

```

import java.awt.*;
import java.awt.event.*;
import javax.swing.*;

public class Problem1 extends JFrame implements ActionListener {
    JLabel lbl;
    JButton btn1, btn2, btn3;
    JPanel panel;

    Problem1() {
        setTitle("惑星名");
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        setSize(300, 300);
        setLocationRelativeTo(null);
        setLayout(new BorderLayout());
        panel = new JPanel();
        lbl = new JLabel();
        btn1 = new JButton("水星");
        btn2 = new JButton("火星");
        btn3 = new JButton("土星");
        btn1.addActionListener(this);
        btn2.addActionListener(this);
        btn3.addActionListener(this);
        panel.add(btn1);
        panel.add(btn2);
        panel.add(btn3);
        Container cont = getContentPane();
        cont.add(lbl, BorderLayout.CENTER);
        cont.add(panel, BorderLayout.SOUTH);
        lbl.setFont(new Font("MS ゴシック", Font.PLAIN, 32));
        lbl.setHorizontalAlignment(JLabel.CENTER);
    }

    public static void main(String[] args) {
        Problem1 p = new Problem1();
        p.setVisible(true);
    }

    public void actionPerformed(ActionEvent e) {
        if(e.getActionCommand().equals("水星"))
            lbl.setText("Mercury");
        else if(e.getActionCommand().equals("火星"))
            lbl.setText("Mars");
        else if(e.getActionCommand().equals("土星"))
            lbl.setText("Saturn");
    }
}

```

```
import java.awt.*;
import java.awt.event.*;
import javax.swing.*;

public class Problem2 extends JFrame implements ActionListener {
    JTextField txt;
    JLabel lbl;
    JButton btn;
    JPanel panell1, panel2;

    Problem2() {
        setTitle("入力文字列の表示");
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        setSize(300, 300);
        setLocationRelativeTo(null);
        setLayout(new BorderLayout());
        panell1 = new JPanel();
        panel2 = new JPanel();
        txt = new JTextField(20);
        lbl = new JLabel();
        btn = new JButton("表示");
        btn.addActionListener(this);
        panell1.add(txt);
        panel2.add(btn);
        Container cont = getContentPane();
        cont.add(panell1, BorderLayout.NORTH);
        cont.add(lbl, BorderLayout.CENTER);
        cont.add(panel2, BorderLayout.SOUTH);
        lbl.setFont(new Font("MS ゴシック", Font.PLAIN, 16));
        lbl.setHorizontalAlignment(JLabel.CENTER);
    }

    public static void main(String[] args) {
        Problem2 p = new Problem2();
        p.setVisible(true);
    }

    public void actionPerformed(ActionEvent e) {
        lbl.setText(txt.getText());
    }
}
```

```

import java.awt.*;
import java.awt.event.*;
import javax.swing.*;

public class Problem3 extends JFrame implements ActionListener {
    JTextField txt1, txt2;
    JLabel lbl1, lbl2, lbl3;
    JButton btn;
    JPanel panel1, panel2, panel3, panel4;

    Problem3() {
        setTitle("お菓子の値段");
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        setSize(300, 300);
        setLocationRelativeTo(null);
        setLayout(new FlowLayout());

        panel1 = new JPanel();
        panel2 = new JPanel();
        panel3 = new JPanel();

        panel1.setPreferredSize(new Dimension(280, 30));
        panel2.setPreferredSize(new Dimension(280, 30));
        panel3.setPreferredSize(new Dimension(280, 100));
        lbl1 = new JLabel("プリン ");
        txt1 = new JTextField(8);
        lbl2 = new JLabel("ワッフル");
        txt2 = new JTextField(8);
        lbl3 = new JLabel();
        lbl3.setPreferredSize(new Dimension(280, 120));
        lbl3.setFont(new Font("MS ゴシック", Font.PLAIN, 32));
        lbl3.setHorizontalAlignment(JLabel.CENTER);
        btn = new JButton("計算");
        btn.addActionListener(this);
        panel1.add(lbl1);
        panel1.add(txt1);
        panel2.add(lbl2);
        panel2.add(txt2);
        panel3.add(btn);
        Container cont = getContentPane();
        cont.add(panel1);
        cont.add(panel2);
        cont.add(lbl3);
        cont.add(panel3);
    }

    public static void main(String[] args) {
        Problem3 p = new Problem3();
        p.setVisible(true);
    }

    public void actionPerformed(ActionEvent e) {
        int pudding = Integer.parseInt(txt1.getText());
        int wafful = Integer.parseInt(txt2.getText());
        int total = 220 * pudding + 140 * wafful;
        lbl3.setText(String.valueOf(total) + "円");
    }
}

