

# 식단관리

201217884

무역학과

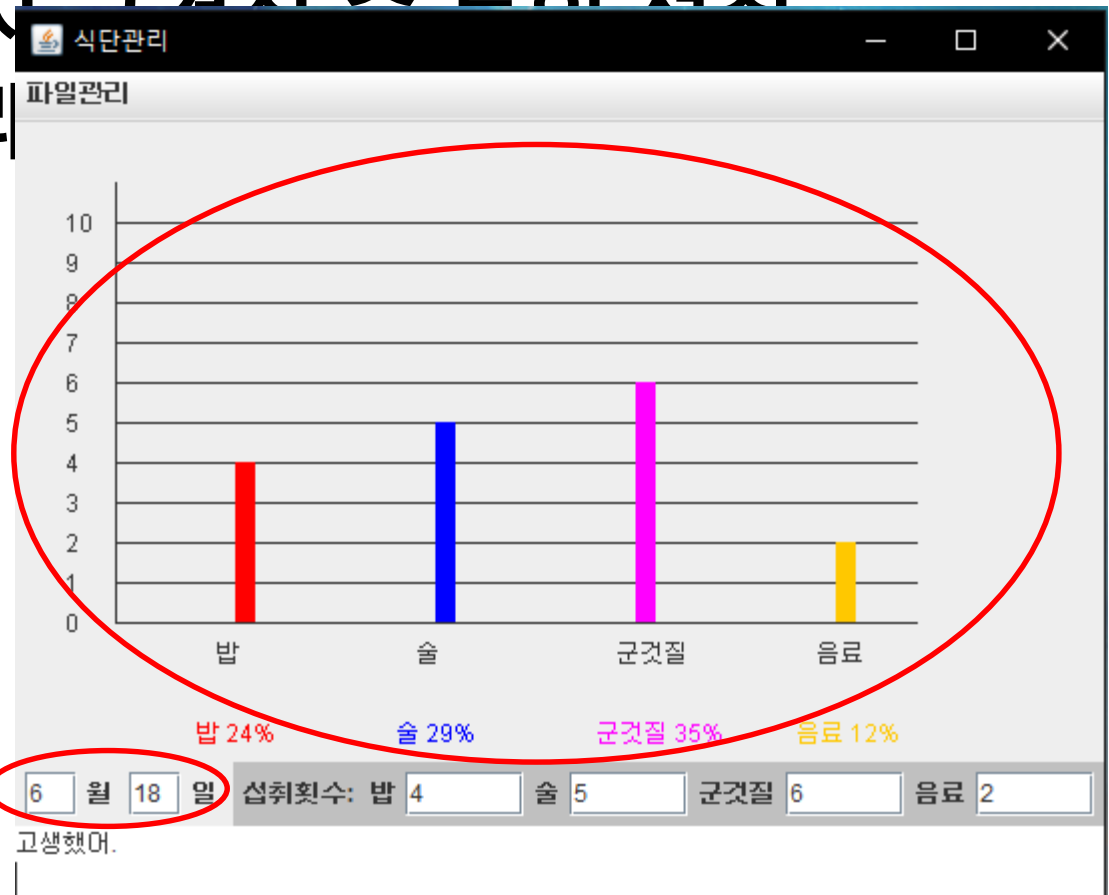
이동석

## 개발 동기

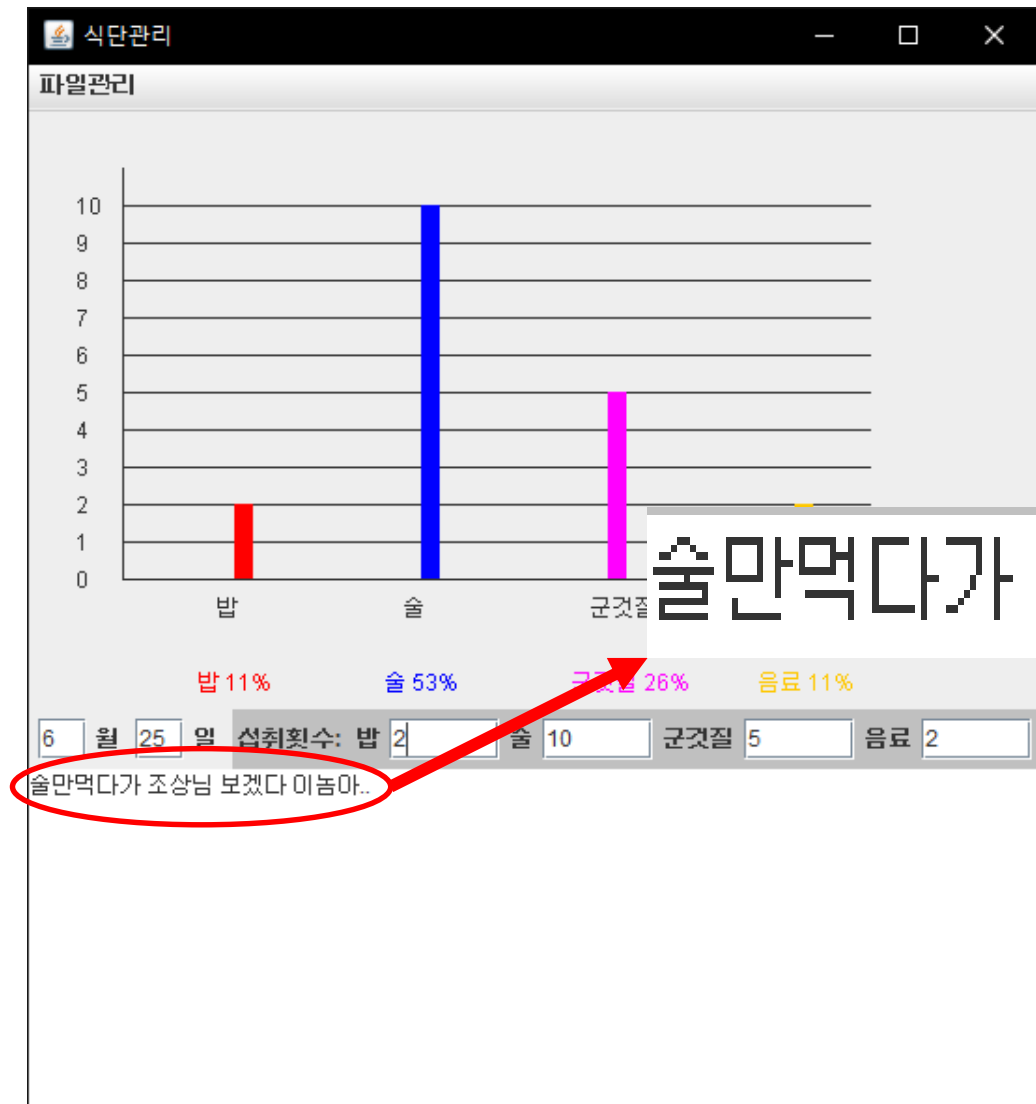
1. 하루하루 나의 식습관을 기록하고 싶어서
2. 식단 관리

## 1. 특정 날짜의 식사 구성비율과 그

3 월 6 일



## 2. 일간 코멘트를 달 수 있는 기능



술만먹다가 조상님 보겠다 이눔아..

