

# JAVA SWINGS BASED- SIH IT PORTAL

---

## - SQL CONNECTIVITY USING JDBC

A

*Report*

*Submitted in partial fulfillment of the*

*Requirements for the award of the*

*Degree of*

## BACHELOR OF ENGINEERING

---

### IN

INFORMATION TECHNOLOGY

By

P HARSHITH <1602-20-737-016 >

Under the guidance of Ms B. Leelavathy



Department of Information Technology Vasavi College of  
Engineering (Autonomous) (Affiliated to Osmania  
University) Ibrahimbagh, Hyderabad-31

## BONAFIDE CERTIFICATE

This is to certify that this project report titled '**COLLEGE NETWORK DATABASE**' is a project work of P HARSHITH bearing roll no. 1602-20-737-016 who carried out the project under my supervision in the IV semester for the academic year 2021- 2022.

Signature

Internal Examiner

Signature

External

## COLLEGE NETWORK DATABASE

### ABSTRACT:

College Network database is designed for establishing the internet network on the campus. College Network database is a project which can be usefully established for the network in any college or campus. In this project, ip address of each specific department is shown. Each department has unique ip address. In this project, number of devices connected to specific department and amount of data used by respective department is shown. In this project, the data usage per each day of whole campus is shown and finally a graph of devices and network usage in the college network is generated.

### REQUIREMENT ANALYSIS:

List of Tables:

- Departments
- Data\_usage
- Graph

### LIST OF ATTRIBUTES WITH THEIR DOMAIN TYPES:

Departments:

```
mysql> desc departments;
```

Field	Type	Null	Key	Default	Extra
did	int	NO	PRI	NULL	
dname	varchar(20)	YES		NULL	
ip_address	varchar(15)	YES		NULL	

Data\_usage:

```
mysql> desc data_usage;
```

Field	Type	Null	Key	Default	Extra
did	int	YES	MUL	NULL	
day	int	NO		NULL	
devices_connected	int	YES		NULL	
data_bandwidth_GB	float(10,5)	YES		NULL	
date	date	YES		NULL	

5 rows in set (0.00 sec)

Graph:

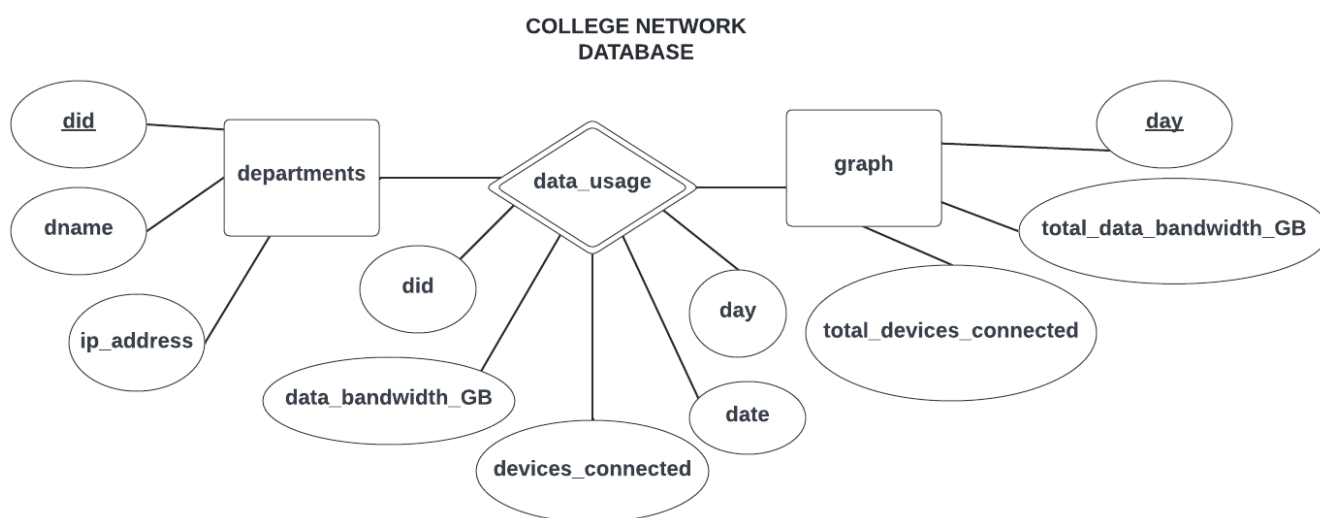
```
mysql> desc graph;
```

Field	Type	Null	Key	Default	Extra
day	int	NO	PRI	NULL	
total_devices_connected	int	YES		NULL	
total_data_bandwidth_GB	float(15,5)	YES		NULL	

3 rows in set (0.01 sec)

## DESIGN

### Entity Relationship Diagram



### DDL OPERATIONS:

1)

```
mysql> create table department(
  -> did int(5), dname varchar(20), ip_address varchar(15), primary key(did));
Query OK, 0 rows affected, 1 warning (0.06 sec)
```

```
mysql> desc department;
```

Field	Type	Null	Key	Default	Extra
did	int	NO	PRI	NULL	
dname	varchar(20)	YES		NULL	
ip_address	varchar(15)	YES		NULL	

3 rows in set (0.01 sec)

2)

```
mysql> create table data_usage(did int(5), day varchar(15), date varchar(15),
devices_connected int(4),data_bandwidth_GB float(10,5),foreign key(did) references departments(did));
Query OK, 0 rows affected, 3 warnings (0.07 sec)
```

```
mysql> desc data_usage
-> ;
```

Field	Type	Null	Key	Default	Extra
did	int	YES	MUL	NULL	
day	varchar(15)	YES		NULL	
date	varchar(15)	YES		NULL	
devices_connected	int	YES		NULL	
data_bandwidth_GB	float(10,5)	YES		NULL	

5 rows in set (0.01 sec)

```
mysql> alter table data_usage drop column date;
Query OK, 0 rows affected (0.05 sec)
Records: 0 Duplicates: 0 Warnings: 0
```

```
mysql> alter table data_usage add(date date);
Query OK, 0 rows affected (0.04 sec)
Records: 0 Duplicates: 0 Warnings: 0
```

```
mysql> desc data_usage
-> ;
```

Field	Type	Null	Key	Default	Extra
did	int	YES	MUL	NULL	
day	varchar(15)	YES		NULL	
devices_connected	int	YES		NULL	
data_bandwidth_GB	float(10,5)	YES		NULL	
date	date	YES		NULL	

5 rows in set (0.01 sec)

### 3)

```
mysql> create table graph(day int(5), devices_connected int(5), total_data_bandwidth_GB float(15,5),
foreign key (day) references data_usage(day));
Query OK, 0 rows affected, 3 warnings (0.06 sec)
```

```
mysql> desc graph;
```

Field	Type	Null	Key	Default	Extra
day	int	YES	MUL	NULL	
devices_connected	int	YES		NULL	
total_data_bandwidth_GB	float(15,5)	YES		NULL	

3 rows in set (0.01 sec)

## DML OPERATIONS:

### 1) Inserting values into departments table:

```
mysql> insert into department values(732,'CIVIL','192.1.2.0');
Query OK, 1 row affected (0.01 sec)
```

```
mysql> insert into department values(733,'CSE','192.1.2.64');
Query OK, 1 row affected (0.01 sec)
```

```
mysql> insert into departments values(736,'MECHANICAL','192.1.2.256');
Query OK, 1 row affected (0.01 sec)

mysql> insert into departments values(737,'IT','192.1.2.320');
Query OK, 1 row affected (0.01 sec)

mysql> insert into departments values(748,'CSE-AIML','192.1.2.384');
Query OK, 1 row affected (0.01 sec)

mysql> insert into departments values(100,'CAFETERIA','192.1.2.448');
Query OK, 1 row affected (0.01 sec)

mysql> insert into departments values(100,'LIBRARY','192.1.2.512');
ERROR 1062 (23000): Duplicate entry '100' for key 'departments.PRIMARY'
mysql> insert into departments values(200,'LIBRARY','192.1.2.512');
Query OK, 1 row affected (0.01 sec)
```

```
mysql> select * from departments;
+-----+-----+-----+
| did | dname      | ip_address |
+-----+-----+-----+
| 100 | CAFETERIA  | 192.1.2.448 |
| 200 | LIBRARY    | 192.1.2.512 |
| 732 | CIVIL      | 192.1.2.0   |
| 733 | CSE        | 192.1.2.64  |
| 734 | EEE        | 192.1.2.128 |
| 735 | ECE        | 192.1.2.192 |
| 736 | MECHANICAL | 192.1.2.256 |
| 737 | IT         | 192.1.2.320 |
| 748 | CSE-AIML   | 192.1.2.384 |
+-----+-----+-----+
9 rows in set (0.01 sec)
```

## 2) Inserting values into data\_usage table:

```
mysql> insert into data_usage values(732,1,46,94.694,'2022-04-17');
Query OK, 1 row affected (0.01 sec)

mysql> insert into data_usage values(736,2,87,60.194,'2022-04-20');
Query OK, 1 row affected (0.01 sec)

mysql> insert into data_usage values(733,2,27,35.7894,'2022-04-20');
Query OK, 1 row affected (0.01 sec)

mysql> insert into data_usage values(100,2,67,45.4673,'2022-04-20');
Query OK, 1 row affected (0.01 sec)

mysql> insert into data_usage values(735,2,54,90.6473,'2022-04-20');
Query OK, 1 row affected (0.01 sec)

mysql> insert into data_usage values(748,3,67,93.3673,'2022-04-22');
Query OK, 1 row affected (0.01 sec)

mysql> insert into data_usage values(737,3,103,207.3373,'2022-04-22');
Query OK, 1 row affected (0.01 sec)

mysql> insert into data_usage values(734,3,73,67.6473,'2022-04-22');
Query OK, 1 row affected (0.01 sec)

mysql> insert into data_usage values(732,3,89,45.873,'2022-04-22');
Query OK, 1 row affected (0.01 sec)

mysql> insert into data_usage values(748,4,67,37.6793,'2022-04-24');
Query OK, 1 row affected (0.01 sec)

mysql> insert into data_usage values(736,4,89,82.6793,'2022-04-24');
Query OK, 1 row affected (0.01 sec)
```

```
mysql> select * from data_usage;
```

did	day	devices_connected	data_bandwidth_GB	date
737	1	201	233.94000	2022-04-17
735	1	101	69.96000	2022-04-17
100	1	53	70.31500	2022-04-17
200	1	22	13.89400	2022-04-17
732	1	46	94.69400	2022-04-17
736	2	87	60.19400	2022-04-20
733	2	27	35.78940	2022-04-20
100	2	67	45.46730	2022-04-20
735	2	54	90.64730	2022-04-20
748	3	67	93.36730	2022-04-22
737	3	103	207.33730	2022-04-22
734	3	73	67.64730	2022-04-22
732	3	89	45.87300	2022-04-22
748	4	67	37.67930	2022-04-24
736	4	89	82.67930	2022-04-24
734	4	105	156.74930	2022-04-24
100	4	17	36.52930	2022-04-24
737	5	120	224.68930	2022-04-27
200	5	12	24.95930	2022-04-27
733	5	122	248.23640	2022-04-27
735	5	408	512.71710	2022-04-27

```
21 rows in set (0.01 sec)
```