```

```

import java.awt.*;
import java.awt.event.*;
import javax.swing.*;

public class Problem4 extends JFrame implements ActionListener {
    JTextField txt;
    JLabel lbl1, lbl2, lbl3;
    JButton btn;
    JPanel panell1, panel2;

    Problem4() {
        setTitle("小児運賃");
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        setSize(300, 300);
        setLocationRelativeTo(null);
        setLayout(new BorderLayout());
        panell1 = new JPanel();
        panel2 = new JPanel();
        lbl1 = new JLabel("普通運賃");
        lbl2 = new JLabel("円");
        lbl3 = new JLabel();
        txt = new JTextField(6);
        btn = new JButton("計算");
        btn.addActionListener(this);
        panell1.add(lbl1);
        panell1.add(txt);
        panell1.add(lbl2);
        panel2.add(btn);
        Container cont = getContentPane();
        cont.add(panell1, BorderLayout.NORTH);
        cont.add(lbl3, BorderLayout.CENTER);
        cont.add(panel2, BorderLayout.SOUTH);
        lbl3.setFont(new Font("MS ゴシック", Font.PLAIN, 32));
        lbl3.setHorizontalAlignment(JLabel.CENTER);
    }

    public static void main(String[] args) {
        Problem4 p = new Problem4();
        p.setVisible(true);
    }

    public void actionPerformed(ActionEvent e) {
        int fee = Integer.parseInt(txt.getText());
        int child = (int)Math.ceil(fee / 2.0 / 10) * 10;
        lbl3.setText(Integer.toString(child) + "円");
    }
}

```

```

import java.awt.*;
import java.awt.event.*;
import javax.swing.*;

public class Problem5 extends JFrame implements ActionListener {
    JTextField txt;
    JLabel lbl1, lbl2, lbl3;
    JButton btn;
    JPanel panel1, panel2;

    Problem5() {
        setTitle("テストの点数");
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        setSize(300, 300);
        setLocationRelativeTo(null);
        setLayout(new BorderLayout());
        panel1 = new JPanel();
        panel2 = new JPanel();
        lbl1 = new JLabel("点数");
        lbl2 = new JLabel("点");
        lbl3 = new JLabel();
        txt = new JTextField(6);
        btn = new JButton("判定");
        btn.addActionListener(this);
        panel1.add(lbl1);
        panel1.add(txt);
        panel1.add(lbl2);
        panel2.add(btn);
        Container cont = getContentPane();
        cont.add(panel1, BorderLayout.NORTH);
        cont.add(lbl3, BorderLayout.CENTER);
        cont.add(panel2, BorderLayout.SOUTH);
        lbl3.setFont(new Font("MS ゴシック", Font.PLAIN, 32));
        lbl3.setHorizontalAlignment(JLabel.CENTER);
    }

    public static void main(String[] args) {
        Problem5 p = new Problem5();
        p.setVisible(true);
    }

    public void actionPerformed(ActionEvent e) {
        int point = Integer.parseInt(txt.getText());
        if(point >= 60)
            lbl3.setText("合格です");
        else
            lbl3.setText("不合格です");
    }
}

```

```

import java.awt.*;
import java.awt.event.*;
import javax.swing.*;

public class Problem6 extends JFrame implements ActionListener {
    JTextField txt;
    JLabel lbl1, lbl2, lbl3;
    JButton btn;
    JPanel panell1, panel2;

    Problem6() {
        setTitle("世紀の表示");
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        setSize(300, 300);
        setLocationRelativeTo(null);
        setLayout(new BorderLayout());
        panell1 = new JPanel();
        panel2 = new JPanel();
        lbl1 = new JLabel("西暦");
        lbl2 = new JLabel("年");
        lbl3 = new JLabel();
        txt = new JTextField(6);
        btn = new JButton("表示");
        btn.addActionListener(this);
        panell1.add(lbl1);
        panell1.add(txt);
        panell1.add(lbl2);
        panel2.add(btn);
        Container cont = getContentPane();
        cont.add(panell1, BorderLayout.NORTH);
        cont.add(lbl3, BorderLayout.CENTER);
        cont.add(panel2, BorderLayout.SOUTH);
        lbl3.setFont(new Font("MS ゴシック", Font.PLAIN, 32));
        lbl3.setHorizontalAlignment(JLabel.CENTER);
    }

    public static void main(String[] args) {
        Problem6 p = new Problem6();
        p.setVisible(true);
    }

    public void actionPerformed(ActionEvent e) {
        int year = Integer.parseInt(txt.getText());
        if(year == 0)
            lbl2.setText("西暦年はありません");
        else {
            int century = (Math.abs(year) - 1)/100 + 1;
            if(year < 0)
                lbl3.setText("紀元前" + century + "世紀です");
            else
                lbl3.setText(century + "世紀です");
        }
    }
}