## 3. 코멘트 및 수치 저장

식단관리

파일관리

OneDrive > 전북대학교 > 2017년 > JAVA > 프로젝트 > project2017 > MemoData

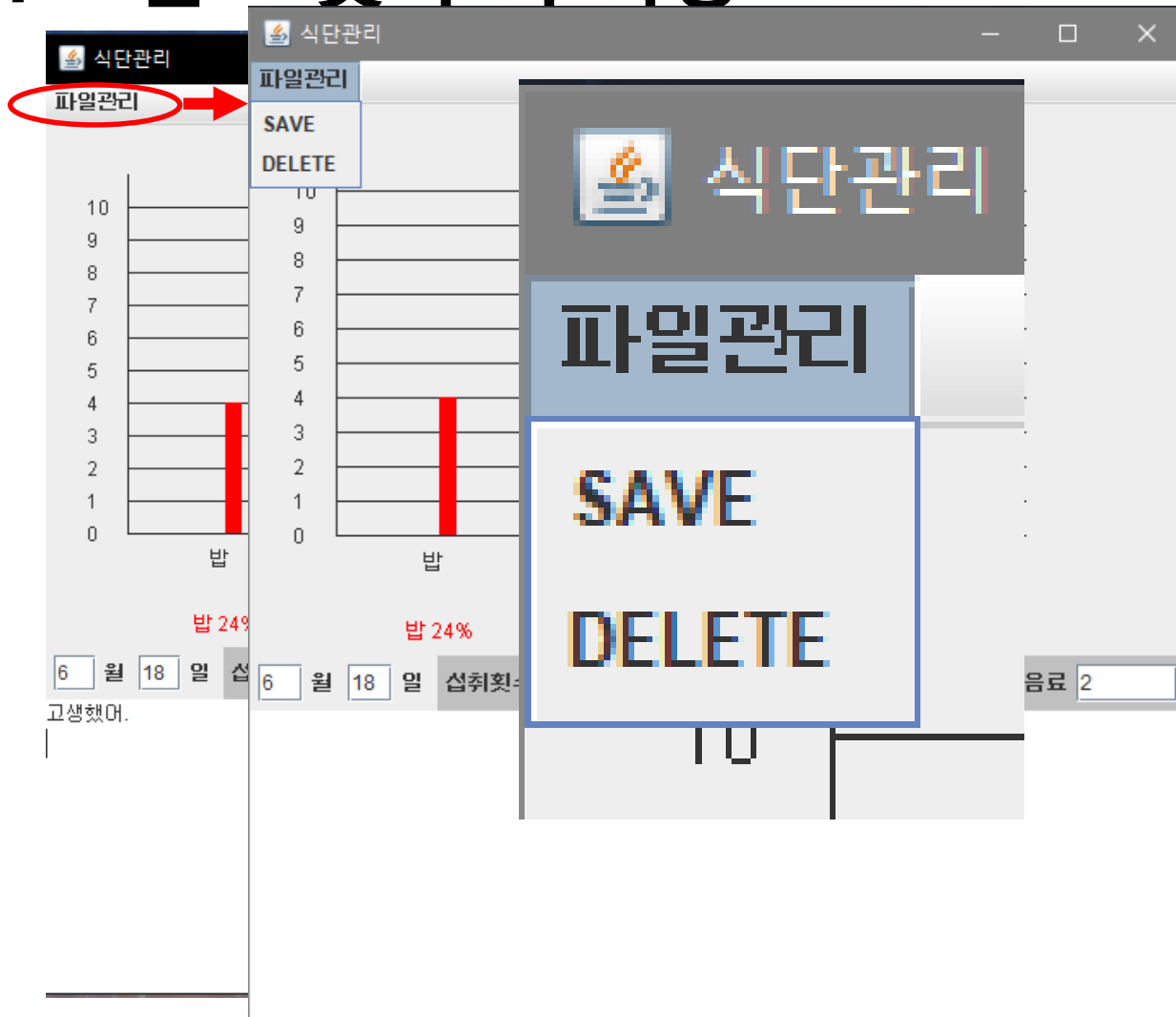
이름	수정한 날짜	유형	크기
<u>20170618</u>	2017-06-18 오후...	텍스트 문서	1KB
<u>20170618num</u>	2017-06-18 오후...	텍스트 문서	1KB
20170620num	2017-06-18 오후...	텍스트 문서	1KB
20170621num	2017-06-18 오후...	텍스트 문서	1KB
20170625	2017-06-18 오후...	텍스트 문서	1KB
20170625num	2017-06-18 오후...	텍스트 문서	1KB

밥 24% 술 29% 군것질 35% 음료 12%

6 월 18 일 섭취횟수: 밥 4 술 5 군것질 6 음료 2

고생했어.

## 3. 코멘트 및 수치 저장



## 4. 숫자만 입력가능

식단관리

파일관리

10

9

8

7

6

5

4

3

2

1

0

밥

밥 24%

6월18일

섭취횟수

고생했어.

```
public class NumberField extends JTextField implements KeyListener { // 텍스트필드에 숫자만 넣게 하기
    private static final long serialVersionUID = 1;

    public NumberField() {
        addKeyListener(this);
    }

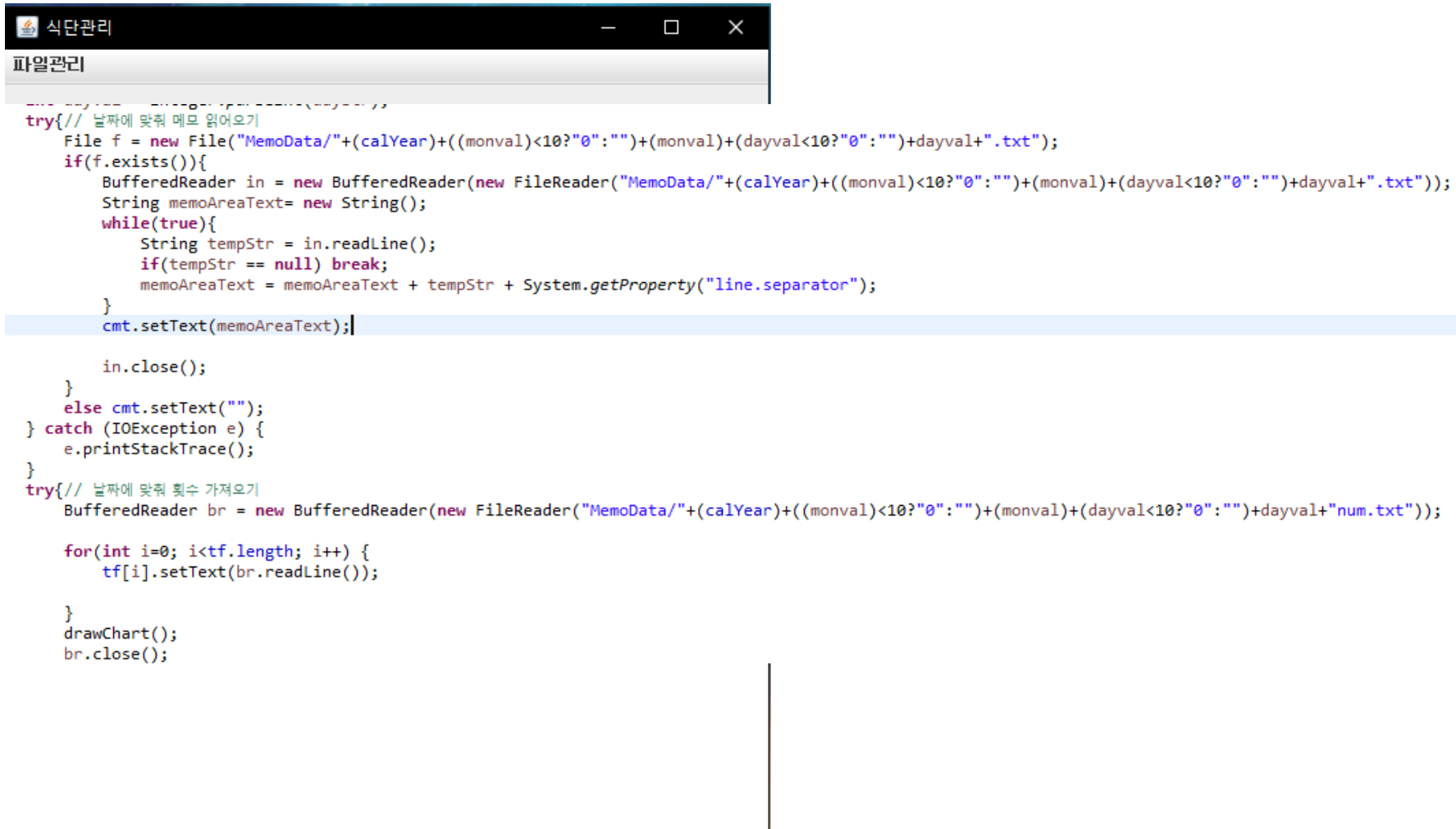
    public void keyPressed(KeyEvent e) {
    }

    public void keyReleased(KeyEvent e) {
    }

    public void keyTyped(KeyEvent e) {
        // Get the current character you typed...
        char c = e.getKeyChar();

        if (!Character.isDigit(c)) {
            e.consume();
            return;
        }
    }
}
```

