```
import java.sql.*;
import java.util.*;
 class DBMD1
{
   public static void main(String[] args) throws Exception
   {
      String dburl="jdbc:mysql://localhost:3306/lju";
      String dbuser="root";
      String dbpass="";
      String driverName="com.mysql.cj.jdbc.Driver";
      //Step:1: Load Driver
      Class.forName(driverName);
      //Step:2:Connect with db
      Connection con=DriverManager.getConnection(dburl,dbuser,dbpass);
      //check is connection Establihed?
      if(con!=null)
      {
         System.out.println();
         System.out.println("Connection Success");
         System.out.println();
      }
      else
      {
         System.out.println("Connection failed");
      }

      DatabaseMetaData dbmd=con.getMetaData();

      /*(1) String getDatabaseProductName() throws SQLException;*/
      // @return database product name

      System.out.println(dbmd.getDatabaseProductName());//MySQL

      /*(2) String getDatabaseProductVersion() throws SQLException;*/
      //@return database version number

      System.out.println(dbmd.getDatabaseProductVersion());
      //5.5.5-10.4.28-MariaDB
      System.out.println(dbmd.getDatabaseMajorVersion());
      //5
      System.out.println(dbmd.getDatabaseMinorVersion());
      //5
```

/*(3) int getMaxColumnsInTable() throws SQLException;*/
// @return the maximum number of columns allowed;

System.out.println(dbmd.getMaxColumnsInTable());
//512

/*(4) boolean supportsStoredProcedures() throws SQLException; */
//@return {@code true} if so; {@code false} otherwise

System.out.println(dbmd.supportsStoredProcedures());
//true

/*(5) String getURL() throws SQLException; */
// @return the URL for this DBMS or {@code null} if it cannot be generated

System.out.println(dbmd.getURL());
//jdbc:mysql://localhost:3306/lju

/*(6) String getDriverName() throws SQLException;*/
//@return JDBC driver name

System.out.println(dbmd.getDriverName());
//MySQL Connector/J

/*(7) String getDriverVersion() throws SQLException;*/
//@return JDBC driver version

System.out.println(dbmd.getDriverVersion());
//mysql-connector-j-8.0.32
System.out.println(dbmd.getDriverMajorVersion());
//8
System.out.println(dbmd.getDriverMinorVersion());
//0

/*(8)String getUserName() throws SQLException; */
// @return the database user name

System.out.println(dbmd.getUserName());
//root@localhost

```
        //Get Table Name
        //returns name of tabels made by user an phpadmin
        //Note:Some databases may not return information for all tables.

        String table[]={"TABLE"};
        ResultSet rs=dbmd.getTables(null,null,null,table);
        while(rs.next())
        {
           System.out.println("Table Name="+rs.getString(3));
        }

    }

}
```

```
import java.sql.*;
import java.util.*;
//--> all code same till connection
//
class RSMD
{
    public static void main(String[] args) throws Exception
    {
        String dburl="jdbc:mysql://localhost:3306/lju";
        String dbuser="root";
        String dbpass="";
        String driverName="com.mysql.cj.jdbc.Driver";
        //Step:1: Load Driver
        Class.forName(driverName);
        //Step:2:Connect with db
        Connection con=DriverManager.getConnection(dburl,dbuser,dbpass);
        //check is connection Establihed?
        if(con!=null)
        {
           System.out.println();
           System.out.println("Connection Success");
           System.out.println();
        }
        else
        {
           System.out.println("Connection failed");
```

```
        }
        String sql="select* from faculty";
        PreparedStatement pst=con.prepareStatement(sql);
        ResultSet rs=pst.executeQuery();
        ResultSetMetaData rsmd=rs.getMetaData();
        System.out.println("Tabe Name="+rsmd.getTableName(1));
        System.out.println("total columns="+rsmd.getColumnCount());
        System.out.println("1st colName="+rsmd.getColumnName(1));
        System.out.println("1st col type="+rsmd.getColumnTypeName(1));
        System.out.println("2nd colName="+rsmd.getColumnName(2));
        System.out.println("2nd col type="+rsmd.getColumnTypeName(2));

          System.out.println(rsmd.getCatalogName(1));//lju

        System.out.println(rsmd.getColumnClassName(1));//java.lang.Integer
        System.out.println(rsmd.getColumnClassName(2));//java.lang.String
        System.out.println(rsmd.getColumnClassName(3));//java.lang.Float

        System.out.println(rsmd.getColumnLabel(1));//facId
        System.out.println(rsmd.getColumnLabel(2));//facName

        System.out.println(rsmd.getColumnDisplaySize(1));//10
        //return the normal maximum number of characters allowed as the width
    }

}
```