```

```

import java.awt.*;
import java.awt.event.*;
import javax.swing.*;

public class Problem7 extends JFrame implements ActionListener {
    JTextField txt;
    JLabel lbl1, lbl2, lbl3;
    JButton btn;
    JPanel panell1, panel2;

    Problem7() {
        setTitle("うるう年の判定");
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        setSize(300, 300);
        setLocationRelativeTo(null);
        setLayout(new BorderLayout());
        panell1 = new JPanel();
        panel2 = new JPanel();
        lbl1 = new JLabel("西暦");
        lbl2 = new JLabel("年");
        lbl3 = new JLabel();
        txt = new JTextField(6);
        btn = new JButton("判定");
        btn.addActionListener(this);
        panell1.add(lbl1);
        panell1.add(txt);
        panell1.add(lbl2);
        panel2.add(btn);
        Container cont = getContentPane();
        cont.add(panell1, BorderLayout.NORTH);
        cont.add(lbl3, BorderLayout.CENTER);
        cont.add(panel2, BorderLayout.SOUTH);
        lbl3.setFont(new Font("MS ゴシック", Font.PLAIN, 24));
        lbl3.setHorizontalAlignment(JLabel.CENTER);
    }

    public static void main(String[] args) {
        Problem7 p = new Problem7();
        p.setVisible(true);
    }

    public void actionPerformed(ActionEvent e) {
        int year = Integer.parseInt(txt.getText());
        if(year % 4 == 0 && year % 100 != 0 || year % 400 == 0)
            lbl3.setText("うるう年です");
        else
            lbl3.setText("うるう年ではありません");
    }
}

```

```

import java.awt.*;
import java.awt.event.*;
import javax.swing.*;

public class Problem8 extends JFrame implements ActionListener {
    JTextField txt;
    JLabel lbl1, lbl2;
    JButton btn;
    JPanel panell1, panel2;

    Problem8() {
        setTitle("月と季節");
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        setSize(300, 300);
        setLocationRelativeTo(null);
        setLayout(new BorderLayout());
        panell1 = new JPanel();
        panel2 = new JPanel();
        lbl1 = new JLabel("月");
        lbl2 = new JLabel();
        txt = new JTextField(4);
        btn = new JButton("判定");
        btn.addActionListener(this);
        panell1.add(txt);
        panell1.add(lbl1);
        panel2.add(btn);
        Container cont = getContentPane();
        cont.add(panell1, BorderLayout.NORTH);
        cont.add(lbl2, BorderLayout.CENTER);
        cont.add(panel2, BorderLayout.SOUTH);
        lbl2.setFont(new Font("MS ゴシック", Font.PLAIN, 32));
        lbl2.setHorizontalAlignment(JLabel.CENTER);
    }

    public static void main(String[] args) {
        Problem8 p = new Problem8();
        p.setVisible(true);
    }

    public void actionPerformed(ActionEvent e) {
        int month = Integer.parseInt(txt.getText());
        switch(month) {
            case 12:
            case 1:
            case 2:
                lbl2.setText("冬です");
                break;
            case 3:
            case 4:
            case 5:
                lbl2.setText("春です");
                break;
            case 6:
            case 7:
            case 8:
                lbl2.setText("夏です");
                break;
            case 9:
            case 10:
            case 11:
                lbl2.setText("秋です");
            default :
                lbl2.setFont(new Font("MS ゴシック", Font.PLAIN, 20));
                lbl2.setText("1~12の月を入力してください");
        }
    }
}

```

```

import java.awt.*;
import java.awt.event.*;
import javax.swing.*;

public class Problem9 extends JFrame implements ActionListener {
    JCheckBox coffee, cake, lunch;
    JLabel lbl;
    JButton btn;
    JPanel panell1, panel2;

    Problem9() {
        setTitle("ランチセット");
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        setSize(300, 300);
        setLocationRelativeTo(null);
        setLayout(new FlowLayout());
        panell1 = new JPanel();
        panel2 = new JPanel();
        coffee = new JCheckBox("コーヒー");
        cake = new JCheckBox("ケーキ");
        lunch = new JCheckBox("ランチ");
        lbl = new JLabel();
        lbl.setPreferredSize(new Dimension(280, 160));
        lbl.setFont(new Font("MS ゴシック", Font.PLAIN, 24));
        lbl.setHorizontalAlignment(JLabel.CENTER);
        btn = new JButton("計算");
        btn.addActionListener(this);
        panell1.add(coffee);
        panell1.add(cake);
        panell1.add(lunch);
        panel2.add(btn);
        Container cont = getContentPane();
        cont.add(panell1);
        cont.add(lbl);
        cont.add(panel2);
    }

    public static void main(String[] args) {
        Problem9 p = new Problem9();
        p.setVisible(true);
    }

    public void actionPerformed(ActionEvent e) {
        int price = 0;
        if(cake.isSelected()==true && coffee.isSelected()==false)
            price = 500;
        if(coffee.isSelected()) {
            if(cake.isSelected())
                price = 650;
            else
                price = 300;
        }
        if(lunch.isSelected()) {
            if(price > 0)
                price += 650;
            else
                price = 850;
        }
        lbl.setText("お会計は" + price + "円です");
    }
}