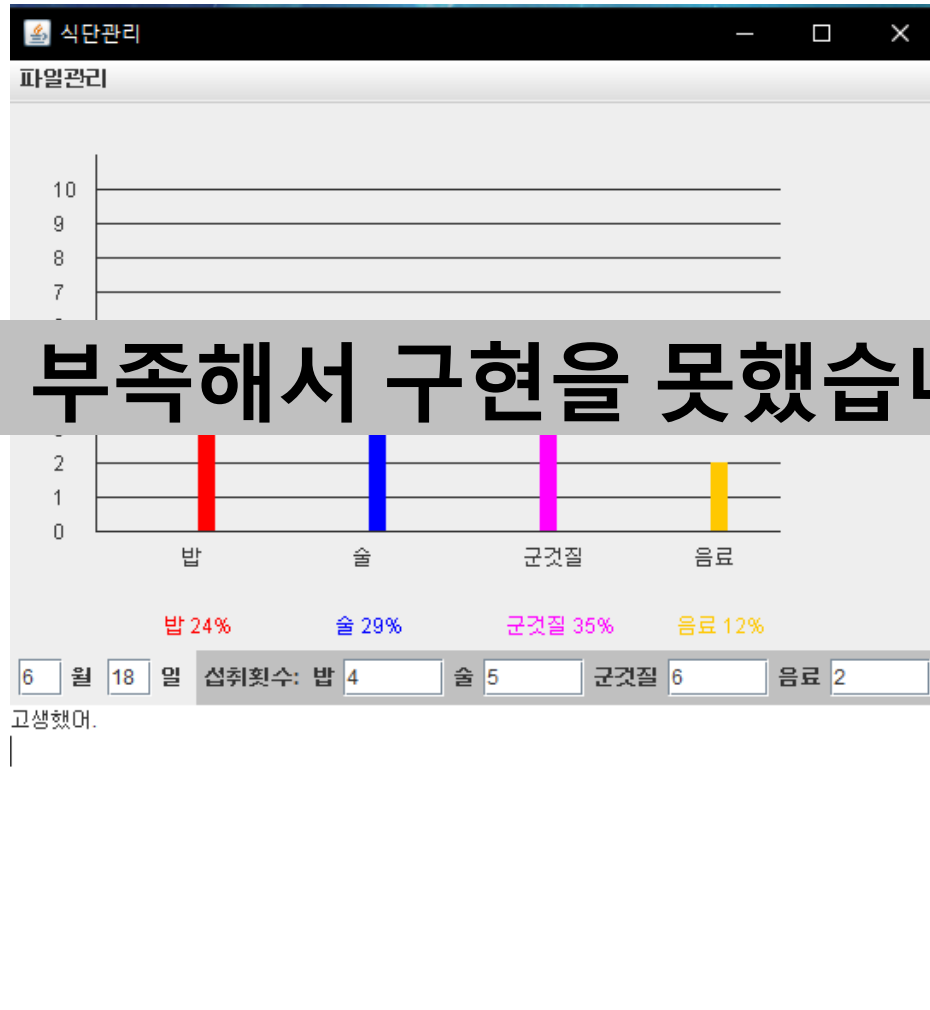
## 5. 해당날짜 텍스트 파일이 있으면 바로 불러오기





# 구현 하고 싶었지만 못한 것

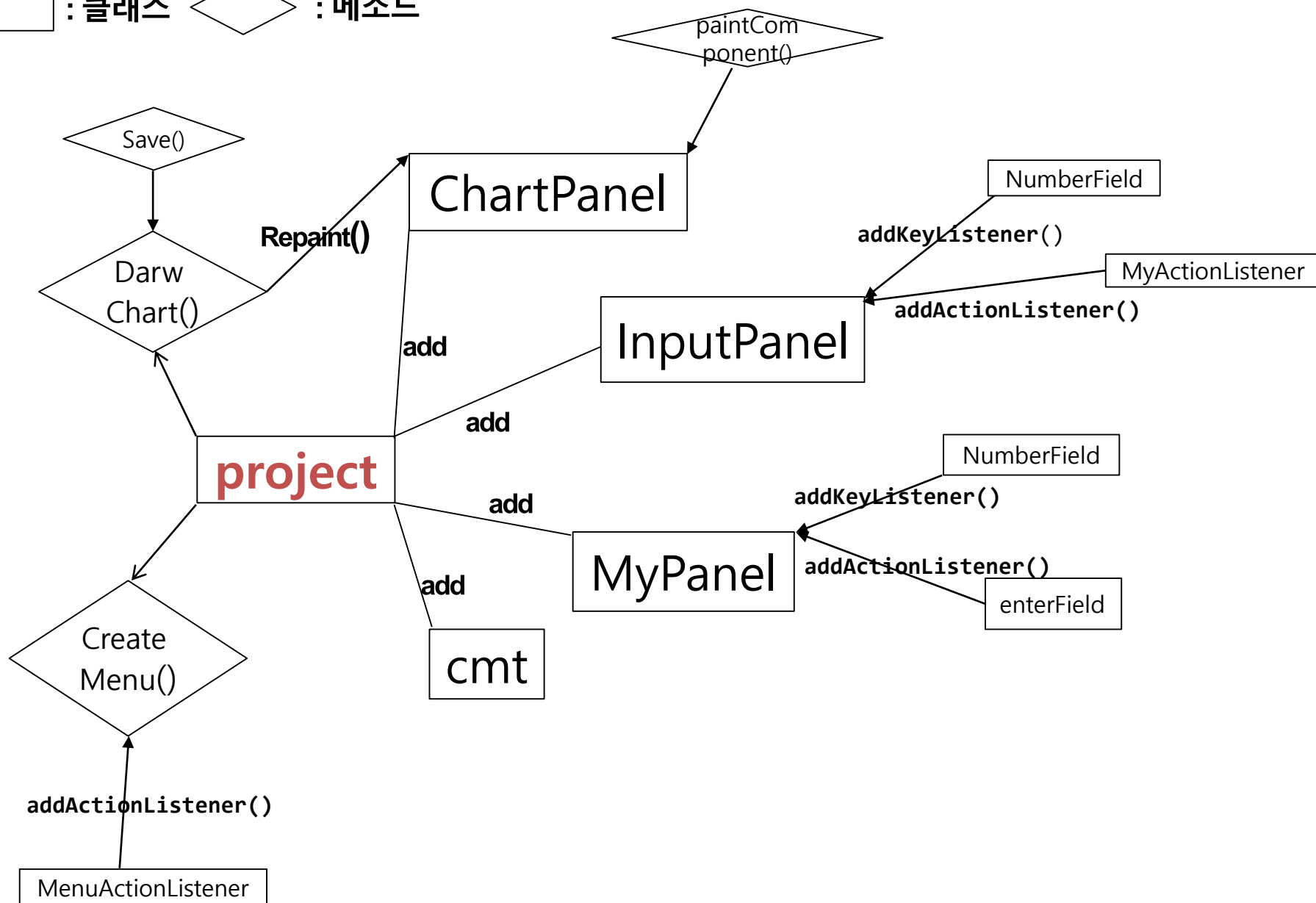
## 6. 주간,월간 통계 분석



시간이 부족해서 구현을 못했습니다.. ㅠㅠ

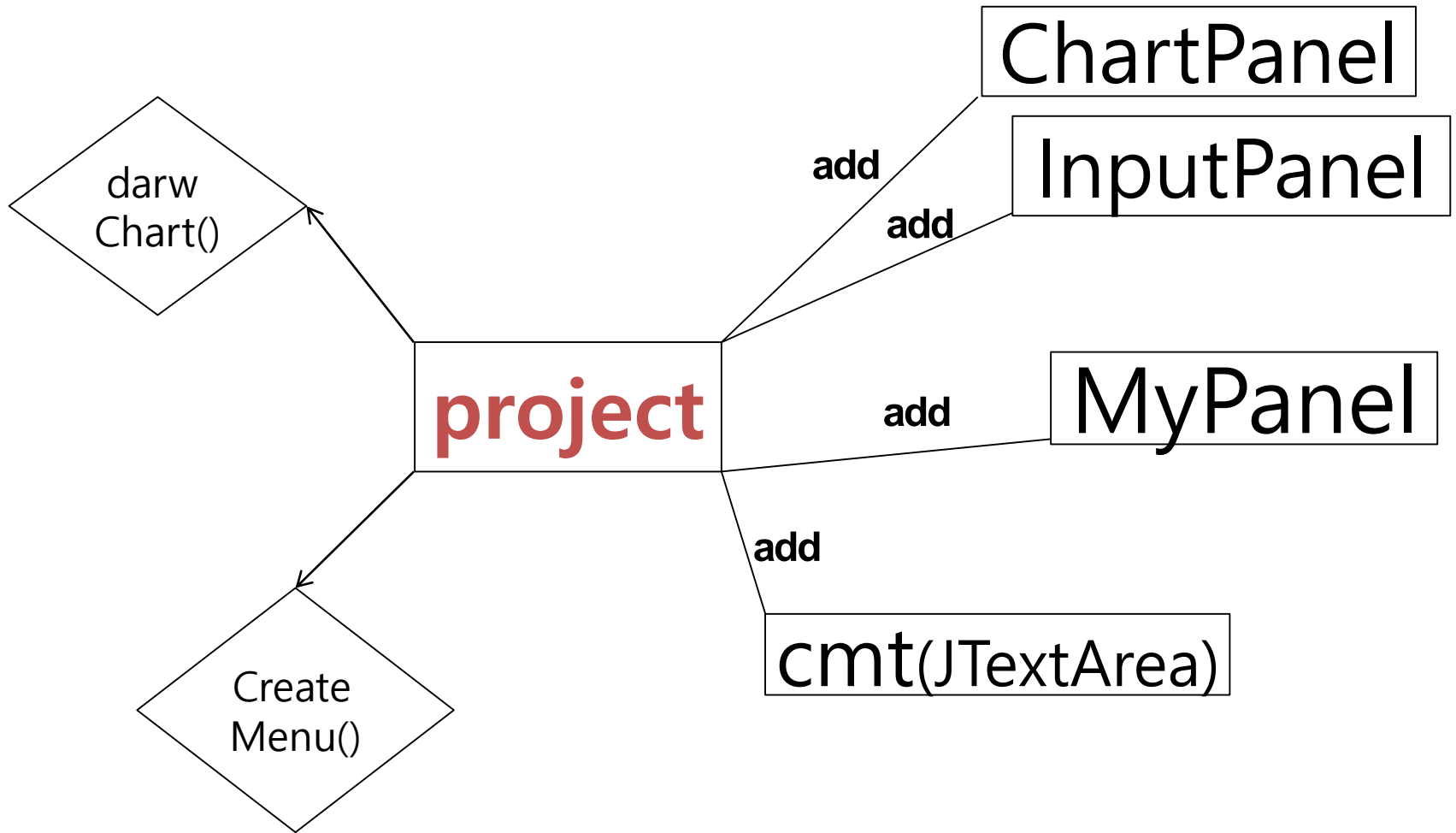
# 클래스 및 메소드 설계

□ : 클래스    ◇ : 메소드



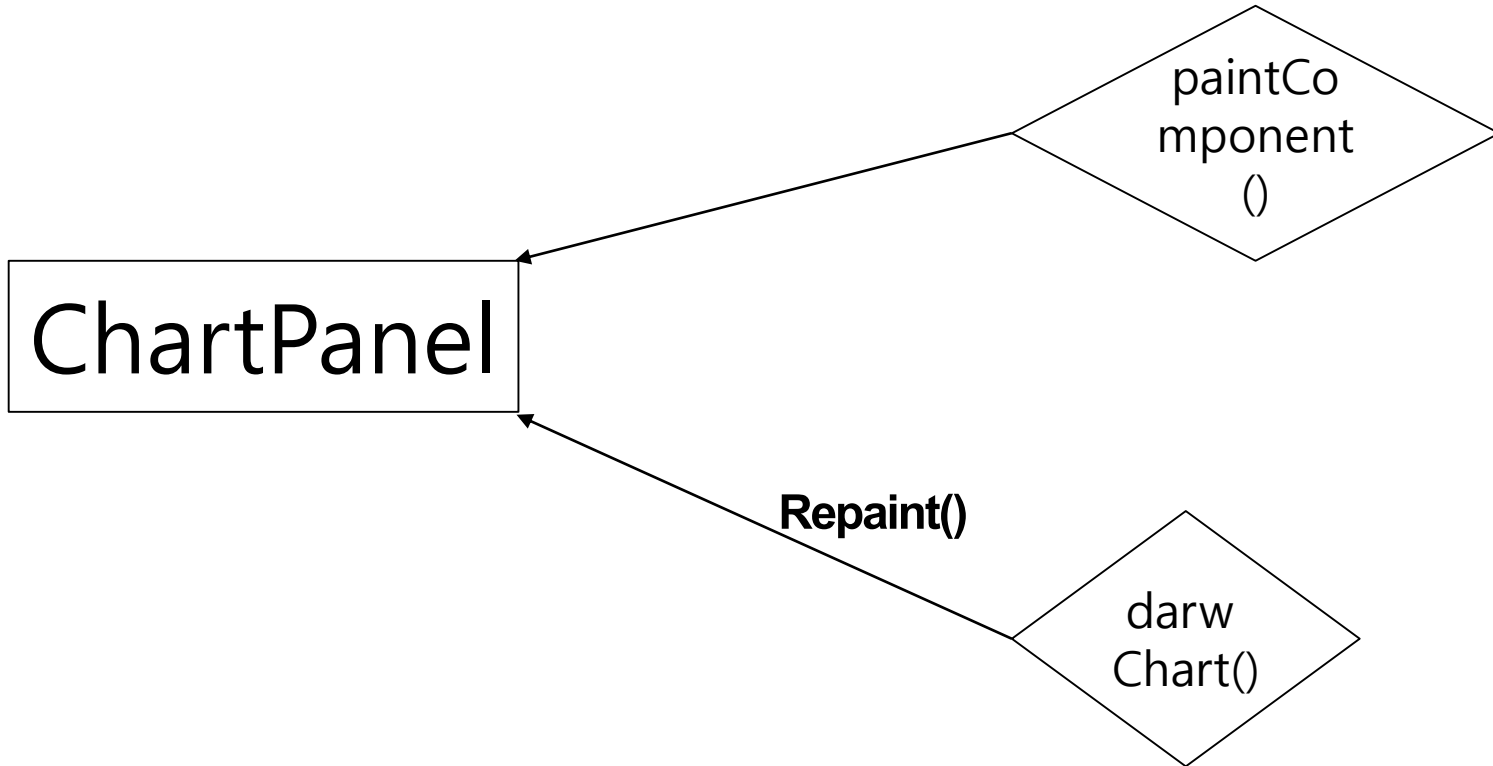
# 클래스 및 메소드 설계

□ : 클래스    ◇ : 메소드



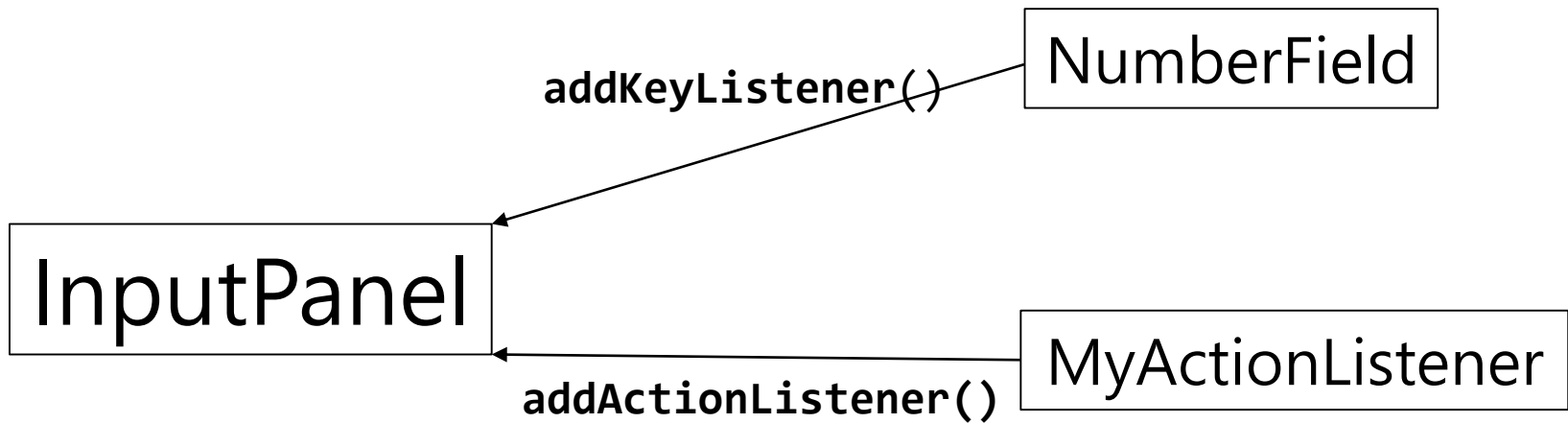
# 클래스 및 메소드 설계

□ : 클래스    ◇ : 메소드



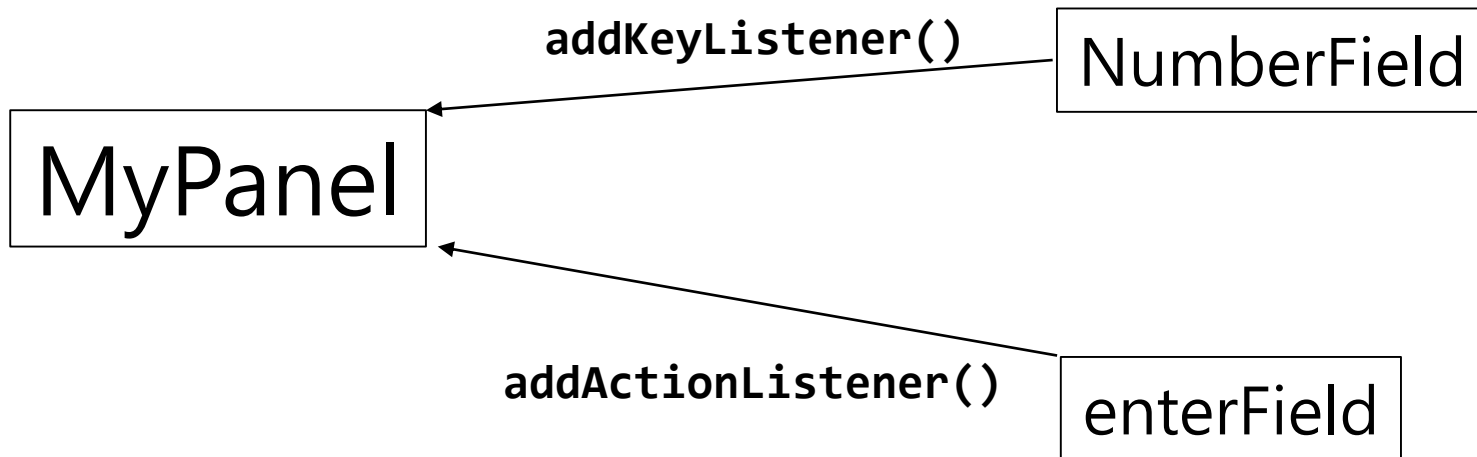
# 클래스 및 메소드 설계

□ : 클래스    ◇ : 메소드