### 3) Inserting values into graph table:

```
mysql> insert into graph values(1,423,482.803);
Query OK, 1 row affected (0.01 sec)

mysql> insert into graph values(2,235,232.098);
Query OK, 1 row affected (0.01 sec)

mysql> insert into graph values(3,332,414.2249);
Query OK, 1 row affected (0.01 sec)

mysql> insert into graph values(4,278,313.6372);
Query OK, 1 row affected (0.01 sec)

mysql> insert into graph values(5,662,1010.6021);
Query OK, 1 row affected (0.01 sec)

mysql> commit;
Query OK, 0 rows affected (0.00 sec)
```

```
mysql> select * from graph;
+-----+-----+-----+
| day | total_devices_connected | total_data_bandwidth_GB |
+-----+-----+-----+
| 1 | 423 | 482.80301 |
| 2 | 235 | 232.09801 |
| 3 | 332 | 414.22491 |
| 4 | 278 | 313.63721 |
| 5 | 662 | 1010.60211 |
+-----+-----+-----+
5 rows in set (0.00 sec)
```

## IMPLEMENTATION:

### Front end programs and its connectivity

**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.

The connection to the database can be performed using Java programming (JDBC API) as:

```
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.Statement;
```



```

public class DB {
    public static Connection getConnection() {
        Connection con=null;
        try{
            Class.forName("com.mysql.cj.jdbc.Driver");

            con=DriverManager.getConnection("jdbc:mysql://localhost:3306/harshithdb","root","Hema@7171");
        } catch (Exception e) {System.out.println(e);}
        return con;
    }
}

```

## Code:

### Menu.java:

```

import java.awt.EventQueue;
import javax.swing.JFrame;
import javax.swing.JPanel;
import javax.swing.border.EmptyBorder;
import javax.swing.GroupLayout;
import javax.swing.GroupLayout.Alignment;
import javax.swing.JLabel;
import java.awt.Font;
import java.awt.Color;
import javax.swing.JButton;
import java.awt.event.ActionListener;
import java.awt.event.ActionEvent;

public class Menu extends JFrame {
    private static final long serialVersionUID = 1L;
    static Menu frame;
    private JPanel contentPane;
    static String idd;

    /**
     * Launch the application.
     */
    public static void main(String[] args) {
        EventQueue.invokeLater(new Runnable() {
            public void run() {
                try {
                    frame = new Menu();
                    frame.setTitle("College Network Database");
                    frame.setVisible(true);
                } catch (Exception e) {
                    e.printStackTrace();
                }
            }
        });
    }

    public static void id(String id) {
        idd=id;
    }

    /**
     * Create the frame.
     */
    public Menu() {
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
    }
}

```

```

setBounds(100, 100, 450, 433);
contentPane = new JPanel();
contentPane.setForeground(Color.GRAY);
contentPane.setBorder(new EmptyBorder(5, 5, 5, 5));
setContentPane(contentPane);

JLabel lblLibrarianSection = new JLabel("COLLEGE NETWORK DATABASE ");
lblLibrarianSection.setForeground(Color.RED);
lblLibrarianSection.setFont(new Font("Academy Engraved LET", Font.BOLD ,
25));

JButton btnBInfo = new JButton("Departments Info");
btnBInfo.addActionListener(new ActionListener() {
    public void actionPerformed(ActionEvent e) {
        Department.main(new String[]{});
        frame.dispose();
    }
});
btnBInfo.setFont(new Font("Tahoma", Font.PLAIN, 13));

JButton btnSInfo = new JButton("Data Usage Info");
btnSInfo.addActionListener(new ActionListener() {
    public void actionPerformed(ActionEvent arg0) {
        Data_usage.main(new String[]{});
        frame.dispose();
    }
});
btnSInfo.setFont(new Font("Tahoma", Font.PLAIN, 13));

JButton btnIssueBook = new JButton("Graphs Info");
btnIssueBook.addActionListener(new ActionListener() {
    public void actionPerformed(ActionEvent e) {
        Graph.main(new String[]{});
        frame.dispose();
    }
});

GroupLayout gl_contentPane = new GroupLayout(contentPane);
gl_contentPane.setHorizontalGroup(
    gl_contentPane.createParallelGroup(Alignment.LEADING)
        .addGroup(Alignment.TRAILING,
gl_contentPane.createSequentialGroup()
            .addContainerGap(81, Short.MAX_VALUE)
            .addComponent(lblLibrarianSection)
            .addGap(54)
            .addGroup(gl_contentPane.createSequentialGroup()
                .addGap(132)

                .addGroup(gl_contentPane.createParallelGroup(Alignment.LEADING)

                    .addComponent(btnIssueBook,
GroupLayout.PREFERRED_SIZE, 191, GroupLayout.PREFERRED_SIZE)
                    .addComponent(btnSInfo, GroupLayout.PREFERRED_SIZE,
191, GroupLayout.PREFERRED_SIZE)
                    .addComponent(btnBInfo, GroupLayout.PREFERRED_SIZE,
191, GroupLayout.PREFERRED_SIZE)
                )
                .addContainerGap(101, Short.MAX_VALUE)
            )
        );
gl_contentPane.setVerticalGroup(
    gl_contentPane.createParallelGroup(Alignment.LEADING)
        .addGroup(gl_contentPane.createSequentialGroup()
            .addContainerGap()

```

```

        .addComponent(lblLibrarianSection)
        .addGap(18)
        .addComponent(btnBInfo, GroupLayout.PREFERRED_SIZE, 37,
GroupLayout.PREFERRED_SIZE)
        .addGap(18)
        .addComponent(btnSInfo, GroupLayout.PREFERRED_SIZE, 37,
GroupLayout.PREFERRED_SIZE)
        .addGap(18)
        .addComponent(btnIssueBook, GroupLayout.PREFERRED_SIZE,
37, GroupLayout.PREFERRED_SIZE)
        .addGap(18)

        .addContainerGap(16, Short.MAX_VALUE))
    );
    contentPane.setLayout(gl_contentPane);
}

}

```

## Department.java:

```

import java.awt.BorderLayout;
import java.awt.EventQueue;

import javax.swing.JFrame;
import javax.swing.JPanel;
import javax.swing.JScrollPane;
import javax.swing.JTable;
import javax.swing.ListSelectionModel;
import javax.swing.border.EmptyBorder;
import javax.swing.event.ListSelectionEvent;
import javax.swing.event.ListSelectionListener;
import javax.swing.table.TableColumnModel;
import javax.swing.GroupLayout;
import javax.swing.GroupLayout.Alignment;
import javax.swing.JLabel;
import java.awt.Font;
import java.awt.Color;
import javax.swing.JButton;
import java.awt.event.ActionListener;
import java.sql.Connection;
import java.sql.PreparedStatement;
import java.sql.ResultSet;
import java.sql.ResultSetMetaData;
import java.awt.event.ActionEvent;

public class Department extends JFrame {
    static Department frame;
    private JPanel contentPane;
    private JTable table;

    public static void main(String[] args) {
        EventQueue.invokeLater(new Runnable() {
            public void run() {
                try {
                    frame = new Department();
                    frame.setTitle("Data Usage Info");
                    frame.setVisible(true);
                } catch (Exception e) {
                    e.printStackTrace();
                }
            }
        });
    }
}

```

```

    });
}

public Department() {
    setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
    setBounds(15, 30, 980, 650);

    contentPane = new JPanel();
    contentPane.setForeground(Color.MAGENTA);
    contentPane.setBorder(new EmptyBorder(5, 5, 5, 5));
    setContentPane(contentPane);

    JLabel lblDepartmentinfo = new JLabel("Department Information");
    lblDepartmentinfo.setForeground(Color.RED);
    lblDepartmentinfo.setFont(new Font("Tahoma", Font.PLAIN, 22));

    JButton btnNewButton = new JButton("Add Department");
    btnNewButton.addActionListener(new ActionListener() {
        public void actionPerformed(ActionEvent e) {
            Adddepartment.main(new String[]{});
            frame.dispose();
        }
    });
    btnNewButton.setFont(new Font("Tahoma", Font.PLAIN, 13));

    JButton btnDeleteDepartment = new JButton("Delete Data Usage details");
    btnDeleteDepartment.addActionListener(new ActionListener() {
        public void actionPerformed(ActionEvent e) {

            Deletedepartment.main(new String[]{});
            frame.dispose();
        }
    });
    btnDeleteDepartment.setFont(new Font("Tahoma", Font.PLAIN, 13));

    JButton btnViewDepartment = new JButton("View All Data usage");
    btnViewDepartment.addActionListener(new ActionListener() {
        public void actionPerformed(ActionEvent e) {
            String data[][]=null;
            String column[]=null;
            try{
                Connection con=DB.getConnection();
                PreparedStatement ps=con.prepareStatement("select * from
departments",ResultSet.TYPE_SCROLL_SENSITIVE,ResultSet.CONCUR_UPDATABLE);
                ResultSet rs=ps.executeQuery();

                ResultSetMetaData rsmd=rs.getMetaData();
                int cols=rsmd.getColumnCount();
                column=new String[cols];
                for(int i=1;i<=cols;i++){
                    column[i-1]=rsmd.getColumnName(i);
                }

                rs.last();
                int rows=rs.getRow();
                rs.beforeFirst();

                data=new String[rows][cols];
                int count=0;

```

```

        while(rs.next()){
            for(int i=1;i<=cols;i++){
                data[count][i-1]=rs.getString(i);
            }
            count++;
        }
        con.close();
    }catch(Exception e1){System.out.println(e1);}

    table = new JTable(data,column);
    table.setDefaultEditor(Object.class, null);

    TableColumnModel columnModel = table.getColumnModel();

    columnModel.getColumn(0).setPreferredWidth(20);
    columnModel.getColumn(1).setPreferredWidth(20);

    JScrollPane sp=new JScrollPane(table);
    add(sp);
    sp.setBounds(400,100,500,350);

//    setBounds(70, 30, 1200, 650);

    ListSelectionModel model = table.getSelectionModel();
    model.addListSelectionListener((ListSelectionListener) new
ListSelectionListener() {

        public void valueChanged(ListSelectionEvent e) {
            int sel = model.getMinSelectionIndex();
//            tfid3.setText(""+(sel+1));
            System.out.println(table.getValueAt(sel, 0));

            Deletedeartment.gettf().setText(""+table.getValueAt(sel, 0));
        }

    });

//    ViewDepartment.main(new String[]{});
}
});
btnViewDepartment.setFont(new Font("Tahoma", Font.PLAIN, 13));

JButton btnBack = new JButton("Back");
btnBack.addActionListener(new ActionListener() {
    public void actionPerformed(ActionEvent e) {
        Menu.main(new String[]{});
        frame.dispose();
    }
});
});