```

```

import java.awt.*;
import java.awt.event.*;
import javax.swing.*;

public class Problem10 extends JFrame implements ActionListener {
    JRadioButton rb1, rb2, rb3;
    ButtonGroup grp;
    JLabel lbl;
    JButton btn;
    JPanel panell1, panel2;

    Problem10() {
        setTitle("惑星の名前");
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        setSize(300, 300);
        setLocationRelativeTo(null);
        setLayout(new FlowLayout());
        panell1 = new JPanel();
        panel2 = new JPanel();
        rb1 = new JRadioButton("水星");
        rb2 = new JRadioButton("火星");
        rb3 = new JRadioButton("土星");
        grp = new ButtonGroup();
        grp.add(rb1);
        grp.add(rb2);
        grp.add(rb3);
        lbl = new JLabel();
        lbl.setPreferredSize(new Dimension(280, 160));
        lbl.setFont(new Font("MS ゴシック", Font.PLAIN, 32));
        lbl.setHorizontalAlignment(JLabel.CENTER);
        btn = new JButton("表示");
        btn.addActionListener(this);
        panell1.add(rb1);
        panell1.add(rb2);
        panell1.add(rb3);
        panel2.add(btn);
        Container cont = getContentPane();
        cont.add(panell1);
        cont.add(lbl);
        cont.add(panel2);
    }

    public static void main(String[] args) {
        Problem10 p = new Problem10();
        p.setVisible(true);
    }

    public void actionPerformed(ActionEvent e) {
        if(rb1.isSelected())
            lbl.setText("Mercury");
        else if(rb2.isSelected())
            lbl.setText("Mars");
        else if(rb3.isSelected())
            lbl.setText("Saturn");
    }
}

```

```

import java.awt.*;
import java.awt.event.*;
import javax.swing.*;

public class Problem11 extends JFrame implements ActionListener {
    JRadioButton rb11, rb12, rb13, rb21, rb22, rb23;
    ButtonGroup grp1, grp2;
    JLabel lbl;
    JButton btn;
    JPanel panel1, panel2, panel3;

    Problem11() {
        setTitle("色の名前");
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        setSize(300, 300);
        setLocationRelativeTo(null);
        setLayout(new FlowLayout());
        panel1 = new JPanel();
        panel2 = new JPanel();
        panel3 = new JPanel();
        rb11 = new JRadioButton("白");
        rb12 = new JRadioButton("赤");
        rb13 = new JRadioButton("緑");
        rb21 = new JRadioButton("英語");
        rb22 = new JRadioButton("フランス語");
        rb23 = new JRadioButton("スペイン語");
        grp1 = new ButtonGroup();
        grp2 = new ButtonGroup();
        grp1.add(rb11);
        grp1.add(rb12);
        grp1.add(rb13);
        grp2.add(rb21);
        grp2.add(rb22);
        grp2.add(rb23);
        lbl = new JLabel();
        lbl.setPreferredSize(new Dimension(280, 120));
        lbl.setFont(new Font("MS ゴシック", Font.PLAIN, 32));
        lbl.setHorizontalAlignment(JLabel.CENTER);
        btn = new JButton("表示");
        btn.addActionListener(this);
        panel1.add(rb11);
        panel1.add(rb12);
        panel1.add(rb13);
        panel2.add(rb21);
        panel2.add(rb22);
        panel2.add(rb23);
        panel3.add(btn);
        Container cont = getContentPane();
        cont.add(panel1);
        cont.add(panel2);
        cont.add(lbl);
        cont.add(panel3);
    }

    public static void main(String[] args) {
        Problem11 p = new Problem11();
        p.setVisible(true);
    }
}

