JAVA AWT BASED- Online MOOC's year wise student database management system - SQL CONNECTIVITY USING JDBC

A

Report

Submitted in partial fulfilment of the

Requirements for the award of the Degree of

BACHELOR OF ENGINEERING

IN

INFORMATION TECHNOLOGY

By

D. ABHIRAJ GOUD<1602-18-737-061>

Department of Information Technology

Vasavi College of Engineering (Autonomous)

Ibrahimbagh, Hyderabad-31

BONAFIDE CERTIFICATE

This to certify that the project report titled "AIRLINE QUALITY AND INFORMATION MANAGEMENT SYSTEM" project work of Mr.D.ABHIRAJ GOUD bearing Roll.no:1602-18-737-061 who carried out this project under my supervision in the IV semester for the academic year 2019-2020.

B.LEELAVATHY

Signature
external examine

Signature
internal examine

ABSTRACT:

The analysis of student performance in aptitude quiz exam can be done using this application as a mediator interface.

The student details are entered along with the examination center details of aptitude quiz exam. Everyone gets a score card wherein they get the marks of each section and get the result whether they are passed or failed.

The database consists of the student details and the score card of each and every student in that particular list.

Requirement Analysis:

List of tables:

- *Student details*
- *writes/attempts*
- *Aptitude quiz exam*
- *Scores*
- *Marks scored*

List of attributes with their domain types:

Student_details:

NAME:	VARCHAR2(20)
HALLTICKET:	NUMBER (5)
MAIL_ID:	VARCHAR2(20)
CLASS:	VARCHAR2(20)

Apptitude_quiz_exam:

DURATION	NUMBER(5)
HALLTICKET	VARCHAR2(20)
EXAM_CENTER	VARCHAR2(20)

Marks_scored:

LOGICAL_REASONING	NUMBER(5)
VERBAL_ABILITY	NUMBER(5)
QUANTITATIVE_APTITUDE	NUMBER(5)
RESULT	VARCHAR2(10)
SHEET_ID	VARCHAR2(10)

APTITUDE_QUIZ_MARKS:

RANK	NUMBER(10)
SHEET_ID	VARCHAR2(10)
HALLTICKET	VARCHAR2(10)

STUDENT_APTITUDE_QUIZ_EXAM:

EXAM_DATE	DATE
MAIL_ID	VARCHAR2(10)
HALLTICKET	VARCHAR2(10)

ER DIAGRAM:

CLASS

NAME

MAIL_ID

STUDENT_DETAILS

MAIL_ID

PARTIAL

STUDENT_APTITUDE_QUIZ_EXAM

EXAM_DATE

HALLTICKET

TOTAL

DURATION

HALLTICKET

APTITUDE_QUIZ_EXAM

EXAM_CENTRE

TOTAL

SHEET_ID

HALLTICKET

APTITUDE_QUIZ MARKS

RANK

TOTAL

SHEET_ID

LOGICAL_REASONING_SCORE

MARKS_SCORED

QUANTITATIVE_APTITUDE_SCORE

RESULT

VERBAL_ABILITY_SCORE
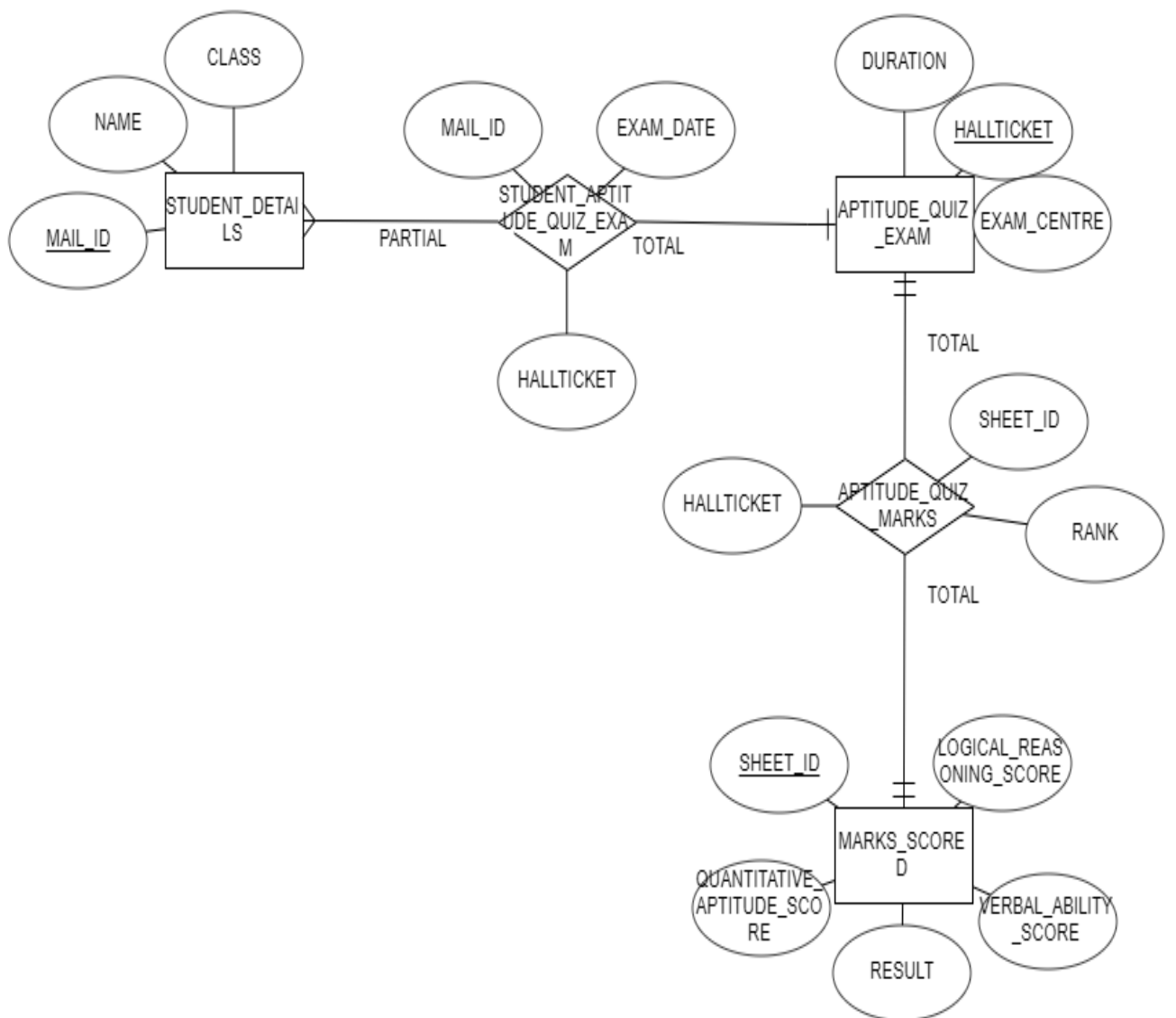
MAPPING CARDINALITIES AND PARTICIPATION CONSTRAINTS:

