DEVELOPER'S PORTFOLIO

TASK - 5

INTRODUCTION TO DEVELOPER'S PORTFOLIO

Developers build portfolio websites as full-stack developer sample projects to showcase their skills and impress clients. As a student or professional learning web development, you must practice making portfolio websites to gain knowledge and experience in efficient front end web development technology.

Campus Name :- Ajeenkya DY Patil University Pune
Name :- Devendra Singh
Batch :- Bachelor's In Computer Application (BCA)
Year :- 1st year
URN :- 2022-B-09072004C

ABOUT TASK 5

JDBC CONNECTION

- CREATE DATABASE
- CREATE TABLE
- INSERT DATA INTO TABLE
- UPDATE VALUES OF THE TABLE

JDBC INTRODUCTION

JDBC stands for Java Database Connectivity. It is a Java API that allows Java programs to connect and interact with various databases. With JDBC, Java developers can write database applications that can run on various platforms and interact with different database management systems (DBMS) such as MySQL, Oracle, SQL Server, and many others. The JDBC API provides a set of classes and interfaces that enable Java applications to send SQL statements to a database, execute them, and retrieve the results.

JDBC CONNECTION CODE

IMPORT STATEMENT

TRY AND CATCH

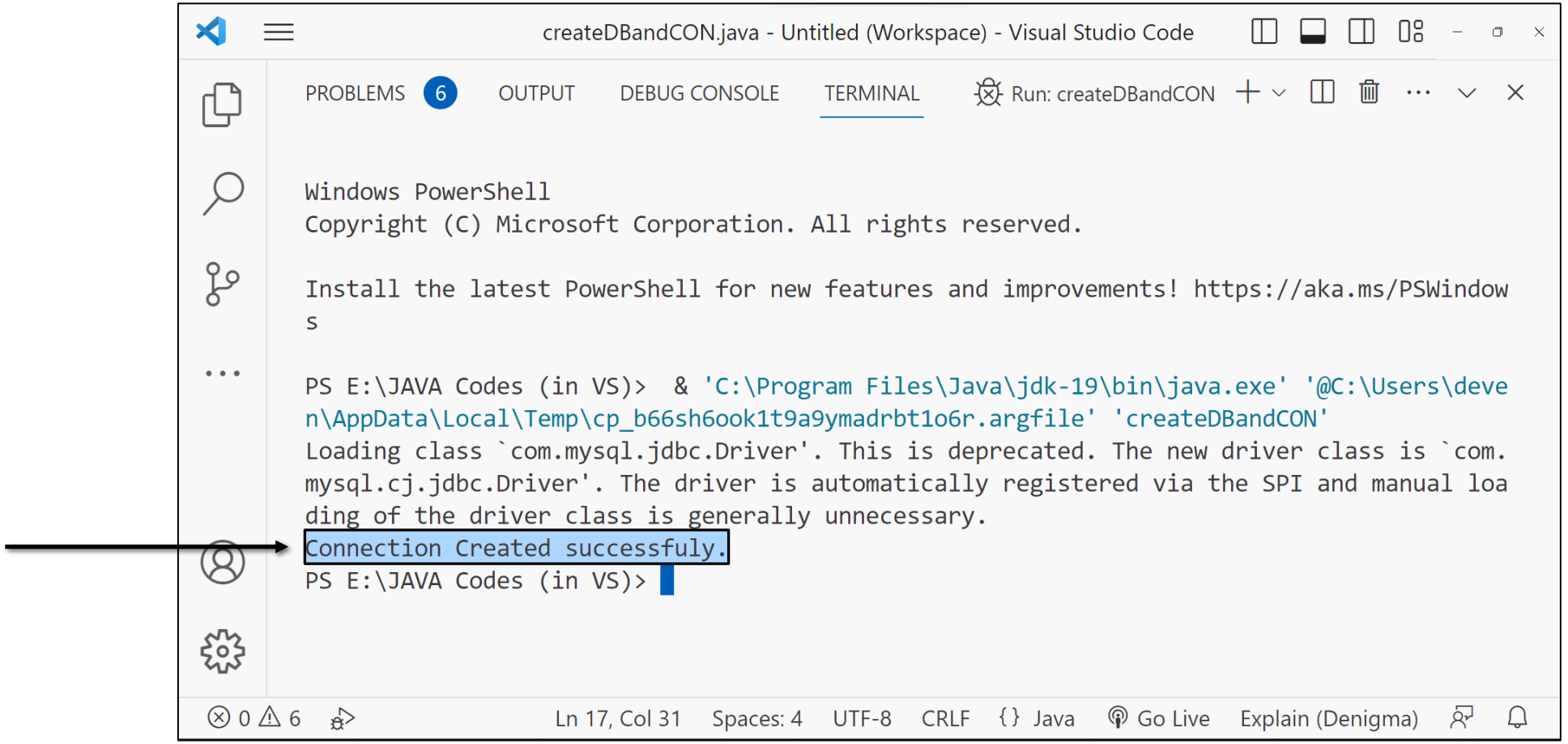
SQL USERNAME

SQL PASSWORD

CONNECTION STATEMENT

```java
import java.sql.*;
public class createDBandCON{
    Run | Debug
    public static void main(String[] args) {
        try{
            Class.forName(className:"com.mysql.jdbc.Driver");
            // Creating a Connection
            String url="jdbc:mysql://localhost:3306";
            String username="root";
            String password="dev@singh7465";
            Connection con=DriverManager.getConnection(url,username,password);
            if(con.isClosed()){
                System.out.println(x:"Connection failed !");
            }
            else{
                System.out.println(x:"Connection Created successfuly.");
            }
            // SQL COMMANDS...
        }
        catch(Exception e){
            e.printStackTrace();
        }
    }
}
```