```

```
public void actionPerformed(ActionEvent e) {
    if(rb11.isSelected()) {
        if(rb21.isSelected())
            lbl.setText("white");
        else if(rb22.isSelected())
            lbl.setText("blanc");
        else if(rb23.isSelected())
            lbl.setText("blanco");
    } else if(rb12.isSelected()) {
        if(rb21.isSelected())
            lbl.setText("red");
        else if(rb22.isSelected())
            lbl.setText("rouge");
        else if(rb23.isSelected())
            lbl.setText("rojo");
    } else if(rb13.isSelected()) {
        if(rb21.isSelected())
            lbl.setText("green");
        else if(rb22.isSelected())
            lbl.setText("vert");
        else if(rb23.isSelected())
            lbl.setText("verde");
    }
}
}
```

```

import java.awt.*;
import java.awt.event.*;
import javax.swing.*;

public class Problem12 extends JFrame implements ActionListener {
    JTextField txt;
    JLabel lbl1, lbl2;
    JButton btn;
    JPanel panell1, panel2;

    Problem12() {
        setTitle("テストの点数");
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        setSize(300, 300);
        setLocationRelativeTo(null);
        setLayout(new BorderLayout());
        panell1 = new JPanel();
        panel2 = new JPanel();
        lbl1 = new JLabel("点数");
        lbl2 = new JLabel("点");
        txt = new JTextField(6);
        btn = new JButton("判定");
        btn.addActionListener(this);
        panell1.add(lbl1);
        panell1.add(txt);
        panell1.add(lbl2);
        panel2.add(btn);
        Container cont = getContentPane();
        cont.add(panell1, BorderLayout.NORTH);
        cont.add(panel2, BorderLayout.SOUTH);
    }

    public static void main(String[] args) {
        Problem12 p = new Problem12();
        p.setVisible(true);
    }

    public void actionPerformed(ActionEvent e) {
        int point = Integer.parseInt(txt.getText());
        int result;
        if(point >= 70)
            JOptionPane.showConfirmDialog(null, "合格です", "テストの点数",
                JOptionPane.DEFAULT_OPTION, JOptionPane.PLAIN_MESSAGE);
        else if(point >= 50) {
            result = JOptionPane.showConfirmDialog(null, "レポートを提出しましたか",
                "テストの点数", JOptionPane.YES_NO_OPTION,
                JOptionPane.PLAIN_MESSAGE);
            if(result == JOptionPane.YES_OPTION)
                JOptionPane.showConfirmDialog(null, "合格です", "テストの点数",
                    JOptionPane.DEFAULT_OPTION, JOptionPane.PLAIN_MESSAGE);
            else
                JOptionPane.showConfirmDialog(null, "不合格です", "テストの点数",
                    JOptionPane.DEFAULT_OPTION, JOptionPane.PLAIN_MESSAGE);
        } else
            JOptionPane.showConfirmDialog(null, "不合格です", "テストの点数",
                JOptionPane.DEFAULT_OPTION, JOptionPane.PLAIN_MESSAGE);
    }
}

```

```

import java.awt.*;
import java.awt.event.*;
import javax.swing.*;

public class Problem13 extends JFrame implements ActionListener {
    JButton btn1, btn2, btn3;
    JPanel panel;
    String shape = "";

    Problem13() {
        setTitle("図形描画");
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        setSize(300, 300);
        setLocationRelativeTo(null);
        setLayout(new BorderLayout());
        panel = new JPanel();
        btn1 = new JButton("正方形");
        btn2 = new JButton(" 円 ");
        btn3 = new JButton("クリア");
        btn1.addActionListener(this);
        btn2.addActionListener(this);
        btn3.addActionListener(this);
        panel.add(btn1);
        panel.add(btn2);
        panel.add(btn3);
        Container cont = getContentPane();
        cont.add(panel, BorderLayout.NORTH);
    }

    public static void main(String[] args) {
        Problem13 p = new Problem13();
        p.setVisible(true);
    }

    public void actionPerformed(ActionEvent e) {
        shape = e.getActionCommand();
        repaint();
    }

    public void paint(Graphics g) {
        super.paint(g);
        if(shape.equals("正方形"))
            g.drawRect(100,100,100,100);
        else if(shape.equals(" 円 "))
            g.drawOval(100,100,100,100);
    }
}

```

```

import java.awt.*;
import java.awt.event.*;
import javax.swing.*;

public class Problem14 extends JFrame implements ActionListener {
    JButton btn1, btn2;
    JLabel lbl;
    JPanel panel;
    String shape = "";

    Problem14() {
        setTitle("画像表示");
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        setSize(300, 300);
        setLocationRelativeTo(null);
        setLayout(new BorderLayout());
        panel = new JPanel();
        btn1 = new JButton("表示");
        btn2 = new JButton("消去");
        btn1.addActionListener(this);
        btn2.addActionListener(this);
        lbl = new JLabel(new ImageIcon("welcome.jpg"));
        panel.add(btn1);
        panel.add(btn2);
        Container cont = getContentPane();
        cont.add(panel, BorderLayout.NORTH);
        cont.add(lbl, BorderLayout.CENTER);
        lbl.setVisible(false);
    }

    public static void main(String[] args) {
        Problem14 p = new Problem14();
        p.setVisible(true);
    }

    public void actionPerformed(ActionEvent e) {
        if(e.getActionCommand().equals("表示"))
            lbl.setVisible(true);
        else if(e.getActionCommand().equals("消去"))
            lbl.setVisible(false);
    }
}