- One student can give any number of exams, so many to one cardinality suit writes.
- Each student who writes the exam has one score card, so one to one cardinality suit scores.

→ All the students might not write the exam so the student details is partial.

→ All the exams written are related to students so it is total.

→ Marks scored by students in the exam must be related to the exam so it is total participation.

DDL AND DML COMMANDS:

SQL>create table student_details(

    2 hallticket number(5) primary key,

    3 name varchar2(20),

    4 mail_id varchar2(20),

    5 class varchar2(20));

Table created.

SQL> create table aptitude_quiz_exam(

  2 duration number(5),

  3 hallticket varchar2(20),

  4 exam_center varchar2(20));

Table created.

SQL> create table marks_scored(

  2 logical_reasoning number(5),

  3 verbal_ability number(5),

  4 quantitative_aptitude number(5),

  5 result varchar2(20),

  6 hallticket number(5));

Table created.

SQL> create table APTITUDE_QUIZ_MARKS(

  2 rank number(10));

 3 sheet_id varchar2(20);

4 hallticket varchar2(20);

Table created.

SQL> create table STUDENT_APTITUDE_QUIZ_EXAM (

  2 exam_date date);

Table created.

SQL> desc student_details;

Name	Null?	Type
HALLTICKET	NOT NULL	NUMBER(5)
NAME		VARCHAR2(20)
MAIL_ID		VARCHAR2(20)
CLASS		VARCHAR2(20)

```
SQL> desc aptitude_quiz_exam;
Name                                    Null?   Type
--------------------------------------- ------- -------------------
-------
DURATION                                        NUMBER(5)
HALLTICKET                                      VARCHAR2(20)
EXAM_CENTER                                     VARCHAR2(20)

SQL> desc marks_scored;
Name                                    Null?   Type
--------------------------------------- ------- -------------------
-------
LOGICAL_REASONING                               NUMBER(5)
VERBAL_ABILITY                                  NUMBER(5)
QUANTITATIVE_APTITUDE                           NUMBER(5)
RESULT                                          VARCHAR2(20)
HALLTICKET                                      NUMBER(5)
```

```
SQL> desc APTITUDE_QUIZ_MARKS;
Name                                    Null?   Type
--------------------------------------- ------- --------------------
-------
RANK                                            NUMBER(10)
HALLTICKET                                      VARCHAR2(20)
SHEET_ID                                        VARCHAR2(20)


SQL> DESC  STUDENT_APTITUDE_QUIZ_EXAM;
Name                                    Null?   Type
--------------------------------------- ------- --------------------
-------
MAIL_ID                                         VARCHAR2(20)
HALLTICKET                                      VARCHAR2(20)
EXAM_DATE                                       VARCHAR2(20)
SQL> alter table scores(
  2 alter table scores add (hallticket number(5));

Table altered.
```

SQL> alter table writes add(hallticket number(5));

Table altered.

SQL> insert into student_details values(&hallticket,'&name','&mail_id','&class');

Enter value for hallticket: 061

Enter value for name: abhi

Enter value for mail_id: abhi123@gmail.com

Enter value for class: btech 2nd year

old   1: insert into student_details values(&hallticket,'&name','&mail_id','&class')

new   1: insert into student_details values(061,'abhi','abhi123@gmail.com','btech 2nd year')

1 row created.

SQL> /

Enter value for hallticket: 071

Enter value for name: ajay

Enter value for mail_id: ajayband@yahoo.com

Enter value for class: btech 4th year

old   1: insert into student_details values(&hallticket,'&name','&mail_id','&class')

new   1: insert into student_details values(071,'ajay','ajayband@yahoo.com','btech 4th year')

1 row created.

SQL> /

Enter value for hallticket: 88

Enter value for name: badri

Enter value for mail_id: badri143@gmail.com

Enter value for class: btech 3rd year

old   1: insert into student_details values(&hallticket,'&name','&mail_id','&class')

new   1: insert into student_details values(88,'badri','badri143@gmail.com','btech 3rd year')

1 row created.

SQL> /

Enter value for hallticket: 98

Enter value for name: ram

Enter value for mail_id: ramnath@gmail.com

Enter value for class: btech2nd year

old   1: insert into student_details values(&hallticket,'&name','&mail_id','&class')

new   1: insert into student_details values(98,'ram','ramnath@gmail.com','btech2nd year')

1 row created.

