Q1.Wap to establish simple connection with database using JDBC.

import java.sql.\*;

public class Establishingconnection{

private static final String url="jdbc:mysql://localhost:3306/";

private static final String username="root";

private static final String password="Shiv@#1511";

public static void main(String args[]) {

try{

Class.forName("com.mysql.cj.jdbc.Driver");

}catch(ClassNotFoundException e){

System.out.println(e.getMessage());

}

try{

Connection conn=DriverManager.getConnection(url,username,password);

System.out.println("connection established ");

System.out.println("Shivkant pandey ");

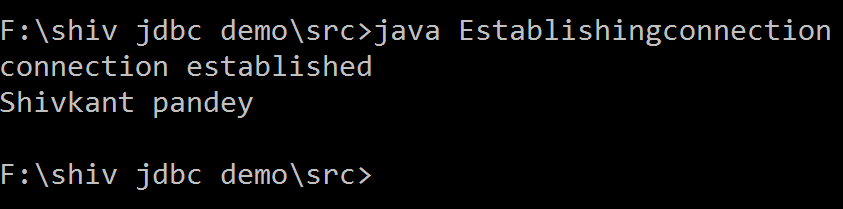
}catch(SQLException f){

System.out.println(f.getMessage());

}

}

}



Q2.Wap to retrieve data from database table using JDBC.

import java.sql.\*;

public class Retrievingdata{

public static void main(String args[]){

try{

Class.forName("com.mysql.cj.jdbc.Driver");

}catch(Exception e){

System.out.println(e.getMessage());}

String url="jdbc:mysql://localhost:3306/shivjdbc";

String username="root";

String password="Shiv@#1511";

try{

Connection con=DriverManager.getConnection(url,username,password);

Statement s=con.createStatement();

ResultSet rs=s.executeQuery("Select \* from students");

while(rs.next()){

int id=rs.getInt("id");

String name=rs.getString("name");

int age=rs.getInt("age");

double marks=rs.getDouble("marks");

System.out.println("ID : "+id);

System.out.println("NAME : "+name);

System.out.println("AGE : "+age);

System.out.println("MARKS : "+marks);}

}catch(Exception e){

System.out.println(e.getMessage());

}

}

}

A screen shot of a computer

Description automatically generated

Q3.Wap to create a table in the database using JDBC.

import java.sql.\*;

public class Creatingtable{

public static void main(String args[]){

String url="jdbc:mysql://localhost:3306/shivjdbc";

String username="root";

String password="Shiv@#1511";

try{

Class.forName("com.mysql.cj.jdbc.Driver");

}catch(Exception e){

System.out.println(e.getMessage());

}

try{

Connection c=DriverManager.getConnection(url,username,password);

Statement s=c.createStatement();

s.executeUpdate("create table students1(id int auto\_increment primary key,name varchar(255) not null,age int not null,marks double not null)");

System.out.println("table created");

}

catch(Exception e){

System.out.println(e.getMessage());

}

}

}

A computer screen shot of a program

Description automatically generated

Q4.Wap to insert data into the table using JDBC.

import java.sql.\*;

public class Insertingdata{

public static void main(String args[]){

try{

Class.forName("com.mysql.cj.jdbc.Driver");

}catch(Exception e){

System.out.println(e.getMessage());

}

String url="jdbc:mysql://localhost:3306/shivjdbc";

String user="root";

String password="Shiv@#1511";

try{

Connection con=DriverManager.getConnection(url,user,password);

Statement s=con.createStatement();

String query=String.format("insert into students(name,age,marks) values('%s',%o,%f)","aqib",23,98.99);

int queryaffected=s.executeUpdate(query);

if (queryaffected>0){

System.out.println("data inserted succesfully ");

}

else{

System.out.println("Data insertion failed");