OUTPUT FOR JDBC CONNECTION CODE



CREATE DATABASE USING JDBC

CREATE DATABASE QUERY

UPDATE STATEMENT

OUTPUT STATEMENT

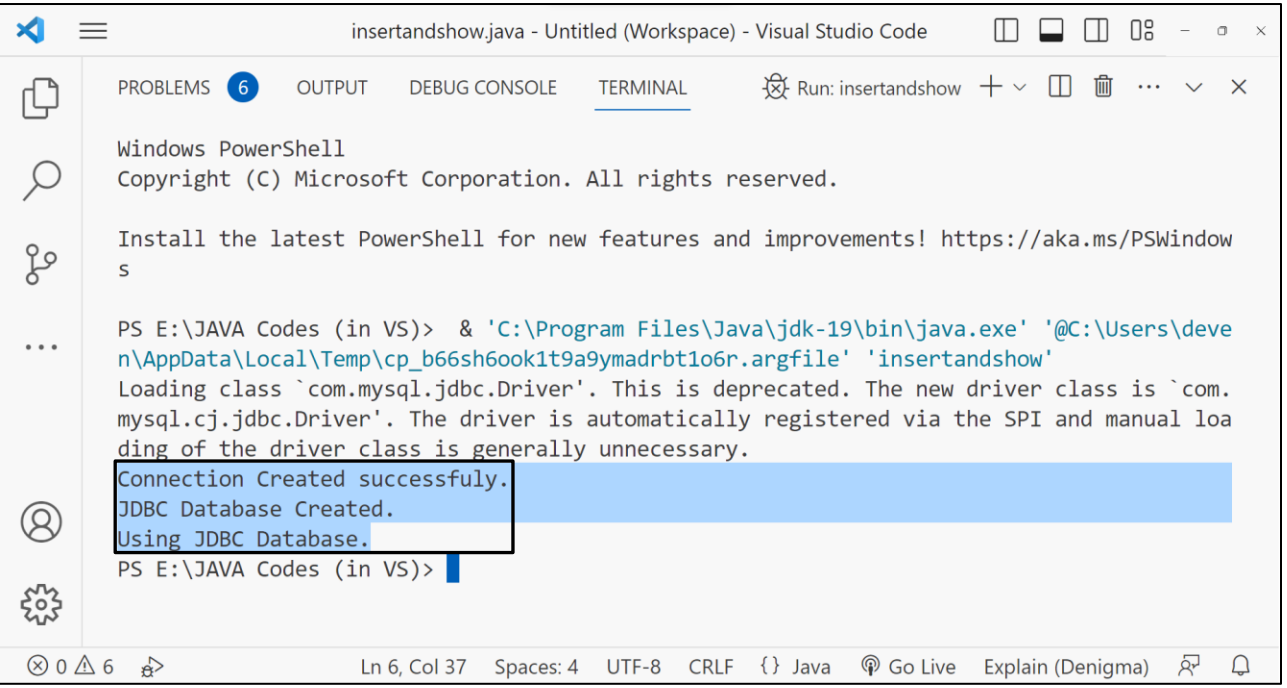
USE DATABASE QUERY

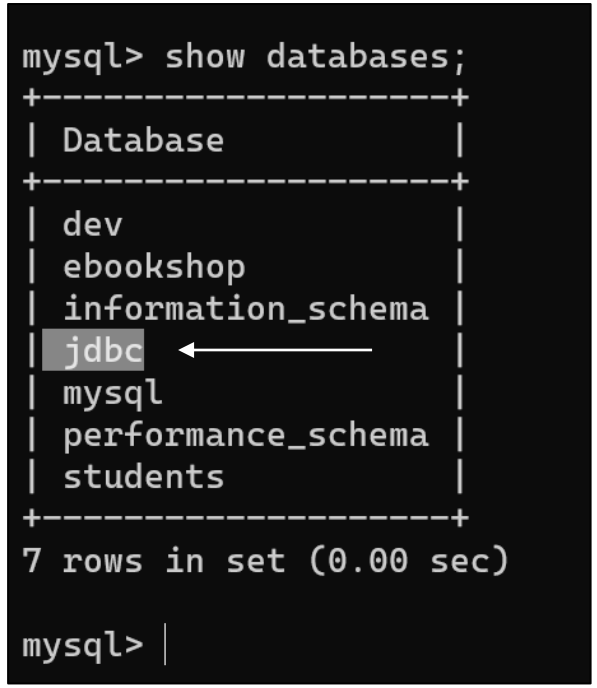
UPDATE STATEMENT

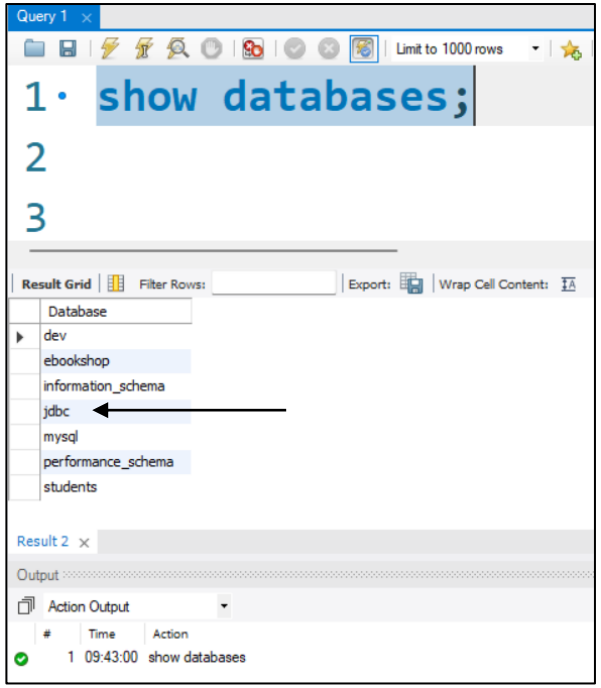
OUTPUT STATEMENT

```
//Creating Database
String createdb = "CREATE DATABASE JDBC";
Statement stmt=con.createStatement();
stmt.executeUpdate(createdb);
System.out.println(x:"JDBC Database Created.");

// USE Database
String usedbquery = "USE JDBC";
Statement stmt1=con.createStatement();
stmt1.executeUpdate(usedbquery);
System.out.println(x:"Using JDBC Database.");
```

