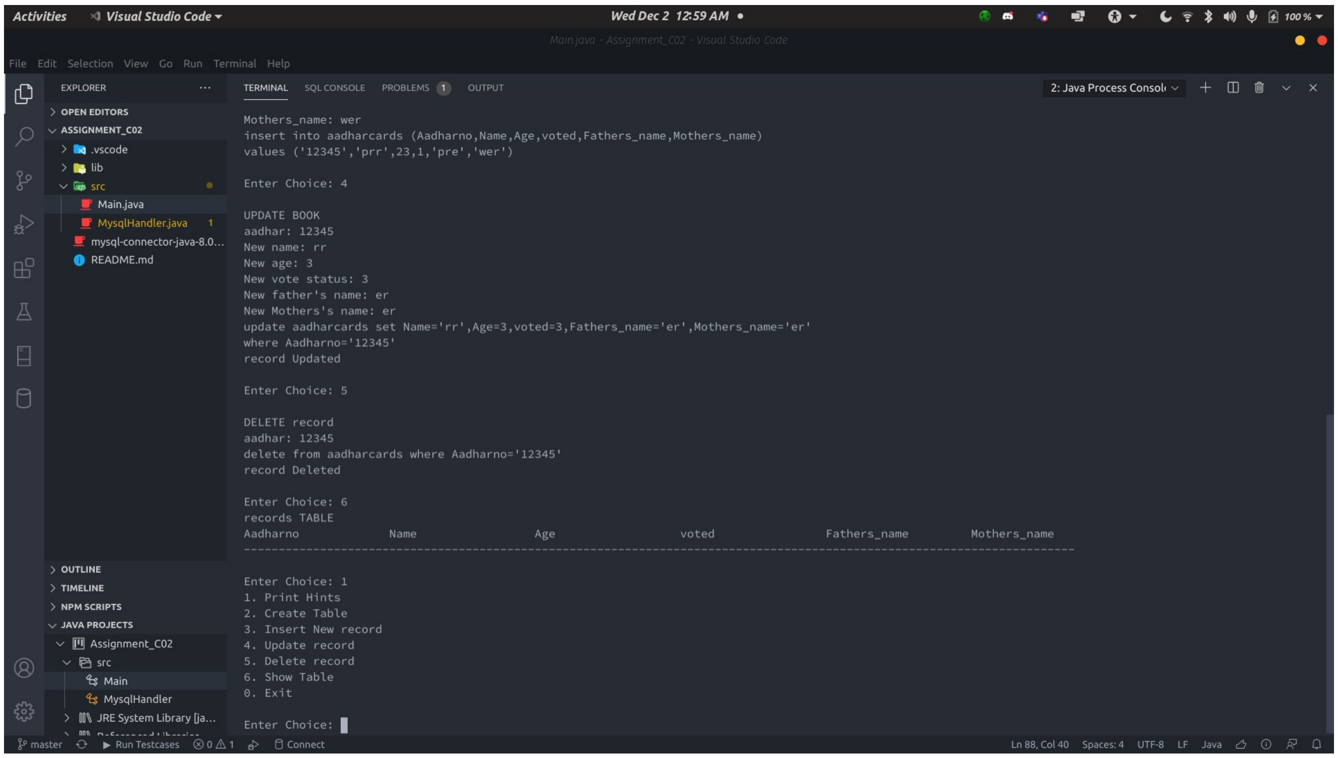
```
import java.util.Scanner;

public class Main {
    private static Scanner input = new Scanner(System.in);
    private static String query;
    private static String name, fname, mname;
    private static int aadhar, age, vote;

    static {
        MysqlHandler.main(new String[0]);
        System.out.println();
    }

    public static void main(String[] args) {
        int rowsAffected;
        int choice = 1;
        while (choice != 0) {
            switch (choice){
                case 1:
                    printHints();
                break;

                case 2:
                    System.out.println("\nCREATING TABLE");
                    query = "create table aadharcards(\n" +
                            "   Aadharno int not null, \n" +
                            "   Name varchar(32) not null, \n" +
                            "   Age int not null, \n" +
                            "   voted int not null, \n" +
                            "   Fathers_name varchar(32) not null, \n" +
                            "   Mothers_name varchar(32) not null \n" +
                            ")";
                    System.out.println(query);
                    MysqlHandler.execute(query);
                break;

                case 3:
                    System.out.println("\nINSERT NEW RECORD");
                    System.out.print("Aadharno: ");
                    aadhar = Integer.parseInt(input.nextLine());
                    System.out.print("Name: ");
                    name = input.nextLine();
                    System.out.print("Age: ");
                    age = Integer.parseInt(input.nextLine());
                    System.out.print("voted: ");
                    vote = Integer.parseInt(input.nextLine());
                    System.out.print("Fathers_name: ");
                    fname = input.nextLine();
                    System.out.print("Mothers_name: ");
                    mname = input.nextLine();

                    query = String.format(
                            "insert into aadharcards (Aadharno,Name,Age,voted,Fathers_name,Mothers_name)\
    n" +
                            "values ('%s','%s',%s,%s,'%s','%s')",aadhar,name,age,vote,fname,mname
                    );
                    System.out.println(query);
                    MysqlHandler.executeUpdate(query);
                break;

                case 4:
                    System.out.println("\nUPDATE BOOK");
                    System.out.print("aadhar: ");
                    aadhar = Integer.parseInt(input.nextLine());
                    System.out.print("New name: ");
                    name = input.nextLine();
                    System.out.print("New age: ");
                    age = Integer.parseInt(input.nextLine());
                    System.out.print("New vote status: ");
                    vote = Integer.parseInt(input.nextLine());
                    System.out.print("New father's name: ");
                    fname = input.nextLine();
                    System.out.print("New Mothers's name: ");
                    mname = input.nextLine();

                    query = String.format(
                            "update aadharcards set Name='%s',Age=%s,voted=
    %s,Fathers_name='%s',Mothers_name='%s'\n" +
                            "where Aadharno='%s'",name,age,vote,fname,mname,aadhar
                    );
                    System.out.println(query);
                    rowsAffected = MysqlHandler.executeUpdate(query);
                    if (rowsAffected == 0)
                        System.out.println("Aadhar no: "+aadhar+" Not Found");
                    else
                        System.out.println("record Updated");
                break;

                case 5:
                    System.out.println("\nDELETE record");
                    System.out.print("aadhar: ");
                    aadhar = Integer.parseInt(input.nextLine());

                    query = String.format("delete from aadharcards where Aadharno='%s'",aadhar);
                    System.out.println(query);
                    rowsAffected = MysqlHandler.executeUpdate(query);
                    if (rowsAffected == 0)
                        System.out.println("aadhar no: "+aadhar+" Not Found");
                    else
                        System.out.println("record Deleted");

                break;

                case 6:
                    System.out.println("records TABLE");
                    MysqlHandler.executeQuery("select * from aadharcards");
                break;

                default:
                    System.out.println("Option ("+choice+") not found");
            }

            System.out.println();
            System.out.print("Enter Choice: ");
            choice = Integer.parseInt(input.nextLine());
        }

        System.out.println("exit()");
    }

    public static void printHints(){
        System.out.println("1. Print Hints");
        System.out.println("2. Create Table");
        System.out.println("3. Insert New record");
        System.out.println("4. Update record");
        System.out.println("5. Delete record");
        System.out.println("6. Show Table");
        System.out.println("0. Exit");
    }

    }


    import java.sql.*;

    public abstract class MysqlHandler {
        static Connection connection = null;
        static Statement statement = null;
        static ResultSet resultSet = null;

        static final String username = "root";
        static final String password = "Hello@123";
        static final String url = "jdbc:mysql://localhost:3306/C02";

        public static void main(String[] args) {

            try{
```

