8) MetaData:- MetaData means extra data about data or extra information of original data.

JDBC provided following types of MetaData.

8.1) DatabaseMetaData.

8.2) ResultSetMetaData.

8.3) Parameter MetaData.

Usually, these are useful in development of **“GUI DB Tools” like “SQL Developer”.**

**8.1) DatabaseMetaData**:- It is interface present in java.sql package. Driver software vendor is responsible to provide implementation.

We use Database metadata to get information about connected Database Or Underlying Database. The information is product name,product Major and Minor version, max no.of tables the database supports, max no.of columns each table can allows,Max no.of connection DB supports, Max length of user name, max length of table name …etc.

We use Database metadata to check whether a particular feature is supported by DB or not like stored procedures,full joins etc.

We get Database metadata object using getMetaData() method of Connection Interface.

DatabaseMetaData d= con.getMetaData();

Example:-

-----------

import java.sql.Connection;

import java.sql.Statement;

import java.sql.ResultSet;

import java.sql.SQLException;

import java.sql.DatabaseMetaData;

class Type4

{

public static void main(String arg[])

{

try

{

Connection con=DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:ORCL","scott","sukumar");

DatabaseMetaData d=con.getMetaData();

System.out.println(" DB product Name:"+ d.getDatabaseProductName());

System.out.println(" Driver Name:"+d.getDriverName());

System.out.println("DB URL:"+ d.getURL());

System.out.println("DB User:"+ d.getUserName());

System.out.println("Maximum columns Table:"+d.getMaxColumnsInTable());

System.out.println("Key words:"+d.getSQLKeywords());

System.out.println("Numeric Functions:"+d.getNumericFunctions());

System.out.println("String Functions:"+d.getStringFunctions());

}

catch(Exception e){System.out.println(e.getMessage());}

}

}

Output:

---------

D:\>java Type4

DB product Name:Oracle

Driver Name:Oracle JDBC driver

DB URL:jdbc:oracle:thin:@localhost:1521:ORCL

DB User:SCOTT

Maximum columns Table:1000

Max Tables in select statement:0

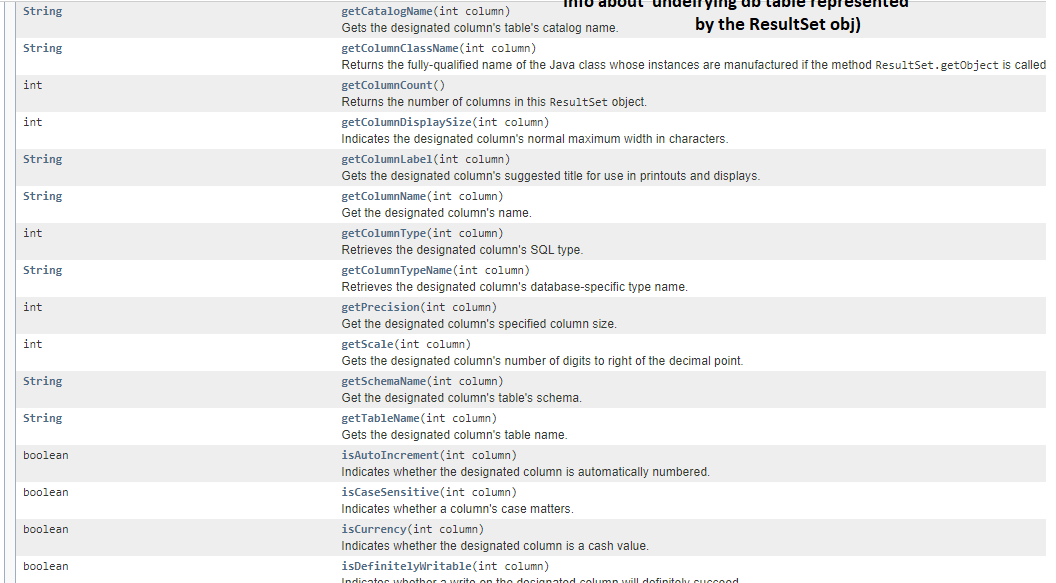
Key words:ACCESS, ADD, ALTER, AUDIT, CLUSTER, COLUMN, COMMENT, COMPRESS, CONNECT, DATE, DROP, EXCLUSIVE, FILE, IDENTIFIED, IMMEDIATE, INCREMENT, INDEX, INITIAL, INTERSECT, LEVEL, LOCK, LONG, MAXEXTENTS, MINUS, MODE, NOAUDIT, NOCOMPRESS, NOWAIT, NUMBER, OFFLINE, ONLINE, PCTFREE, PRIOR, all\_PL\_SQL\_reserved\_ words

