

Object Oriented Programming Language-11

Vamika Mahajan
R2142211138

500096495

Create a database named "MeetingDB". Create a java program with JDBC to perform the following

- 1. Create a table named "Meeting" having the columns (MeetingID, ScheduledTime, ScheduledDate, participantID, NameOfParticipant, participantDateBirth, Email, Mobile). "participantID" is set auto increment.**

```
import java.sql.*;
import java.util.*;
public class J1
{
    public static void main(String args[])
    {
        String query1 = "create table Meeting(MeetingID, ScheduledTime,
ScheduledDate, participantID AUTO_INCREMENT, NameOfParticipant,
participantDateBirth, Email, Mobile)";
        try
        {
            Class.forName("com.mysql.jdbc.Driver");
            Connection con =
DriverManager.getConnection("jdbc:mysql://localhost:3306/MeetingDB", "root", ""); //3
            System.out.println("Connection Okay");

            Statement stmt = con.createStatement();
            stmt.execute(query1);
            con.close();
        }
        catch(Exception e)
        {
            System.out.println(e);
        }
    }
}
```

2. Create a stored procedure that can be invoked to insert records to the table with proper data (5 records).

```
import java.sql.*;
import java.util.*;
public class J2
{
    public static void main(String args[])
    {

        CREATE PROCEDURE Meeting1(MeetingID, ScheduledTime,
ScheduledDate, participantID AUTO_INCREMENT, NameOfParticipant,
participantDateBirth, Email, Mobile)

BEGIN

INSERT INTO MEETING(MeetingID, ScheduledTime, ScheduledDate, participantID
AUTO_INCREMENT, NameOfParticipant, participantDateBirth, Email, Mobile) VALUES
(?,?,?,?,?,?,?)

INSERT INTO MEETING(MeetingID, ScheduledTime, ScheduledDate, participantID
AUTO_INCREMENT, NameOfParticipant, participantDateBirth, Email, Mobile) VALUES
(?,?,?,?,?,?,?)

INSERT INTO MEETING(MeetingID, ScheduledTime, ScheduledDate, participantID
AUTO_INCREMENT, NameOfParticipant, participantDateBirth, Email, Mobile) VALUES
(?,?,?,?,?,?,?)

INSERT INTO MEETING(MeetingID, ScheduledTime, ScheduledDate, participantID
AUTO_INCREMENT, NameOfParticipant, participantDateBirth, Email, Mobile) VALUES
(?,?,?,?,?,?,?)

INSERT INTO MEETING(MeetingID, ScheduledTime, ScheduledDate, participantID
AUTO_INCREMENT, NameOfParticipant, participantDateBirth, Email, Mobile) VALUES
(?,?,?,?,?,?,?)

    }
}
```

3. Get the "MeetingID" from the user. Display the names of all the participants for the meeting with the given MeetingID on the console, preceded by the message, "Names of participants in meeting 1144".

```
import java.sql.*;
public class Ques3{
    public static void main(String args[])throws ClassNotFoundException,SQLException
    {
        Class.forName("com.mysql.cj.jdbc.Driver");
        Connection
        con=DriverManager.getConnection("jdbc:mysql://localhost:3306/Meeting","root","1234");
        Statement st=con.createStatement();

        st.executeUpdate(
            "create table Meeting(NameOfParticipant varchar(20),MeetingID int,ScheduledTime
            varchar(20),Email varchar(30)" + "Mobile varchar(10));");
        st.executeUpdate("insert into Meeting
        values('Vamika',1144,'Tuesday','Vamika@gmail.com','1234567891')");
        st.executeUpdate("insert into Meeting
        values('Ram',1144,'Monday','ram@gmail.com','1234567880')");
        st.executeUpdate("insert into Meeting
        values(\"Shyam\",1144,\"Tuesday\", \"Shyam@gmail.com\", \"1234567690\")");
        st.executeUpdate("insert into Meeting
        values(\"Ashish\",1105,\"Wednesday\", \"Ashish@gmail.com\", \"1234569876\")");
        st.executeUpdate("insert into Meeting
        Values(\"Rohit\",1105,\"Tuesday\", \"Rohit@gmail.com\", \"1234554321\")");

        ResultSet rs=st.executeQuery("Select * from Meeting");
        ResultSetMetaData rmd=rs.getMetaData();

        System.out.println("Number of columns:" +rmd.getColumnCount());
        System.out.println("Column names in the table Meeting :");
        for(int i=1; i<=rmd.getColumnCount(); i++) {
            System.out.println(rmd.getColumnName(i));
        }
    }
}
```

- 4. Insert a new participant using only the participant's name. Do not provide the "participantID". Determine the ID that was assigned to the new participant and display it on the console (the statement that displays the ID should display the name of the new participant as well).**

```
import java.sql.*;
public class Ques3{
    public static void main(String args[])throws ClassNotFoundException,SQLException
    {
        Class.forName("com.mysql.cj.jdbc.Driver");
        Connection
        con=DriverManager.getConnection("jdbc:mysql://localhost:3306/Meeting","root","1234");
        Statement st=con.createStatement();

        st.executeUpdate(
            "create table Meeting(NameOfParticipant varchar(20),MeetingID int,ScheduledTime
            varchar(20),Email varchar(30)+ "Mobile varchar(10));");
        st.executeUpdate("insert into Meeting
        values('Vamika',1144,'Tuesday','Vamika@gmail.com','1234567891')");
        st.executeUpdate("insert into Meeting
        values('Ram','1144','Monday','ram@gmail.com','1234567880')");
        st.executeUpdate("insert into Meeting
        values(\"Shyam\",1144,\"Tuesday\", \"Shyam@gmail.com\", \"1234567690\")");
        st.executeUpdate("insert into Meeting
        values(\"Ashish\",1105,\"Wednesday\", \"Ashish@gmail.com\", \"1234569876\")");
        st.executeUpdate("insert into Meeting
        Values(\"Rohit\",1105,\"Tuesday\", \"Rohit@gmail.com\", \"1234554321\")");

        System.out.println("Names of participants in meeting 1144 :");
        while(rs.next()) {
            if (rs.getInt(2)==1144)
            {
                System.out.println(rs.getString(1));
                System.out.println(rs.getString(1));
            }
        }
    }
}
```