}

}

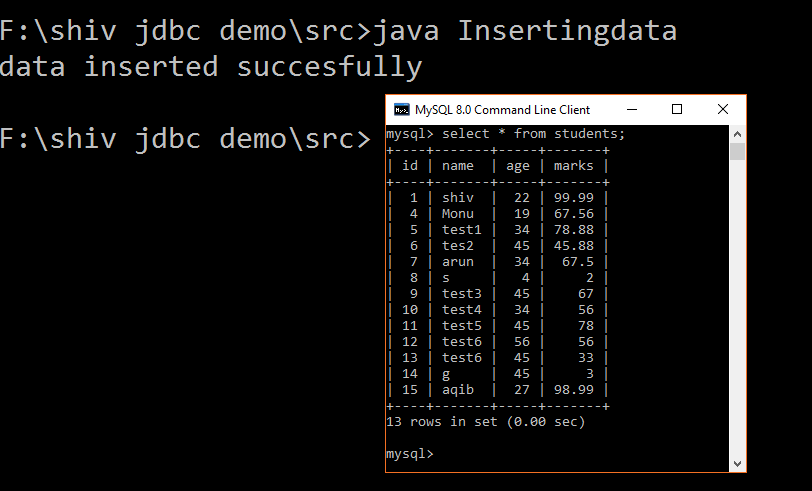
catch(Exception e){

System.out.println(e.getMessage());

}

}

}



Q5.Wap to insert values into the database table using prepared statement.

import java.sql.\*;

import java.util.\*;

public class Preparedinsert{

public static void main(String args[]){

try{

Class.forName("com.mysql.cj.jdbc.Driver");

}catch(Exception e){

System.out.println(e.getMessage());}

String url="jdbc:mysql://localhost:3306/shivjdbc";

String user="root";

String pass="Shiv@#1511";

try{

Connection c=DriverManager.getConnection(url,user,pass);

String query="insert into students(name,age,marks) values(?,?,?)";

PreparedStatement ps=c.prepareStatement(query);

Scanner sc=new Scanner(System.in);

System.out.println("Enter name");

ps.setString(1,sc.nextLine());

System.out.println("Enter age");

ps.setInt(2,sc.nextInt());

System.out.println("Enter marks");

ps.setDouble(3,sc.nextDouble());

int rowsaffect=ps.executeUpdate();

if (rowsaffect>0){System.out.println("Inserted data using prepared");

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

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

}

}

A screen shot of a computer screen

Description automatically generated

Q6.Wap to insert the values into the table using the prepared statement (loop).

import java.sql.\*;

import java.util.\*;

public class Preparedinsertloop{

public static void main(String args[]){

try{

Class.forName("com.mysql.cj.jdbc.Driver");

}

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

String url="jdbc:mysql://localhost:3306/shivjdbc";

String user="root";

String pass="Shiv@#1511";

try{

Connection c=DriverManager.getConnection(url,user,pass);

String query="insert into students(name,age,marks) values(?,?,?)";

PreparedStatement ps=c.prepareStatement(query);

do{

Scanner sc=new Scanner(System.in);

System.out.println("Enter name");

ps.setString(1,sc.nextLine());

System.out.println("Enter age");

ps.setInt(2,sc.nextInt());

System.out.println("Enter marks");

ps.setDouble(3,sc.nextDouble());

int rowsaffect=ps.executeUpdate();

if (rowsaffect>0){

System.out.println("Inserted data using prepared");

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

}

Scanner sx=new Scanner(System.in);

System.out.println("Do you want to insert more rows :(y/n)");

String yn=sx.nextLine();

if (yn.startsWith("n")){break;}}

while(true);

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

}

}

A screenshot of a computer screen

Description automatically generated

A screenshot of a computer program

Description automatically generated

Q7.Wap to show the use of stored procedure.

import java.sql.\*;

public class Callingstoreprocedure{

public static void main(String args[]){

try{

Class.forName("com.mysql.cj.jdbc.Driver");

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

String url="jdbc:mysql://localhost:3306/shivjdbc";

String user="root";

String pass="Shiv@#1511";

try{

Connection c=DriverManager.getConnection(url,user,pass);

CallableStatement cb=c.prepareCall("{call sp()}");

cb.execute();