SQL> /

Enter value for hallticket: 110

Enter value for name: vaishnavi

Enter value for mail_id: vaisho4@gmail.com

Enter value for class: btech 2nd year

old   1: insert into student_details values(&hallticket,'&name','&mail_id','&class')

new   1: insert into student_details values(110,'vaishnavi','vaisho4@gmail.com','btech 2nd year')

1 row created.

```
SQL> insert into aptitude_quiz_exam values
(&duration,'&hallticket','&exam_center');

Enter value for duration: 3

Enter value for hallticket: 061

Enter value for exam_center: banjara hills

old   1: insert into aptitude_quiz_exam values
(&duration,'&hallticket','&exam_center')

new   1: insert into aptitude_quiz_exam values
(3,'061','banjara hills')

1 row created.

SQL> /

Enter value for duration: 3

Enter value for hallticket: 071

Enter value for exam_center: narsingi

old   1: insert into aptitude_quiz_exam values
(&duration,'&hallticket','&exam_center')

new   1: insert into aptitude_quiz_exam values
(3,'071','narsingi')

1 row created.
```

```
SQL> /
Enter value for duration: 3
Enter value for hallticket: 88
Enter value for exam_center: mehdipatnam
old   1: insert into aptitude_quiz_exam values (&duration,'&hallticket','&exam_center')
new   1: insert into aptitude_quiz_exam values (3,'88','mehdipatnam')

1 row created.

SQL> /
Enter value for duration: 3
Enter value for hallticket: 98
Enter value for exam_center: miyapur
old   1: insert into aptitude_quiz_exam values (&duration,'&hallticket','&exam_center')
new   1: insert into aptitude_quiz_exam values (3,'98','miyapur')

1 row created.
```

SQL> /

Enter value for duration: 3

Enter value for hallticket: 110

Enter value for exam_center: jubliee hills

old   1: insert into aptitude_quiz_exam values (&duration,'&hallticket','&exam_center')

new   1: insert into aptitude_quiz_exam values (3,'110','jubliee hills')

1 row created.

SQL> insert into marks_scored values(&logical_reasoning,&verbal_ability,&quantitative_aptitude,'&result',&hallticket);

Enter value for logical_reasoning: 22

Enter value for verbal_ability: 21

Enter value for quantitative_aptitude: 20

Enter value for result: pass

Enter value for hallticket: 061

old   1: insert into marks_scored values(&logical_reasoning,&verbal_ability,&quantitative_aptitude,'&result',&hallticket)

new   1: insert into marks_scored values(22,21,20,'pass',061)

1 row created.

SQL> /
Enter value for logical_reasoning: 09
Enter value for verbal_ability: 11
Enter value for quantitative_aptitude: 1
Enter value for result: fail
Enter value for hallticket: 071
old   1: insert into marks_scored values(&logical_reasoning,&verbal_ability,&quantitative_aptitude,'&result',&hallticket)
new   1: insert into marks_scored values(09,11,1,'fail',071)

1 row created.

SQL> /
Enter value for logical_reasoning: 19
Enter value for verbal_ability: 16
Enter value for quantitative_aptitude: 14
Enter value for result: pass

Enter value for hallticket: 88

old 1: insert into marks_scored values(&logical_reasoning,&verbal_ability,&quantitative_aptitude,'&result',&hallticket)

new 1: insert into marks_scored values(19,16,14,'pass',88)

1 row created.

SQL> /

Enter value for logical_reasoning: 3

Enter value for verbal_ability: 2

Enter value for quantitative_aptitude: 4

Enter value for result: fail

Enter value for hallticket: 98

old 1: insert into marks_scored values(&logical_reasoning,&verbal_ability,&quantitative_aptitude,'&result',&hallticket)

new 1: insert into marks_scored values(3,2,4,'fail',98)

1 row created.

SQL> /

Enter value for logical_reasoning: 24

Enter value for verbal_ability: 25

Enter value for quantitative_aptitude: 23

Enter value for result: pass

Enter value for hallticket: 110

old   1: insert into marks_scored values(&logical_reasoning,&verbal_ability,&quantitative_aptitude,'&result',&hallticket)

new   1: insert into marks_scored values(24,25,23,'pass',110)

1 row created.

SQL> insert into writes values('&exam_date',&hall_ticket);

Enter value for exam_date: 03-FEB-2020

Enter value for hall_ticket: 061

old   1: insert into writes values('&exam_date',&hall_ticket)

new   1: insert into writes values('03-FEB-2020',061)

1 row created.

SQL> /

Enter value for exam_date: 03-FEB-2020

Enter value for hall_ticket: 071

old   1: insert into writes values('&exam_date',&hall_ticket)

new   1: insert into writes values('03-FEB-2020',071)

1 row created.

SQL> /

Enter value for exam_date: 03-FEB-2020

Enter value for hall_ticket: 88

old   1: insert into writes values('&exam_date',&hall_ticket)

new   1: insert into writes values('03-FEB-2020',88)

1 row created.

SQL> /

Enter value for exam_date: 19-OCT-2019

Enter value for hall_ticket: 98

old   1: insert into writes values('&exam_date',&hall_ticket)

new   1: insert into writes values('19-OCT-2019',98)

1 row created.

SQL> /

Enter value for exam_date: 03-FEB-2020

Enter value for hall_ticket: 110

old   1: insert into writes values('&exam_date',&hall_ticket)

new   1: insert into writes values('03-FEB-2020',110)

1 row created.

SQL> INSERT INTO scores values(&rank,&hallticket);

Enter value for rank: 1

Enter value for hallticket: 061

old   1: INSERT INTO scores values(&rank,&hallticket)

new   1: INSERT INTO scores values(1,061)

1 row created.

SQL> /

Enter value for rank: 9999

Enter value for hallticket: 071

old   1: INSERT INTO scores values(&rank,&hallticket)

new   1: INSERT INTO scores values(9999,071)

1 row created.

SQL> /

Enter value for rank: 179767246

Enter value for hallticket: 88

old   1: INSERT INTO scores values(&rank,&hallticket)

new   1: INSERT INTO scores values(179767246,88)

1 row created.

SQL> /

Enter value for rank: 97536

Enter value for hallticket: 98

old   1: INSERT INTO scores values(&rank,&hallticket)

new   1: INSERT INTO scores values(97536,98)

1 row created.

SQL> /

Enter value for rank: 432

Enter value for hallticket: 110

old   1: INSERT INTO scores values(&rank,&hallticket)

new   1: INSERT INTO scores values(432,110)

1 row created.

