MDX Query for OLAP Server

This example demonstrates how to use MDX to query OLAP server via OLAP4J (http://www.olap4j.org/)

<u>Prerequisite:</u>

The OLAP server must support XML for Analysis.

(Example: Mondrian OLAP Server: http://mondrian.pentaho.com)

<u>Steps:</u>

1. *Download OLAP4J at http://www.olap4j.org/*
2. *Unzip the distribution and copy all the jar files located at olap4j-1.x.x.xxx\lib (except **olap4j-jdk14-1.x.x.xxx.jar**) to \\Repertoire installation directory\ext*
3. *Create an object datasource:*
    a. *Create the data columns. The name and type of the columns will have to be derived from the MDX Query.*
    b. *Paste the following code under the JavaScript tab.*
4. *Modify the two parameters ${URL} & ${MDX}:*
    a. *${URL} - The service end point.*

       For example: jdbc:xmla:Server=http://localhost:8080/mondrian/xmla

    b. *${MDX} - The MDX Query.*
5. *Run the datasource.*

----

importClass(org.olap4j.OlapConnection);

importClass(org.olap4j.OlapStatement);

importClass(org.olap4j.CellSet);

importClass(org.olap4j.Position);

importClass(java.sql.Connection);

importClass(java.sql.DriverManager);

function pushTo(/*PushContext*/ cxt, /*DataListener*/ dl)

{

```
new java.lang.Class.forName("org.olap4j.driver.xmla.XmlaOlap4jDriver");


var connection = new java.sql.DriverManager.getConnection("${URL}");

var olapConnection =  connection.unwrap(org.olap4j.OlapConnection);

var statement = olapConnection.createStatement();


var cellSet = statement.executeOlapQuery("${MDX##}");


dl.startData(this);


for (var i=0;i<cellSet.getAxes().get(1).getPositionCount();i++)

{

  var rec = this.newRecordInstance();

  var data = rec.getData();


   //print dimension
        for (var j =0; j < cellSet.getAxes().get(1).getPositions().get(0).getMembers().size(); j++)

{

data[j] = cellSet.getAxes().get(1).getPositions().get(i).getMembers().get(j).getName();

}


var cnt = j+1;


   //print measures

for (var k =0; k < cellSet.getAxes().get(0).getPositions().size(); k++)

{

data[j] = cellSet.getCell(cellSet.getAxes().get(0).getPositions().get(k),
cellSet.getAxes().get(1).getPositions().get(i)).getFormattedValue()

j++;
```

```
    }


  dl.processRecord(rec);

 }


dl.endData(this);


}
```