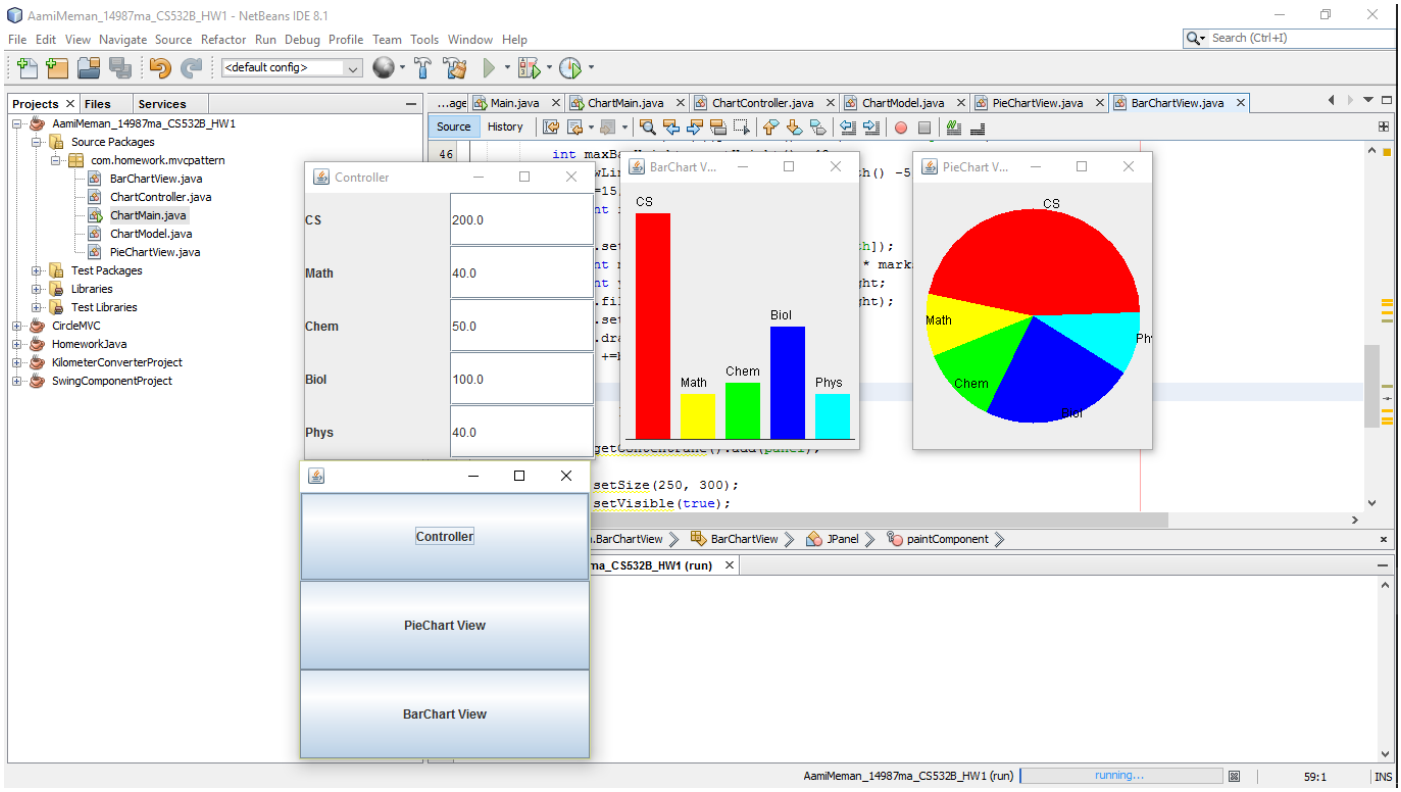

-

ChartMain.java

* Student Info: Name=Meman Aamir, ID=14987ma

* Subject: CS532B_HWNo1_Summer_2016

* Author: meman

* Filename: ChartMain.java

* Date and Time: May 26, 2016 7:40:49 PM

* Project Name: AamiMeman_14987ma_CS532B_HW1

```
package com.homework.mvcpattern;

import javax.swing.*;

import javax.swing.event.*;

import java.util.*;

import java.awt.*;

import java.awt.event.ActionEvent;
```

```
import java.awt.event.ActionListener;
public class ChartMain extends JFrame {
    JButton btncontroller = new JButton("Controller");
    JButton btnpiechart = new JButton("PieChart View");
    JButton btnbarchart = new JButton("BarChart View");
    ChartModel model = new ChartModel();
  public ChartMain() {
        this.setLayout(new GridLayout(3, 1));
        this.getContentPane().add(btncontroller);
        this.getContentPane().add(btnpiechart);
        this.getContentPane().add(btnbarchart);
        btncontroller.addActionListener(new ActionListener() {
            public void actionPerformed(ActionEvent e) {
                ChartController controller = new ChartController(model);
            }
        });
        btnpiechart.addActionListener(new ActionListener() {
            public void actionPerformed(ActionEvent e) {
                PieChartView piechart = new PieChartView(model);
            }
        });
        btnbarchart.addActionListener(new ActionListener() {
            public void actionPerformed(ActionEvent e) {
                BarChartView barchart = new BarChartView(model);
            }
```

```
        });
        this.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        this.setSize(300, 300);
        this.setVisible(true);
    }
    public static void main(String[] args) {
        ChartMain frame = new ChartMain();
    }
}
```

ChartModel.java

```
 * Student Info: Name=Meman Aamir, ID=14987ma
 * Subject: CS532B_HWNo1_Summer_2016
 * Author: meman
 * Filename: ChartModel.java
 * Date and Time: May 26, 2016 7:54:45 PM
 * Project Name: AamiMeman_14987ma_CS532B_HW1
package com.homework.mvcpattern;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import java.util.ArrayList;
class ChartModel {
    private String[] dataName = {"CS", "Math", "Chem", "Biol", "Phys"};
    private double[] data = {200, 40, 50, 100, 40};