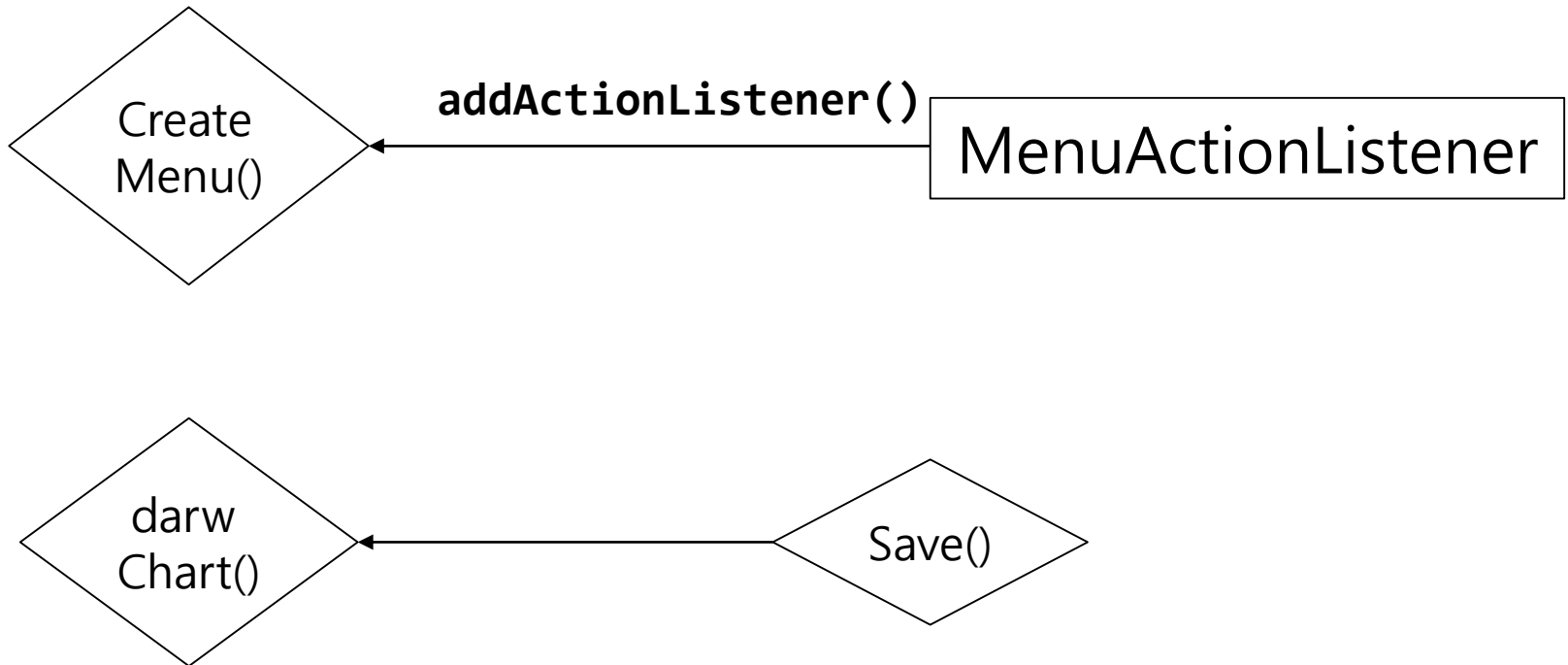
# 클래스 및 메소드 설계

□ : 클래스    ◇ : 메소드



# ≡ 클래스 및 메소드 설계

□ : 클래스    ◇ : 메소드



## BorderLayout

The image shows a Java Swing window titled "식단관리" (Diet Management) with a standard Mac OS X title bar. The window is divided into four regions by BorderLayout:

- BorderLayout.NORTH:** The top region, containing a bar chart showing the percentage distribution of food items.
- BorderLayout.CENTER:** The middle region, containing a row of input fields for food items and their quantities.