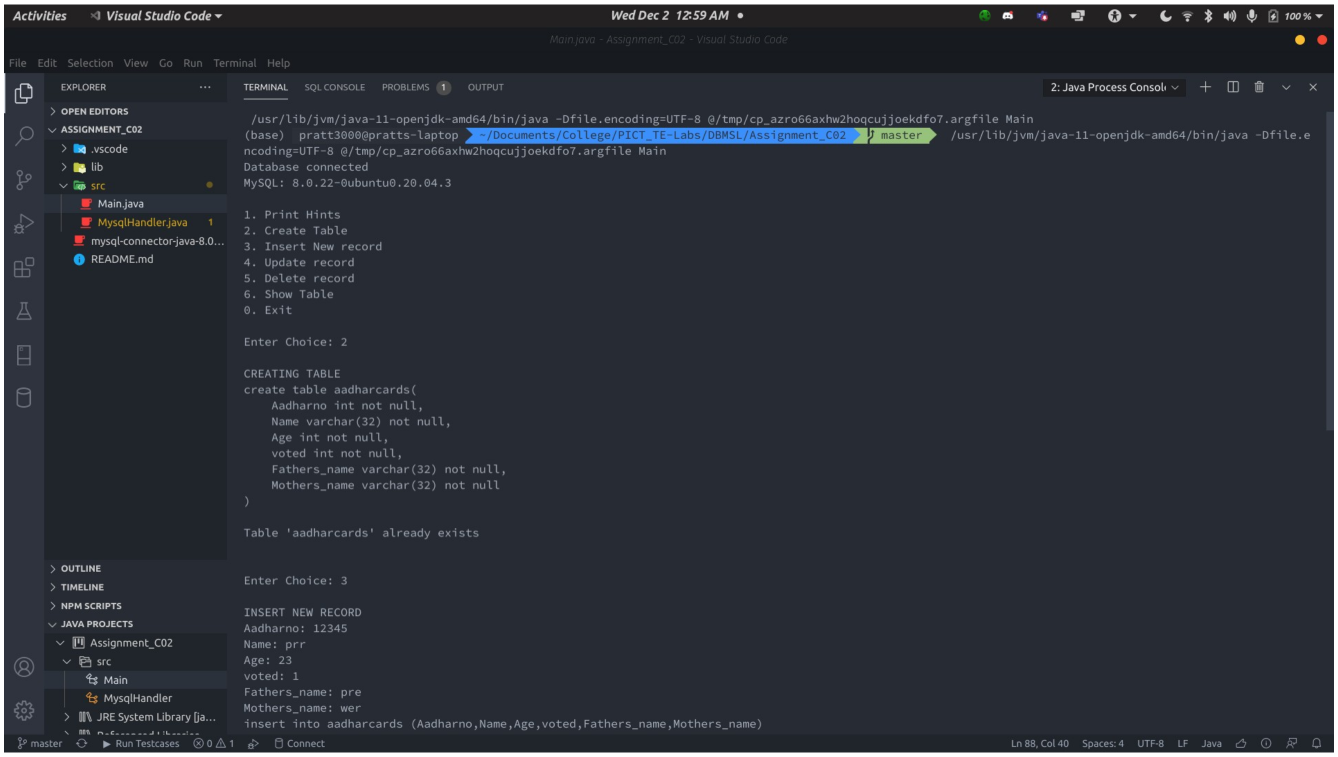
```
        connection = DriverManager.getConnection(url,username,password);
        statement = connection.createStatement();
        resultSet = statement.executeQuery("select version()");
        System.out.println("Database connected");

        while(resultSet.next())
            System.out.println("MySQL: "+resultSet.getString(1));


    }catch (Exception e){
        System.out.println(e.getMessage());
    }finally {
        try {
            if(resultSet != null)
                resultSet.close();
            if(statement != null)
                statement.close();
            if(connection != null)
                connection.close();
        }catch (Exception e){
            System.out.println(e);
        }
    }
}

static boolean execute(String query){
    try{
        connection = DriverManager.getConnection(url,username,password);
        statement = connection.createStatement();
        return statement.execute(query);

    }catch (SQLException e){
        System.out.println();
        System.out.println(e.getMessage());
        System.out.println();
    }catch (Exception e){
        System.out.println(e);
    }finally {
        try {
            if(statement != null)
                statement.close();
            if(connection != null)
                connection.close();
        }catch (Exception e){
            System.out.println(e);
        }
    }
    return false;
}
```

```
static int executeUpdate(String query){
    try{
        connection = DriverManager.getConnection(url,username,password);
        statement = connection.createStatement();
        return statement.executeUpdate(query);

    }catch (SQLException e){
        System.out.println();
        System.out.println(e.getMessage());
        System.out.println();
    }catch (Exception e){
        System.out.println(e);
    }finally {
        try {
            if(statement != null)
                statement.close();
            if(connection != null)
                connection.close();
        }catch (Exception e){
            System.out.println(e);
        }
    }
    return -1;
}

static void executeQuery(String query){
    try{
        connection = DriverManager.getConnection(url,username,password);
        statement = connection.createStatement();
        resultSet = statement.executeQuery(query);
        int columns = resultSet.getMetaData().getColumnCount();

        String headers="";
        for(int i=1;i<=columns;i++){
            String a="";
            headers += String.format("%-20s",resultSet.getMetaData().getColumnName(i))+" ";
        }