OUTPUT FOR CREATE DATABASE CODE







CREATE TABLE USING JDBC

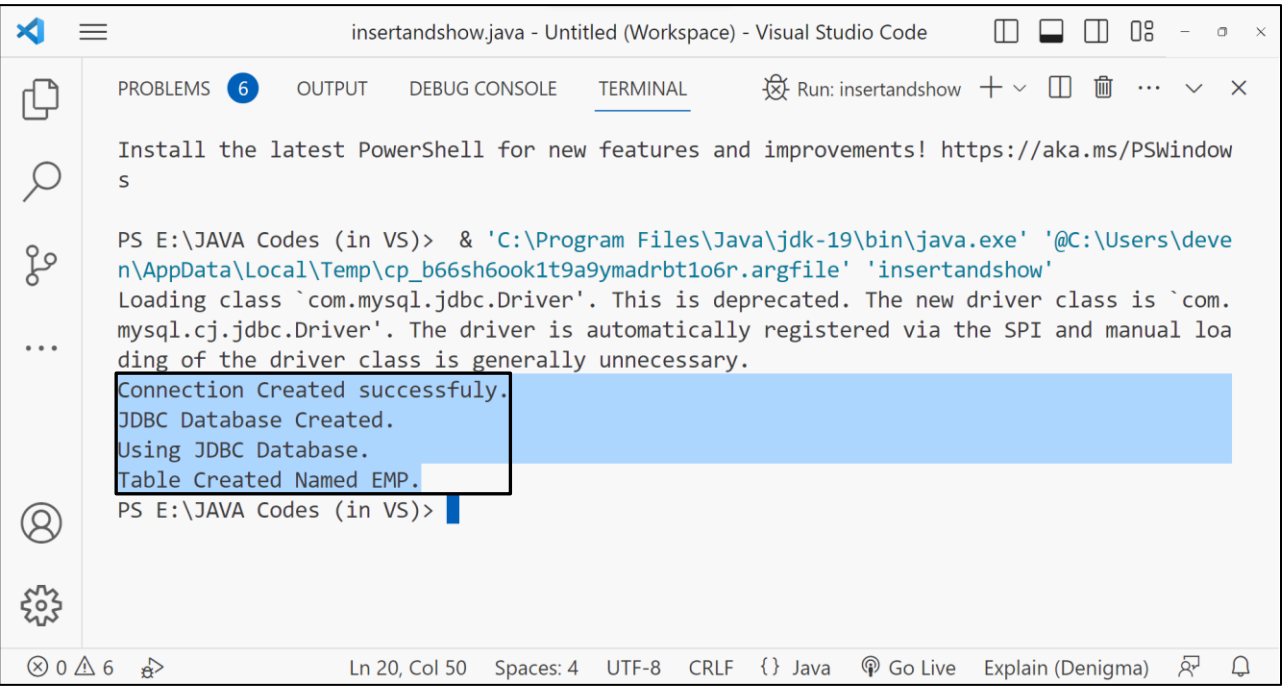
CREATE TABLE QUERY

UPDATE STATEMENT

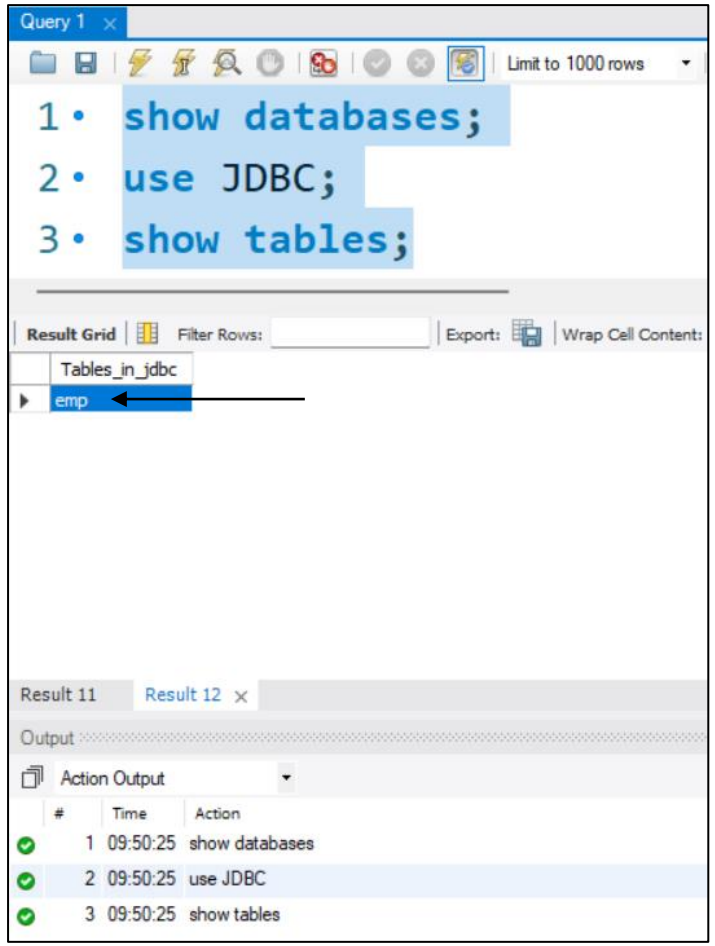
OUTPUT STATEMENT

```
// Creating Table EMP
String createtable = "CREATE TABLE EMP(EMP_ID int(4) ,
EMP_NAME varchar(10))";
Statement stmt2=con.createStatement();
stmt2.executeUpdate(createtable);
System.out.println(x:"Table Created Named EMP.");
```

OUTPUT FOR TABLE CREATION CODE




```
mysql> show databases;
+--------------------+
| Database           |
+--------------------+
| dev                |
| ebookshop          |
| information_schema |
| jdbc               |
| mysql              |
| performance_schema |
| students           |
+--------------------+
7 rows in set (0.00 sec)

mysql> use jdbc;
Database changed
mysql> show tables;
+----------------+
| Tables_in_jdbc |
+----------------+
| emp            |
+----------------+
1 row in set (0.00 sec)
```



