DBMS LAB CHIT SOLUTIONS

1. SQL JDBC Connectivity

```
package my_jdbc;

import java.util.*;
import java.sql.*;

public class my_conn {
//    Function to Display
      public void Display(){
            try{
                  Class.forName("com.mysql.jdbc.Driver");
                  Connection con =
DriverManager.getConnection("jdbc:mysql://localhost:3306/my_db", "root", "MH!$bf5292");

                  Statement stmt=con.createStatement();
                  ResultSet rs=stmt.executeQuery("select * from student");

                  while(rs.next()) {
                        System.out.println(rs.getInt(1)+" "+rs.getString(2)+"
"+rs.getInt(3));
                  }

                  con.close();
            }
            catch(Exception e){
                  System.out.println(e);
            }
      }

//    Function to Insert
      public void Insert(int id, String name, int marks){
            try{
                  Class.forName("com.mysql.jdbc.Driver");
                  Connection con =
DriverManager.getConnection("jdbc:mysql://localhost:3306/my_db", "root", "MH!$bf5292");

                  String query = "insert into student (id, name, marks)" + "
values("+id+",'"+name+"',"+marks+")";

                  Statement stmt=con.createStatement();
                  stmt.executeUpdate(query);

                  con.close();
            }
            catch(Exception e){
                  System.out.println(e);
            }
      }

//    Function to Update
      public void Update(){
            try{
                  Class.forName("com.mysql.jdbc.Driver");
                  Connection con =
DriverManager.getConnection("jdbc:mysql://localhost:3306/my_db", "root", "MH!$bf5292");

                  String query = "update student set name='Raj' where id=2";

                  Statement stmt=con.createStatement();
                  stmt.executeUpdate(query);
```

```
                con.close();
            }
            catch(Exception e){
                System.out.println(e);
            }
        }

//      Function to Delete
        public void Delete(){
            try{
                Class.forName("com.mysql.jdbc.Driver");
                Connection con =
DriverManager.getConnection("jdbc:mysql://localhost:3306/my_db", "root", "MH!$bf5292");

                String query = "delete from student where id=2";

                Statement stmt=con.createStatement();
                stmt.executeUpdate(query);

                con.close();
            }
            catch(Exception e){
                System.out.println(e);
            }
        }

        public static void main(String args[]) {
            try {
                Scanner sc=new Scanner(System.in);
                my_conn obj1=new my_conn();

                int ch;

                String name;
                int id, marks;

                do
                {
                    System.out.println("----------------------------------------
-");
                    System.out.println("    1.Display all Records");
                    System.out.println("    2.Insert new Record");
                    System.out.println("    3.Update a Record");
                    System.out.println("    4.Delete a Record");
                    System.out.println("    5.Exit");
                    System.out.println("----------------------------------------
-");
                    System.out.println("    Enter your Choice");

                    ch=sc.nextInt();

                    switch(ch){
                        case 1: obj1.Display();
                            break;
                        case 2: System.out.println("Enter Student Id: ");
                            id=sc.nextInt();

                            System.out.println("Enter Student Name: ");
                            name=sc.next();

                            System.out.println("Enter Student Marks: ");
                            marks=sc.nextInt();

                            obj1.Insert(id, name, marks);

                            break;
```

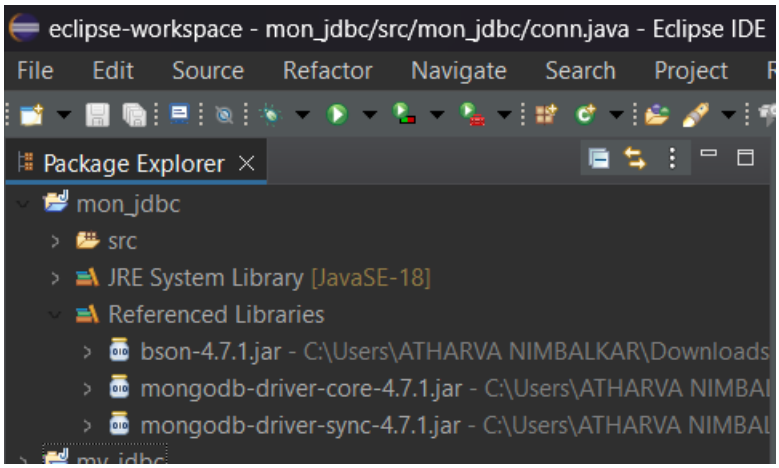
```
                    case 3: obj1.Update();
                          break;
                    case 4: obj1.Delete();
                          break;
              }
       }while(ch!=5);

       System.out.println("Exitting...");

       sc.close();
}
catch(Exception e) {
       System.out.println(e);
```