- BorderLayout.WEST:** The left region, containing a date picker and a text area.
- BorderLayout.SOUTH:** The bottom region, containing a text area.

The bar chart displays the following data:

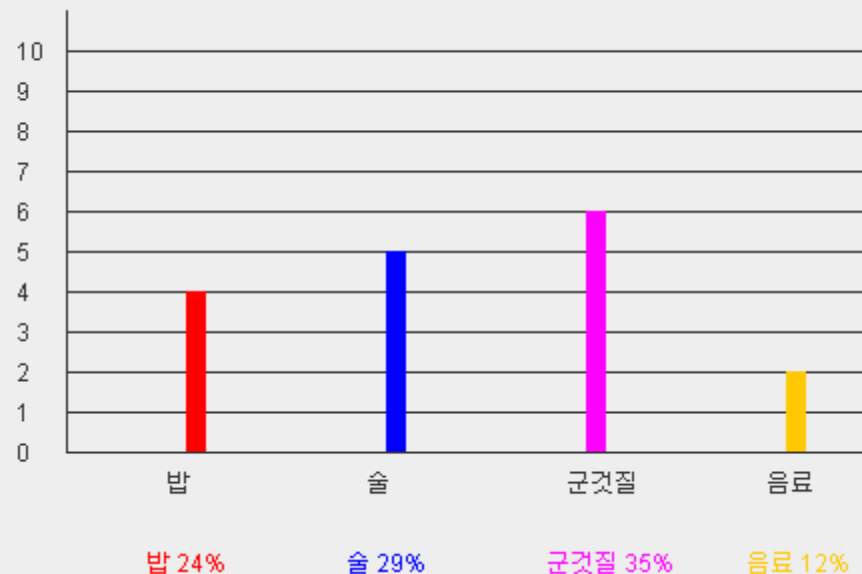
Food Item	Percentage
밥 (Rice)	24%
술 (Alcohol)	29%
군것질 (Snacks)	35%
음료 (Beverages)	12%

The input fields below the chart show the following values:

Food Item	Quantity
밥 (Rice)	4
술 (Alcohol)	5
군것질 (Snacks)	6
음료 (Beverages)	2

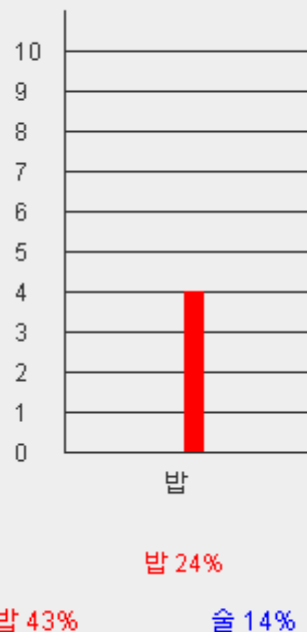


# 클래스 및 메소드 설계



```
public class PieChartFrame extends JFrame {  
    Container contentPane;  
    int [] data = {0,0,0,0};  
    int [] arcAngle = new int [4];  
    Color [] color = {Color.RED, Color.BLUE, Color.MAGENTA, Color.ORANGE};  
    String [] itemName = {"밥", "술", "군것질", "음료"}; // 식단 목록  
    JTextField [] tf = new JTextField [4]; // 섭취횟수  
    ChartPanel chartPanel = new ChartPanel();  
}
```