```

```

import java.awt.*;
import java.awt.event.*;
import javax.swing.*;

public class Problem15 extends JFrame implements MouseListener {
    JLabel lbl;

    Problem15() {
        setTitle("クリック座標");
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        setSize(300, 300);
        setLocationRelativeTo(null);
        setLayout(new BorderLayout());
        lbl = new JLabel();
        lbl.setFont(new Font("MS ゴシック", Font.PLAIN, 32));
        lbl.setHorizontalAlignment(JLabel.CENTER);
        Container cont = getContentPane();
        cont.add(lbl, BorderLayout.CENTER);
        addMouseListener(this);
    }

    public static void main(String[] args) {
        Problem15 p = new Problem15();
        p.setVisible(true);
    }

    public void mouseClicked(MouseEvent e) {
        if(e.getY() > 150)
            lbl.setForeground(Color.red);
        else
            lbl.setForeground(Color.black);
        lbl.setText("x : " + e.getX() + ", y : " + e.getY());
    }

    public void mouseEntered(MouseEvent e) { }

    public void mouseExited(MouseEvent e) { }

    public void mousePressed(MouseEvent e) { }

    public void mouseReleased(MouseEvent e) { }
}

```

```

import java.awt.*;
import java.awt.event.*;
import javax.swing.*;

public class Problem16 extends JFrame implements ActionListener {
    JLabel lbl;
    JButton btn1, btn2, btn3;
    JPanel panel;
    Timer t;
    int sec = 0;

    Problem16() {
        setTitle("タイマー");
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        setSize(300, 300);
        setLocationRelativeTo(null);
        setLayout(new BorderLayout());
        panel = new JPanel();
        lbl = new JLabel();
        btn1 = new JButton("スタート");
        btn2 = new JButton("ストップ");
        btn3 = new JButton("リセット");
        btn1.addActionListener(this);
        btn2.addActionListener(this);
        btn3.addActionListener(this);
        panel.add(btn1);
        panel.add(btn2);
        panel.add(btn3);
        Container cont = getContentPane();
        cont.add(panel, BorderLayout.NORTH);
        lbl.setFont(new Font("Arial", Font.BOLD, 128));
        lbl.setHorizontalAlignment(JLabel.CENTER);
        cont.add(lbl, BorderLayout.CENTER);
        lbl.setText(Integer.toString(sec));
        t = new Timer(1000, this);
        t.setActionCommand("タイマー");
    }

    public static void main(String[] args) {
        Problem16 p = new Problem16();
        p.setVisible(true);
    }

    public void actionPerformed(ActionEvent e) {
        if(e.getActionCommand().equals("スタート"))
            t.start();
        else if(e.getActionCommand().equals("ストップ"))
            t.stop();
        else if(e.getActionCommand().equals("リセット"))
            sec = 0;
        else if(e.getActionCommand().equals("タイマー"))
            sec++;
        lbl.setText(Integer.toString(sec));
    }
}

```

```

import java.awt.*;
import java.awt.event.*;
import javax.swing.*;

public class Problem17 extends JFrame implements ActionListener {
    JComboBox cmbyear, cmbmonth;
    JButton btn;
    JLabel lbl, lblyear, lblmonth;
    JPanel panel;
    String[] year = { " ", "2009", "2010", "2011" };
    String[] month = { " ", " 1", " 2", " 3", " 4", " 5", " 6", " 7", " 8", " 9",
        "10", "11", "12" };

    Problem17() {
        setTitle("カレンダー");
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        setSize(450, 400);
        setLocationRelativeTo(null);
        setLayout(new BorderLayout());
        panel = new JPanel();
        cmbyear = new JComboBox(year);
        cmbmonth = new JComboBox(month);
        btn = new JButton("表示");
        btn.addActionListener(this);
        lblyear = new JLabel("年");
        lblmonth = new JLabel("月");
        lbl = new JLabel();
        panel.add(cmbyear);
        panel.add(lblyear);
        panel.add(cmbmonth);
        panel.add(lblmonth);
        panel.add(btn);
        Container cont = getContentPane();
        cont.add(panel, BorderLayout.NORTH);
        cont.add(lbl, BorderLayout.CENTER);
        lbl.setFont(new Font("MS ゴシック", Font.BOLD, 24));
        lbl.setHorizontalAlignment(JLabel.CENTER);
    }

    public static void main(String[] args) {
        Problem17 p = new Problem17();
        p.setVisible(true);
    }

    public void actionPerformed(ActionEvent e) {
        int[] days = { 0, 31, 28, 31, 30, 31, 30, 31, 31, 30, 31, 30, 31 };
        int year = Integer.parseInt(((String)cmbyear.getSelectedItem());
        int month =
            Integer.parseInt(((String)cmbmonth.getSelectedItem()).trim());
        if(year % 4 == 0 && year % 100 != 0 || year % 400 == 0)
            days[2] = 29;
        int bias = year + (year - 1)/4 - (year - 1)/100 + (year - 1)/400;
        int s = 0;
        for(int i = 1; i < month; i++)
            s += days[i];
        int week = (bias + s) % 7;
        StringBuffer cal = new StringBuffer("<html><pre> <font color='red'>
            Sun</font> Mon Tue Wed Thu Fri <font color='blue'>Sat</font><br>");
        for(int i = 1; i <= week; i++)
            cal.append(" ");
        for(int i = 1; i <= days[month]; i++) {
            cal.append((i < 10 ? " " + i : " " + i));
            if((week + i) % 7 == 0)
                cal.append("<br>");
        }
        cal.append("</pre></html>");
        lbl.setText(cal.toString());
    }
}

