THE UNIVERSITY OF DODOMA

COURSE: OBJECT ORIENTED PROGRAMMING IN JAVA

INSTRUCTOR: MR EVERYJUSTUS BARONGO

COURSE CODE: CP 215

INDIVISUAL ASSIGNMENT

NAME: ANUARI IDDI ISSA

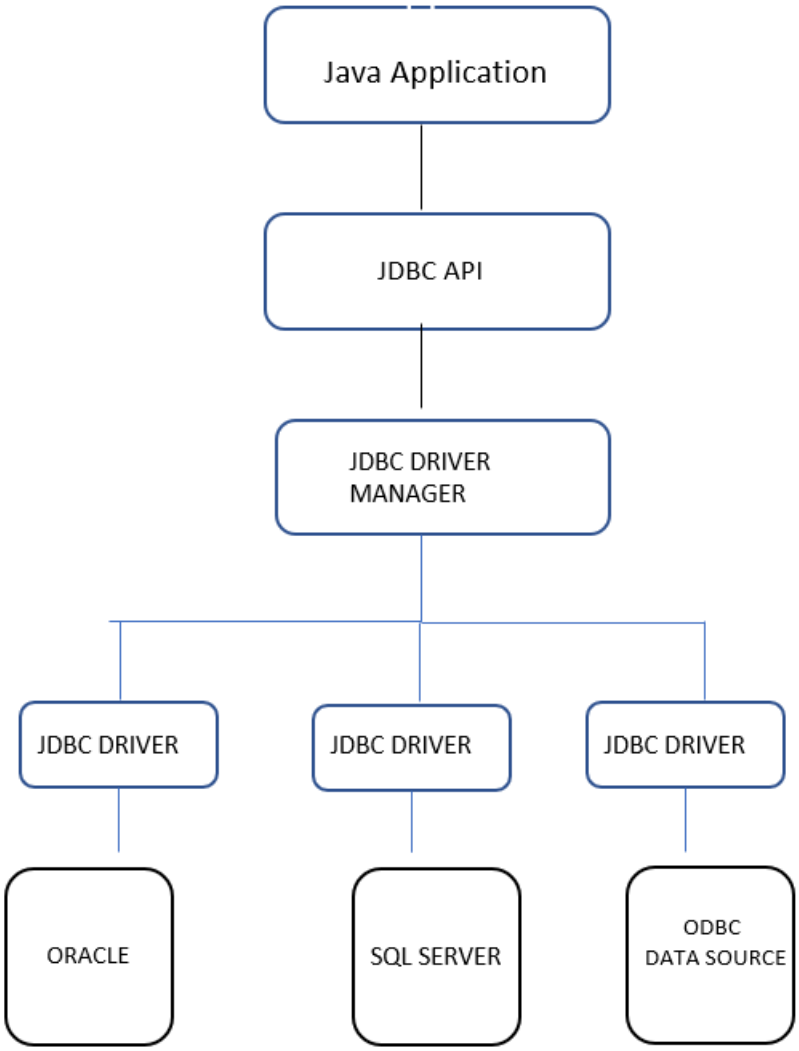
REG NO: T/UDOM/2020/00345

COURSE BCS CE

Question 01.

In diagram, indicate the relationship between Java code, JDBC API and Database Driver.

JDBC stand for Java Database Connectivity is the Java API that manages connecting to a database, executing the queries and commands and handling the result sets obtained from the database. JDBC provides the mechanics of the java applications communicating with a database. The JDBC provide two layers which are JDBC API and JDBC driver.



Question 02.

Itemize requirements necessary to use JDBC and any DBMS.

