R4	R5	R6	R7
getPlanesByColor	X				5.1		X

0.10 getPlanesByLivery(x)

0.10.1 Purpose

Returns all planes which match one of the liveries supplied by array x. Searches in 4_data.xml

0.10.2 Return data


```
{
    "@attributes": {
        "registration": "DAJ7178"
    },
    "Color": "maroon",
    "Livery": "eurowhite",
    "Bars": {}
}
```


0.10.3 Covered Requirements

Operation	R1	R2	R3	R4	R5	R6	R7
getPlanesByLivery	X				5.1	X	

0.11 getProtocolByDate(x)

0.11.1 Purpose

Returns protocols of given date x in 2_data.xml. Example date ist '2017-06-12'.

0.11.2 Return data

```
<Protocol protocolid="UFGUPCARTE">
    <Testdate>2017-06-12</Testdate>
    <results>
        <test>wings</test>
        <vibrations loc="cockpit">
            <scale>low</scale>
            <critical>false</critical>
        </vibrations>
        <vibrations loc="cabin">
            <scale>negligible</scale>
            <critical>false</critical>
        </vibrations>
        <detached-wing side="left">none</detached-wing>
        <detached-wing side="right">none</detached-wing>
    </results>
</Protocol>
```

0.11.3 Covered Requirements

Operation	R1	R2	R3	R4	R5	R6	R7
getProtocolByDate				X			X

0.12 getToiletsBetweenFlowrates(x)

0.12.1 Purpose

Returns all toilet-specs of toilets which flowrate falls between the min/max value supplied by x in 3_data.xml. Example interval is minRate = 2.0, maxRate = 2.5.

0.12.2 Return data

```
<ToiletSpecs unitid="db4e6218-b7ec-40fa-a771-bb321b25b8d4">
    <Capacity>8.75</Capacity>
    <Flowrate>2.45</Flowrate>
</ToiletSpecs>
```

0.12.3 Covered Requirements

Operation	R1	R2	R3	R4	R5	R6	R7
getToiletsBetweenFlowrates		X			5.2		X

0.13 getCustomerOfPlane(x)

0.13.1 Purpose

Returns customer of plane with registration = x. Example registration is 'KPTDTWVNUZ'.

0.13.2 Return data


```
{
    "Firstname": "Sila",
    "Lastname": "Hunter",
    "Address": {
        "@attributes": {
            "zip": "21360",
            "city": "South Erintown"
        },
        "Street": "Spencer Shoals",
        "Housenumber": "360"
    }
}
```


0.13.3 Covered Requirements

Operation	R1	R2	R3	R4	R5	R6	R7
getCustomerOfPlane			X			X	

0.14 getTestdateOfProtocol(x)

0.14.1 Purpose

Returns test date of protocol with protocolId = x.

0.14.2 Return data

"2017-02-19"

0.14.3 Covered Requirements

Operation	R1	R2	R3	R4	R5	R6	R7
getTestdateOfProtocol			X			X	

Requirement summary

Operation	R1	R2	R3	R4	R5	R6	R7
getTeamsByEmployees		X					X
getTeamsByCountries	X				5.1		X
getSmallestTeamID							
getCustomerByID		X				X	
getCustomersByZip		X				X	
getStreetsByZip		X				X	
getBarsOfPlane			X			X	
getBarMinifridgeAndColor					5.3		X
getPlanesByColor	X				5.1		X
getPlanesByLivery	X				5.1	X	
getProtocolByDate				X			X
getToiletsBetweenFlowrates		X			5.2		X
getCustomerOfPlane			X			X	
getTestdateOfProtocol			X			X	