```

```

        btnBack.setFont(new Font("Tahoma", Font.PLAIN, 13));

        GroupLayout gl_contentPane = new GroupLayout(contentPane);
        gl_contentPane.setHorizontalGroup(
            gl_contentPane.createParallelGroup(Alignment.LEADING)
                .addGroup(Alignment.CENTER,
                    gl_contentPane.createSequentialGroup()
                        .addComponent(lblDepartmentinfo)
                        .addGap(100)
                        .addGroup(Alignment.CENTER
                            ,gl_contentPane.createSequentialGroup()
                                .addGap(132)

                                .addGroup(gl_contentPane.createParallelGroup(Alignment.LEADING)
                                    .addComponent(btnBack, GroupLayout.PREFERRED_SIZE,
                                        191, GroupLayout.PREFERRED_SIZE)
                                    .addComponent(btnViewDepartment,
                                        GroupLayout.PREFERRED_SIZE, 191, GroupLayout.PREFERRED_SIZE)
                                    .addComponent(btnDeleteDepartment,
                                        GroupLayout.PREFERRED_SIZE, 191, GroupLayout.PREFERRED_SIZE)
                                    // .addComponent(btnUpdateDepartment,
                                        GroupLayout.PREFERRED_SIZE, 191, GroupLayout.PREFERRED_SIZE)
                                    .addComponent(btnNewButton,
                                        GroupLayout.PREFERRED_SIZE, 191, GroupLayout.PREFERRED_SIZE))
                                    .addContainerGap(101, Short.MAX_VALUE))
                                );
            gl_contentPane.setVerticalGroup(
                gl_contentPane.createParallelGroup(Alignment.LEADING)
                    .addGroup(gl_contentPane.createSequentialGroup()
                        .addContainerGap()
                        .addComponent(lblDepartmentinfo)
                        .addGap(38)
                        .addComponent(btnNewButton, GroupLayout.PREFERRED_SIZE,
                            37, GroupLayout.PREFERRED_SIZE)
                        .addGap(18)
                        // .addComponent(btnUpdateDepartment,
                            GroupLayout.PREFERRED_SIZE, 37, GroupLayout.PREFERRED_SIZE)
                        // .addGap(18)
                        .addComponent(btnDeleteDepartment,
                            GroupLayout.PREFERRED_SIZE, 37, GroupLayout.PREFERRED_SIZE)
                        .addGap(18)
                        .addComponent(btnViewDepartment,
                            GroupLayout.PREFERRED_SIZE, 37, GroupLayout.PREFERRED_SIZE)
                        .addGap(18)
                        .addComponent(btnBack, GroupLayout.PREFERRED_SIZE, 37,
                            GroupLayout.PREFERRED_SIZE)
                        .addContainerGap(16, Short.MAX_VALUE))
                    );
            contentPane.setLayout(gl_contentPane);
        }
    }
}

```

## Adddepartment.java:

```

import java.awt.BorderLayout;
import java.awt.EventQueue;

import javax.swing.JFrame;
import javax.swing.JPanel;
import javax.swing.border.EmptyBorder;
import javax.swing.GroupLayout;
import javax.swing.GroupLayout.Alignment;

```

```

import javax.swing.JLabel;
import javax.swing.JOptionPane;

import java.awt.Font;
import java.awt.Color;
import javax.swing.JTextField;
import javax.swing.JButton;
import java.awt.event.ActionListener;
import java.awt.event.ActionEvent;
import javax.swing.LayoutStyle.ComponentPlacement;

public class Adddepartment extends JFrame {
    private static final long serialVersionUID = 1L;
    static Adddepartment frame;
    private JPanel contentPane;
    private JTextField textField;
    private JTextField textField_1;
    private JTextField textField_2;

    /**
     * Launch the application.
     */
    public static void main(String[] args) {
        EventQueue.invokeLater(new Runnable() {
            public void run() {
                try {
                    frame = new Adddepartment();
                    frame.setTitle("Department Info.");
                    frame.setVisible(true);
                } catch (Exception e) {
                    e.printStackTrace();
                }
            }
        });
    }

    /**
     * Create the frame.
     */
    public Adddepartment() {
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        setBounds(975, 425, 400, 300);
        contentPane = new JPanel();
        contentPane.setBorder(new EmptyBorder(5, 5, 5, 5));
        setContentPane(contentPane);

        JLabel lblAdddepartments = new JLabel("Add Department Details");
        lblAdddepartments.setForeground(Color.BLUE);
        lblAdddepartments.setFont(new Font("Tahoma", Font.PLAIN, 18));

        JLabel lblIdid = new JLabel("Department ID :");

        JLabel lblIdname = new JLabel("Department Name:");

        JLabel lblIp_address = new JLabel("IP Address :");

        textField = new JTextField();
        textField.setColumns(10);

        textField_1 = new JTextField();
        textField_1.setColumns(10);
    }

```

```

textField_2 = new JTextField();
textField_2.setColumns(10);

JButton btnAdddepartments = new JButton("Submit");
btnAdddepartments.addActionListener(new ActionListener() {
    public void actionPerformed(ActionEvent e) {

        String dname=textField_1.getText();
        String ip_address=textField_2.getText();
        //int did = textField.getText();
        int did = Integer.parseInt(textField.getText());

        if(dname.equals("") || ip_address.equals("") )
        {
            JOptionPane.showMessageDialog(Adddepartment.this,"Fields Cannot
be Blank!");
        }
        else {
            int id=Integer.parseInt(textField.getText());
            if(Departmentdetails.checkdid(id)) {
                JOptionPane.showMessageDialog(Adddepartment.this,"Department
with Same Department ID is present already\nInsertion Failed !!!");
            }
            else {
                int i=Departmentdetails.insertDepartment(did,dname,ip_address);
                if(i>0){
                    JOptionPane.showMessageDialog(Adddepartment.this,"Department
added successfully!");
                }else{
                    JOptionPane.showMessageDialog(Adddepartment.this,"Unknown Error
!!!\nInsertion not completed");
                }
            }
        }
    }
});

GroupLayout gl_contentPane = new GroupLayout(contentPane);
gl_contentPane.setHorizontalGroup(
    gl_contentPane.createParallelGroup(Alignment.TRAILING)
        .addGroup(gl_contentPane.createSequentialGroup()

.addGroup(gl_contentPane.createParallelGroup(Alignment.LEADING)
            .addGroup(gl_contentPane.createSequentialGroup()
                .addGap(150)
                .addComponent(lblAdddepartments))
            .addGroup(gl_contentPane.createSequentialGroup()
                .addGap(18)

.addGroup(gl_contentPane.createParallelGroup(Alignment.LEADING, false)
                    .addComponent(lbldid)
                    .addComponent(lbllname)
                    .addComponent(lblip_address)

                )
                .addGap(47)

.addGroup(gl_contentPane.createParallelGroup(Alignment.LEADING)
                    .addComponent(textField_2,
GroupLayout.PREFERRED_SIZE, 167, GroupLayout.PREFERRED_SIZE)

```



```

        .addComponent(textField_1,
GroupLayout.PREFERRED_SIZE, 167, GroupLayout.PREFERRED_SIZE)
        .addComponent(textField,
GroupLayout.PREFERRED_SIZE, 167, GroupLayout.PREFERRED_SIZE)))
        .addContainerGap(125, Short.MAX_VALUE))
        .addGroup(Alignment.LEADING,
gl_contentPane.createSequentialGroup())
        .addGap(50)
        .addComponent(btnAdddepartments)
        .addGap(30)
//        .addComponent(btnBack)
        .addContainerGap())
    );
    gl_contentPane.setVerticalGroup(
        gl_contentPane.createParallelGroup(Alignment.LEADING)
            .addGroup(gl_contentPane.createSequentialGroup()
                .addComponent(lblAdddepartments)
                .addGap(18)

            .addGroup(gl_contentPane.createParallelGroup(Alignment.BASELINE)
                .addComponent(lblDid)
                .addComponent(textField,
GroupLayout.PREFERRED_SIZE, GroupLayout.DEFAULT_SIZE, GroupLayout.PREFERRED_SIZE))
                .addGap(18)

            .addGroup(gl_contentPane.createParallelGroup(Alignment.BASELINE)
                .addComponent(lblDname)
                .addComponent(textField_1,
GroupLayout.PREFERRED_SIZE, GroupLayout.DEFAULT_SIZE, GroupLayout.PREFERRED_SIZE))
                .addGap(18)

            .addGroup(gl_contentPane.createParallelGroup(Alignment.BASELINE)
                .addComponent(lblip_address)
                .addComponent(textField_2,
GroupLayout.PREFERRED_SIZE, GroupLayout.DEFAULT_SIZE, GroupLayout.PREFERRED_SIZE))
                /*.addGap(18)

            .addGroup(gl_contentPane.createParallelGroup(Alignment.BASELINE)
                .addComponent(lblEmail)
                .addComponent(textField_3,
GroupLayout.PREFERRED_SIZE, GroupLayout.DEFAULT_SIZE, GroupLayout.PREFERRED_SIZE))*/
                .addGap(18)

            .addGroup(gl_contentPane.createParallelGroup(Alignment.BASELINE)
                )
                .addGap(30)

            .addGroup(gl_contentPane.createParallelGroup(Alignment.BASELINE)
                .addComponent(btnAdddepartments,
GroupLayout.PREFERRED_SIZE, 31, GroupLayout.PREFERRED_SIZE)

//                .addComponent(btnBack,
GroupLayout.PREFERRED_SIZE, 31, GroupLayout.PREFERRED_SIZE)
                )

    )
    );
    contentPane.setLayout(gl_contentPane);
}
}

```

## Deletedeartment.java:

```

import java.awt.BorderLayout;
import java.awt.EventQueue;

import javax.swing.JFrame;
import javax.swing.JPanel;
import javax.swing.border.EmptyBorder;
import javax.swing.GroupLayout;
import javax.swing.GroupLayout.Alignment;
import javax.swing.JLabel;
import javax.swing.JOptionPane;
import javax.swing.JTextField;
import javax.swing.JButton;
import java.awt.Font;
import java.awt.event.ActionListener;
import java.awt.event.ActionEvent;

public class Deletedepartment extends JFrame {
    static Deletedepartment frame;
    private JPanel contentPane;
    private static JTextField textField;

    public static JTextField gettf()
    {
        return textField;
    }

    /**
     * Launch the application.
     */
    public static void main(String[] args) {
        EventQueue.invokeLater(new Runnable() {
            public void run() {
                try {
                    frame = new Deletedepartment();
                    frame.setTitle("Delete department Info.");
                    frame.setVisible(true);
                } catch (Exception e) {
                    e.printStackTrace();
                }
            }
        });
    }

    /**
     * Create the frame.
     */
    public Deletedepartment() {
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        setBounds(1000, 100, 350, 150);

        contentPane = new JPanel();
        contentPane.setBorder(new EmptyBorder(5, 5, 5, 5));

        setContentPane(contentPane);

        JLabel lblEnterDid = new JLabel("Enter did number :");

        textField = new JTextField();
        textField.setColumns(10);

        JButton btnDelete = new JButton("Delete");
        btnDelete.addActionListener(new ActionListener() {