        System.out.println(headers);

        for(int i=0;i<columns*20;i++)
            System.out.print("-");
        System.out.println();

        while(resultSet.next()){
            String row = "";
            for(int i=1;i<=columns;i++) {
                row += String.format("%-20s",resultSet.getString(i))+" ";
            }
```

```
                System.out.println(row);
            }

        }catch (Exception e){
            System.out.println(e);
        }finally {
            try {
                if(statement != null)
                    statement.close();
                if(connection != null)
                    connection.close();
            }catch (Exception e){
                System.out.println(e);
            }
        }
    }
}
```



```
Mothers_name: wer
insert into aadharcards (Aadharno,Name,Age,voted,Fathers_name,Mothers_name)
values ('12345','prr',23,1,'pre','wer')

Enter Choice: 4

UPDATE BOOK
aadhar: 12345
New name: rr
New age: 3
New vote status: 3
New father's name: er
New Mothers's name: er
update aadharcards set Name='rr',Age=3,voted=3,Fathers_name='er',Mothers_name='er'
where Aadharno='12345'
record Updated

Enter Choice: 5

DELETE record
aadhar: 12345
delete from aadharcards where Aadharno='12345'
record Deleted

Enter Choice: 6
records TABLE
Aadharno            Name                Age                 voted               Fathers_name        Mothers_name
------------------------------------------------------------------------------------------------------------------------

Enter Choice: 1
1. Print Hints
2. Create Table
3. Insert New record
4. Update record
5. Delete record
6. Show Table
0. Exit

Enter Choice: 
```



```
 /usr/lib/jvm/java-11-openjdk-amd64/bin/java -Dfile.encoding=UTF-8 @/tmp/cp_azro66axhw2hoqcujjoekdfo7.argfile Main
(base)  pratt3000@pratts-laptop  ~/Documents/College/PICT_TE-Labs/DBMSL/Assignment_C02  master  /usr/lib/jvm/java-11-openjdk-amd64/bin/java -Dfile.e
ncoding=UTF-8 @/tmp/cp_azro66axhw2hoqcujjoekdfo7.argfile Main
Database connected
MySQL: 8.0.22-0ubuntu0.20.04.3

1. Print Hints
2. Create Table
3. Insert New record
4. Update record
5. Delete record
6. Show Table
0. Exit

Enter Choice: 2

CREATING TABLE
create table aadharcards(
    Aadharno int not null,
    Name varchar(32) not null,
    Age int not null,
    voted int not null,
    Fathers_name varchar(32) not null,
    Mothers_name varchar(32) not null
)

Table 'aadharcards' already exists


Enter Choice: 3

INSERT NEW RECORD
Aadharno: 12345
Name: prr
Age: 23
voted: 1
Fathers_name: pre
Mothers_name: wer
insert into aadharcards (Aadharno,Name,Age,voted,Fathers_name,Mothers_name)
```