```

```

import java.awt.*;
import javax.swing.*;

public class Problem18 extends JFrame {

    static final int WIDTH = 500;
    static final int HEIGHT = 300;
    static final int OBJECTS = 1;
    MyThread[] mythread = new MyThread[OBJECTS];
    Circle[] circle = new Circle[OBJECTS];

    Problem18() {
        int radius = 15;
        circle[0] = new Circle(radius, Color.red);
        circle[0].setXY(0, HEIGHT / 2);
        circle[0].setMove(8);
        circle[0].setSpeed(60);
        circle[0].setCage(WIDTH, HEIGHT);

        for(int i = 0; i < OBJECTS; i++) {
            mythread[i] = new MyThread(i, circle[i].getSleeptime());
            mythread[i].start();
        }

        setTitle("円の移動");
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        setSize(WIDTH , HEIGHT);
        setLocationRelativeTo(null);
    }

    public static void main(String[] args) {
        Problem18 p = new Problem18();
        p.setVisible(true);
    }

    public void paint(Graphics g) {
        super.paint(g);
        for(int i = 0; i < OBJECTS; i++) {
            circle[i].paint(g);
        }
    }

    class MyThread extends Thread {

        int index, sleeptime;

        MyThread(int index, int sleeptime) {
            this.index = index;
            this.sleeptime = sleeptime;
        }

        public void run() {
            Thread thisThread = Thread.currentThread();
            while(mythread[index] == thisThread) {
                circle[index].nextPosition();
                repaint();
                try {
                    this.sleep(sleeptime);
                } catch (InterruptedException e) { }
            }
        }
    }
}

```

```

class Circle {

    int width, height;
    int radius;
    int x, y;
    int dx;
    int speed;
    Color color;

    Circle(int radius, Color color) {
        this.radius = radius;
        this.color = color;
    }

    public void setXY(int x, int y) {
        this.x = x;
        this.y = y;
    }

    public void setMove(int dx) {
        this.dx = dx;
    }

    public void setCage(int width, int height) {
        this.width = width;
        this.height = height;
    }

    public void setSpeed(int speed) {
        this.speed = speed;
    }

    public int getSleeptime() {
        return 100 > speed ? 100 - speed : 10;
    }

    public void paint(Graphics g) {
        g.setColor(color);
        g.fillOval(x - radius, y - radius, 2 * radius, 2 * radius);
    }

    public void nextPosition() {
        setXY(x + dx, y);
        if(x > width + radius)
            setXY(0, y);
    }
}

```

```

import java.awt.*;
import javax.swing.*;

public class Problem19 extends JFrame {

    static final int WIDTH = 300;
    static final int HEIGHT = 500;
    static final int OBJECTS = 1;
    MyThread[] mythread = new MyThread[OBJECTS];
    Circle[] circle = new Circle[OBJECTS];

    Problem19() {
        int radius = 15;
        circle[0] = new Circle(radius, Color.red);
        circle[0].setXY(WIDTH / 2, HEIGHT / 2);
        circle[0].setMove(8);
        circle[0].setSpeed(60);
        circle[0].setCage(WIDTH, HEIGHT);

        for(int i = 0; i < OBJECTS; i++) {
            mythread[i] = new MyThread(i, circle[i].getSleeptime());
            mythread[i].start();
        }

        setTitle("球の落下");
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        setSize(WIDTH, HEIGHT);
        setLocationRelativeTo(null);
    }

    public static void main(String[] args) {
        Problem19 p = new Problem19();
        p.setVisible(true);
    }

    public void paint(Graphics g) {
        super.paint(g);
        for(int i = 0; i < OBJECTS; i++) {
            circle[i].paint(g);
        }
    }

    class MyThread extends Thread {

        int index, sleeptime;

        MyThread(int index, int sleeptime) {
            this.index = index;
            this.sleeptime = sleeptime;
        }

        public void run() {
            Thread thisThread = Thread.currentThread();
            while(mythread[index] == thisThread) {
                circle[index].nextPosition();
                repaint();
                try {
                    this.sleep(sleeptime);
                } catch (InterruptedException e) { }
            }
        }
    }
}