ResultSet rs=cb.getResultSet();

System.out.println("Student with lowest marks 60 ");

while(rs.next()){

int id=rs.getInt("id");

String name=rs.getString("name");

int age=rs.getInt("age");

double marks=rs.getDouble("marks");

System.out.println("ID : "+id);

System.out.println("NAME : "+name);

System.out.println("AGE : "+age);

System.out.println("MARKS : "+marks);}

cb.close();

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

}

}

A screen shot of a computer

Description automatically generated

A screen shot of a computer

Description automatically generated

Q8.Wap to scroll the data of a table using scrollable resultset.

import java .sql.\*;

public class Scrollableresultset{

public static void main(String args[]){

try{

Class.forName("com.mysql.cj.jdbc.Driver");

}catch(ClassNotFoundException e){

System.out.println(e.getMessage());

}

String url="jdbc:mysql://localhost:3306/shivjdbc";

String user="root";

String pass="Shiv@#1511";

try(Connection c=DriverManager.getConnection(url,user,pass)){

String sql="Select \* from students";

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

ResultSet rs=s.executeQuery(sql);

rs.first();

readsinfo("first",rs);

rs.relative(3);

readsinfo("relative(3)",rs);

rs.previous();

readsinfo("previous",rs);

rs.absolute(4);

readsinfo("absolute(4)",rs);

rs.last();

readsinfo("last",rs);

rs.relative(-2);

readsinfo("relative(-2)",rs);

}catch(SQLException e){

e.printStackTrace();

}

}

private static void readsinfo(String position,ResultSet rs){

try{

int id=rs.getInt("id");

String name=rs.getString("name");

int age=rs.getInt("age");

double marks=rs.getInt("marks");

String sinfo="%s: %d-%s-%d-%f\n";

System.out.format(sinfo,position,id,name,age,marks);

}catch(SQLException f){

System.out.println(f.getMessage());

}

}

}

A screenshot of a computer program

Description automatically generated

Q9.Wap to insert , update and delete the data in a row of a table using updatable resultset.

import java.sql.\*;

import java.io.\*;

import java.util.\*;

public class Updatableresultset{

public static void main(String args[]){

try{

Class.forName("com.mysql.cj.jdbc.Driver");

}catch(Exception e){

System.out.println(e.getMessage());}

String url="jdbc:mysql://localhost:3306/shivjdbc";

String user="root";

String pass="Shiv@#1511";

try(Connection c=DriverManager.getConnection(url,user,pass)){

Console console = System.console();

DatabaseMetaData md=c.getMetaData();

boolean isUpdatable=md.supportsResultSetConcurrency(

ResultSet.TYPE\_SCROLL\_INSENSITIVE,ResultSet.CONCUR\_UPDATABLE);

if(!isUpdatable){

System.out.println("The database not supports updatable");

}

String sql="select \* from students";

Statement s=c.createStatement(ResultSet.TYPE\_SCROLL\_INSENSITIVE,ResultSet.CONCUR\_UPDATABLE);

ResultSet rs=s.executeQuery(sql);

int row=-1;

while(row!=0){

row=Integer.parseInt(console.readLine("Enter row number : "));

if (rs.absolute(row)){

readsinfo("student at row "+row+":",rs);

String ans=console.readLine("Do you want to update this row(Y/N):");

if(ans.toUpperCase().equals("Y")){

String name=console.readLine("\tUpdate name:");

int age=Integer.parseInt(console.readLine("\tUpdate age:"));

double marks=Double.parseDouble(console.readLine("\tUpdate marks:"));

rs.updateRow();

System.out.println("The student record has been updated");

}

ans = console.readLine("Do you want to delete this row(Y/N)?: ");

if (ans.equalsIgnoreCase("Y")) {

rs.deleteRow();

System.out.println("The student at row " + row + " has been deleted.");

}

ans =console.readLine("Do you want to insert new row (Y/N):");

if(ans.toUpperCase().equals("Y")){

rs.moveToInsertRow();

