

- 1.) i) PreparedStatements objects are used to execute repetitive SQL Statements.
- ii) Compared to statement object execution, Prepared Statement object creation is faster. The reason is the object is precompiled, by eliminating the compilation task by DBMS.
- iii) The Prepared Statement object can be used by just replacing the parameters.

- 2.) i) `public boolean next()`
Used to move the cursor to the one row next from the current position.
- ii) `public boolean previous()`
Used to move the cursor to the row previous from the current position.
- iii) `public boolean first()`
Used to move the cursor to the first row in ResultSet object.
- iv) `public boolean last()`
Used to move the cursor to the last row in ResultSet object.
- v) `public int getInt (int index)` or `public int getInt (String column):`
Used to return the data of the specified index or column name.

vi) public ~~int~~ String getString(int index)
or public String getString(String
column-name)

Used to return the data
of specified column, index or
column name.

3) i) Since a Prepared Statements object
represents only one SQL statement
at a time, we can execute only
one statement by one prepared
statement object.

ii) To prevent injection attacks it does not
allow more than one value to a
place holder.

```

import java.sql.*;

class Exp19a {
    public String database = "C:\\Users\\deong\\College\\Java\\Manual-
Programs\\Experiment19\\MSBTE.accdb";

    private Connection conn;

    // Create Connection
    public void createConnection() {
        try {
            conn = DriverManager.getConnection("jdbc:ucanaccess://" + database);
        } catch (SQLException e) {
            System.out.println("Connection Failed");
            System.exit(1);
        }
    }

    public void closeConnection() {
        try {
            conn.close();
        } catch (SQLException e) {
            System.out.println("Close Connection Failed ?");
        }
    }

    public void updateQuery(String query) {
        try {
            Statement statement = conn.createStatement();
            statement.executeUpdate(query);
        } catch (SQLException e) {
            System.out.println("Error in updateQuery()");
        }
    }

    public static void main(String[] args) {
        Exp19a dbconn = new Exp19a();
        try {
            Class.forName("net.ucanaccess.jdbc.UcanaccessDriver");
        } catch (Exception e) {
            System.out.println("Error in Loading Driver");
        }
        dbconn.createConnection();
        dbconn.updateQuery("UPDATE Students SET FirstName = 'Deon' , LastName = 'Gracias' WHERE
ID = 3;");
    }
}

```

ID	First Name	Last Name	Address	City	State	Phone No	Email
1	Rebecca	Didio	171 E 24th St	Leith	TAS	03-8174-9123	rebecca.didio@hcm.edu.au
2	Stevie	Hallo	22222 Acoma St	Proston	QLD	07-9997-3366	stevie.hallo@hcm.edu.au
3	Mariko	Stayer	534 Schoenborr	Hamel	WA	08-5558-9019	mariko_stayer@hcm.edu.au

ID	FirstName	LastName	Address	City	State	PhoneNo	Email
1	Rebecca	Didio	171 E 24th St	Leith	TAS	03-8174-9123	rebecca.didio@hcm.edu.au
2	Stevie	Hallo	22222 Acoma St	Proston	QLD	07-9997-3366	stevie.hallo@hcm.edu.au
3	Deon	Gracias	534 Schoenborr	Hamel	WA	08-5558-9019	mariko_stayer@hcm.edu.au

```
import java.sql.*;

class Exp19b {
    public String database = "C:\\Users\\deong\\College\\Java\\Manual-
Programs\\Experiment19\\SampleDatabase.accdb";

    private Connection conn;

    // Create Connection
    public void createConnection() {
        try {
            conn = DriverManager.getConnection("jdbc:ucanaccess://" + database);
        } catch (SQLException e) {
            System.out.println("Connection Failed");
            System.exit(1);
        }
    }

    public void closeConnection() {
        try {
            conn.close();
        } catch (SQLException e) {
            System.out.println("Close Connection Failed ?");
        }
    }

    public void updateQuery(int id, String firstName, String lastName) {
        try {
            PreparedStatement stmt = conn.prepareStatement("insert into student values(?,?,?)");
            stmt.setInt(1, id);
            stmt.setString(2, firstName);
            stmt.setString(3, lastName);
            int i = stmt.executeUpdate();
            System.out.println(i + " records inserted");
        } catch (SQLException e) {
            System.out.println("Error in updateQuery()");
        }
    }

    public static void main(String[] args) {
        Exp19b dbconn = new Exp19b();
    }
}
```



```

try {
    Class.forName("net.ucanaccess.jdbc.UcanaccessDriver");
} catch (Exception e) {
    System.out.println("Error in Loading Driver");
}
dbconn.createConnection();
dbconn.updateQuery(101, "Abhishek", "Yadav");
}
}

```

1 records inserted

Student		
id	firstname	lastname
1	Deon	Gracias
101	Abhishek	Yadav

```

import java.sql.*;

public class Exp19c {
    public String database = "C:\\Users\\deong\\College\\Java\\Manual-
Programs\\Experiment19\\MSBTE.accdb";

    private Connection conn;

    // Create Connection
    public void createConnection() {
        try {
            conn = DriverManager.getConnection("jdbc:ucanaccess://" + database);
        } catch (SQLException e) {
            System.out.println("Connection Failed");
            System.exit(1);
        }
    }

    public void closeConnection() {
        try {
            conn.close();
        } catch (SQLException e) {
            System.out.println("Close Connection Failed ?");
        }
    }

    public void selectQuery() {
        try {
            Statement stmt = conn.createStatement();
            ResultSet rs = stmt.executeQuery("SELECT ID, first_name, last_name FROM Student;");
            System.out.println("-----");
            System.out.printf("%5s | %13s | %13s\n", "ID", "FirstName", "LastName");
            System.out.println("-----");
            while (rs.next()) {

```

```

        System.out.printf("%5s | %13s | %13s\n", Integer.toString(rs.getInt("ID")),
rs.getString("first_name"),
        rs.getString("last_name"));

    }
    } catch (SQLException e) {
        e.printStackTrace();
        System.out.println("Error in selectQuery()");
    }
}

public static void main(String[] args) {
    Exp19c dbconn = new Exp19c();
    try {
        Class.forName("net.ucanaccess.jdbc.UcanaccessDriver");
    } catch (Exception e) {
        System.out.println("Error in Loading Driver");
    }
    dbconn.createConnection();
    dbconn.selectQuery();
}
}

```

ID	FirstName	LastName
1	Rebbeca	Didio
2	Stevie	Hallo
3	Deon	Gracias
4	Gerardo	Woodka
5	Mayra	Bena
6	Idella	Scotland
7	Sherill	Klar
8	Ena	Desjardiws
9	Vince	Siena
10	Theron	Jarding