# ResultSet interface

The object of ResultSet maintains a cursor pointing to a row of a table. Initially, cursor points to before the first row.

#### *By default, ResultSet object can be moved forward only and it is not updatable.*

But we can make this object to move forward and backward direction by passing either TYPE\_SCROLL\_INSENSITIVE or TYPE\_SCROLL\_SENSITIVE in createStatement(int,int) method as well as we can make this object as updatable by:

Statement stmt = con.createStatement(ResultSet.TYPE\_SCROLL\_INSENSITIVE,

                     ResultSet.CONCUR\_UPDATABLE);

### Commonly used methods of ResultSet interface

|  |  |
| --- | --- |
| ****1) public boolean next():**** | is used to move the cursor to the one row next from the current position. |
| ****2) public boolean previous():**** | is used to move the cursor to the one row previous from the current position. |
| ****3) public boolean first():**** | is used to move the cursor to the first row in result set object. |
| ****4) public boolean last():**** | is used to move the cursor to the last row in result set object. |
| ****5) public boolean absolute(int row):**** | is used to move the cursor to the specified row number in the ResultSet object. |
| ****6) public boolean relative(int row):**** | is used to move the cursor to the relative row number in the ResultSet object, it may be positive or negative. |
| ****7) public int getInt(int columnIndex):**** | is used to return the data of specified column index of the current row as int. |
| ****8) public int getInt(String columnName):**** | is used to return the data of specified column name of the current row as int. |
| ****9) public String getString(int columnIndex):**** | is used to return the data of specified column index of the current row as String. |
| ****10) public String getString(String columnName):**** | is used to return the data of specified column name of the current row as String. |

### Example of Scrollable ResultSet

Let’s see the simple example of ResultSet interface to retrieve the data of 3rd row.

**import** java.sql.\*;

**class** FetchRecord{

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

Class.forName("oracle.jdbc.driver.OracleDriver");

Connection con=DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:xe","system","oracle");

Statement stmt=con.createStatement(ResultSet.TYPE\_SCROLL\_SENSITIVE,ResultSet.CONCUR\_UPDATABLE);

ResultSet rs=stmt.executeQuery("select \* from emp765");

*//getting the record of 3rd row*

rs.absolute(3);

System.out.println(rs.getString(1)+" "+rs.getString(2)+" "+rs.getString(3));

con.close();

}}