```

```

        public void actionPerformed(ActionEvent e) {
            String id=textField.getText();
            if(id==null||id.trim().equals("")){
                JOptionPane.showMessageDialog(Deleteddepartment.this,"Id
can't be blank");
            }else{
                int did=Integer.parseInt(textField.getText());
                int i=Departmentdetails.deleteDepartment(did);
                if(i>0){

                    JOptionPane.showMessageDialog(Deleteddepartment.this," Record deleted
successfully!");
                }else{

                    JOptionPane.showMessageDialog(Deleteddepartment.this,"Unable to delete given
id!");
                }
            }
        }
    });
    btnDelete.setFont(new Font("Tahoma", Font.PLAIN, 13));

    GroupLayout gl_contentPane = new GroupLayout(contentPane);
    gl_contentPane.setHorizontalGroup(
        gl_contentPane.createParallelGroup(Alignment.LEADING)
            .addGroup(gl_contentPane.createSequentialGroup()
                .addGap(39)
                .addComponent(lblEnterDid)
                .addGap(57)
                .addComponent(textField, GroupLayout.PREFERRED_SIZE, 100,
GroupLayout.PREFERRED_SIZE)
                .addGap(107, Short.MAX_VALUE)
                .addGroup(gl_contentPane.createSequentialGroup()
                    .addGap(100)
                    .addComponent(btnDelete)
                    .addGap(30)
                    .addComponent(btnNewButton)
                )
            );
    gl_contentPane.setVerticalGroup(
        gl_contentPane.createParallelGroup(Alignment.CENTER)
            .addGroup(gl_contentPane.createSequentialGroup()
                .addGap(19)

                .addGroup(gl_contentPane.createParallelGroup(Alignment.LEADING)
                    .addComponent(textField,
GroupLayout.PREFERRED_SIZE, GroupLayout.DEFAULT_SIZE, GroupLayout.PREFERRED_SIZE)
                    .addComponent(lblEnterDid)
                    .addGap(33)
                    .addGroup(gl_contentPane.createParallelGroup()
                        .addComponent(btnDelete)
                        .addComponent(btnNewButton)
                    )
                )
            );
    contentPane.setLayout(gl_contentPane);
}
}

```

## Departmentdetails.java:

```

import java.sql.Connection;
import java.sql.PreparedStatement;

```

```

import java.sql.ResultSet;
public class Departmentdetails{

    public static boolean checkdid(int did){
        boolean status=false;
        try{
            Connection con=DB.getConnection();
            PreparedStatement ps=con.prepareStatement("select * from departments
where did=?");
            ps.setInt(1,did);
            ResultSet rs=ps.executeQuery();
            status=rs.next();
            con.close();
        }catch(Exception e){System.out.println(e);}
        return status;
    }

    public static int insertDepartment(int did,String dname,String ip_address){
        int status=0;
        try{
            Connection con=DB.getConnection();
            PreparedStatement ps=con.prepareStatement("insert into
departments(did,dname,ip_address) values(?,?,?)");
            ps.setInt(1,did);
            ps.setString(2,dname);
            ps.setString(3,ip_address);

            status=ps.executeUpdate();
            con.close();
        }catch(Exception e){System.out.println(e);}
        return status;
    }

    public static int deleteDepartment(int id){
        int status=0;
        try{
            Connection con=DB.getConnection();
            PreparedStatement ps=con.prepareStatement("delete from departments
where did=?");
            ps.setInt(1,id);
            status=ps.executeUpdate();
            con.close();
        }catch(Exception e){System.out.println(e);}
        return status;
    }
}

```

## Viewdepartment.java:

```

import java.awt.BorderLayout;
import java.awt.EventQueue;
import java.sql.Connection;
import java.sql.PreparedStatement;
import java.sql.ResultSet;
import java.sql.ResultSetMetaData;

import javax.swing.JFrame;
import javax.swing.JPanel;
import javax.swing.JScrollPane;
import javax.swing.border.EmptyBorder;
import javax.swing.JTable;

public class Viewdepartment extends JFrame {

```

```

private JPanel contentPane;
private JTable table;

public static void main(String[] args) {
    EventQueue.invokeLater(new Runnable() {
        public void run() {
            try {
                Viewdepartment frame = new Viewdepartment();
                frame.setTitle("department");
                frame.setVisible(true);
            } catch (Exception e) {
                e.printStackTrace();
            }
        }
    });
}

/**
 * Create the frame.
 */
public Viewdepartment() {
    setDefaultCloseOperation(JFrame.DISPOSE_ON_CLOSE);
    setBounds(100, 100, 450, 300);
    contentPane = new JPanel();
    contentPane.setBorder(new EmptyBorder(5, 5, 5, 5));
    contentPane.setLayout(new BorderLayout(0, 0));
    setContentPane(contentPane);

    String data[][]=null;
    String column[]=null;
    try{
        Connection con=DB.getConnection();
        PreparedStatement ps=con.prepareStatement("select * from
departments",ResultSet.TYPE_SCROLL_SENSITIVE,ResultSet.CONCUR_UPDATABLE);
        ResultSet rs=ps.executeQuery();

        ResultSetMetaData rsmd=rs.getMetaData();
        int cols=rsmd.getColumnCount();
        column=new String[cols];
        for(int i=1;i<=cols;i++){
            column[i-1]=rsmd.getColumnName(i);
        }

        rs.last();
        int rows=rs.getRow();
        rs.beforeFirst();

        data=new String[rows][cols];
        int count=0;
        while(rs.next()){
            for(int i=1;i<=cols;i++){
                data[count][i-1]=rs.getString(i);
            }
            count++;
        }
        con.close();
    }catch(Exception e){System.out.println(e);}

    table = new JTable(data,column);
    table.setDefaultEditor(Object.class, null);
    JScrollPane sp=new JScrollPane(table);

    contentPane.add(sp, BorderLayout.CENTER);
}

```

```
}
```

## Data\_usage.java:

```
import java.awt.BorderLayout;
import java.awt.EventQueue;

import javax.swing.JFrame;
import javax.swing.JPanel;
import javax.swing.JScrollPane;
import javax.swing.JTable;
import javax.swing.ListSelectionModel;
import javax.swing.border.EmptyBorder;
import javax.swing.event.ListSelectionEvent;
import javax.swing.event.ListSelectionListener;
import javax.swing.table.TableColumnModel;
import javax.swing.GroupLayout;
import javax.swing.GroupLayout.Alignment;
import javax.swing.JLabel;
import java.awt.Font;
import java.awt.Color;
import javax.swing.JButton;
import java.awt.event.ActionListener;
import java.sql.Connection;
import java.sql.PreparedStatement;
import java.sql.ResultSet;
import java.sql.ResultSetMetaData;
import java.awt.event.ActionEvent;

public class Data_usage extends JFrame {
    static Data_usage frame;
    private JPanel contentPane;
    private JTable table;

    public static void main(String[] args) {
        EventQueue.invokeLater(new Runnable() {
            public void run() {
                try {
                    frame = new Data_usage();
                    frame.setTitle("Data Usage Info");
                    frame.setVisible(true);
                } catch (Exception e) {
                    e.printStackTrace();
                }
            }
        });
    }

    public Data_usage() {
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        setBounds(15, 30, 980, 650);

        contentPane = new JPanel();
        contentPane.setForeground(Color.MAGENTA);
        contentPane.setBorder(new EmptyBorder(5, 5, 5, 5));
        setContentPane(contentPane);

        JLabel lblData_usageinfo = new JLabel("Data Usage Information");
        lblData_usageinfo.setForeground(Color.RED);
        lblData_usageinfo.setFont(new Font("Tahoma", Font.PLAIN, 22));
    }
}
```

```

        JButton btnNewButton = new JButton("Add Data Usage");
        btnNewButton.addActionListener(new ActionListener() {
            public void actionPerformed(ActionEvent e) {
                Adddata_usage.main(new String[]{});
                frame.dispose();
            }
        });
        btnNewButton.setFont(new Font("Tahoma", Font.PLAIN, 13));

        JButton btnDeleteData_usage = new JButton("Delete Data Usage details");
        btnDeleteData_usage.addActionListener(new ActionListener() {
            public void actionPerformed(ActionEvent e) {

                Deletedata_usage.main(new String[]{});
                frame.dispose();
            }
        });
        btnDeleteData_usage.setFont(new Font("Tahoma", Font.PLAIN, 13));

        JButton btnViewData_usage = new JButton("View All Data usage");
        btnViewData_usage.addActionListener(new ActionListener() {
            public void actionPerformed(ActionEvent e) {
                String data[][]=null;
                String column[]=null;
                try{
                    Connection con=DB.getConnection();
                    PreparedStatement ps=con.prepareStatement("select * from
data_usage",ResultSet.TYPE_SCROLL_SENSITIVE,ResultSet.CONCUR_UPDATABLE);
                    ResultSet rs=ps.executeQuery();

                    ResultSetMetaData rsmd=rs.getMetaData();
                    int cols=rsmd.getColumnCount();
                    column=new String[cols];
                    for(int i=1;i<=cols;i++){
                        column[i-1]=rsmd.getColumnName(i);
                    }

                    rs.last();
                    int rows=rs.getRow();
                    rs.beforeFirst();

                    data=new String[rows][cols];
                    int count=0;
                    while(rs.next()){
                        for(int i=1;i<=cols;i++){
                            data[count][i-1]=rs.getString(i);
                        }
                        count++;
                    }
                    con.close();
                }catch (Exception e1){System.out.println(e1);}

                table = new JTable(data,column);
                table.setDefaultEditor(Object.class, null);
            }
        });
    }
}

```

```

        TableColumnModel columnModel = table.getColumnModel();

        columnModel.getColumn(0).setPreferredWidth(20);
        columnModel.getColumn(1).setPreferredWidth(20);

        JScrollPane sp=new JScrollPane(table);
        add(sp);
        sp.setBounds(400,100,500,350);

//        setBounds(70, 30, 1200, 650);

        ListSelectionModel model = table.getSelectionModel();
        model.addListSelectionListener((ListSelectionListener) new
ListSelectionListener() {

            public void valueChanged(ListSelectionEvent e) {
                int sel = model.getMinSelectionIndex();
//                tfid3.setText(""+(sel+1));
                System.out.println(table.getValueAt(sel, 0));

                Deletedata_usage.gettf().setText(""+table.getValueAt(sel, 1));
            }

        });

//        Viewdata_usage.main(new String[]{});
    }
});
btnViewData_usage.setFont(new Font("Tahoma", Font.PLAIN, 13));

JButton btnBack = new JButton("Back");
btnBack.addActionListener(new ActionListener() {
    public void actionPerformed(ActionEvent e) {
        Menu.main(new String[]{});
        frame.dispose();
    }
});
btnBack.setFont(new Font("Tahoma", Font.PLAIN, 13));

GroupLayout gl_contentPane = new GroupLayout(contentPane);