String name=console.readLine("\tUpdate name:");

int age=Integer.parseInt(console.readLine("\tUpdate age:"));

double marks=Double.parseDouble(console.readLine("\tUpdate marks:"));

rs.updateString("name",name);

rs.updateInt("age",age);

rs.updateDouble("marks",marks);

rs.moveToCurrentRow();

System.out.println("The student record has been inserted");

}

}

else {

System.out.println("There is no student at row " + row);

}

}

} catch (SQLException ex) {

ex.printStackTrace();

}

}

private static void readsinfo(String position, ResultSet rs){

try{

String name=rs.getString("name");

int age=rs.getInt("age");

double marks=rs.getDouble("marks");

String studentinfo="%s:%s-%d-%f\n";

System.out.format(studentinfo,position,name,age,marks);

}catch(SQLException e){

System.out.println(e.getMessage());

}

}

}

A screenshot of a computer program

Description automatically generated

Q10.Wap to update the age of students using updatable resultset.

import java.sql.\*;

public class Updatecolumn{

public static void main(String args[]){

try{

Class.forName("com.mysql.cj.jdbc.Driver");

}catch(Exception e){

System.out.println(e.getMessage());}

String url="jdbc:mysql://localhost:3306/shivjdbc";

String user="root";

String pass="Shiv@#1511";

try{

Connection connection=DriverManager.getConnection(url,user,pass);

Statement statement = connection.createStatement(ResultSet.TYPE\_SCROLL\_INSENSITIVE,ResultSet.CONCUR\_UPDATABLE);

String sql = "Select \* from students";

ResultSet results = statement.executeQuery(sql);

System.out.println("Details of the students: ");

results.beforeFirst();

while(results.next()){

System.out.println("id"+results.getInt("id"));

System.out.println(",name: "+results.getString("name"));

System.out.println(",age: "+results.getInt("age"));

System.out.println(",marks: "+results.getDouble("marks"));

}

System.out.println();

results.beforeFirst();

while(results.next()){

int newage = results.getInt("age")+ 5;

results.updateInt("age",newage);

results.updateRow();

}

System.out.println("Details of the students after increasing age");

results.beforeFirst();

while(results.next()){

System.out.println("id"+results.getInt("id"));

System.out.println(",name: "+results.getString("name"));

System.out.println(",age: "+results.getInt("age"));

System.out.println(",marks: "+results.getDouble("marks"));

}

System.out.println();