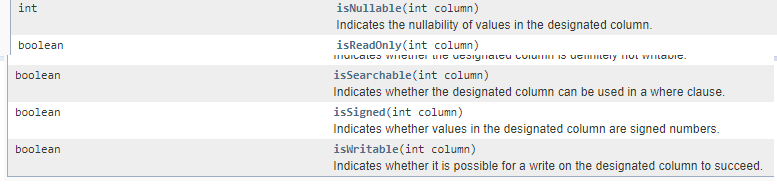
Numeric Functions:ABS,ACOS,ASIN,ATAN,ATAN2,CEILING,COS,EXP,FLOOR,LOG,LOG10,MOD,PI,POWER,ROUND,SIGN,SIN,SQRT,TAN,TRUNCATE

String Functions:ASCII,CHAR,CHAR\_LENGTH,CHARACTER\_LENGTH,CONCAT,LCASE,LENGTH,LTRIM,OCTET\_LENGTH,REPLACE,RTRIM,SOUNDEX,SUBSTRING,UCASE

**8.2) ResultSetMetaData:-** It is an interface present in java.sql package.Driver software vendor is responsible to provide implementation. It provides information about data base table represented by ResultSet object.The information is no.of column,column name,column type,precision , scale , Nullable column or not,Auto incremental or not …etc.

We can get ResultSetMetaData object by using getMetaData() method of ResultSet interface.





Ex:-

import java.sql.DriverManager;

import java.sql.Connection;

import java.sql.Statement;

import java.sql.ResultSet;

import java.sql.SQLException;

import java.sql.ResultSetMetaData;

class Type4

{

public static void main(String arg[])

{

try

{

Connection con=DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:ORCL","scott","sukumar");

Statement s=con.createStatement();

ResultSet r1=s.executeQuery("select \* from sukumar");

ResultSetMetaData r=r1.getMetaData();

int a=r.getColumnCount();

System.out.println("Total no.ofcolumns in ResultSet:"+a);

for(int i=1;i<=a;i++)

{

System.out.println("Column number:"+ i);

System.out.println("Column Name:"+r.getColumnName(i));

System.out.println("Column type:"+ r.getColumnType(i));

System.out.println("Column Size:"+r.getColumnDisplaySize(i));

System.out.println(" ----------------------------------------");

}

}

catch(Exception e){System.out.println(e.getMessage());}

}

}

Output:

---------

D:\>java Type4

Total no.ofcolumns in ResultSet:2

Column number:1

Column Name:SNO

Column type:2

Column Size:22

----------------------------------------

Column number:2

Column Name:SNAME

Column type:12

Column Size:10

----------------------------------------

**8.3. Parameter MetaData:-** It gives more information about the (?) param which is placed in SQL query of Prepared Statement object or Callable Statement object.

The information is parameter type, parameter count, patameter mode(in,out,inout), parameter nullable or not, parameter precesion, parameter scale ,…etc.

Most of the JDBC Driver s/w s are not providing implementation class to patameterMetaData Interface. So we get UNSUPPORTED EXCEPTION in most of the JDBC drivers.

We get parameter MetaData object using getParameterMetaData() method of PreparedStatement or Callable Statement.