SQL> select * from student_details;

HALLTICKET	NAME	MAIL_ID	CLASS
61	abhi	abhi123@gmail.com	btech 2nd year
71	ajay	ajayband@yahoo.com	btech 4th year
88	badri	badri143@gmail.com	btech 3rd year
98	ram	ramnath@gmail.com	btech2ndyear
110	vaishnavi	vaish04@gmail.com	btech2ndyear

SQL> select * from aptitude_quiz_exam;

DURATION	HALLTICKET	EXAM_CENTER
3	061	banjara hills
3	071	narsingi
3	88	mehdipatnam

3	98	miyapur
3	110	jubliee hills

SQL> select * from marks_scored;

```
LOGICAL_REASONING VERBAL_ABILITY
QUANTITATIVE_APTITUDE RESULT
--------------- ------------- ------------------ ----------------
--
HALLTICKET
----------
```

LOGICAL_REASONING	VERBAL_ABILITY	QUANTITATIVE_APTITUDE	RESULT
22	21	20	pass
61			
9	11	1	fail
71			
19	16	14	pass
88			

```
LOGICAL_REASONING VERBAL_ABILITY
QUANTITATIVE_APTITUDE RESULT
```

```
---------------- ------------ ------------------ ----------------
--

HALLTICKET

----------

         3         2              4 fail
    98


        24        25             23 pass
    110
```

SQL> select * from STUDENT_APTITUDE_QUIZ_EXAM;

```
MAIL_ID            HALLTICKET         EXAM_DATE

------------------ ------------------ ------------------
```
abhi123@gmail.com   061   03-FEB-2020

badri143@gmail.com   88   03-FEB-2020

ajayband@yahoo.com   071   03-FEB-2020

ramnath@gmail.com   98   27-FEB-2020

vaish04@gmail.com   110   03-FEB-2020

SQL> select * from APTITUDE_QUIZ_MARKS;

```
     RANK HALLTICKET           SHEET_ID
---------- ------------------- -------------------
         1         061          123
      9999         071          134
 179767246          88          145
     97536          98          189
       432         110          210
```

IMPLEMENTATION

FRONT END PROGRAMS

1)insert aptitude quiz exam details:

```java
package aptitude_analysis;

import java.awt.FlowLayout;

import java.awt.GridLayout;

import java.awt.TextArea;

import java.awt.event.ActionEvent;

import java.awt.event.ActionListener;

import java.sql.Connection;

import java.sql.DriverManager;

import java.sql.SQLException;

import javax.swing.JButton;

import javax.swing.JFrame;

import javax.swing.JLabel;

import javax.swing.JOptionPane;

import javax.swing.JPanel;

import javax.swing.JTextField;

public class insert_aptitude_quiz_exam extends JFrame {
```

```java
/**
 *
 */
private static final long serialVersionUID = 1L;
JPanel jp1,jp2,jp3;
Connection con;
int i;
java.sql.Statement stmt;
JLabel du;
JLabel ht;
JLabel ec;
JTextField d,h,e;
TextArea ta;
JButton in;
void displaySQLErrors(SQLException e) {
ta.append("\nSQLException:" + e.getMessage() +"\n");
ta.append("SQLState:   "+ e.getSQLState() + "\n");
ta.append("VendorError:  " + e.getErrorCode() + "\n");

}
```

```java
public insert_aptitude_quiz_exam()
{
    try {
        Class.forName("oracle.jdbc.driver.OracleDriver");
    } catch (ClassNotFoundException e) {
        // TODO Auto-generated catch block
        e.printStackTrace();
    }
    try {

con=DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:ORCL","project","vasavi");
    } catch (SQLException e) {
        // TODO Auto-generated catch block
        displaySQLErrors(e);
    }
    try {
        stmt=con.createStatement();
    } catch (SQLException e) {
        // TODO Auto-generated catch block
```

```java
    displaySQLErrors(e);
}
du=new JLabel("Duration of exam");
d=new JTextField(10);
ht=new JLabel("Hall-ticket");
h=new JTextField(10);
ec=new JLabel("Exam-center");
e=new JTextField(10);
ta=new TextArea(20,100);
in=new JButton("submit");
jp1=new JPanel(new FlowLayout());
jp2=new JPanel(new FlowLayout());
jp3=new JPanel(new FlowLayout());
jp1.add(du);
jp1.add(d);
jp1.add(ht);
jp1.add(h);
jp1.add(ec);
jp1.add(e);
jp2.add(in);
```

```java
            jp3.add(ta);
            add(jp1);
            add(jp2);
            add(jp3);
            setVisible(true);
            setSize(2000,1000);
            setTitle("Enter following details:");
            setLayout(new FlowLayout());
            pack();
    in.addActionListener(new ActionListener() {

            @Override
            public void actionPerformed(ActionEvent argo) {
                // TODO Auto-generated method stub
                try {
                    String duration=d.getText();
                    if(checkint(duration)==0) {
                        JOptionPane.showMessageDialog(null,
"Duration should be an Integer");
                        throw new Exception();
                    }
```

```java
            i=stmt.executeUpdate("insert into
aptitude_quiz_exam values ("+ d.getText()
+",'"+h.getText()+"','"+e.getText()+"')");
        } catch (SQLException e) {
            // TODO Auto-generated catch block
            displaySQLErrors(e);
        } catch (Exception e1) {
            // TODO Auto-generated catch block
            e1.printStackTrace();
        }
        ta.append("\n Inserted "+i+"rows successfully");
    }
    });
}
public int checkint(String rank) {
try {
    Integer.parseInt(rank);
    return 1;
}
catch(Exception e) {
    return 0;
```

```
            }


            }


}
```