# 클래스 및 메소드 설계



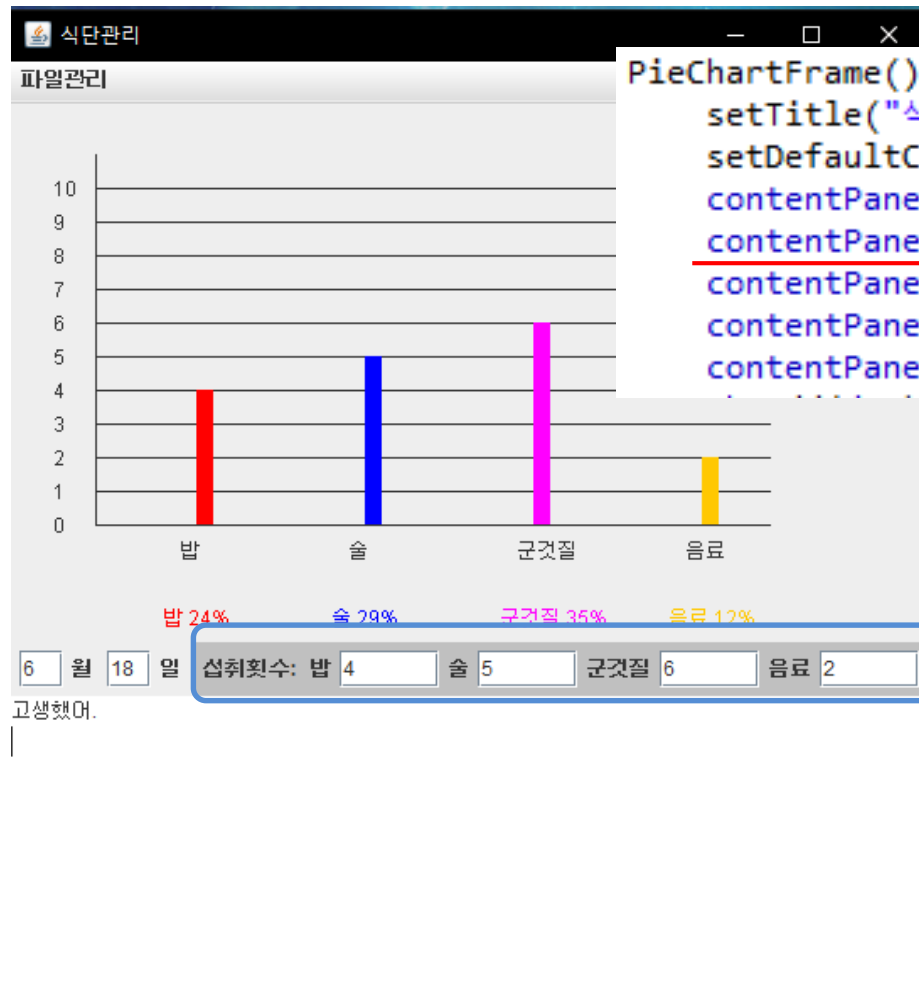
```
class ChartPanel extends JPanel {
    ChartPanel()
    {
        this.setPreferredSize(new Dimension(200,320));
    }
    public void paintComponent(Graphics g) {
        super.paintComponent(g);
        g.drawString(" ", 25, 255);
        g.drawLine(50, 250, 450, 250);
        g.drawLine(50, 30, 50, 250);
        for (int cnt = 1; cnt < 11; cnt++) {
            g.drawString(cnt*1 + " ", 25, 255 - 20*cnt);
            g.drawLine(50, 250 - 20*cnt, 450, 250 - 20*cnt);
        } // 배경에 표기선

        for(int i=0; i<data.length;i++)
        {

            g.drawString(itemName[i], 100 + i*100, 270);
        } // 차트 바르 밑에 표기

        for(int i=0; i<data.length; i++) {
            g.setColor(color[i]);
            g.drawString(itemName[i]+" "+Math.round(arcAngle[i]*100./360.)+"%", 90+i*100, 310);
        } // 식단과 퍼센트
        for(int i=0; i<data.length; i++) {
            g.setColor(color[i]);
            g.fillRect(110 + i*100, 250 - data[i]*20 ,10,data[i]*20);
        } // 차트 그리기
    }
}
```

# 클래스 및 메소드 설계



```
PieChartFrame() {  
    setTitle("식단관리");  
    setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);  
    contentPane = getContentPane();  
    contentPane.add(new InputPanel(), BorderLayout.CENTER);  
    contentPane.add(chartPanel, BorderLayout.NORTH);  
    contentPane.add(cmt, BorderLayout.SOUTH);  
    contentPane.add(new MyPanel(), BorderLayout.WEST);  
}
```