```

```

class Circle {

    int width, height;
    int radius;
    int x, y;
    int dy;
    int speed;
    Color color;

    Circle(int radius, Color color) {
        this.radius = radius;
        this.color = color;
    }

    public void setXY(int x, int y) {
        this.x = x;
        this.y = y;
    }

    public void setMove(int dy) {
        this.dy = dy;
    }

    public void setCage(int width, int height) {
        this.width = width;
        this.height = height;
    }

    public void setSpeed(int speed) {
        this.speed = speed;
    }

    public int getSleeptime() {
        return 100 > speed ? 100 - speed : 10;
    }

    public void paint(Graphics g) {
        g.setColor(color);
        g.fillOval(x - radius, y - radius, 2 * radius, 2 * radius);
    }

    public void nextPosition() {
        setXY(x, y + dy);
        if(y < height * 2 / 10 || y > height - (radius + 4))
            dy = -dy;
    }
}

```

```

import java.awt.*;
import javax.swing.*;

public class Problem20 extends JFrame {

    static final int WIDTH = 500;
    static final int HEIGHT = 300;
    static final int OBJECTS = 3;
    MyThread[] mythread = new MyThread[OBJECTS];
    Circle[] circle = new Circle[OBJECTS];

    Problem20() {
        int radius = 15;
        circle[0] = new Circle(radius, Color.red);
        circle[0].setXY(100, 100);
        circle[0].setDirection(-6, 8);
        circle[0].setSpeed(90);
        circle[0].setCage(WIDTH, HEIGHT);

        circle[1] = new Circle(radius, Color.blue);
        circle[1].setXY(300, 200);
        circle[1].setDirection(-8, 6);
        circle[1].setSpeed(70);
        circle[1].setCage(WIDTH, HEIGHT);

        circle[2] = new Circle(radius, Color.green);
        circle[2].setXY(400, 250);
        circle[2].setDirection(10, -3);
        circle[2].setSpeed(50);
        circle[2].setCage(WIDTH, HEIGHT);

        for(int i = 0; i < OBJECTS; i++) {
            mythread[i] = new MyThread(i, circle[i].getSleeptime());
            mythread[i].start();
        }

        setTitle("球のクッション");
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        setSize(WIDTH, HEIGHT);
        setLocationRelativeTo(null);
    }

    public static void main(String[] args) {
        Problem20 p = new Problem20();
        p.setVisible(true);
    }

    public void paint(Graphics g) {
        super.paint(g);
        for(int i = 0; i < OBJECTS; i++) {
            circle[i].paint(g);
        }
    }
}

```

```

class MyThread extends Thread {

    int index, sleeptime;

    MyThread(int index, int sleeptime) {
        this.index = index;
        this.sleeptime = sleeptime;
    }

    public void run() {
        Thread thisThread = Thread.currentThread();
        while(mythread[index] == thisThread) {
            circle[index].nextPosition();
            repaint();
            try {
                this.sleep(sleeptime);
            } catch (InterruptedException e) { }
        }
    }
}

class Circle {
    int width, height;
    int radius;
    int x, y;
    int dx, dy;
    int speed;
    Color color;

    Circle(int radius, Color color) {
        this.radius = radius;
        this.color = color;
    }
    public void setXY(int x, int y) {
        this.x = x;
        this.y = y;
    }
    public void setDirection(int dx, int dy) {
        this.dx = dx;
        this.dy = dy;
    }
    public void setCage(int width, int height) {
        this.width = width;
        this.height = height;
    }
    public void setSpeed(int speed) {
        this.speed = speed;
    }
    public int getSleeptime() {
        return 100 > speed ? 100 - speed : 10;
    }
    public void paint(Graphics g) {
        g.setColor(color);
        g.fillOval(x - radius, y - radius, 2 * radius, 2 * radius);
    }
    public void nextPosition() {
        setXY(x + dx, y + dy);
        if(x < 4 + radius || x > width - (radius + 4))
            dx = -dx;
        if(y < 30 + radius || y > height - (radius + 4))
            dy = -dy;
    }
}

```