Select DBMS as Mysql

IDE as Netbeans

Connector as mysql-connector-java-8.0.28

Question 03.

```
import com.mysql.jdbc.Connection;
import com.mysql.jdbc.Statement;
import java.sql.DriverManager;
import java.sql.ResultSet;
import java.sql.SQLException;


public class Mykazi {
        public static void main(String[] args) {
       try{
           Connection conn=(Connection) DriverManager.getConnection("jdbc:mysql://localhost:3306/carrental", "root","");
           Statement st=(Statement) conn.createStatement();
           ResultSet rs=st.executeQuery("select * from registration");
           while(rs.next()){
             System.out.println("Name"+rs.getString(1)+"Adderess"+rs.getString(2)+"gender"+rs.getString(3)+"Last"+rs.getString(4)
             +"Phone "+rs.getString(5)+"Passwd"+rs.getString(6));
           }
       }
       catch(SQLException e){
       e.printStackTrace();
       }
    }
}
```

Question 05.

```
CREATE DATABASE StudentData;
CREATE TABLE student(
   regNo int (50) NOT NULL AUTO_INCREMENT PRIMARY KEY,
   names VARCHAR(50) NOT NULL,
   Address VARCHAR(50) NULL,
);

CREATE TABLE course(
   courseID int (20) NOT NULL AUTO_INCREMENT PRIMARY KEY,
   course_code VARCHAR(10) NOT NULL,
   course_name VARCHAR(200) NOT NULL,
   studID int(10)
);
```

QN6

The java program for insert data into the database.

```
import com.mysql.jdbc.Connection;
import com.mysql.jdbc.Statement;
import java.sql.DriverManager;
import java.sql.PreparedStatement;
import java.sql.ResultSet;
import java.sql.SQLException;
import static javafx.scene.input.KeyCode.T;


public class StudentData {
        public static void main(String[] args) {
       try{
           Connection conn=(Connection) DriverManager.getConnection("jdbc:mysql://localhost:3306/StudentData", "root","");
           String qry="INSERT INTO student VALUES(?,?,?)";
           PreparedStatement pst=conn.prepareStatement(qry);
           pst.setString(1, "Anuary");
           pst.setString(2, "T/UDOM/2020");
           pst.setString(3, "00345");
           pst.executeUpdate();
            System.out.println("Values inserted..");
       }
       catch(SQLException e){
       e.printStackTrace();
       }
    }
}
```

Java program to select data into the database

```
import java.sql.PreparedStatement;
import java.sql.ResultSet;
import java.sql.SQLException;


public class AnotherStudent {
    public static void main(String[] args) {
        try{
            Connection conn=(Connection) DriverManager.getConnection("jdbc:mysql://localhost:3306/StudentData", "root","");
            String qry="select *from student";
            Statement st=(Statement) conn.createStatement();
            ResultSet rs=st.executeQuery(qry);
            while(rs.next()){
              System.out.println("Name"+rs.getString(1)+"Adderess"+rs.getString(2)+"gender"+rs.getString(3)
              );
            }

        }
        catch(SQLException e){
        e.printStackTrace();
        }
    }
}
```

Java program delete data into the database

```
import com.mysql.jdbc.Connection;
import com.mysql.jdbc.Statement;
import java.sql.DriverManager;
import java.sql.PreparedStatement;
import java.sql.ResultSet;
import java.sql.SQLException;


public class Delete {
        public static void main(String[] args) {
    try{
        Connection conn=(Connection) DriverManager.getConnection("jdbc:mysql://localhost:3306/StudentData", "root","");
        String qry="delete from student";
        PreparedStatement pst=conn.prepareStatement(qry);
        pst.executeUpdate();
            System.out.println("Values Deleted..");
            conn.close();
    }
    catch(SQLException e){
    e.printStackTrace();
    }
  }
}
```

QUESTION 7

```
import com.mysql.jdbc.Statement;
import java.sql.Connection;
import java.sql.DriverManager;
import static java.sql.DriverManager.getConnection;
import java.sql.SQLException;



public class Update {
        public static void main(String[] args) {
              try{
             Connection conn=(Connection) DriverManager.getConnection("jdbc:mysql://localhost:3306/StudentData", "root","");
    String [] queries = {
        "insert into student values ('A', 'X', '8765')",
        "insert into student values ('A', 'X', '876')",
        "insert into student values ('A', 'X', '865')",
        };
        Statement stat = (Statement) conn.createStatement();
for (String query : queries) {
        stat.addBatch(query);
}
         stat.executeBatch();
         stat.close();
         conn.close();

              }
              catch(SQLException e){
               e.printStackTrace();
                   }
        }
}
```