    private ArrayList<ActionListener> actionListenerList = new ArrayList<>();
```

```
    public void addActionListener(ActionListener l) {
        actionListenerList.add(l);
    }
    public void removeActionListener(ActionListener l) {
        actionListenerList.remove(l);
    }
    private void processEvent(ActionEvent e) {
        for (ActionListener listener : actionListenerList) {
            listener.actionPerformed(e);
        }
    }
    public String[] getDataName() {
        return dataName;
    }
    public double[] getData() {
        return data;
    }
    public void setChartData(String[] dataName, double[] data) {
        this.dataName = dataName;
        this.data = data;
        processEvent(new ActionEvent(this, ActionEvent.ACTION_PERFORMED, "data"));
    }
}
```

ChartController.java

```
 * Student Info: Name=Meman Aamir, ID=14987ma
 * Subject: CS532B_HWNo1_Summer_2016
 * Author: meman
 * Filename: ChartController.java
 * Date and Time: May 26, 2016 7:56:14 PM
 * Project Name: AamiMeman_14987ma_CS532B_HW1
package com.homework.mvcpattern;
import java.awt.GridLayout;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import javax.swing.JFrame;
import javax.swing.JLabel;
import javax.swing.JTextField;
class ChartController extends JFrame {
    ChartModel model;
    JTextField dataList;
    JLabel dataLabel;
    ChartController(ChartModel model) {
        this.model = model;
        this.setTitle("Controller");
        this.setLayout(new GridLayout(5, 2));
        this.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        this.setSize(300, 300);
        this.setVisible(true);
```

```
        String[] list = model.getDataName();
        double[] data = model.getData();
        int i = 0;
        for (double value : data) {
            dataList = new JTextField(Double.toString(value));
            dataLabel = new JLabel(list[i]);
            this.getContentPane().add(dataLabel);
            this.getContentPane().add(dataList);
            int j = i;
            dataList.addActionListener(new ActionListener() {
                public void actionPerformed(ActionEvent e) {
                    double foo = Double.parseDouble(dataList.getText());
                    data[j] = foo;
                    model.setChartData(list, data);
                }
            });
            i++;
        }
    }
}
```