---

charFileStoring in DataBase

Table structure | Relation view

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	file_Id	int(10)			No	*None*		AUTO_INCREMENT	Change Drop More
2	file_Name	varchar(20)	utf8mb4_general_ci		No	*None*			Change Drop More
3	file_Content	longtext	utf8mb4_general_ci		No	*None*			Change Drop More

```
import java.sql.*;
import java.io.*;
import java.util.*;
 class FileStorageEx
{
   public static void main(String[] args) throws Exception
   {
      String dburl="jdbc:mysql://localhost:3306/lju";
      String dbuser="root";
      String dbpass="";
      String driverName="com.mysql.cj.jdbc.Driver";
      //Step:1: Load Driver
      Class.forName(driverName);
      //Step:2:Connect with db
      Connection con=DriverManager.getConnection(dburl,dbuser,dbpass);
      //check is connection Establihed?
      if(con!=null)
      {
         System.out.println();
         System.out.println("Connection Success");
         System.out.println();
      }
      else
      {
         System.out.println("Connection failed");
      }
      String sql="insert into charfileStorage(file_Name,file_Content) values(?,?)";

      PreparedStatement pst=con.prepareStatement(sql);

      pst.setString(1,"Test Char File");

      File f=new File("D://p1.java");
      FileReader fr=new FileReader(f);

      pst.setCharacterStream(2,fr,f.length());

      int r=pst.executeUpdate();
      if (r>0)
      {
         System.out.println();
```

```
            System.out.println("File Storage Success");
            System.out.println();
        }
        else
        {
            System.out.println("File Storage failed");
        }

    }
}
```

```
import java.sql.*;
import java.io.*;
import java.util.*;
class FilewRetrival
{
    public static void main(String[] args) throws Exception
    {
        String dburl="jdbc:mysql://localhost:3306/lju";
        String dbuser="root";
        String dbpass="";
        String driverName="com.mysql.cj.jdbc.Driver";
        //Step:1: Load Driver
        Class.forName(driverName);
        //Step:2:Connect with db
        Connection con=DriverManager.getConnection(dburl,dbuser,dbpass);
        //check is connection Establihed?
        if(con!=null)
        {
            System.out.println();
            System.out.println("Connection Success");
            System.out.println();
        }
        else
        {
            System.out.println("Connection failed");
        }

        String sql="select * from charfilestorage";
        PreparedStatement pst=con.prepareStatement(sql);
        ResultSet rst=pst.executeQuery();
        while(rst.next())
        {
```

```
            String fileName=rst.getString("file_Name");

            Clob fClob=rst.getClob("file_Content");

            Reader r=fClob.getCharacterStream();

            FileWriter fw=new FileWriter("D://"+fileName+".txt");
            int i;
            while((i=r.read())!=-1)
            {
               fw.write((char)i);
            }
           System.out.println("File stored in D: drive with Name="+fileName);
            r.close();
            fw.close();
        }
    }
}
```

BinaryFileStorage

	#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
☐	1	fileId	int(10)			No	None		AUTO_INCREMENT	Change Drop More
☐	2	fileName	varchar(50)	utf8mb4_general_ci		No	None			Change Drop More
☐	3	fileSizeKb	bigint(50)			No	None			Change Drop More
☐	4	fileExtension	varchar(50)	utf8mb4_general_ci		No	None			Change Drop More
☐	5	fileContent	longblob			No	None			Change Drop More

//Write a program that is used to store audio , video,image
//file in Dastabase using java

```
import java.sql.*;
import java.io.*;
import java.util.*;
 class BinaryStorage
{
   public static void main(String[] args) throws Exception
   {
```

```
String dburl="jdbc:mysql://localhost:3306/lju";
String dbuser="root";
String dbpass="";
String driverName="com.mysql.cj.jdbc.Driver";
//Step:1: Load Driver
Class.forName(driverName);
//Step:2:Connect with db
Connection con=DriverManager.getConnection(dburl,dbuser,dbpass);
//check is connection Establihed?
if(con!=null)
{
   System.out.println();
   System.out.println("Connection Success");
   System.out.println();
}
else
{
   System.out.println("Connection failed");
}
```

String sql="insert into bfstoage (fileName,fileSizeKb,fileExtension,fileContent) values(?,?,?,?)";

```
PreparedStatement pst=con.prepareStatement(sql);
```

File f=new File("D://AP.JPG");

String fileName=f.getName();
Long fileLength=f.length();
long fileSizeKb=(fileLength/1024);
String fileExtension=fileName.substring(fileName.lastIndexOf("."));

pst.setString(1,fileName);
pst.setLong(2,fileSizeKb);
pst.setString(3,fileExtension);

```
FileInputStream fis=new FileInputStream(f);
```

pst.setBinaryStream(4,fis);

```
int r=pst.executeUpdate();
if (r>0)
{
   System.out.println();
```

```
            System.out.println("Insert Success");
            System.out.println();
        }
        else
        {
            System.out.println("Insert failed");
        }
    }
}
```

```
import java.sql.*;
import java.io.*;
import java.util.*;

class BinaryFileRetival
{
    public static void main(String[] args) throws Exception
    {
        String dburl="jdbc:mysql://localhost:3306/lju";
        String dbuser="root";
        String dbpass="";
        String driverName="com.mysql.cj.jdbc.Driver";
        //Step:1: Load Driver
        Class.forName(driverName);
        //Step:2:Connect with db
        Connection con=DriverManager.getConnection(dburl,dbuser,dbpass);
        //check is connection Establihed?
        if(con!=null)
        {
            System.out.println();
            System.out.println("Connection Success");
            System.out.println();
        }
        else
        {
            System.out.println("Connection failed");
        }

        String sql="select * from bfstoage";

        PreparedStatement pst=con.prepareStatement(sql);
        ResultSet rs=pst.executeQuery();
```

```
        while(rs.next())
        {
           //First extract blob obj from table
           Blob b=rs.getBlob("fileContent");

           //Convert blob obj to byte[]
           byte [] arr=b.getBytes(1,(int)b.length());

           //create file with new name and write
           String fileName="D://retrived"+rs.getString("fileName");

           FileOutputStream fos= new FileOutputStream(fileName);
           System.out.println(fileName+" is FileRetrived");
           fos.write(arr);
           fos.close();
        }
     }

}
```