```



```

        gl_contentPane.setHorizontalGroup(
            gl_contentPane.createParallelGroup(Alignment.LEADING)
                .addGroup(Alignment.CENTER,
                    gl_contentPane.createSequentialGroup()
                        .addComponent(lblData_usageinfo)
                        .addGap(100))
                .addGroup(Alignment.CENTER
                    ,gl_contentPane.createSequentialGroup()
                        .addGap(132)

                .addGroup(gl_contentPane.createParallelGroup(Alignment.LEADING)
                    .addComponent(btnBack, GroupLayout.PREFERRED_SIZE, 191,
                        GroupLayout.PREFERRED_SIZE)
                        .addComponent(btnViewData_usage,
                            GroupLayout.PREFERRED_SIZE, 191, GroupLayout.PREFERRED_SIZE)
                        .addComponent(btnDeleteData_usage,
                            GroupLayout.PREFERRED_SIZE, 191, GroupLayout.PREFERRED_SIZE)
                        // .addComponent(btnUpdateData_usage,
                            GroupLayout.PREFERRED_SIZE, 191, GroupLayout.PREFERRED_SIZE)
                        .addComponent(btnNewButton,
                            GroupLayout.PREFERRED_SIZE, 191, GroupLayout.PREFERRED_SIZE))
                        .addContainerGap(101, Short.MAX_VALUE))
        );
        gl_contentPane.setVerticalGroup(
            gl_contentPane.createParallelGroup(Alignment.LEADING)
                .addGroup(gl_contentPane.createSequentialGroup()
                    .addContainerGap()
                    .addComponent(lblData_usageinfo)
                    .addGap(38)
                    .addComponent(btnNewButton, GroupLayout.PREFERRED_SIZE,
37, GroupLayout.PREFERRED_SIZE)
                        .addGap(18)
                        // .addComponent(btnUpdateData_usage,
                            GroupLayout.PREFERRED_SIZE, 37, GroupLayout.PREFERRED_SIZE)
                        // .addGap(18)
                        .addComponent(btnDeleteData_usage,
                            GroupLayout.PREFERRED_SIZE, 37, GroupLayout.PREFERRED_SIZE)
                        .addGap(18)
                        .addComponent(btnViewData_usage,
                            GroupLayout.PREFERRED_SIZE, 37, GroupLayout.PREFERRED_SIZE)
                        .addGap(18)
                        .addComponent(btnBack, GroupLayout.PREFERRED_SIZE, 37,
                            GroupLayout.PREFERRED_SIZE)
                        .addContainerGap(16, Short.MAX_VALUE))
        );
        contentPane.setLayout(gl_contentPane);
    }
}

```

## Adddata\_usage.java:

```

import java.awt.BorderLayout;
import java.awt.EventQueue;

import javax.swing.JFrame;
import javax.swing.JPanel;
import javax.swing.border.EmptyBorder;
import javax.swing.GroupLayout;
import javax.swing.GroupLayout.Alignment;
import javax.swing.JLabel;
import javax.swing.JOptionPane;

```

```

import java.awt.Font;
import java.awt.Color;
import javax.swing.JTextField;
import javax.swing.JButton;
import java.awt.event.ActionListener;
import java.awt.event.ActionEvent;
import javax.swing.LayoutStyle.ComponentPlacement;

import java.text.SimpleDateFormat;
import java.util.Date;

public class Adddata_usage extends JFrame {
    private static final long serialVersionUID = 1L;
    static Adddata_usage frame;
    private JPanel contentPane;
    private JTextField textField;
    private JTextField textField_1;
    private JTextField textField_2;
    private JTextField textField_3;
    private JTextField textField_4;

    /**
     * Launch the application.
     */
    public static void main(String[] args) {
        EventQueue.invokeLater(new Runnable() {
            public void run() {
                try {
                    frame = new Adddata_usage();
                    frame.setTitle("Data Usage Info.");
                    frame.setVisible(true);
                } catch (Exception e) {
                    e.printStackTrace();
                }
            }
        });
    }

    /**
     * Create the frame.
     */
    public Adddata_usage() {
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        setBounds(928, 380, 450, 350);
        contentPane = new JPanel();
        contentPane.setBorder(new EmptyBorder(5, 5, 5, 5));
        setContentPane(contentPane);

        JLabel lblAdddata_usage = new JLabel("Add Data Usage Details");
        lblAdddata_usage.setForeground(Color.MAGENTA);
        lblAdddata_usage.setFont(new Font("Tahoma", Font.PLAIN, 18));

        JLabel lblidid = new JLabel("Department ID :");

        JLabel lblday = new JLabel("Day Number:");

        JLabel lbldevices_connected = new JLabel("No. of devices connected :");

        JLabel lbldata_bandwidth_GB = new JLabel("Amount of Data Consumed(GB) :");

        JLabel lbldate = new JLabel("Date :");
    }

```

```

textField = new JTextField();
textField.setColumns(10);

textField_1 = new JTextField();
textField_1.setColumns(10);

textField_2 = new JTextField();
textField_2.setColumns(10);

textField_3 = new JTextField();
textField_3.setColumns(10);

textField_4 = new JTextField();
textField_4.setColumns(10);

JButton btnAdddata_usage = new JButton("Submit");
btnAdddata_usage.addActionListener(new ActionListener() {
    public void actionPerformed(ActionEvent e) {

        int did = Integer.parseInt(textField.getText());
        int day = Integer.parseInt(textField_1.getText());
        int devices_connected = Integer.parseInt(textField_2.getText());
        float data_bandwidth_GB = Float.parseFloat(textField_3.getText());

        //String date = df.format(dateChooserFrom.getText());
        String date=textField_4.getText();

        if(date.equals("") /*|| ip_address.equals("")*/ )
        {
            JOptionPane.showMessageDialog(Adddata_usage.this,"Fields Cannot
be Blank!");
        }
        else {
            int id=Integer.parseInt(textField.getText());
            /*if(Data_usagedetails.checkdid(id)) {
                JOptionPane.showMessageDialog(Adddata_usage.this,"Department
with Same Department ID is present already\nInsertion Failed !!!");
            }
            else {
                */{int
i=Data_usagedetails.insertData_usage(did,day,devices_connected,data_bandwidth_GB,date);
                if(i>0){
                    JOptionPane.showMessageDialog(Adddata_usage.this,"Department
added successfully!");
                }else{
                    JOptionPane.showMessageDialog(Adddata_usage.this,"Unknown Error
!!!\nInsertion not completed");
                }
            }
        }
    });

    GroupLayout gl_contentPane = new GroupLayout(contentPane);
    gl_contentPane.setHorizontalGroup(
        gl_contentPane.createParallelGroup(Alignment.TRAILING)
            .addGroup(gl_contentPane.createSequentialGroup()

```

```

        .addGroup(gl_contentPane.createParallelGroup(Alignment.LEADING)
            .addGroup(gl_contentPane.createSequentialGroup()
                .addGap(150)
                .addComponent(lblAdddata_usage))
            .addGroup(gl_contentPane.createSequentialGroup()
                .addGap(18)

        .addGroup(gl_contentPane.createParallelGroup(Alignment.LEADING, false)
            .addComponent(lbldid)
            .addComponent(lblday)
            .addComponent(lbldevices_connected)
            .addComponent(lbldata_bandwidth_GB)
            .addComponent(lbldate)

        )
        .addGap(47)

        .addGroup(gl_contentPane.createParallelGroup(Alignment.LEADING)

            .addComponent(textField_4,
                GroupLayout.PREFERRED_SIZE, 167, GroupLayout.PREFERRED_SIZE)
            .addComponent(textField_3,
                GroupLayout.PREFERRED_SIZE, 167, GroupLayout.PREFERRED_SIZE)
            .addComponent(textField_2,
                GroupLayout.PREFERRED_SIZE, 167, GroupLayout.PREFERRED_SIZE)
            .addComponent(textField_1,
                GroupLayout.PREFERRED_SIZE, 167, GroupLayout.PREFERRED_SIZE)
            .addComponent(textField,
                GroupLayout.PREFERRED_SIZE, 167, GroupLayout.PREFERRED_SIZE)))
        .addContainerGap(125, Short.MAX_VALUE)
        .addGroup(Alignment.LEADING,
gl_contentPane.createSequentialGroup()
            .addGap(50)
            .addComponent(btnAdddata_usage)
            .addGap(30)
//            .addComponent(btnBack)
            .addContainerGap())

    );
    gl_contentPane.setVerticalGroup(
        gl_contentPane.createParallelGroup(Alignment.LEADING)
            .addGroup(gl_contentPane.createSequentialGroup()
                .addComponent(lblAdddata_usage)
                .addGap(18)

        .addGroup(gl_contentPane.createParallelGroup(Alignment.BASELINE)
            .addComponent(lbldid)
            .addComponent(textField,
                GroupLayout.PREFERRED_SIZE, GroupLayout.DEFAULT_SIZE, GroupLayout.PREFERRED_SIZE))
            .addGap(18)

        .addGroup(gl_contentPane.createParallelGroup(Alignment.BASELINE)
            .addComponent(lblday)
            .addComponent(textField_1,
                GroupLayout.PREFERRED_SIZE, GroupLayout.DEFAULT_SIZE, GroupLayout.PREFERRED_SIZE))
            .addGap(18)

        .addGroup(gl_contentPane.createParallelGroup(Alignment.BASELINE)
            .addComponent(lbldevices_connected)
            .addComponent(textField_2,
                GroupLayout.PREFERRED_SIZE, GroupLayout.DEFAULT_SIZE, GroupLayout.PREFERRED_SIZE))
            .addGap(18)

```

```

        .addGroup(gl_contentPane.createParallelGroup(Alignment.BASELINE)
            .addComponent(lbldata_bandwidth_GB)
            .addComponent(textField_3,
                GroupLayout.PREFERRED_SIZE, GroupLayout.DEFAULT_SIZE, GroupLayout.PREFERRED_SIZE))

        .addGap(18)

        .addGroup(gl_contentPane.createParallelGroup(Alignment.BASELINE)
            .addComponent(lbldate)
            .addComponent(textField_4,
                GroupLayout.PREFERRED_SIZE, GroupLayout.DEFAULT_SIZE, GroupLayout.PREFERRED_SIZE))

        .addGap(18)

        .addGroup(gl_contentPane.createParallelGroup(Alignment.BASELINE)
            .addGap(30)

        .addGroup(gl_contentPane.createParallelGroup(Alignment.BASELINE)
            .addComponent(btnAdddata_usage,
                GroupLayout.PREFERRED_SIZE, 31, GroupLayout.PREFERRED_SIZE)

        )

    )
    );
    contentPane.setLayout(gl_contentPane);
}
}