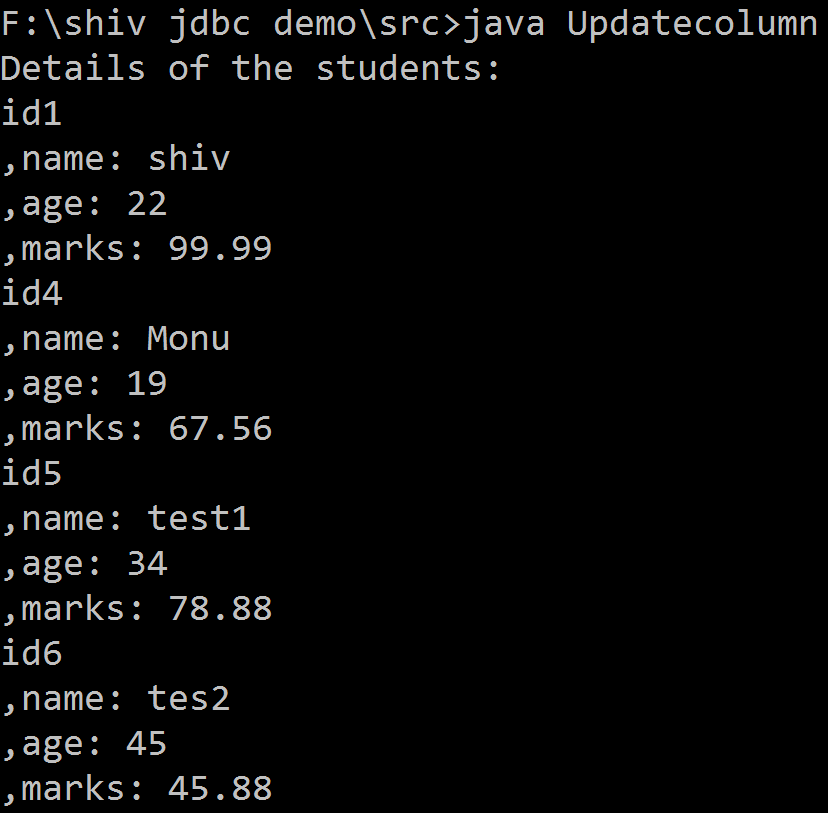
}

catch (SQLException e){System.out.println(e);

}

}

}



A screenshot of a computer

Description automatically generated

Q11.Wap to create table in the database that can store image as one of its attributes using jdbc and show how image can be stored.

import java.io.File;

import java.io.FileInputStream;

import java.io.InputStream;

import java.sql.\*;

public class Photodb{

public static void main(String[]args){

String url = "jdbc:mysql://localhost:3306/shivjdbc";

String user = "root";

String password = "Shiv@#1511";

String filepath ="C://Users//shiv//Downloads//12.jpg";

try(Connection conn =DriverManager.getConnection(url,user,password)){

String sql = "Insert into photos(photoname,photo) values(?,?)";

PreparedStatement pstmt = conn.prepareStatement(sql);

pstmt.setString(1,"Shiv");

InputStream inputStream = new FileInputStream(new File(filepath));

pstmt.setBlob(2,inputStream);

int row = pstmt.executeUpdate();

if (row>0){

System.out.println("done");

}

}catch(Exception e){

System.out.println(e);

} } }

A black screen with white text

Description automatically generated

A screenshot of a computer

Description automatically generated

Q12.Wap to create a connection class and connection method which can be used by other JDBC file to establish connection to database.

import java.sql.\*;

public class DBUtil{

private static final String url ="jdbc:mysql://localhost:3306/shivjdbc";

private static final String user = "root";

private static final String password = "Shiv@1511";

private static Connection connection = null;

static {

try {

connection = DriverManager.getConnection(url, user, password);

} catch (Exception e) {

e.printStackTrace();

}

}

public static Connection getConnection(){

return connection;

}

}

A black background with white text

Description automatically generated

Q13.Wap to create table which can store text file with file name ,size,extension and then store data in it using JDBC.

import java.io.File;

import java.io.FileReader;

import java.io.IOException;

import java.nio.file.Files;

import java.nio.file.Path;

import java.nio.file.Paths;

import java.sql.Connection;

import java.sql.DriverManager;

import java.sql.PreparedStatement;

import java.sql.SQLException;

import java.util.List;

import java.util.stream.Collectors;

import java.util.stream.Stream;

public class Filesstorer {

public static void main(String[] args) {

String SQL = "INSERT INTO filewarehouse(fname,fsizeKB,fext,fcontent)values(?,?,?,?)";

Path dir = Paths.get("F://uploading tofile warehouse//");

String url="jdbc:mysql://localhost:3306/shivjdbc";

String user="root";

String pass="Shiv@#1511";

try(Stream <Path> list = Files.list(dir);

Connection connection = DriverManager.getConnection(url,user,pass);

PreparedStatement ps =connection.prepareStatement(SQL)){

List <Path> PathList = list.collect(Collectors.toList());

System.out.println("Following files are saved in database");

for(Path path : PathList){

System.out.println(path.getFileName());

File file = path.toFile();

String fileName = file.getName();

long fileLength = file.length();

long fileLengthinkb = fileLength/1024;

ps.setString(1, fileName);

ps.setLong(2, fileLengthinkb);

ps.setString(3, fileName.substring(fileName.lastIndexOf(" ")+ 1 ));

ps.setCharacterStream(4, new FileReader(file), fileLength);

ps.addBatch();

}

int [] executeBatch = ps.executeBatch();

for(int i : executeBatch){

System.out.println(i);