INSERT VALUES USING JDBC

INSERT QUERY

RECORD 1

RECORD 2

RECORD 3

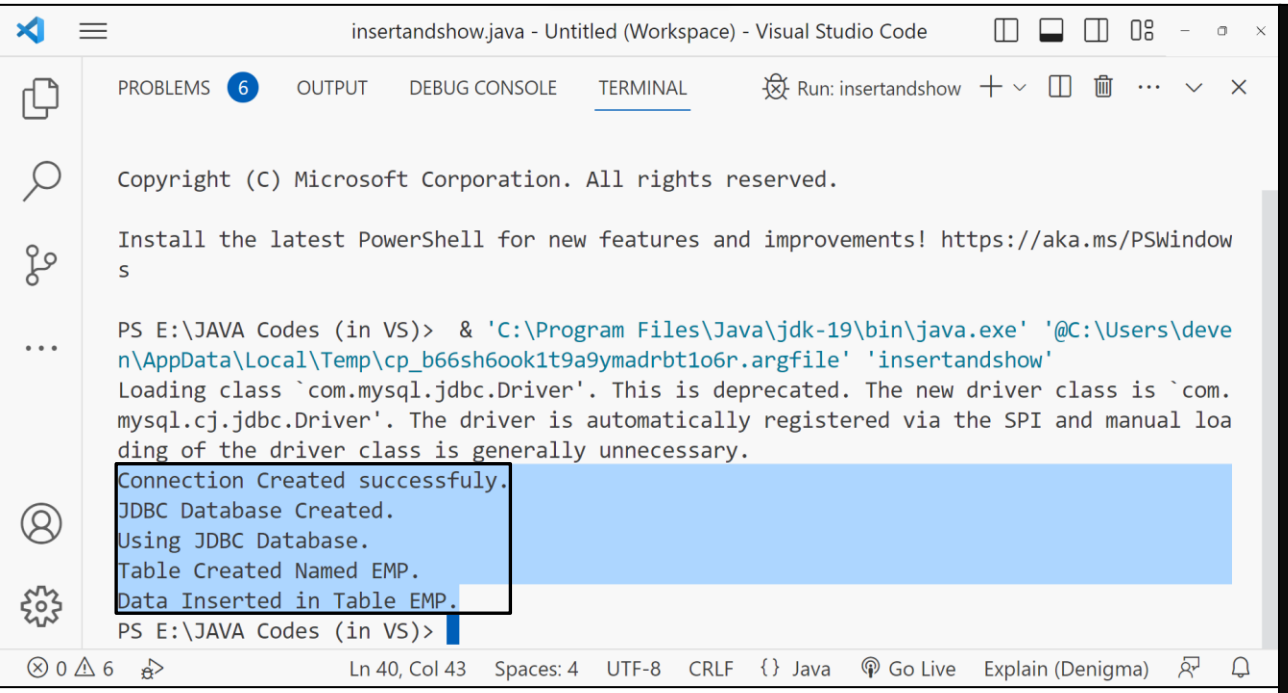
RECORD 4

RECORD 5

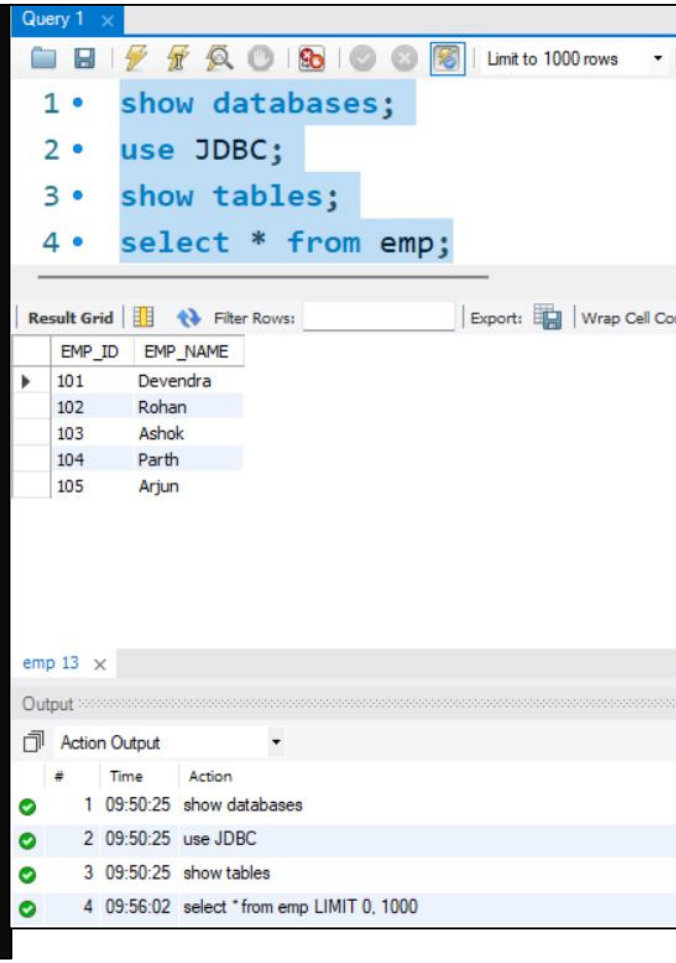
OUTPUT STATEMENT

```
//Insert Command
String insert = "insert into emp(EMP_ID,EMP_NAME) values(?,?)";
PreparedStatement pstmt = con.prepareStatement(insert);
pstmt.setInt(parameterIndex:1,x:101);
pstmt.setString(parameterIndex:2,x:"Devendra");
pstmt.executeUpdate();
pstmt.setInt(parameterIndex:1,x:102);
pstmt.setString(parameterIndex:2,x:"Rohan");
pstmt.executeUpdate();
pstmt.setInt(parameterIndex:1,x:103);
pstmt.setString(parameterIndex:2,x:"Ashok");
pstmt.executeUpdate();
pstmt.setInt(parameterIndex:1,x:104);
pstmt.setString(parameterIndex:2,x:"Parth");
pstmt.executeUpdate();
pstmt.setInt(parameterIndex:1,x:105);
pstmt.setString(parameterIndex:2,x:"Arjun");
pstmt.executeUpdate();
System.out.println(x:"Data Inserted in Table EMP.");
```