```

## Deletedata\_usage.java:

```

import java.awt.BorderLayout;
import java.awt.EventQueue;

import javax.swing.JFrame;
import javax.swing.JPanel;
import javax.swing.border.EmptyBorder;
import javax.swing.GroupLayout;
import javax.swing.GroupLayout.Alignment;
import javax.swing.JLabel;
import javax.swing.JOptionPane;
import javax.swing.JTextField;
import javax.swing.JButton;
import java.awt.Font;
import java.awt.event.ActionListener;
import java.awt.event.ActionEvent;

public class Deletedata_usage extends JFrame {
    static Deletedata_usage frame;
    private JPanel contentPane;
    private static JTextField textField;

    public static JTextField gettf()
    {
        return textField;
    }
}

```

```

/**
 * Launch the application.
 */
public static void main(String[] args) {
    EventQueue.invokeLater(new Runnable() {
        public void run() {
            try {
                frame = new Deletedata_usage();
                frame.setTitle("Delete data_usage Info.");
                frame.setVisible(true);
            } catch (Exception e) {
                e.printStackTrace();
            }
        }
    });
}

/**
 * Create the frame.
 */
public Deletedata_usage() {
    setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
    setBounds(1000, 100, 350, 150);

    contentPane = new JPanel();
    contentPane.setBorder(new EmptyBorder(5, 5, 5, 5));

    setContentPane(contentPane);

    JLabel lblEnterDay = new JLabel("Enter day number :");

    textField = new JTextField();
    textField.setColumns(10);

    JButton btnDelete = new JButton("Delete");
    btnDelete.addActionListener(new ActionListener() {
        public void actionPerformed(ActionEvent e) {
            String id=textField.getText();
            if(id==null||id.trim().equals("")){
                JOptionPane.showMessageDialog(Deletedata_usage.this,"Id
can't be blank");
            }else{
                int day=Integer.parseInt(textField.getText());
                //int id=Integer.parseInt(textField.getText());
                int i=Data_usagedetails.deleteData_usage(day);
                if(i>0){

                    JOptionPane.showMessageDialog(Deletedata_usage.this," Record deleted
successfully!");
                }else{

                    JOptionPane.showMessageDialog(Deletedata_usage.this,"Unable to delete given
id!");
                }
            }
        }
    });
    btnDelete.setFont(new Font("Tahoma", Font.PLAIN, 13));

    GroupLayout gl_contentPane = new GroupLayout(contentPane);
    gl_contentPane.setHorizontalGroup(
        gl_contentPane.createParallelGroup(Alignment.LEADING)

```

```

        .addGroup(gl_contentPane.createSequentialGroup())
        .addGap(39)
        .addComponent(lblEnterDay)
        .addGap(57)
        .addComponent(textField, GroupLayout.PREFERRED_SIZE, 100,
GroupLayout.PREFERRED_SIZE)
        .addContainerGap(107, Short.MAX_VALUE)
        .addGroup(gl_contentPane.createSequentialGroup())
        .addGap(100)
        .addComponent(btnDelete)
        .addGap(30)
        .addComponent(btnNewButton)
        )
    );
    gl_contentPane.setVerticalGroup(
        gl_contentPane.createParallelGroup(Alignment.CENTER)
        .addGroup(gl_contentPane.createSequentialGroup())
        .addGap(19)

        .addGroup(gl_contentPane.createParallelGroup(Alignment.LEADING)
        .addComponent(textField,
GroupLayout.PREFERRED_SIZE, GroupLayout.DEFAULT_SIZE, GroupLayout.PREFERRED_SIZE)
        .addComponent(lblEnterDay))
        .addGap(33)
        .addGroup(gl_contentPane.createParallelGroup())
        .addComponent(btnDelete)
        .addComponent(btnNewButton)
        )
    );
    contentPane.setLayout(gl_contentPane);
}
}

```

## Data\_usagedetails.java:

```

import java.sql.Connection;
import java.sql.PreparedStatement;
import java.sql.ResultSet;
public class Data_usagedetails{

    public static boolean checkdid(int did){
        boolean status=false;
        try{
            Connection con=DB.getConnection();
            PreparedStatement ps=con.prepareStatement("select * from data_usage
where did=?");
            ps.setInt(1,did);
            ResultSet rs=ps.executeQuery();
            status=rs.next();
            con.close();
        }catch(Exception e){System.out.println(e);}
        return status;
    }

    public static int insertData_usage(int did,int day,int devices_connected,float
data_bandwidth_GB,String date){
        int status=0;
        try{
            Connection con=DB.getConnection();
            PreparedStatement ps=con.prepareStatement("insert into
data_usage(did,day,devices_connected,data_bandwidth_GB,date) values(?,?,?,?,?)");
            ps.setInt(1,did);
            ps.setInt(2,day);
            ps.setInt(3,devices_connected);
            ps.setFloat(4,data_bandwidth_GB);

```

```

        ps.setString(5,date);

        status=ps.executeUpdate();
        con.close();
    }catch(Exception e){System.out.println(e);}
    return status;
}

public static int deleteData_usage(int id){
    int status=0;
    try{
        Connection con=DB.getConnection();
        PreparedStatement ps=con.prepareStatement("delete from data_usage
where day=?");

        ps.setInt(1,id);
        status=ps.executeUpdate();
        con.close();
    }catch(Exception e){System.out.println(e);}
    return status;
}
}

```

## Viewdata\_usage.java:

```

import java.awt.BorderLayout;
import java.awt.EventQueue;
import java.sql.Connection;
import java.sql.PreparedStatement;
import java.sql.ResultSet;
import java.sql.ResultSetMetaData;

import javax.swing.JFrame;
import javax.swing.JPanel;
import javax.swing.JScrollPane;
import javax.swing.border.EmptyBorder;
import javax.swing.JTable;

public class Viewdata_usage extends JFrame {

    private JPanel contentPane;
    private JTable table;

    public static void main(String[] args) {
        EventQueue.invokeLater(new Runnable() {
            public void run() {
                try {
                    Viewdata_usage frame = new Viewdata_usage();
                    frame.setTitle("Data Usage");
                    frame.setVisible(true);
                } catch (Exception e) {
                    e.printStackTrace();
                }
            }
        });
    }

    /**
     * Create the frame.
     */
    public Viewdata_usage() {
        //
        setDefaultCloseOperation(JFrame.DISPOSE_ON_CLOSE);
    }
}

```



```

        setBounds(450, 150, 800, 500);
        contentPane = new JPanel();
        contentPane.setBorder(new EmptyBorder(5, 5, 5, 5));
        contentPane.setLayout(new BorderLayout(0, 0));
        setContentPane(contentPane);

        String data[][]=null;
        String column[]=null;
        try{
            Connection con=DB.getConnection();
            PreparedStatement ps=con.prepareStatement("select * from
data_usage",ResultSet.TYPE_SCROLL_SENSITIVE,ResultSet.CONCUR_UPDATABLE);
            ResultSet rs=ps.executeQuery();

            ResultSetMetaData rsmd=rs.getMetaData();
            int cols=rsmd.getColumnCount();
            column=new String[cols];
            for(int i=1;i<=cols;i++){
                column[i-1]=rsmd.getColumnName(i);
            }

            rs.last();
            int rows=rs.getRow();
            rs.beforeFirst();

            data=new String[rows][cols];
            int count=0;
            while(rs.next()){
                for(int i=1;i<=cols;i++){
                    data[count][i-1]=rs.getString(i);
                }
                count++;
            }
            con.close();
        }catch(Exception e){System.out.println(e);}

        table = new JTable(data,column);
        table.setDefaultEditor(Object.class, null);
        JScrollPane sp=new JScrollPane(table);

        contentPane.add(sp, BorderLayout.CENTER);
    }
}

```

## Graph.java:

```

import java.awt.BorderLayout;
import java.awt.EventQueue;

import javax.swing.JFrame;
import javax.swing.JPanel;
import javax.swing.JScrollPane;
import javax.swing.JTable;
import javax.swing.ListSelectionModel;
import javax.swing.border.EmptyBorder;
import javax.swing.event.ListSelectionEvent;
import javax.swing.event.ListSelectionListener;
import javax.swing.table.TableColumnModel;
import javax.swing.GroupLayout;
import javax.swing.GroupLayout.Alignment;
import javax.swing.JLabel;
import java.awt.Font;

```

```

import java.awt.Color;
import javax.swing.JButton;
import java.awt.event.ActionListener;
import java.sql.Connection;
import java.sql.PreparedStatement;
import java.sql.ResultSet;
import java.sql.ResultSetMetaData;
import java.awt.event.ActionEvent;

public class Graph extends JFrame {
    static Graph frame;
    private JPanel contentPane;
    private JTable table;

    public static void main(String[] args) {
        EventQueue.invokeLater(new Runnable() {
            public void run() {
                try {
                    frame = new Graph();
                    frame.setTitle("Data Usage Info");
                    frame.setVisible(true);
                } catch (Exception e) {
                    e.printStackTrace();
                }
            }
        });
    }

    public Graph() {
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        setBounds(15, 30, 980, 650);

        contentPane = new JPanel();
        contentPane.setForeground(Color.MAGENTA);
        contentPane.setBorder(new EmptyBorder(5, 5, 5, 5));
        setContentPane(contentPane);

        JLabel lblGraphinfo = new JLabel("Graph Information");
        lblGraphinfo.setForeground(Color.RED);
        lblGraphinfo.setFont(new Font("Tahoma", Font.PLAIN, 22));

        JButton btnNewButton = new JButton("View Bar Graph");
        btnNewButton.addActionListener(new ActionListener() {
            public void actionPerformed(ActionEvent e) {
                // 
            }
        });
        btnNewButton.setFont(new Font("Tahoma", Font.PLAIN, 13));

        JButton btnViewGraph = new JButton("View Graph Details");
        btnViewGraph.addActionListener(new ActionListener() {
            public void actionPerformed(ActionEvent e) {
                String data[][]=null;
                String column[]=null;
                try{
                    Connection con=DB.getConnection();
                    PreparedStatement ps=con.prepareStatement("select * from
graph",ResultSet.TYPE_SCROLL_SENSITIVE,ResultSet.CONCUR_UPDATABLE);
                    ResultSet rs=ps.executeQuery();

```

```

        ResultSetMetaData rsmd=rs.getMetaData();
        int cols=rsmd.getColumnCount();
        column=new String[cols];
        for(int i=1;i<=cols;i++){
            column[i-1]=rsmd.getColumnName(i);
        }

        rs.last();
        int rows=rs.getRow();
        rs.beforeFirst();

        data=new String[rows][cols];
        int count=0;
        while(rs.next()){
            for(int i=1;i<=cols;i++){
                data[count][i-1]=rs.getString(i);
            }
            count++;
        }
        con.close();
    }catch(Exception e1){System.out.println(e1);}

    table = new JTable(data,column);
    table.setDefaultEditor(Object.class, null);

    TableColumnModel columnModel = table.getColumnModel();

    columnModel.getColumn(0).setPreferredWidth(20);
    columnModel.getColumn(1).setPreferredWidth(20);

    JScrollPane sp=new JScrollPane(table);
    add(sp);
    sp.setBounds(400,100,500,350);

//        setBounds(70, 30, 1200, 650);

    ListSelectionModel model = table.getSelectionModel();
    model.addListSelectionListener((ListSelectionListener) new
ListSelectionListener() {

        public void valueChanged(ListSelectionEvent e) {
            int sel = model.getMinSelectionIndex();
            tfid3.setText(""+(sel+1));
            System.out.println(table.getValueAt(sel, 0));

//
//
//        DeleteGraph.gettf().setText(""+table.getValueAt(sel, 0));
        }

    });