2)update aptitude quiz exam details

package aptitude_analysis;

```java
import java.awt.FlowLayout;

import java.awt.GridLayout;

import java.awt.List;

import java.awt.TextArea;

import java.awt.event.ActionEvent;

import java.awt.event.ActionListener;

import java.awt.event.ItemEvent;

import java.awt.event.ItemListener;

import java.sql.Connection;

import java.sql.DriverManager;

import java.sql.ResultSet;

import java.sql.SQLException;


import javax.swing.JButton;

import javax.swing.JFrame;

import javax.swing.JLabel;

import javax.swing.JPanel;

import javax.swing.JTextField;

public class update_aptitude_quiz_exam extends JFrame{
```

```java
/**
 *
 */
private static final long serialVersionUID = 1L;
JPanel jp1,jp2,jp3;
Connection con;
int i;
java.sql.Statement stmt;
JLabel du;
JLabel ht;
JLabel ec;
JTextField d,h,e;
TextArea ta;
JButton in;
List lis;
ResultSet rs;
String sel;
void displaySQLErrors(SQLException e) {
    ta.append("\nSQLException:" + e.getMessage() +"\n");
    ta.append("SQLState:    "+ e.getSQLState() + "\n");
```

```java
        ta.append("VendorError:  " + e.getErrorCode() + "\n");

    }

    public update_aptitude_quiz_exam()
    {
        try {
            Class.forName("oracle.jdbc.driver.OracleDriver");
        } catch (ClassNotFoundException e) {
            // TODO Auto-generated catch block
            e.printStackTrace();
        }
        try {

con=DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:ORCL","project","vasavi");
        } catch (SQLException e) {
            // TODO Auto-generated catch block
            e.printStackTrace();
        }
        try {
```

```java
        stmt=con.createStatement();
    } catch (SQLException e) {
        // TODO Auto-generated catch block
        e.printStackTrace();
    }
    lis=new List();
    du=new JLabel("Duration of exam");
    d=new JTextField(10);
    ht=new JLabel("Hall-ticket");
    h=new JTextField(10);
    ec=new JLabel("Exam-center");
    e=new JTextField(10);
    ta=new TextArea(20,100);
    in=new JButton("submit");
    jp1=new JPanel(new FlowLayout());
    jp2=new JPanel(new FlowLayout());
    jp3=new JPanel(new FlowLayout());
    jp1.add(du);
    jp1.add(d);
    jp1.add(ht);
```

```java
        jp1.add(h);
        jp1.add(ec);
        jp1.add(e);
        jp2.add(in);
        jp3.add(ta);
        add(jp1);
        add(jp2);
        add(jp3);
        add(lis);
        try {
            rs=stmt.executeQuery("select hallticket from aptitude_quiz_exam");
            while(rs.next()) {
                lis.add(rs.getString(1));
            }
        } catch (SQLException e) {
            // TODO Auto-generated catch block
            displaySQLErrors(e);
        }
        lis.addItemListener(new ItemListener() {
```

```java
@Override
public void itemStateChanged(ItemEvent argo) {
    // TODO Auto-generated method stub
    try {
        sel=lis.getSelectedItem();
        rs=stmt.executeQuery("select duration,hallticket,exam_center from aptitude_quiz_exam where hallticket="+lis.getSelectedItem()+"");
        if(rs.next()) {
            d.setText(rs.getString(1));
            h.setText(rs.getString(2));
            e.setText(rs.getString(3));
        }
    } catch (SQLException e) {
        // TODO Auto-generated catch block
        displaySQLErrors(e);
    }
}
});
setVisible(true);
```

```java
        setSize(2000,1000);

        setTitle("Enter following details:");

        setLayout(new FlowLayout());

        pack();


in.addActionListener(new ActionListener() {


        @Override
        public void actionPerformed(ActionEvent argo) {
            // TODO Auto-generated method stub
            try {
                i=stmt.executeUpdate("update
aptitude_quiz_exam set
duration="+d.getText()+",exam_center='"+e.getText()+"'wh
ere hallticket='"+h.getText()+"'");

            } catch (SQLException e) {
                // TODO Auto-generated catch block
                displaySQLErrors(e);
            }
            ta.append("\n Updated "+i+"rows successfully");
          }
        });
```

```
    }

}
```

3)delete aptitude quiz exam details

```java
package aptitude_analysis;

import java.awt.FlowLayout;

import java.awt.GridLayout;

import java.awt.List;

import java.awt.TextArea;

import java.awt.event.ActionEvent;

import java.awt.event.ActionListener;

import java.awt.event.ItemEvent;

import java.awt.event.ItemListener;

import java.sql.Connection;

import java.sql.DriverManager;

import java.sql.ResultSet;

import java.sql.SQLException;

import javax.swing.JButton;

import javax.swing.JFrame;
```

```java
import javax.swing.JLabel;
import javax.swing.JPanel;
import javax.swing.JTextField;

public class delete_aptitude_quiz_exam extends JFrame{

    /**
     *
     */
    private static final long serialVersionUID = 1L;
    JPanel jp1,jp2,jp3;
    Connection con;
    int i;
    java.sql.Statement stmt;
    JLabel du;
    JLabel ht;
    JLabel ec;
    JTextField d,h,e;
    TextArea ta;
    JButton in;
    List lis;
```

```java
ResultSet rs;
String sel;

public delete_aptitude_quiz_exam()
{
    try {
        Class.forName("oracle.jdbc.driver.OracleDriver");
    } catch (ClassNotFoundException e) {
        // TODO Auto-generated catch block
        e.printStackTrace();
    }
    try {

con=DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:ORCL","project","vasavi");
    } catch (SQLException e) {
        // TODO Auto-generated catch block
        e.printStackTrace();
    }
    try {
        stmt=con.createStatement();
```

```java
} catch (SQLException e) {
    // TODO Auto-generated catch block
    e.printStackTrace();
}
lis=new List();
du=new JLabel("Duration of exam");
d=new JTextField(10);
ht=new JLabel("Hall-ticket");
h=new JTextField(10);
ec=new JLabel("Exam-center");
e=new JTextField(10);
ta=new TextArea(20,100);
in=new JButton("Delete");
jp1=new JPanel(new FlowLayout());
jp2=new JPanel(new FlowLayout());
jp3=new JPanel(new FlowLayout());
jp1.add(du);
jp1.add(d);
jp1.add(ht);
jp1.add(h);
```

```java
        jp1.add(ec);

        jp1.add(e);

        jp2.add(in);

        jp3.add(ta);

        add(jp1);

        add(jp2);

        add(jp3);

        add(lis);

        try {

            rs=stmt.executeQuery("select hallticket from
aptitude_quiz_exam");

            while(rs.next()) {

                lis.add(rs.getString(1));

            }

        } catch (SQLException e) {

            // TODO Auto-generated catch block

            e.printStackTrace();

        }

        lis.addItemListener(new ItemListener() {

            @Override
```

```java
public void itemStateChanged(ItemEvent argo) {
    // TODO Auto-generated method stub
    try {
        sel=lis.getSelectedItem();
        rs=stmt.executeQuery("select duration,hallticket,exam_center from aptitude_quiz_exam where hallticket="+lis.getSelectedItem()+"");
        if(rs.next()) {
            d.setText(rs.getString(1));
            h.setText(rs.getString(2));
            e.setText(rs.getString(3));
        }
    } catch (SQLException e) {
        // TODO Auto-generated catch block
        e.printStackTrace();
    }
}

});
setVisible(true);
setSize(2000,1000);
```

```java
            setTitle("Enter following details:");
            setLayout(new FlowLayout());
            pack();

    in.addActionListener(new ActionListener() {

            @Override
            public void actionPerformed(ActionEvent argo) {
                // TODO Auto-generated method stub
                try {
                    i=stmt.executeUpdate("delete from
aptitude_quiz_exam where hallticket='"+h.getText()+"'");
                } catch (SQLException e) {
                    // TODO Auto-generated catch block
                    e.printStackTrace();
                }
                ta.append("\n deleted "+i+"rows successfully");
            }
        });
    }
```

```java
}
```