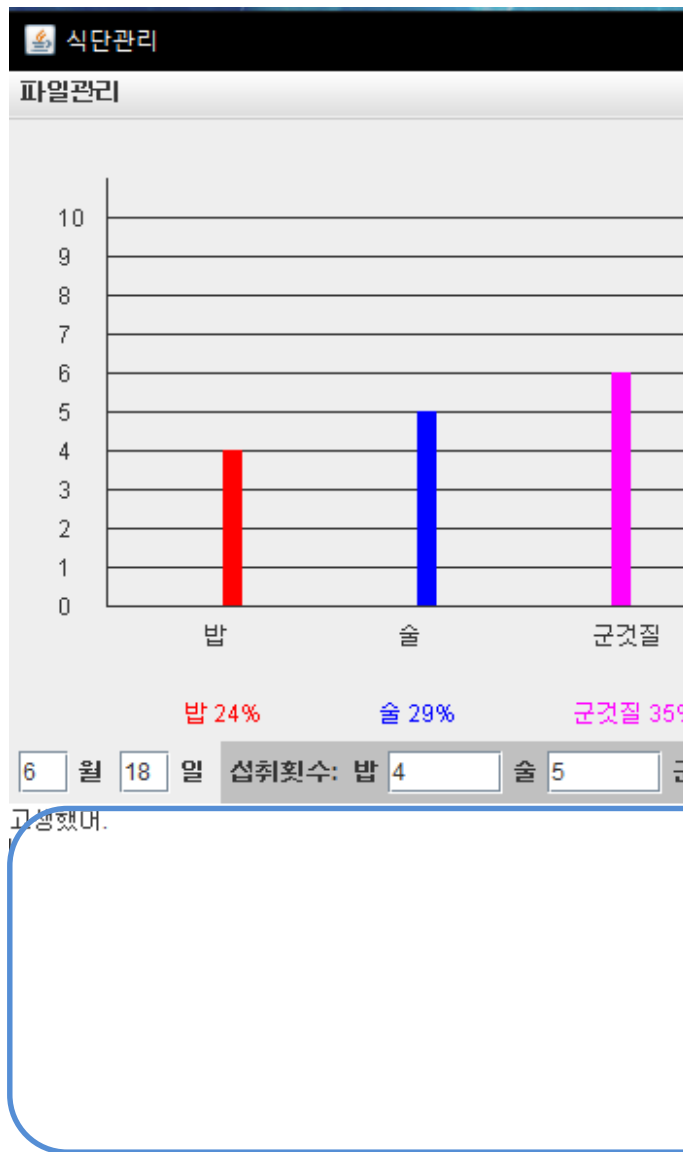
# 클래스 및 메소드 설계

식단관리

```
class InputPanel extends JPanel {  
    InputPanel() { // 선택 횟수 입력패널
```

```
class InputPanel extends JPanel { // 선택 횟수 입력패널  
    InputPanel() {  
        this.setBackground(Color.LIGHT_GRAY);  
        String monthstr = days0.getText();  
        String daystr = days1.getText();  
        int monval = Integer.parseInt(monthstr);  
        int dayval = Integer.parseInt(daystr);  
        add(new JLabel("선택횟수: "));  
  
        try{  
            BufferedReader br = new BufferedReader(new FileReader("MemoData/"+(calYear)+((monval)<10?"0":"")+monval+(dayval<10?"0":"")+dayval+"num.txt"));  
  
            for(int i=0; i<tf.length; i++) {  
                tf[i] = new JTextField(br.readLine(),5);  
                tf[i].addActionListener(new MyActionListener());  
                tf[i].addKeyListener(new NumberField());  
                add(new JLabel(itemName[i])); // 메뉴 삽입  
                add(tf[i]);  
            }  
            br.close();  
        }catch (FileNotFoundException e)  
        {  
            for(int i=0; i<tf.length; i++) {  
                tf[i] = new JTextField("0", 5);  
                tf[i].addActionListener(new MyActionListener());  
                tf[i].addKeyListener(new NumberField());  
                add(new JLabel(itemName[i])); // 메뉴 삽입  
                add(tf[i]);  
            }  
            e.printStackTrace();  
        }  
        catch (IOException e)  
        {  
            e.printStackTrace();  
        }  
    }  
}
```

# 클래스 및 메소드 설계

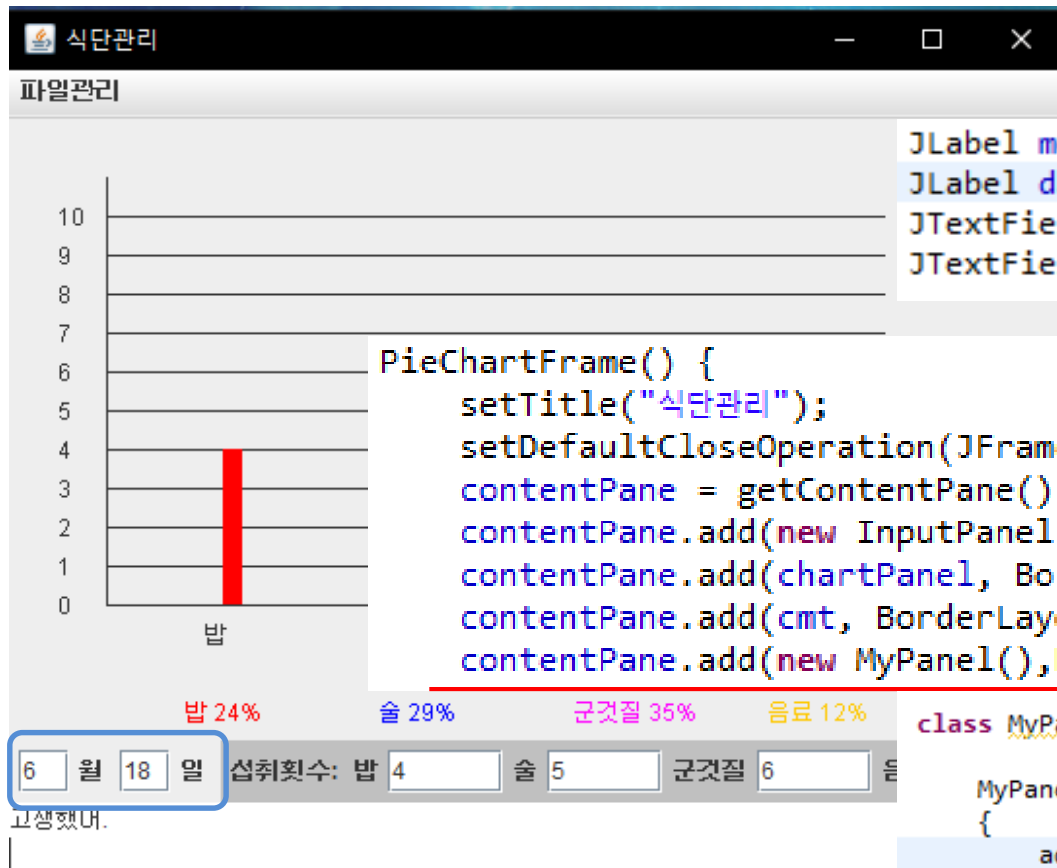


```
public class PieChartFrame extends JFrame {
    Container contentPane;
    int [] data = {0,0,0,0};
    int [] arcAngle = new int [4];
    Color [] color = {Color.RED, Color.BLUE, Color.MAGENTA, Color.ORANGE};
    String [] itemName = {"밥", "술", "군것질", "음료"}; // 식단 목록
    JTextField [] tf = new JTextField [4]; // 섭취횟수
    ChartPanel chartPanel = new ChartPanel();

    JTextArea cmt = new JTextArea(10,10); // 코멘트

    PieChartFrame() {
        setTitle("식단관리");
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        contentPane = getContentPane();
        contentPane.add(new InputPanel(), BorderLayout.CENTER);
        contentPane.add(chartPanel, BorderLayout.NORTH);
        contentPane.add(cmt, BorderLayout.SOUTH);
        contentPane.add(new MyPanel(), BorderLayout.WEST);
    }
}
```

# 클래스 및 메소드 설계



```
JLabel month1 = new JLabel("월 ");
JLabel day1 = new JLabel("일 ");
JTextField days0 = new JTextField(calMonth1, 2);
JTextField days1 = new JTextField(calday1, 2);
```

```
class MyPanel extends JPanel{// 서쪽 월 일 표시
```

```
MyPanel()
```

```
{
```

```
    add(days0);
```

```
    add(month1);
```

```
    add(days1);
```

```
    add(day1);
```

```
    days0.addKeyListener(new NumberField());
```

```
    days1.addKeyListener(new NumberField());
```

```
    days0.addActionListener(new enterField());
```

```
    days1.addActionListener(new enterField());
```

```
}
```

```
}
```

## 개발 일정

기간		세부내용
1일차	6/8	기능 및 디자인 자료 수집
2일차	6/9	클래스 및 메소드 설계
3일차	6/15~16	이벤트 처리 설계
4일차	6/17~18	디버깅
5일차	6/19	프로젝트 최종 발표

## 명품 JAVA Essential

10장 Open Challenge(p.424)

11장 Open Challenge(p.457)

[http://castleone88.blogspot.kr/2016/02/javatextarea\\_12.html](http://castleone88.blogspot.kr/2016/02/javatextarea_12.html)

<http://blog.naver.com/azure0777>

<http://javablue.tistory.com/entry/숫자만-입력받는-JTextField-구현>  
[\[Java Blue\]](#)

<https://wikidocs.net/227>

<http://blog.naver.com/highkrs/220462036333>



**감사합니다.**