PieChart View.java

```
 * Student Info: Name=Meman Aamir, ID=14987ma
 * Subject: CS532B_HWNo1_Summer_2016
* Author: meman
* Filename: PieChartView.java
* Date and Time: Jun 1, 2016 2:39:24 PM
* Project Name: AamiMeman_14987ma_CS532B_HW1
package com.homework.mvcpattern;
import java.awt.Color;
import java.awt.Graphics;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import javax.swing.JFrame;
import javax.swing.JPanel;
public class PieChartView extends JFrame implements ActionListener {
    JPanel panel;
    ChartModel model;
    Color[] colors = {Color.RED, Color.YELLOW, Color.GREEN,
        Color.BLUE, Color.CYAN, Color.MAGENTA, Color.ORANGE,
        Color.PINK, Color.darkGray};
    public PieChartView(ChartModel model) {
        this.model = model;
        model.addActionListener(this);
        this.setTitle("PieChart View");
        this.panel = new JPanel() {
```

```
protected void paintComponent(Graphics g) {
    super.paintComponent(g);
    String[] subject = model.getDataName();
    double[] marks = model.getData();
    int xCenter = getWidth() / 2;
    int yCenter = getHeight() / 2;
    int radius = (int) (Math.min(getWidth(), getHeight()) * 0.5 * 0.9);
    int x = xCenter - radius;
    int y = yCenter - radius;
    double total = 0;
    for (int i = 0; i < marks.length; i++) {
        total += marks[i];
    }
    int angle1 = 0;
    int angle2 = 0;
    for (int i = 0; i < marks.length; i++) {
        angle1 = angle1 + angle2;
        angle2 = (int) Math.ceil(360 * marks[i] / total);
        g.setColor(colors[i % colors.length]);
        g.fillArc(x, y, 2 * radius, 2 * radius, angle1, angle2);
        g.setColor(Color.BLACK);
        g.drawString(subject[i], (int) (getWidth() / 2 + radius * Math.cos((angle1 + angle2 /
2) * 2 * Math.PI / 360)), (int) (getHeight() / 2 - radius * Math.sin((angle1 + angle2 / 2) * 2 *
Math.PI / 360)));


    }
```

```
            }
        };
        this.getContentPane().add(panel);
        this.setSize(250, 300);
        this.setVisible(true);
    }
    @Override
    public void actionPerformed(ActionEvent e) {
        this.panel.repaint();
    }
}
```

BarChart View.java

```
 * Student Info: Name=Meman Aamir, ID=14987ma
 * Subject: CS532B_HWNo1_Summer_2016
 * Author: meman
 * Filename: BarChartView.java
 * Date and Time: Jun 3, 2016 9:05:09 PM
 * Project Name: AamiMeman_14987ma_CS532B_HW1
package com.homework.mvcpattern;
import java.awt.Color;
import java.awt.Graphics;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import javax.swing.JFrame;
```

```
import javax.swing.JPanel;
public class BarChartView extends JFrame implements ActionListener {
 JPanel panel;
 ChartModel model;
  Color[] colors = {Color.RED, Color.YELLOW, Color.GREEN,
      Color.BLUE, Color.CYAN, Color.MAGENTA, Color.ORANGE,
      Color.PINK, Color.darkGray};
   BarChartView(ChartModel model) {
      this.model = model;
      model.addActionListener(this);
      this.setTitle("BarChart View");
      this.panel = new JPanel() {
         protected void paintComponent(Graphics g) {
            super.paintComponent(g);
            String[] subject = model.getDataName();
            double[] marks = model.getData();
                  double max=marks[0];
      for(int i=1;i<marks.length;i++)
         max=Math.max(max, marks[i]);
      int barWidth=(int)((getWidth()-10.0)/marks.length -10);
      int maxBarHeight = getHeight() -40;
      g.drawLine(5, getHeight() -10, getWidth() -5, getHeight()-10);
      int x=15;
      for(int i=0; i < marks.length; i++)
      {
```

```
            g.setColor(colors[i % colors.length]);
            int newHeight= (int)(maxBarHeight * marks[i] / max);
            int y = getHeight () -10 - newHeight;
            g.fillRect(x, y, barWidth, newHeight);
            g.setColor(Color.BLACK);
            g.drawString(subject[i], x, y-7);
            x +=barWidth + 10;
        }
                }
        };
        this.getContentPane().add(panel);
        this.setSize(250, 300);
        this.setVisible(true);
    }
    @Override
    public void actionPerformed(ActionEvent e) {
         this.panel.repaint();
    }
}
```