}

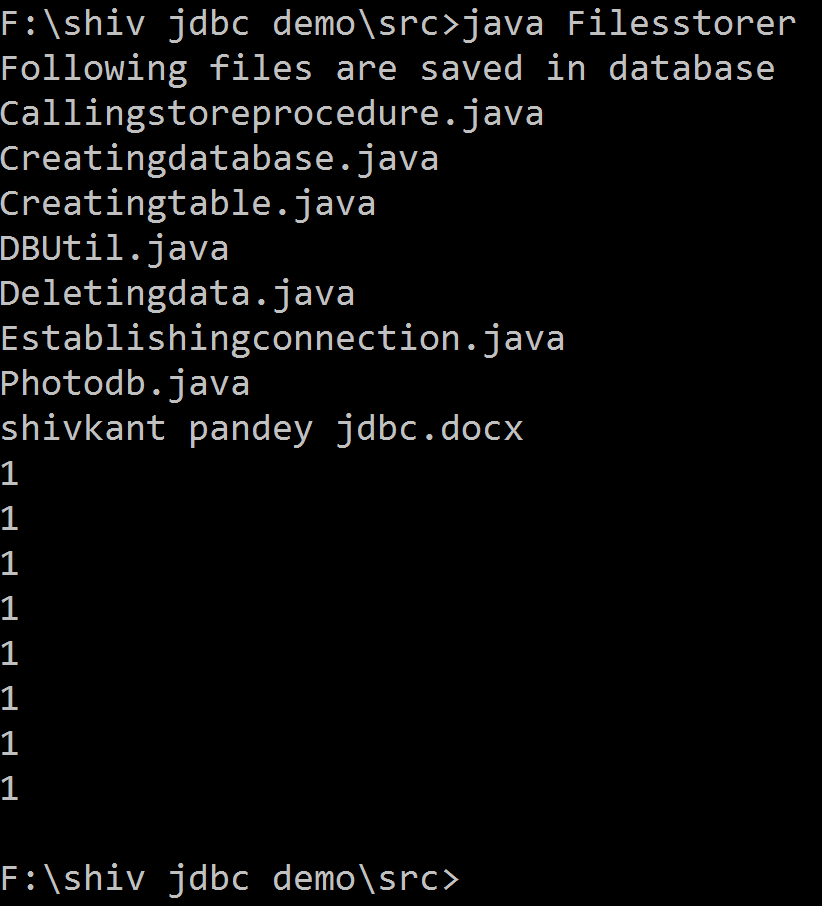
}

catch(Exception e){

e.printStackTrace();

}

} }



Q14.Wap to download multimedia files from a table in the database using JDBC.

import java.io.IOException;

import java.io.InputStream;

import java.nio.file.Files;

import java.nio.file.Paths;

import java.sql.Blob;

import java.sql.Connection;

import java.sql.DriverManager;

import java.sql.PreparedStatement;

import java.sql.ResultSet;

import java.sql.SQLException;

public class DownloadMultiMediaFiles{

public static void main(String[] args) {

try{

downloadBinaryFilesFromDatabase();

}catch(SQLException e){

System.out.println(e.getMessage());

}

}

private static void downloadBinaryFilesFromDatabase() throws SQLException {

String SQL = "SELECT \* FROM filewarehouse";

String url="jdbc:mysql://localhost:3306/shivjdbc";

String user="root";

String pass="Shiv@#1511";

try(Connection connection = DriverManager.getConnection(url,user,pass);

PreparedStatement ps = connection.prepareStatement(SQL);

ResultSet rs = ps.executeQuery()){

System.out.println("Following files are downloaded from databse. ");

while(rs.next()){

int fileId = rs.getInt("fileid");

String fileName = rs.getString("fname");

long filesizeinKb = rs.getLong("fsizeKB");

String fileExtension = rs.getString("fext");

System.out.println("File Extension: " + fileExtension);

Blob blob = rs.getBlob("fcontent");

InputStream inputStream = blob.getBinaryStream();