Assignment C2

- Title: Java MySQL Connectivity

- Problem Statement
  Implement MySQL database connectivity with Java. Implement Database navigation operations (add, delete, edit) using JDBC.

- Objective:
  - Understand concept of database
  - Understand concept of JDBC
  - Understand database operations.

- Outcome:
  Successfully implement & study database connectivity & perform basic operations using it.

- S/W & H/W Requirements:
  MySQL
  Java/Python/PHP
  64 bit OS, IDE
  8GB RAM, hardware.

- Theory:

MySQL:
- MySQL is the most popular open source Relational SQL database Management System.
- MySQL is one of the best RDMS being used for developing various web-based software applications
- MySQL is developed, marketed & supported ~~&~~ by MySQL which is a Swedish Company.

Establishing JDBC connection.
- JDBC is an advancement of ODBC
- Moves data from frontend to backend.
- Consists of classes & interfaces written in JAVA.
- link between code & database.

ODBC was written in C++, Python, etc so wasnt platform independent. JDBC was written in JAVA & thus is platfor independent.

- loading the driver:
  - class.forName ("com.mysql.cj.jdbc.Driver")
  - DriverManager.registerDriver()

classmate
Date
Page

2) create the connection:

Connection con = DriverManager.getConnection (url, user, password)

3) Create a statement:
These enable you to send SQL commands & receive data.
Statement st = con. createStatement();

4) Execute the query:
- Query for updating/ inserting table in database
- Query for retrieving data.

The executeQuery() method of Statement Interface is used to execute queries of retrieving values from database This method returns the object of ResultSet that can be used to get all the records of a ~~ba~~ table.

The executeupdate (sql query) method of Statement interface is used to execute queries of updating / inserting

5) Close connections:
conn. close()

- Conclusion:
  Successfully established database connectivity using JDBC & performed basic operations using it.

Scanned with CamScanner