5. Count the number of people participating in the meeting with ID 1105. Display a message on the console that gives both the "MeetingID" and the number of participants.

```
import java.sql.*;
public class Ques3{
    public static void main(String args[])throws ClassNotFoundException,SQLException
    {
        Class.forName("com.mysql.cj.jdbc.Driver");
        Connection
        con=DriverManager.getConnection("jdbc:mysql://localhost:/3306/Meeting","root","1234");
        Statement st=con.createStatement();

        st.executeUpdate(
            "create table Meeting(NameOfParticipant varchar(20),MeetingID int,ScheduledTime
            varchar(20),Email varchar(30)" + "Mobile varchar(10));");
        st.executeUpdate("insert into Meeting
        values('Vamika','1144','Tuesday','Vamika@gmail.com','1234567891')");
        st.executeUpdate("insert into Meeting
        values('Ram','1144','Monday','ram@gmail.com','1234567880')");
        st.executeUpdate("insert into Meeting
        values(\"Shyam\",1144,\"Tuesday\", \"Shyam@gmail.com\", \"1234567690\")");
        st.executeUpdate("insert into Meeting
        values(\"Ashish\",1105,\"Wednesday\", \"Ashish@gmail.com\", \"1234569876\")");
        st.executeUpdate("insert into Meeting
        Values(\"Rohit\",1105,\"Tuesday\", \"Rohit@gmail.com\", \"1234554321\")");
        st.executeUpdate("insert into
        Meeting(NameOfParticipant,ScheduledTime)values('Suraj',Monday)");
        rs=st.executeQuery("select * from Meeting");
        String s="Suraj";
        System.out.println("ID assigned to the new participant :");
        while (rs.next())
        {
            if(s.compareTo(rs.getString(1))==0)
            {
                System.out.println("rs.getInt(2)");
            }
        }
    }
}
```

6. Determine the names of all who participate in meetings on Tuesdays. Display the names, preceded by the message, “Participants attending Tuesday meetings”.

```
import java.sql.*;
public class Ques3{
public static void main(String args[])throws ClassNotFoundException,SQLException
{
Class.forName("com.mysql.cj.jdbc.Driver");
Connection
con=DriverManager.getConnection("jdbc:mysql://localhost:/3306/Meeting","root","1234");
Statement st=con.createStatement();

st.executeUpdate(
    "create table Meeting(NameOfParticipant varchar(20),MeetingID int,ScheduledTime
varchar(20),Email varchar(30)" + "Mobile varchar(10));");
st.executeUpdate("insert into Meeting
values('Vamika',1144,'Tuesday','Vamika@gmail.com','1234567891')");
st.executeUpdate("insert into Meeting
values('Ram','1144','Monday','ram@gmail.com','1234567880')");
st.executeUpdate("insert into Meeting
values(\"Shyam\",1144,\"Tuesday\", \"Shyam@gmail.com\", \"1234567690\")");
st.executeUpdate("insert into Meeting
values(\"Ashish\",1105,\"Wednesday\", \"Ashish@gmail.com\", \"1234569876\")");
st.executeUpdate("insert into Meeting
Values(\"Rohit\",1105,\"Tuesday\", \"Rohit@gmail.com\", \"1234554321\")");
rs= st.executeQuery("select * from Meeting");
int counter = 0;
while(rs.next())
{
    if (rs.getInt(2)==1105)
    {
        counter++;
    }
}
System.out.println("Meeting ID - 1105\nNumber of people :"+ counter);
System.out.println("\n Part E");
System.out.println("Participants attending Tuesday meetings :");
rs =st.executeQuery("select * from Meeting");
s="Tuesday";
while (rs.next())
{
    if(s.compareTo(rs.getString(3))==0)
    {
System.out.println(rs.getString(1));
    }
}
}}
```

7. Create a function to get the "participantDateBirth", find the participant's age, and print it.

```
import java.sql.*;
import java.util.*;
public class J2
{
    public static void main(String args[])
    {
```

```
CREATE OR REPLACE FUNCTION DOB ()
RETURNS INT
AS
BEGIN
DECLARE Years AS INT
DECLARE BirthdayDate AS DATETIME
DECLARE Age AS INT
```

```
SET Years = DATEDIFF(YY,DateOfBirth,GETDATE())
```

```
SET BirthdayDate = DATEADD(YY,Years,DateOfBirth)
```

```
SET Age = Years -
```

```
CASE
```

```
WHEN BirthdayDate>GETDATE() THEN 1
```

```
ELSE 0
```

```
END
```

```
RETURN Age
```

```
END
```

```
}
```

```
}
```