Dt : 7/3/2025

**imp*

2.PreparedStatement:

=>'PreparedStatement' is an interface from java.sql package and which is used to execute normal queries with IN-Parameters.

=>we use parepareStatement()-method from 'Connection' interface to create implementation Object for 'PreparedStatement' interface,becuase the prepareStatement()-method internally holding 'Anonymous Local innerclass as implementation class of PreparedStatemet Interface and which generate PreparedStatement-Object

Method signature of prepareStatement():

public abstract java.sql.PreparedStatement prepareStatement(java.lang.String) throws java.sql.SQLException;

syntax:

PreparedStatement ps = con.prepareStatement("query-structure");

=>The following are two important methods of PreparedStatement:

(a)executeQuery()

(b)executeUpdate()

(a)executeQuery():

=>executeQuery()-method is used to execute select-queries.

Method Signature of executeQuery():

public abstract java.sql.ResultSet executeQuery() throws java.sql.SQLException;

syntax:

ResultSet rs = ps.executeQuery();

(b)executeUpdate():

=>executeUpdate()-method is used to execute NonSelect queries.

Method signature of executeUpdate():

public abstract int executeUpdate() throws java.sql.SQLException;

syntax:

int k = ps.executeUpdate();

Note:

=>executeQuery() and executeUpdate() methods are with parameter in 'Statement' and without parameter in 'PreparedStatement'

Ex:(Demonstrating PreparedStatement)

DBTable : BankCustomer72(accno,cid,cname,balance,acctype)

primary key : accno

create table BankCustomer72(accno number(15),cid varchar2(15),cname varchar2(15), balance number(10,2),acctype varchar2(15),primary key(accno));

Construct JDBC Application to perform the following operations based on Choice:

1.AddBankCustomer

2.ViewAllBankCustomers

3.Exit

Note:

repeat the process until we perform exit operation

Program : DBCon6.java

```
package test;
import java.sql.*;
import java.util.*;
public class DBCon6
{
	public static void main(String[] args)
	{
	  Scanner s = new Scanner(System.in);
	  try(s;){
		Class.forName("oracle.jdbc.driver.OracleDriver");
		Connection con = DriverManager.getConnection

	("jdbc:oracle:thin:@localhost:1521:xe","system","tiger");
		PreparedStatement ps1 = con.prepareStatement
				("insert into BankCustomer72 values(?,?,?,?,?)");
				   //Compilation process
		PreparedStatement ps2 = con.prepareStatement
				("select * from BankCustomer72");//Compilation
Process
		while(true) {
			System.out.println("******Operations Choice******");
			System.out.println("\t1.AddBankCustomer"
					+ "\n\t2.ViewAllBankCustomers"
					+ "\n\t3.Exit");
			System.out.println("Enter Your Choice:");
			int choice = Integer.parseInt(s.nextLine());
			switch(choice) {
			case 1:
				//read data from console into Local variables
				System.out.println("Enter the CustAccNo:");
				long accNo = Long.parseLong(s.nextLine());
				String cId = "SB"+accNo;
				System.out.println("Enter the CustName:");
				String cName = s.nextLine();
				System.out.println("Enter the Cust-Balance:");
				float balance = Float.parseFloat(s.nextLine());
				System.out.println("Enter the Cust-AccType:");
				String accType = s.nextLine();

				//Load data to PreparedStatement Object using Setter
methods
				ps1.setLong(1, accNo);
```

```
                        ps1.setString(2, cId);
                        ps1.setString(3, cName);
                        ps1.setFloat(4, balance);
                        ps1.setString(5, accType);

                        int k = ps1.executeUpdate();//Execution Process
                        if(k>0) {
                              System.out.println("BankCustomer Added
Successfully...");
                        }

                        break;
                  case 2:
                        ResultSet rs = ps2.executeQuery();//Execution process
                        while(rs.next()) {
                              System.out.println(rs.getLong(1)+"\t"
                                          +rs.getString(2)+"\t"
                                          +rs.getString(3)+"\t"
                                          +rs.getFloat(4)+"\t"
                                          +rs.getString(5));
                        }//end of loop
                        break;
                  case 3:
                        System.out.println("Operations Stopped...");
                        System.exit(0);
                  default:
                        System.out.println("Invalid Choice....");
                  }//end of switch
            }//end of while
        }catch(Exception e) {
            e.printStackTrace();
        }
      }
}
```


o/p:

*******Operations Choice*******

1.AddBankCustomer

2.ViewAllBankCustomers

3.Exit

Enter Your Choice:

2

6123456 SB6123456 Alex 12000.0 Savings

454541234 SB454541234 Ram 16000.0 Savings

*******Operations Choice******

1.AddBankCustomer

2.ViewAllBankCustomers

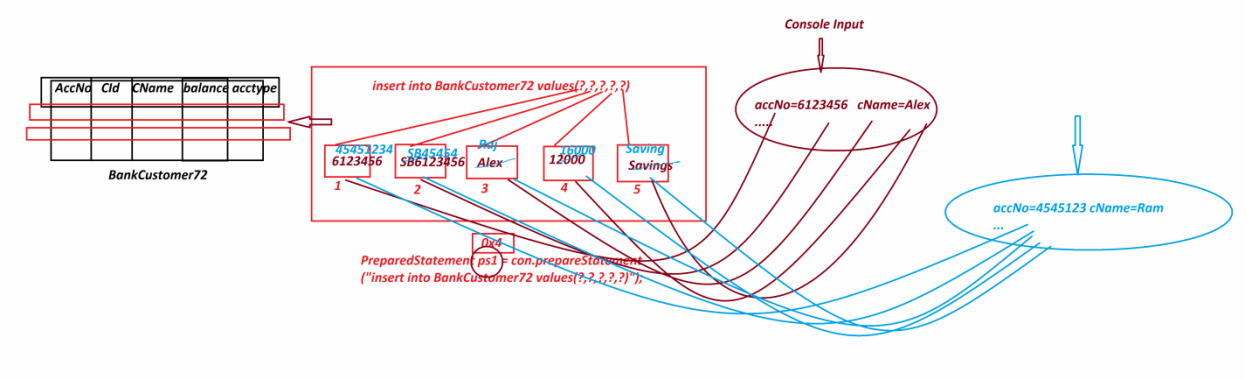
3.Exit

Enter Your Choice:

3

Operations Stopped...

Diagram:



==========