4)Main Method

```java
package aptitude_analysis;

import java.awt.FlowLayout;

import java.awt.event.ActionEvent;

import java.awt.event.ActionListener;

import javax.swing.JFrame;

import javax.swing.JMenu;

import javax.swing.JMenuBar;

import javax.swing.JMenuItem;

public class First_frame extends JFrame {

    /**
     *
     */
    private static final long serialVersionUID = 1L;
    JMenuBar mnubar;
    JMenu m1,m2,m3,m4,m5;
```

```java
    JMenuItem
in1,in2,in3,in4,in5,up1,up2,up3,up4,up5,dl1,dl2,dl3,dl4,dl5;
    public First_frame() {
        mnubar=new JMenuBar();
        m1=new JMenu("Student_Details");
        m2=new JMenu("Aprtitude_Quiz_Exam");
        m3=new JMenu("Marks_scored");
        m4=new JMenu("Aptitude_quiz_marks");
        m5=new JMenu("Student_Aptitude_quiz_exam");
        in1=new JMenuItem("Insert");
        up1=new JMenuItem("Update");
        dl1=new JMenuItem("Delete");
        in2=new JMenuItem("Insert");
        up2=new JMenuItem("Update");
        dl2=new JMenuItem("Delete");
        in3=new JMenuItem("Insert");
        up3=new JMenuItem("Update");
        dl3=new JMenuItem("Delete");
        in4=new JMenuItem("Insert");
        up4=new JMenuItem("Update");
```

```java
dl4=new JMenuItem("Delete");
in5=new JMenuItem("Insert");
up5=new JMenuItem("Update");
dl5=new JMenuItem("Delete");
setVisible(true);
setSize(2000,1000);
setTitle("Aptitude Quiz Performance Analysis");
setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
setLayout(new FlowLayout());
setJMenuBar(mnubar);
mnubar.add(m1);
m1.add(in1);
m1.add(up1);
m1.add(dl1);
mnubar.add(m2);
m2.add(in2);
m2.add(up2);
m2.add(dl2);
mnubar.add(m3);
m3.add(in3);
```

```java
        m3.add(up3);

        m3.add(dl3);

        mnubar.add(m4);

        m4.add(in4);

        m4.add(up4);

        m4.add(dl4);

        mnubar.add(m5);

        m5.add(in5);

        m5.add(up5);

        m5.add(dl5);


        in1.addActionListener(new ActionListener() {

            @Override
            public void actionPerformed(ActionEvent e) {
                // TODO Auto-generated method stub
                new insert_student_details();
            }
        });
        up1.addActionListener(new ActionListener() {
```

```java
            @Override
            public void actionPerformed(ActionEvent argo) {
                // TODO Auto-generated method stub
                new update_student_details();

            }
        });
        dl1.addActionListener(new ActionListener() {

            @Override
            public void actionPerformed(ActionEvent e) {
                // TODO Auto-generated method stub
                new delete_student_details();

            }
        });
        in2.addActionListener(new ActionListener() {

            @Override
            public void actionPerformed(ActionEvent argo) {
                // TODO Auto-generated method stub
```

```java
            new insert_aptitude_quiz_exam();
        }
    });
    up2.addActionListener(new ActionListener() {

        @Override
        public void actionPerformed(ActionEvent argo) {
            // TODO Auto-generated method stub
            new update_aptitude_quiz_exam();

        }
    });
    dl2.addActionListener(new ActionListener() {

        @Override
        public void actionPerformed(ActionEvent e) {
            // TODO Auto-generated method stub
            new delete_aptitude_quiz_exam();

        }
    });
```

```java
in3.addActionListener(new ActionListener() {

    @Override
    public void actionPerformed(ActionEvent argo) {
        // TODO Auto-generated method stub
        new insert_marks_scored();

    }
});
up3.addActionListener(new ActionListener() {

    @Override
    public void actionPerformed(ActionEvent argo) {
        // TODO Auto-generated method stub
        new update_marks_scored();

    }
});
dl3.addActionListener(new ActionListener() {

    @Override
    public void actionPerformed(ActionEvent e) {
```

```java
        // TODO Auto-generated method stub
        new delete_marks_scored();

    }
});
in4.addActionListener(new ActionListener() {

    @Override
    public void actionPerformed(ActionEvent argo) {
        // TODO Auto-generated method stub
        new insert_aptitude_quiz_marks();

    }
});
up4.addActionListener(new ActionListener() {

    @Override
    public void actionPerformed(ActionEvent argo) {
        // TODO Auto-generated method stub
        new update_aptitude_quiz_marks();

    }
```

```java
});
dl4.addActionListener(new ActionListener() {

    @Override
    public void actionPerformed(ActionEvent e) {
        // TODO Auto-generated method stub
        new delete_aptitude_quiz_marks();

    }
});
in5.addActionListener(new ActionListener() {

    @Override
    public void actionPerformed(ActionEvent argo) {
        // TODO Auto-generated method stub
        new insert_student_Aptitude_quiz_exam();

    }
});
up5.addActionListener(new ActionListener() {

    @Override
```

```java
        public void actionPerformed(ActionEvent argo) {
            // TODO Auto-generated method stub
            new update_student_Aptitude_quiz_exam();

        }
    });
    dl5.addActionListener(new ActionListener() {

        @Override
        public void actionPerformed(ActionEvent e) {
            // TODO Auto-generated method stub
            new delete_student_Aptitude_quiz_exam();

        }
    });

}


public static void main(String a[]) {
    new First_frame();
}
```

```
}
```

