**Internet Technologies Study Material 17-03-20 and 18-03-20**

JDBC ResultSet Interface

Bydefault Resultset interface are non scrollable and not updatable

It is classified into two types

* Non-Scrollable ResultSet
* Scrollable ResultSet

By default a ResultSet Interface is Non-Scrollable, In non-scrollable ResultSet we can move only in forward direction (that means from first record to last record), but not in Backward Direction.

So, If we want to move in backward direction, we can make use of **Scrollable Interface**.

Also, in Non-Scrollable ResultSet and Scrollable ResultSet updations are not allowed

**Non-Scrollable ResultSet**

**Non Scrollable ResultSet** vs  **Scrollable ResultSet**

|  |  |
| --- | --- |
| Cursor move only in forward direction | Cursor can move both forward and backward direction |
| Slow performance, If we want to move nth record then we need to n+1 iteration | Fast performance, directly move on any record. |
| Non-Scrollable ResultSet cursor can not move randomly | Scrollable ResultSet cursor can move randomly |

**Create Scrollable ResultSet**

Statement stmt=con.CreateStatement(parameter1, parameter2);

Please note that first argument is type and next argument is mode.

These type are predefined in ResultSet Interface of Jdbc which are static are as follows:

1. TYPE\_FORWARD\_ONLY

It is default setting and it restrict the virtual cursor to downward direction

1. TYPE\_SCROLL\_SENSITIVE

Permits cursor to move in both directions

1. TYPE\_SCROLL\_INSENSITIVE

Permits cursor to move in both directions

These mode are predefined in ResultSet Interface of JDBC which are static are as follows:

Rows in ResultSet can be updated similar to how rows in table can be updated

1. CONCUR\_READ\_ONLY

Prevent ResultSet from being updated

1. CONCUR\_UPDATABLE

It will update ResultSet

For eg:

Statement stmt=con.createStatement (ResultSet.TYPE\_SCROLL\_INSENSITIVE,ResultSet.CONCUR\_READ\_ONLY);

This will make ResultSet cursor to move in both direction and also will prevent Resultset from being updated

**Methods of Scrollable ResultSet (in addition with next())**

Below all methods are used for move the cursor in Scrollable ResultSet.

* **first:**Used to move the virtual cursor to the first row in ResultSet.
* **last():**Used to move the virtual cursor at last row.
* **previous():** used to move cursor to previous row
* **absolute():** position the virtual cursor at the row no specified by the integer passed as the argument
* **relative():** move the virtual cursor to the specified no of rows with respect to current position. The value passed can be positive or negative.
* **getrows():** returns an integer that represents the no of current rows in ResultSet

Example :

Say we have Student table

|  |  |
| --- | --- |
| **RollNo**  | **Name**  |
| 1 | Rohit  |
| 2 | Sanaya  |
| 3 | Rishabh  |
| 4 | Rohan  |
| 5 | Rekha  |
| 6 | Swati  |
| 7 | Neha  |

Import java.sql.\*;

Public class result

{ public static void main(String args[]) throws Exception

{

 Class.forName(“com.mysql.jdbc.Driver”);

Connection con=DriverManager.getConnection(“jdbc:mysql://localhost3306/DB”,”root”, “root”));

Statement st=con.createStatement(ResultSet.TYPE\_SCROLL\_INSENSITIVE,ResultSet.CONCUR\_READ\_ONLY);

ResultSet rs=st.executeQuery(“Select \* from Student”);

while (rs.next())

{System.out.println (rs.getInt(1) +” ”+rs.getString(2));

}

//usage of above mentioned functions

rs.first(); // after displaying every record, it will move cursor again at the first row

System.out.println( rs.getInt(1) + “ ”+ rs.getString(2));

rs.last();// cursor at last row

System.out.println( rs.getInt(1) + “ ”+ rs.getString(2));

rs.absolute(3);// cursor at third row

System.out.println( rs.getInt(1) + “ ”+ rs.getString(2));

rs.previous(); //now cursor last row

rs.relative(-1);

con.close()

}}