
e-Transaction Aggregation & Analysis Layer

**Technical Document
on
Creation of Client Connector**

January 2013



The **specifications** of Client Connector are given below:

Web Method Name: eTransactionCount

Input Parameters: The web method created for the purpose will have following parameters:

SL #	Parameter Name	Format	Example
1	TransactionDate	DD/MM/ YYYY	23/11/ 2012
2	UserName	String	DeiTy
3	Password	String	xYz#\$36F

Response Type: The web method will return either XML or dataset.

XML

```
<?xml version="1.0" encoding="utf-8" ?>
<NeSD_Bihar>
  <response ServiceCode="C200102200003" ServiceCount="207" LocationCode="10212212001"/>
  <response ServiceCode="A200100100004" ServiceCount="7245" LocationCode="10212212002"/>
  <response ServiceCode="D200100200005" ServiceCount="3425" LocationCode="10212213001"/>
</NeSD_Bihar>
```

1 SAMPLE CODE

1.3 WEB SERVICE WRITTEN IN JAVA

```
package in.nic.exchange.action;

import in.nic.exchange.db.DBConnection;

import java.sql.Connection;
import java.sql.PreparedStatement;
import java.sql.ResultSet;
import java.sql.SQLException;
import java.sql.Statement;
import java.text.SimpleDateFormat;
import java.util.ArrayList;
import java.util.Date;
import java.util.HashMap;

public class ExchangeWebService {

public ArrayList<HashMap<String, Float>> countRegistrationsActivityWise(Date date) {

    Connection con = null;
    PreparedStatement pstmt = null;
    ResultSet rs = null;
    System.out.println("date = " + date);
    String query = "select name, sum(transactions) as total_trns from refactivity ra left join
(select * from dashboard where trndate = ?) db on db.activity=ra.code gGroup by ra.code,ra.name order by
ra.code;";
    SimpleDateFormat formatter = new SimpleDateFormat("yyyy-MM-dd");
    String d = formatter.format(date);
    ArrayList<HashMap<String, Float>> result = new ArrayList<HashMap<String, Float>>();
    HashMap<String, Float> map = null;
    try {
        con = DBConnection.getConnection("", "");
        pstmt = con.prepareStatement(query,
ResultSet.TYPE_SCROLL_INSENSITIVE,ResultSet.CONCUR_UPDATABLE);
        pstmt.setString(1, d);
        rs = pstmt.executeQuery(query);
        System.out.println("query = " + query);
        while(rs.next()) {
            map = new HashMap<String, Float>();
            map.put(rs.getString("name"), rs.getFloat("total_trns"));
            result.add(map);
        }
    } catch (SQLException sqle) {
        sqle.printStackTrace();
    } catch (Exception e) {
        e.printStackTrace();
    } finally {
        try {
            rs.close();
            pstmt.close();
            con.close();
        } catch (SQLException e) {
            // TODO Auto-generated catch block
            e.printStackTrace();
        }
    }
}
```

```

        return null;
    }

    public ArrayList<HashMap<String, Float>> countRegistrationsStateWise(Date date) {

        Connection con = null;
        PreparedStatement pstmt = null;
        ResultSet rs = null;
        System.out.println("date = " + date);
        String query = " select name, sum(transactions) as total_trns from refstate rs left join (select *
from dashboard where trndate= ? ) db on db.state=rs.code where acronym='I' group by rs.code,rs.name order by
rs.code;";
        SimpleDateFormat formatter = new SimpleDateFormat("yyyy-MM-dd");
        String d = formatter.format(date);

        ArrayList<HashMap<String, Float>> result = new ArrayList<HashMap<String, Float>>();
        HashMap<String, Float> map = null;
        try {
            con = DBConnection.getConnection("", "");
            pstmt = con.prepareStatement(query,
ResultSet.TYPE_SCROLL_INSENSITIVE,ResultSet.CONCUR_UPDATABLE);
            pstmt.setString(1, d);
            rs = pstmt.executeQuery(query);
            System.out.println("query = " + query);
            .while(rs.next()) {
                map = new HashMap<String, Float>();
                map.put(rs.getString("name"), rs.getFloat("total_trns"));
                result.add(map);
            }
        } catch (SQLException sqle) {
            sqle.printStackTrace();
        } catch (Exception e) {
            e.printStackTrace();
        } finally {
            try {
                rs.close();
                pstmt.close();
                con.close();
            } catch (SQLException e) {
                // TODO Auto-generated catch block
                e.printStackTrace();
            }
        }

        return null;
    }

    public float countRegistrations(Date date) {

        Connection con = null;

```

```

Statement stmt = null;
ResultSet rs = null;
System.out.println("date = " + date);
String query = "select sum(transactions) as trns from dashboard where trndate='";
SimpleDateFormat formatter = new SimpleDateFormat("yyyy-MM-dd");
String d = formatter.format(date);
System.out.println("date = " + date);

try {
    con = DBConnection.getConnection("", "");
    stmt =
con.createStatement(ResultSet.TYPE_SCROLL_INSENSITIVE,ResultSet.CONCUR_UPDATABLE);
    query += d + "'";
    rs = stmt.executeQuery(query);
    System.out.println("query = " + query);
    if(rs.next()) {
        return rs.getFloat("trns");
    } else {
        return 0;
    }

} catch(SQLException sqle) {
    sqle.printStackTrace();
} catch(Exception e) {
    e.printStackTrace();
} finally {
    try {
        rs.close();
        stmt.close();
        con.close();
    } catch (SQLException e) {
        // TODO Auto-generated catch block
        e.printStackTrace();
    }
}

return 0;
}
}

```