Connectivity with the Database:

Java Database Connectivity (JDBC) is an application programming interface (API) for the programming language Java, which defines how a client may access a database. It is a Java-based data access technology used for Java database connectivity. It is part of the Java Standard Edition platform, from Oracle Corporation. It provides methods to query and update data in a database and is oriented towards relational databases.

Block of code for JAVA- SQL connectivity with JDBC:

```
public void connectToDB()
        {
                try
                {

connection=DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:orcl","project","vasavi");

statement=connection.createStatement();
                }
```

```java
        catch(SQLException connectException)
        {

System.out.println(connectException.getMessage());

System.out.println(connectException.getSQLState());

System.out.println(connectException.getErrorCode());
            System.exit(1);
        }
    }
```
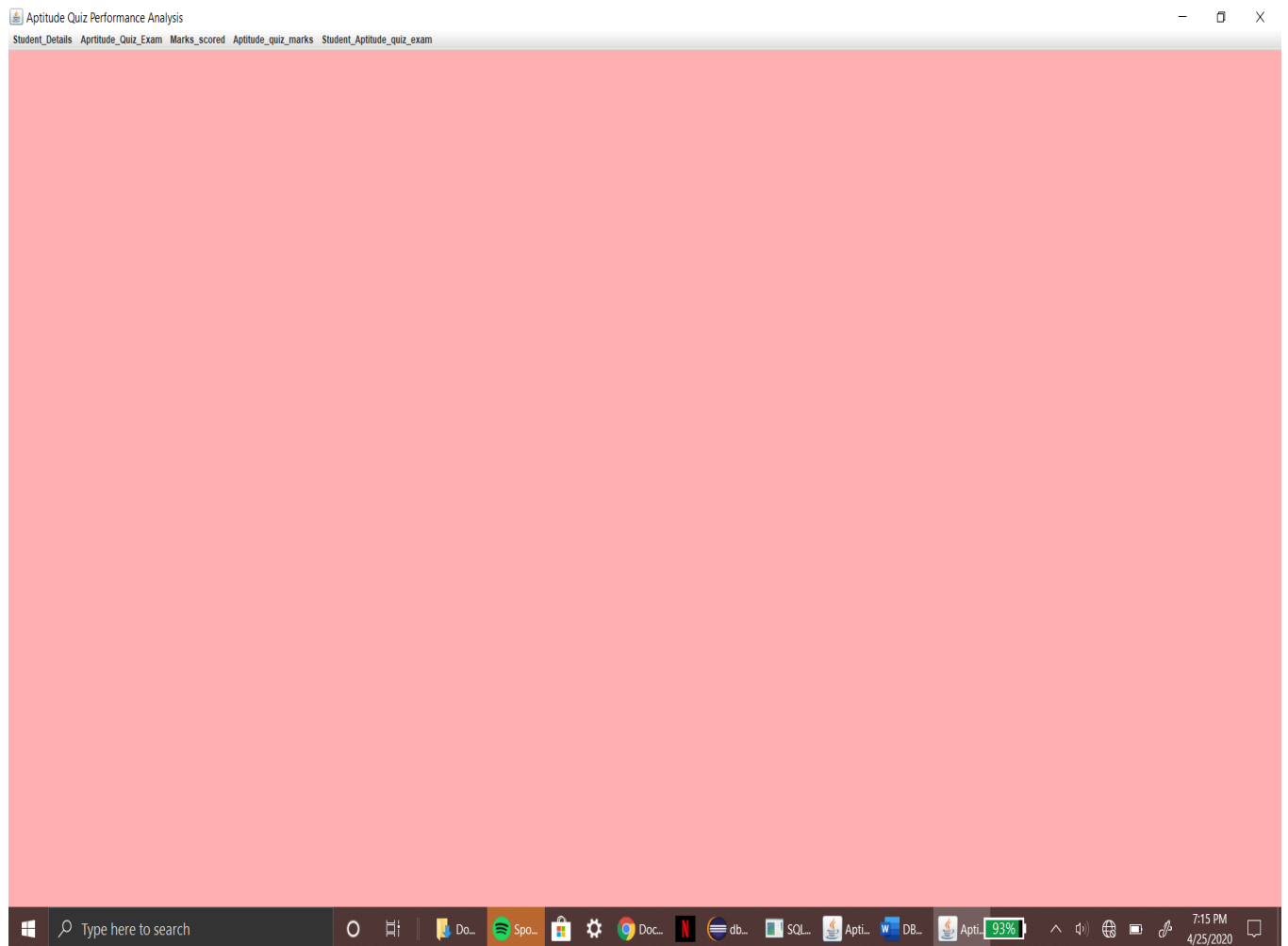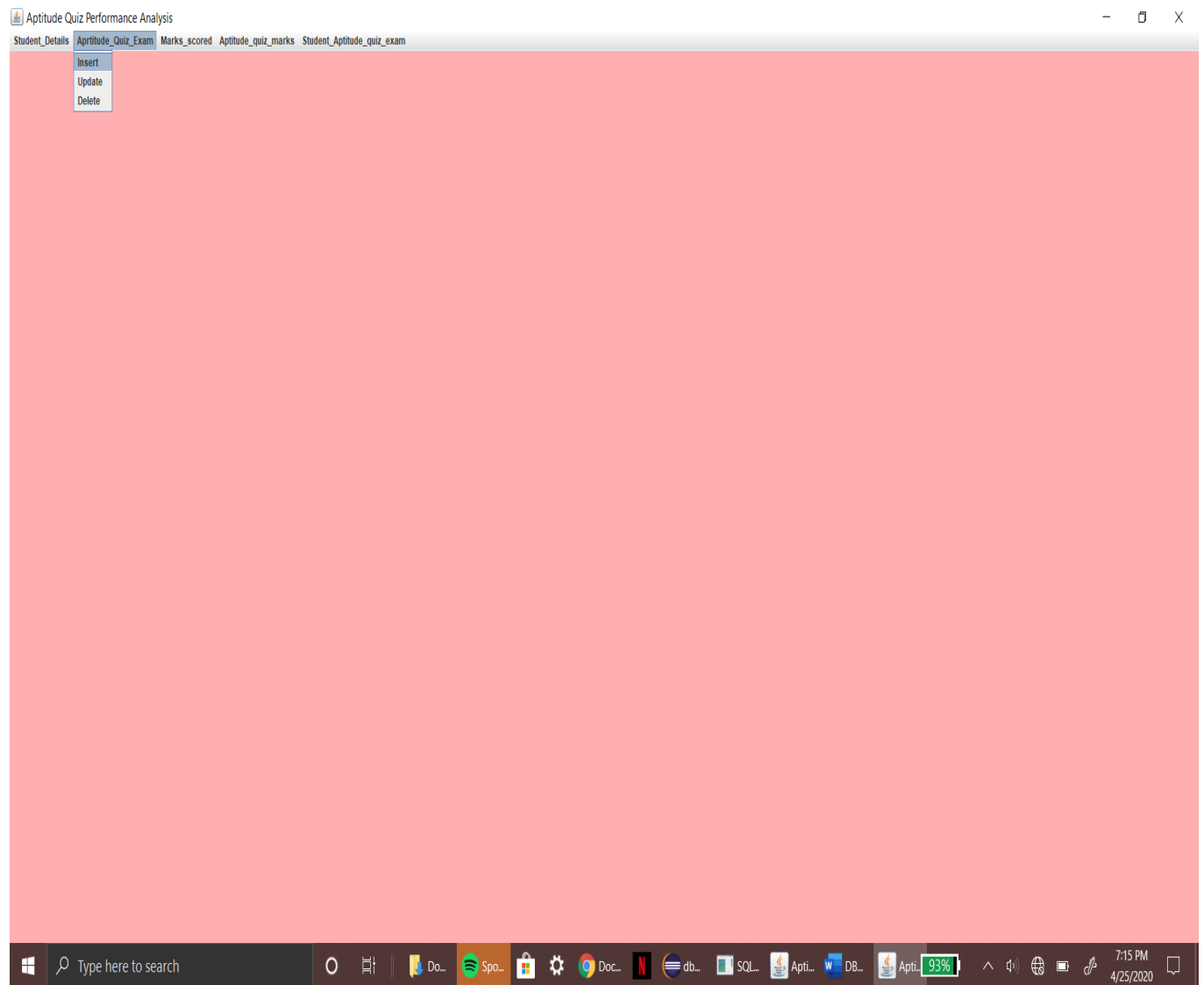
TESTING