```

```

//          ViewGraph.main(new String[]{});
    }
});
btnViewGraph.setFont(new Font("Tahoma", Font.PLAIN, 13));

JButton btnBack = new JButton("Back");
btnBack.addActionListener(new ActionListener() {
    public void actionPerformed(ActionEvent e) {
        Menu.main(new String[]{});
        frame.dispose();
    }
});
btnBack.setFont(new Font("Tahoma", Font.PLAIN, 13));

GroupLayout gl_contentPane = new GroupLayout(contentPane);
gl_contentPane.setHorizontalGroup(
    gl_contentPane.createParallelGroup(Alignment.LEADING)
        .addGroup(Alignment.CENTER,
            gl_contentPane.createSequentialGroup()
                .addComponent(lblGraphinfo)
                .addGap(100)
                .addGroup(Alignment.CENTER
                    ,gl_contentPane.createSequentialGroup()
                        .addGap(132)

                            .addGroup(gl_contentPane.createParallelGroup(Alignment.LEADING)
                                .addComponent(btnBack, GroupLayout.PREFERRED_SIZE,
191, GroupLayout.PREFERRED_SIZE)
                                    .addComponent(btnViewGraph,
GroupLayout.PREFERRED_SIZE, 191, GroupLayout.PREFERRED_SIZE)
//                                .addComponent(btnDeleteGraph,
GroupLayout.PREFERRED_SIZE, 191, GroupLayout.PREFERRED_SIZE)
//                                .addComponent(btnUpdateGraph,
GroupLayout.PREFERRED_SIZE, 191, GroupLayout.PREFERRED_SIZE)
                                    .addComponent(btnNewButton,
GroupLayout.PREFERRED_SIZE, 191, GroupLayout.PREFERRED_SIZE))
                                .addContainerGap(101, Short.MAX_VALUE))
                            );
gl_contentPane.setVerticalGroup(
    gl_contentPane.createParallelGroup(Alignment.LEADING)
        .addGroup(gl_contentPane.createSequentialGroup()
            .addContainerGap()
            .addComponent(lblGraphinfo)
            .addGap(38)
            .addComponent(btnNewButton, GroupLayout.PREFERRED_SIZE,
37, GroupLayout.PREFERRED_SIZE)
                .addGap(18)
                // .addComponent(btnUpdateGraph,
GroupLayout.PREFERRED_SIZE, 37, GroupLayout.PREFERRED_SIZE)
                // .addGap(18)
//                .addComponent(btnDeleteGraph, GroupLayout.PREFERRED_SIZE,
37, GroupLayout.PREFERRED_SIZE)
//                .addGap(18)
                .addComponent(btnViewGraph, GroupLayout.PREFERRED_SIZE,
37, GroupLayout.PREFERRED_SIZE)
                    .addGap(18)
                    .addComponent(btnBack, GroupLayout.PREFERRED_SIZE, 37,
GroupLayout.PREFERRED_SIZE)
                        .addContainerGap(16, Short.MAX_VALUE))
            );
contentPane.setLayout(gl_contentPane);
}

```

```
}
```

## Bargraph.java:

```
import javax.swing.JFrame;
import javax.swing.SwingUtilities;

import org.jfree.chart.ChartFactory;
import org.jfree.chart.ChartPanel;
import org.jfree.chart.JFreeChart;
import org.jfree.chart.plot.PlotOrientation;
import org.jfree.data.category.CategoryDataset;
import org.jfree.data.category.DefaultCategoryDataset;

public class Bargraph extends JFrame {

    private static final long serialVersionUID = 1L;

    public Bargraph(String appTitle) {
        super(appTitle);

        CategoryDataset dataset = createDataset();

        JFreeChart chart=ChartFactory.createBarChart(
            "Bar Chart College Network", //Chart Title
            "Day", // Category axis
            "Frequency", // Value axis
            dataset,
            PlotOrientation.VERTICAL,
            true,true,false
        );

        ChartPanel panel=new ChartPanel(chart);
        setContentPane(panel);
    }

    private CategoryDataset createDataset() {
        DefaultCategoryDataset dataset = new DefaultCategoryDataset();

        dataset.addValue(423, "Devices connected", "1");
        dataset.addValue(482.81, "Data Bandwidth(GB)", "1");

        dataset.addValue(235, "Devices connected", "2");
        dataset.addValue(232.09, "Data Bandwidth(GB)", "2");

        dataset.addValue(332, "Devices connected", "3");
        dataset.addValue(414.22, "Data Bandwidth(GB)", "3");

        dataset.addValue(278, "Devices connected", "4");
        dataset.addValue(313.63, "Data Bandwidth(GB)", "4");

        dataset.addValue(662, "Devices connected", "5");
        dataset.addValue(1010.61, "Data Bandwidth(GB)", "5");

        return dataset;
    }

    public static void main(String[] args) throws Exception {
```

```
SwingUtilities.invokeLater(()->{
    Bargraph example=new Bargraph("Bar Chart Window");
    example.setSize(800, 400);
    example.setLocationRelativeTo(null);


















    example.setVisible(true);
});
}
```


















## GitHub links and folder structure:

<https://github.com/HARSHITH-7171/COLLEGE-DATABASE-NETWORK>

📁 > This PC > Desktop > networkdb >



















Name	Date modified	Type	Size
📁 .settings	27-06-2022 09:47 PM	File folder	
📁 bin	27-06-2022 09:47 PM	File folder	
📁 src	27-06-2022 09:47 PM	File folder	
📄 .classpath	27-06-2022 07:35 PM	CLASSPATH File	1 KB
📄 .project	12-06-2022 02:00 PM	PROJECT File	1 KB

Name	Date modified	Type	Size
 Adddata_usage\$1	27-06-2022 07:35 PM	CLASS File	1 KB
 Adddata_usage\$2	27-06-2022 07:35 PM	CLASS File	2 KB
 Adddata_usage	27-06-2022 07:35 PM	CLASS File	5 KB
 Adddepartment\$1	27-06-2022 07:35 PM	CLASS File	1 KB
 Adddepartment\$2	27-06-2022 07:35 PM	CLASS File	2 KB
 Adddepartment	27-06-2022 07:35 PM	CLASS File	5 KB
 Bargraph	27-06-2022 08:31 PM	CLASS File	3 KB
 Data_usage\$1	27-06-2022 07:35 PM	CLASS File	1 KB
 Data_usage\$2	27-06-2022 07:35 PM	CLASS File	1 KB
 Data_usage\$3	27-06-2022 07:35 PM	CLASS File	1 KB
 Data_usage\$4\$1	27-06-2022 07:35 PM	CLASS File	2 KB
 Data_usage\$4	27-06-2022 07:35 PM	CLASS File	4 KB
 Data_usage\$5	27-06-2022 07:35 PM	CLASS File	1 KB
 Data_usage	27-06-2022 07:35 PM	CLASS File	4 KB
 Data_usagedetails	27-06-2022 07:35 PM	CLASS File	3 KB
 DB	27-06-2022 07:35 PM	CLASS File	1 KB
 Deletedata_usage\$1	27-06-2022 07:35 PM	CLASS File	1 KB

Name	Date modified	Type	Size
 Deletedata_usage\$1	27-06-2022 07:35 PM	CLASS File	1 KB
 Deletedata_usage\$2	27-06-2022 07:35 PM	CLASS File	2 KB
 Deletedata_usage	27-06-2022 07:35 PM	CLASS File	4 KB
 Deleteddepartment\$1	27-06-2022 07:35 PM	CLASS File	1 KB
 Deleteddepartment\$2	27-06-2022 07:35 PM	CLASS File	2 KB
 Deleteddepartment	27-06-2022 07:35 PM	CLASS File	4 KB
 Department\$1	27-06-2022 09:06 PM	CLASS File	1 KB
 Department\$2	27-06-2022 09:06 PM	CLASS File	1 KB
 Department\$3	27-06-2022 09:06 PM	CLASS File	1 KB
 Department\$4\$1	27-06-2022 09:06 PM	CLASS File	2 KB
 Department\$4	27-06-2022 09:06 PM	CLASS File	4 KB
 Department\$5	27-06-2022 09:06 PM	CLASS File	1 KB
 Department	27-06-2022 09:06 PM	CLASS File	4 KB
 Departmentdetails	27-06-2022 07:35 PM	CLASS File	2 KB
 Graph\$1	27-06-2022 09:15 PM	CLASS File	1 KB
 Graph\$2	27-06-2022 09:15 PM	CLASS File	1 KB
 Graph\$3\$1	27-06-2022 09:15 PM	CLASS File	1 KB



› This PC › Desktop › networkdb › bin

Name	Date modified	Type	Size
 Department\$	27-06-2022 09:06 PM	CLASS File	1 KB
 Department	27-06-2022 09:06 PM	CLASS File	4 KB
 Departmentdetails	27-06-2022 07:35 PM	CLASS File	2 KB
 Graph\$1	27-06-2022 09:15 PM	CLASS File	1 KB
 Graph\$2	27-06-2022 09:15 PM	CLASS File	1 KB
 Graph\$3\$1	27-06-2022 09:15 PM	CLASS File	1 KB
 Graph\$3	27-06-2022 09:15 PM	CLASS File	4 KB
 Graph\$4	27-06-2022 09:15 PM	CLASS File	1 KB
 Graph	27-06-2022 09:15 PM	CLASS File	4 KB
 Menu\$1	27-06-2022 07:35 PM	CLASS File	1 KB
 Menu\$2	27-06-2022 07:35 PM	CLASS File	1 KB
 Menu\$3	27-06-2022 07:35 PM	CLASS File	1 KB
 Menu\$4	27-06-2022 09:01 PM	CLASS File	1 KB
 Menu	27-06-2022 07:35 PM	CLASS File	4 KB
 Viewdata_usage\$1	27-06-2022 07:35 PM	CLASS File	1 KB
 Viewdata_usage	27-06-2022 07:35 PM	CLASS File	3 KB
 Viewdepartment\$1	27-06-2022 07:35 PM	CLASS File	1 KB
 Viewdepartment	27-06-2022 07:35 PM	CLASS File	3 KB

## Testing:

## Java GUI Testing:

# COLLEGE NETWORK DATABASE

Departments Info

Data Usage Info

Graphs Info

## Department Information

Add Department

Delete Data Usage details

View All Data usage

Back

## Department Information

[Add Department](#)[Delete Data Usage details](#)[View All Data usage](#)[Back](#)

did	dname	ip_address
100	CAFETERIA	192.1.2.448
143	FASHION DESIGNING	192.0.4.126
200	LIBRARY	192.1.2.512
400	BIO TECHNOLOGY	192.0.4.156
732	CIVIL	192.1.2.0
733	CSE	192.1.2.64
734	EEE	192.1.2.128
735	ECE	192.1.2.192
736	MECHANICAL	192.1.2.256
737	IT	192.1.2.320
748	CSE-AIML	192.1.2.384

Data Usage Info

## Department Information

Add Department

Delete Data Usage details

View All Data usage

Back

did	dname	ip_address
100	CAFETERIA	192.1.2.448
143	FASHION DESIGNING	192.0.4.126
200	LIBRARY	192.1.2.512
400	BIO TECHNOLOGY	192.0.4.156
732	CIVIL	192.1.2.0
733	CSE	192.1.2.64
734	EEE	192.1.2.128
735	ECE	192.1.2.192
736	MECHANICAL	192.1.2.256
737	IT	192.1.2.320
748	CSE-AIML	192.1.2.384

Department Info.