2. MongoDB JDBC Connectivity

JAR FILES




```
package mon_jdbc;

import static com.mongodb.client.model.Filters.eq;

import java.util.*;

import com.mongodb.*;
import com.mongodb.client.*;
import com.mongodb.client.model.*;
import com.mongodb.client.result.*;

import org.bson.*;
import org.bson.conversions.*;
import org.bson.types.ObjectId;

public class conn {
//    Function to Display
      public void Display(){
        String uri = "mongodb://localhost:27017";

        try (MongoClient client = MongoClients.create(uri)) {

            MongoDatabase database = client.getDatabase("my_db");
            MongoCollection<Document> coll = database.getCollection("student");

            Bson projectionFields = Projections.fields(
                    Projections.include("roll_no", "name", "marks"),
                    Projections.excludeId());

            MongoCursor<Document> cursor = coll.find()
                    .projection(projectionFields)
                    .iterator();

            try {
                while(cursor.hasNext()) {
                    System.out.println(cursor.next().toJson());
                }
            } finally {
                cursor.close();
            }
        }
      }

//    Function to Insert
      public void Insert(int roll, String name, int marks){
```

```
            String uri = "mongodb://localhost:27017";

        try (MongoClient client = MongoClients.create(uri)) {

            MongoDatabase db = client.getDatabase("my_db");
            MongoCollection<Document> coll = db.getCollection("student");

            try {
                InsertOneResult result = coll.insertOne(
                        new Document()
                        .append("_id", new ObjectId())
                        .append("roll_no", roll)
                        .append("name", name)
                        .append("marks", marks));

                System.out.println("Success! Inserted document id: " +
result.getInsertedId());
            } catch (MongoException e) {
                System.err.println("Unable to insert due to an error: " + e);
            }
        }
      }

//    Function to Update
      public void Update(){
            String uri = "mongodb://localhost:27017";

        try (MongoClient client = MongoClients.create(uri)) {

            MongoDatabase db = client.getDatabase("my_db");
            MongoCollection<Document> coll = db.getCollection("student");

            Document query = new Document().append("roll_no",  1);

            Bson updates = Updates.combine(
                    Updates.set("name", "Neha"),
                    Updates.currentTimestamp("lastUpdated"));

            UpdateOptions options = new UpdateOptions().upsert(true);

            try {
                UpdateResult result = coll.updateOne(query, updates, options);

                System.out.println("Modified document count: " +
result.getModifiedCount());
                System.out.println("Upserted id: " + result.getUpsertedId()); // only
contains a value when an upsert is performed
            } catch (MongoException e) {
                System.err.println("Unable to update due to an error: " + e);
            }
        }
      }

//    Function to Delete
      public void Delete(){
            String uri = "mongodb://localhost:27017";

        try (MongoClient client = MongoClients.create(uri)) {

            MongoDatabase db = client.getDatabase("my_db");
            MongoCollection<Document> coll = db.getCollection("student");

            Bson query = eq("roll_no", 1);

            try {
                DeleteResult result = coll.deleteOne(query);
```

```
                System.out.println("Deleted document count: " +
result.getDeletedCount());
            } catch (MongoException e) {
                System.err.println("Unable to delete due to an error: " + e);
            }
        }
    }

    public static void main(String args[]) {
        try {
            Scanner sc=new Scanner(System.in);

            conn obj1=new conn();

            int ch;

            String name;
            int id, marks;

            do
            {
                System.out.println("----------------------------------------
-");
                System.out.println("    1.Display all Records");
                System.out.println("    2.Insert new Record");
                System.out.println("    3.Update a Record");
                System.out.println("    4.Delete a Record");
                System.out.println("    5.Exit");
                System.out.println("----------------------------------------
-");
                System.out.println("    Enter your Choice");

                ch=sc.nextInt();

                switch(ch){
                    case 1: obj1.Display();
                        break;
                    case 2: System.out.println("Enter Student Id: ");
                        id=sc.nextInt();

                        System.out.println("Enter Student Name: ");
                        name=sc.next();

                        System.out.println("Enter Student Marks: ");
                        marks=sc.nextInt();

                        obj1.Insert(id, name, marks);

                        break;
                    case 3: obj1.Update();
                        break;
                    case 4: obj1.Delete();
                        break;
                }
            }while(ch!=5);

            System.out.println("Exitting...");

            sc.close();
        }
        catch(Exception e) {
            System.out.println(e);
        }
    }
}
```