The program executes three basic operations those are insert update and delete on 5 different tables. Along with this, it

also has an output column which gives information about how many rows have been edited. Errors syntactical or exceptional will be shown if occurred.

HOME PAGE:

Student_Details Aprtitude_Quiz_Exam Marks_scored Aptitude_quiz_marks Student_Aptitude_quiz_exam

Insert
Update
Delete

INSERT QUIZ DETAILS:

Enter following details:

| Duration of exam | 4 | Hall-ticket | 121 | Exam-center | Kondapur | submit |

Inserted 1rows successfully

```
SQL Plus
SQL> select * from aptitude_quiz_exam;

  DURATION HALLTICKET          EXAM_CENTER
---------- ------------------- -------------------
         4 66                  canada
         3 061                 USA
         3 071                 narsingi
         3 88                  mehdipatnam
         3 98                  miyapur
         3 110                 jubliee hills

6 rows selected.

SQL>
```

UPDATE QUIZ DETAILS:

Enter following details:

Duration of exam	4
Hall-ticket	121
Exam-center	Tarnaka

Updated 1rows successfully

```
061
071
110
121
66
88
98
```

```
SQL> select * from aptitude_quiz_exam;

  DURATION HALLTICKET           EXAM_CENTER
---------- -------------------- --------------------
         4 66                   canada
         4 121                  Tarnaka
         3 061                  USA
         3 071                  narsingi
         3 88                   mehdipatnam
         3 98                   miyapur
         3 110                  jubliee hills

7 rows selected.

SQL>
```

DELETE QUIZ DETAILS:

Enter following details:

Duration of exam	4
Hall-ticket	121
Exam-center	Tarnaka

deleted 1rows successfully

```
061
071
110
121
66
88
98
```

```
QL> select * from aptitude_quiz_exam;

 DURATION HALLTICKET          EXAM_CENTER
--------- -------------------- --------------------
        4 66                   canada
        4 121                  Tarnaka
        3 061                  USA
        3 071                  narsingi
        3 88                   mehdipatnam
        3 98                   miyapur
        3 110                  jubliee hills

 rows selected.

QL>
```

DAILOG BOX:

Enter following details:

| Duration of exam | SIX | Hall-ticket | 200 | Exam-center | UPPAL | submit |

Inserted 0rows successfully

Message

(i) Duration should be an Integer

OK

EXCEPTIONS:

Enter following details:

Duration of exam [6] Hall-ticket [9123455666999999] Exam-center [KARIMNAGAR] [submit]

SQLException:ORA-12899: value too large for column "PROJECT"."APTITUDE_QUIZ_EXAM"."HALLTICKET" (actual: 24, maximu

SQLState: 72000
VendorError: 12899

Inserted 0rows successfully