### Add Department Details

Department ID :

800

Department Name:

HOTEL MANAGEMENT

IP Address :

192.1.2.143

Submit

Message

Department added successfully!

OK

## Department Information

Add Department

Delete Data Usage details

View All Data usage

Back


did	dname	ip_address
100	CAFETERIA	192.1.2.448
143	FASHION DESIGNING	192.0.4.126
200	LIBRARY	192.1.2.512
400	BIO TECHNOLOGY	192.0.4.156
732	CIVIL	192.1.2.0
733	CSE	192.1.2.64
734	EEE	192.1.2.128
735	ECE	192.1.2.192
736	MECHANICAL	192.1.2.256
737	IT	192.1.2.320
748	CSE-AIML	192.1.2.384
800	HOTEL MANAGEMENT	192.1.2.143

Delete department Info.

Enter did number :

Delete

Message

 Record deleted successfully!

OK

## Data Usage Information

Add Data Usage

Delete Data Usage details

View All Data usage

Back

## Data Usage Information

[Add Data Usage](#)[Delete Data Usage details](#)[View All Data usage](#)[Back](#)

did	day	devices_connected	data_bandwidth_GB	date	
737	1	201	233.94	2022-04-17	▲
735	1	101	69.96	2022-04-17	
100	1	53	70.315	2022-04-17	
200	1	22	13.894	2022-04-17	
732	1	46	94.694	2022-04-17	
736	2	87	60.194	2022-04-20	
733	2	27	35.7894	2022-04-20	
100	2	67	45.4673	2022-04-20	
735	2	54	90.6473	2022-04-20	
748	3	67	93.3673	2022-04-22	
737	3	103	207.3373	2022-04-22	
734	3	73	67.6473	2022-04-22	
732	3	89	45.873	2022-04-22	
748	4	67	37.6793	2022-04-24	
736	4	89	82.6793	2022-04-24	
734	4	105	156.7493	2022-04-24	
100	4	17	36.5293	2022-04-24	
737	5	120	224.6893	2022-04-27	
200	5	12	24.9593	2022-04-27	
733	5	122	248.2364	2022-04-27	
735	5	400	512.7174	2022-04-27	▼



Data Usage Info

## Data Usage Information

Add Data Usage
Delete Data Usage details
View All Data usage
Back

did	day	devices_connected	data_bandwidth_GB	date
732	1	46	94.694	2022-04-17
736	2	87	60.194	2022-04-20
733	2	27	35.7894	2022-04-20
100	2	67	45.4673	2022-04-20
735	2	54	90.6473	2022-04-20
748	3	67	93.3673	2022-04-22
737	3	103	207.3373	2022-04-22
734	3	73	67.6473	2022-04-22
732	3	89	45.873	2022-04-22
748	4	67	37.6793	2022-04-24
736	4	89	82.6793	2022-04-24
734	4	105	156.7493	2022-04-24
100	4	17	36.5293	2022-04-24
737	5	120	224.6893	2022-04-27
200	5	12	24.9593	2022-04-27
733	5	122	248.2364	2022-04-27
735	5	408	512.7171	2022-04-27
100	4	785	865.5	2022-06-25
737	7	789	4589.0	2022-06-23
100	20	456	789.56	2022-06-26

Message

Department added successfully!

OK

Add Data Usage Details

Department ID : 737
Day Number: 26
No. of devices connected : 456
Amount of Data Consumed(GB) : 896.0
Date : 2022-06-27

Submit

Data Usage Info

## Data Usage Information

Add Data Usage
Delete Data Usage details
View All Data usage
Back

did	day	devices_connected	data_bandwidth_GB	date
736	2	87	60.194	2022-04-20
733	2	27	35.7894	2022-04-20
100	2	67	45.4673	2022-04-20
735	2	54	90.6473	2022-04-20
748	3	67	93.3673	2022-04-22
737	3	103	207.3373	2022-04-22
734	3	73	67.6473	2022-04-22
732	3	89	45.873	2022-04-22
748	4	67	37.6793	2022-04-24
736	4	89	82.6793	2022-04-24
734	4	105	156.7493	2022-04-24
100	4	17	36.5293	2022-04-24
737	5	120	224.6893	2022-04-27
200	5	12	24.9593	2022-04-27
733	5	122	248.2364	2022-04-27
735	5	408	512.7171	2022-04-27
100	4	785	865.5	2022-06-25
737	7	789	4589.0	2022-06-23
100	20	456	789.56	2022-06-26
737	26	456	896.0	2022-06-27

Delete data\_usage Info.

Enter day number : 20

Delete

Message

Record deleted successfully!

OK

## Graph Information

View Bar Graph

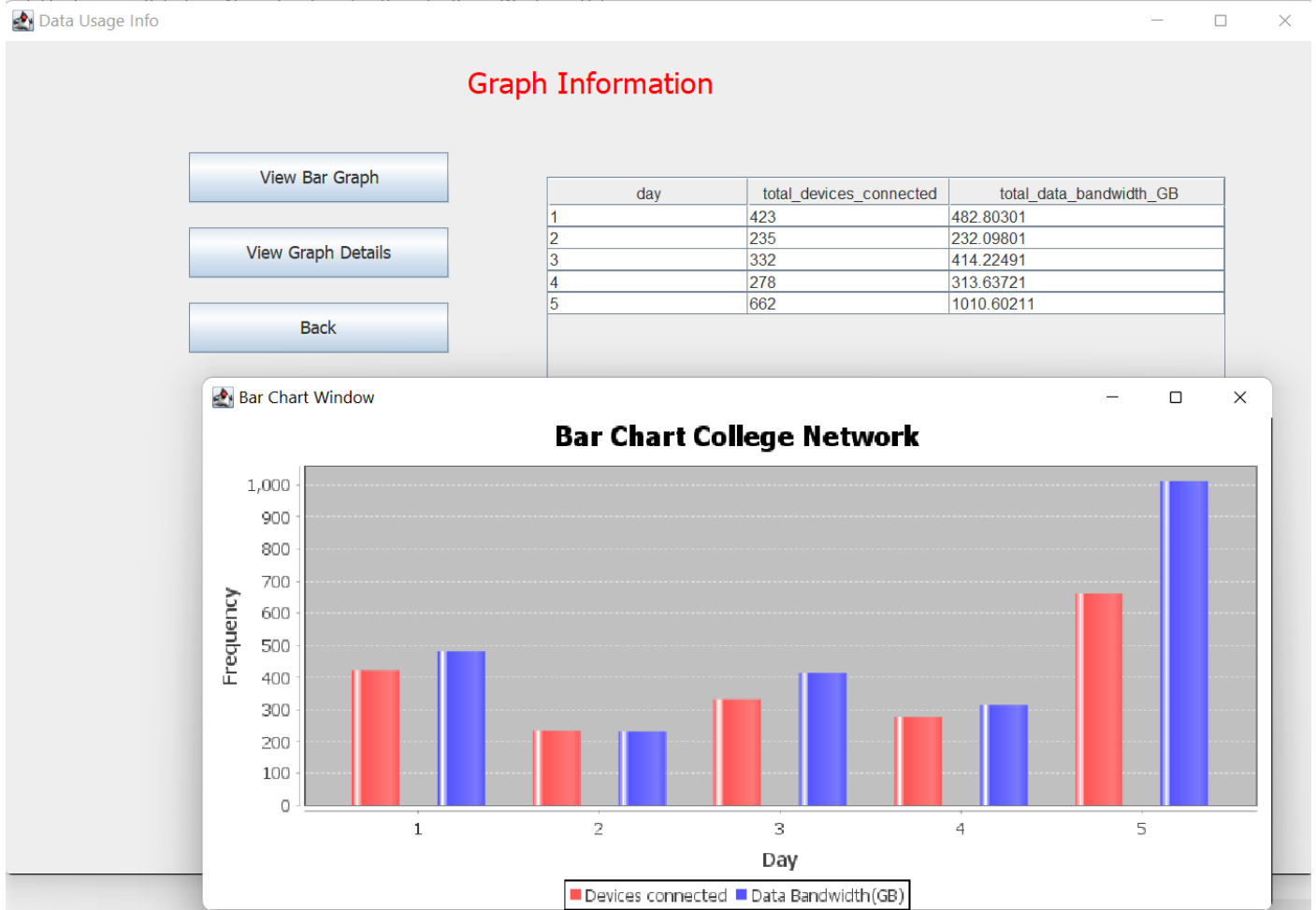
View Graph Details

Back

## Graph Information

[View Bar Graph](#)[View Graph Details](#)[Back](#)

day	total_devices_connected	total_data_bandwidth_GB
1	423	482.80301
2	235	232.09801
3	332	414.22491
4	278	313.63721
5	662	1010.60211



The data entered in the above form is updated in the “departments” table of the MySQL database as:

```
mysql> select * from departments;
```

did	dname	ip_address
100	CAFETERIA	192.1.2.448
200	LIBRARY	192.1.2.512
400	BIO TECHNOLOGY	192.0.4.156
732	CIVIL	192.1.2.0
733	CSE	192.1.2.64
734	EEE	192.1.2.128
735	ECE	192.1.2.192
736	MECHANICAL	192.1.2.256
737	IT	192.1.2.320
748	CSE-AIML	192.1.2.384
800	HOTEL MANAGEMENT	192.1.2.143

---

**Results:** I successfully completed this MINI PROJECT “COLLEGE DATABASE NETWORK”.

**Discussion and Future work:**

While doing this project I got new ideas I understood how to work on projects. Now to further extend this project I want to create a android app by which I can control my project on my hand and connect to it.

**References:**

- <https://docs.oracle.com/javase/7/docs/api/javax/swing/package-summary.html>
- <https://www.javatpoint.com/mysql-connection>
- <https://dev.mysql.com/doc/dev/connector-j/8.0/>

\*\*\*\*\*THE END\*\*\*\*\*