OUTPUT AFTER INSETING VALUES IN TABLE




```
mysql> use jdbc;
Database changed
mysql> show tables;
+---------------+
| Tables_in_jdbc |
+---------------+
| emp           |
+---------------+
1 row in set (0.00 sec)

mysql> select * from emp;
+--------+----------+
| EMP_ID | EMP_NAME |
+--------+----------+
|    101 | Devendra |
|    102 | Rohan    |
|    103 | Ashok    |
|    104 | Parth    |
|    105 | Arjun    |
+--------+----------+
5 rows in set (0.00 sec)
```



UPDATE VALUES USING JDBC

USE DATABASE QUERY

UPDATE STATEMENT

OUTPUT STATEMENT

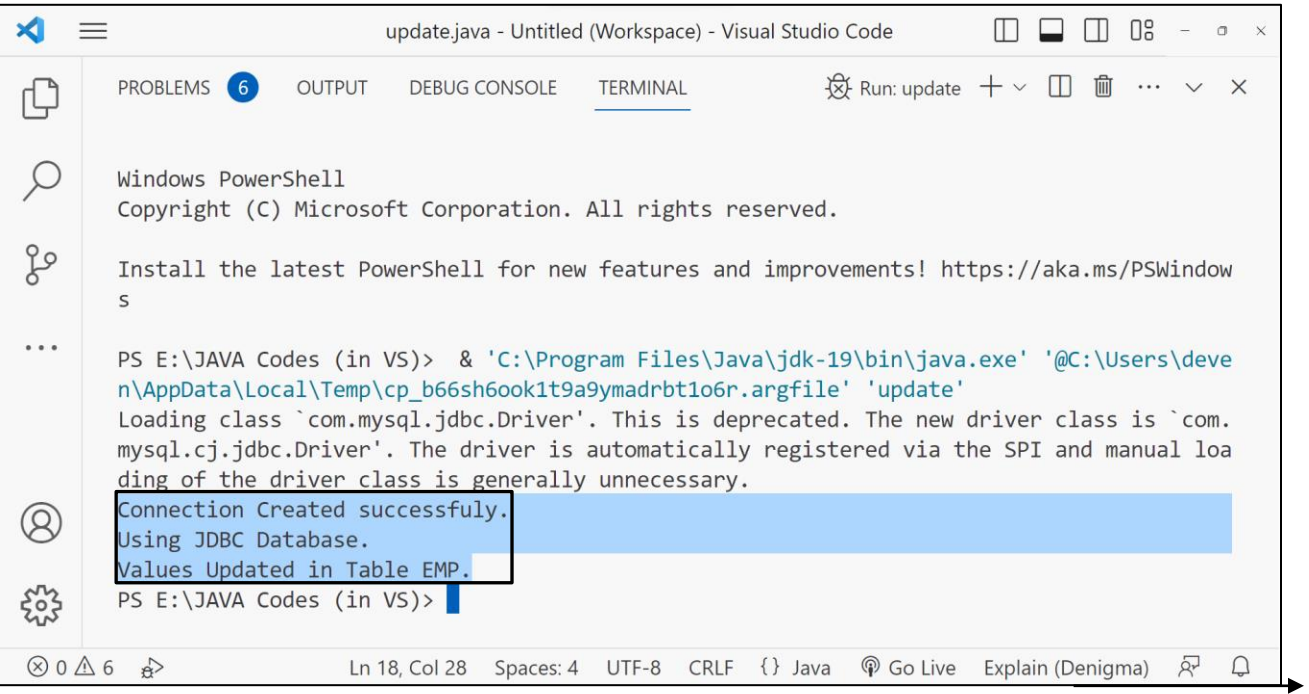
UPDATE QUERY

UPDATE STATEMENT

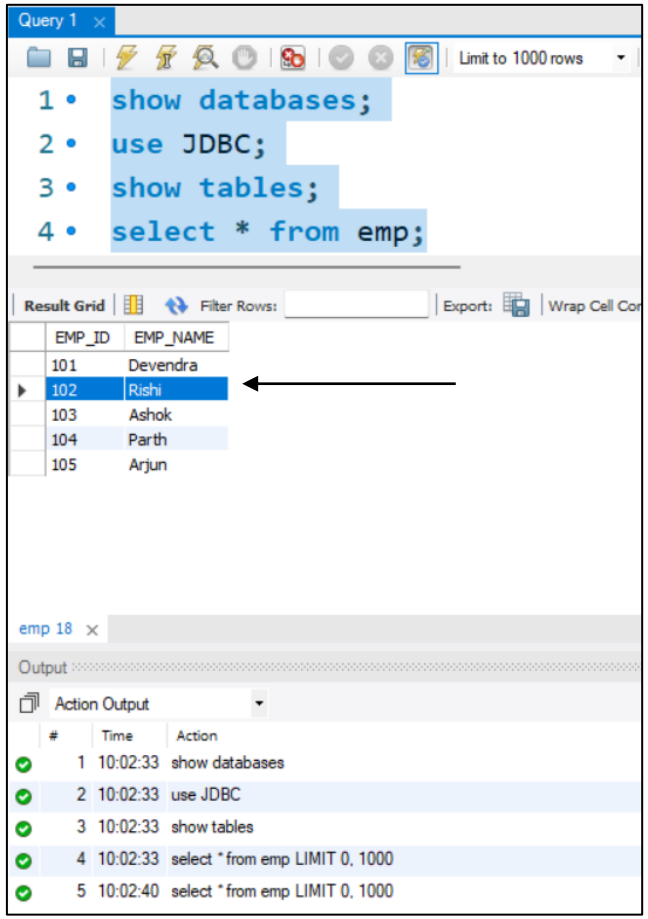
OUTPUT STATEMENT

```
// USE Database
String usedbquery = "USE JDBC";
Statement stmt1=con.createStatement();
stmt1.executeUpdate(usedbquery);
System.out.println(x:"Using JDBC Database.");

// Update Command
String update = "UPDATE EMP SET EMP_NAME = \"Rishi\"
WHERE EMP_ID = 102;";
Statement stmt2=con.createStatement();
stmt2.executeUpdate(update);
System.out.println(x:"Values Updated in Table EMP.");
```

OUTPUT AFTER UPDATING VALUES IN TABLE




```
mysql> select * from emp;
+--------+----------+
| EMP_ID | EMP_NAME |
+--------+----------+
|    101 | Devendra |
|    102 | Rohan    |
|    103 | Ashok    |
|    104 | Parth    |
|    105 | Arjun    |
+--------+----------+
5 rows in set (0.00 sec)

mysql> select * from emp;
+--------+----------+
| EMP_ID | EMP_NAME |
+--------+----------+
|    101 | Devendra |
|    102 | Rishi    |
|    103 | Ashok    |
|    104 | Parth    |
|    105 | Arjun    |
+--------+----------+
5 rows in set (0.00 sec)
```



→ Click Here

THANK YOU