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

Files.copy(inputStream, Paths.get("F://download folder" + fileName));

}

}

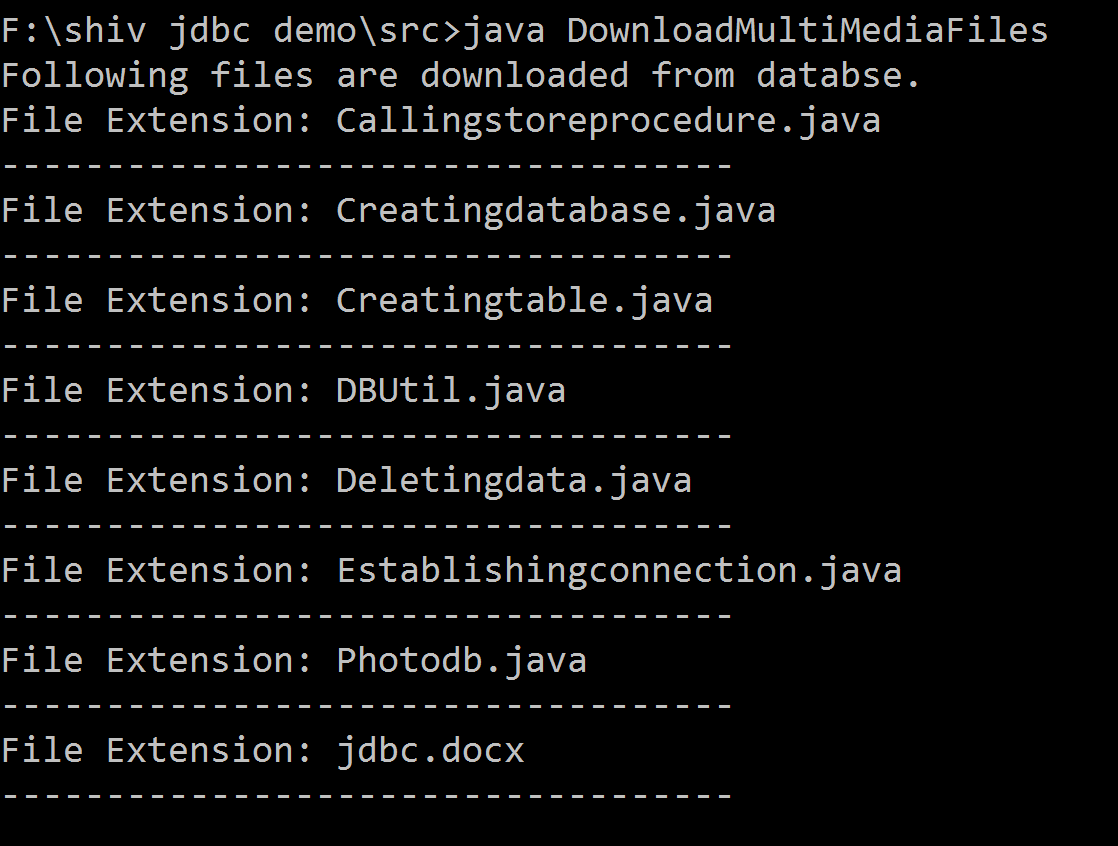
catch(IOException e){

e.printStackTrace();

}

}

}



Q15.Wap to create store procedure in the database using JDBC.

import java.sql.\*;

public class StoredProcedure{

public static void main(String[]args){

String url = "jdbc:mysql://localhost:3306/shivjdbc";

String user = "root";

String password = "Shiv@#1511";

try(Connection conn = DriverManager.getConnection(url,user,password)){

Statement statement = conn.createStatement();

String queryCreate = "Create procedure AddTwoNumbers(IN num1 INT,IN num2 INT, OUT result INT)";

queryCreate+= "Begin ";

queryCreate+=" SET result = num1 + num2; ";

queryCreate+="End";

statement.execute(queryCreate);

System.out.println("created");

}catch(Exception e){

System.out.println(e);

}

}

}

A black background with white text

Description automatically generated