

Swing:

```
import javax.swing.*;
```

Utwórz applet, który zawiera etykietę (JLabel(„Mój applet”)) oraz obiekt klasy Wykres klasy dziedziczącej po klasie JPanel. W klasie Wykres przesłonięta jest metoda paintComponent: w losowo wybranym miejscu rysowany jest okrąg o średnicy 10 pikseli.

Dodatkowe informacje na temat rysowania w Swingu:

<https://docs.oracle.com/javase/tutorial/uiswing/painting/step2.html>

```
import java.applet.Applet;
import java.awt.*;
import javax.swing.*;
public class japplet_jpanel extends JApplet {
    int applet_width=500;
    int applet_height=400;
    Wykres wykr = new Wykres();
    public void init(){
        setSize(applet_width,applet_height);
        JLabel etykieta = new JLabel("Mój applet");
        setLayout(new FlowLayout());
        add(wykr);
        etykieta.setBorder(BorderFactory.createLineBorder(Color.orange));
        add(etykieta);
    }
}
class Wykres extends JPanel{
    int wielkosc=200;
    public Wykres(){setBorder(BorderFactory.createLineBorder(Color.blue));
    }

    public Dimension getPreferredSize()
    {return new Dimension(wielkosc,wielkosc);}

    public void paintComponent(Graphics graf){
        super.paintComponent(graf);
        graf.drawOval((int)(wielkosc*Math.random()),
(int)(wielkosc*Math.random()), 10, 10);
    }
}
```

Dodaj przycisk (JButton(„START")), po wciśnięciu którego odświeżany będzie obiekt klasy Wykres.

```
import java.applet.Applet;
import java.awt.*;
import java.awt.event.ActionEvent;
import java.awt.event.*;
import javax.swing.*;
public class japplet_jpanel_jbutton extends JApplet implements ActionListener {
    int applet_width=500;
    int applet_height=400;
```

```

Wykres wyk = new Wykres();
    public void init(){
        setSize(applet_width,applet_height);
        JLabel etykieta = new JLabel("Mój aplet");
        setLayout(new FlowLayout());
        add(wykr);
        etykieta.setBorder(BorderFactory.createLineBorder(Color.orange));
        JButton przycisk = new JButton("START");
        przycisk.addActionListener(this);
        przycisk.setBorder(BorderFactory.createLineBorder(Color.green));
        add(etykieta);
        add(przycisk);
    }
    public void actionPerformed(ActionEvent e){
        wyk.repaint();
    }
}

```

Odświeżyć chcemy obiekt wyk i na jego rzecz wywołujemy repaint() czyli wyk.repaint().

Dodaj animację: Na początku paintComponent nic nie rysuje. Dopiero wciśnięcie przycisku wywołuje rysowanie, które powtarza się cyklicznie (repaint()).

```

import java.applet.Applet;
import java.awt.*;
import java.awt.event.ActionEvent;
import java.awt.event.*;
import javax.swing.*;
public class japplet_animation extends JApplet implements ActionListener {
    int applet_width=500;
    int applet_height=400;
    Wykres1 wyk = new Wykres1();
    public void init(){
        setSize(applet_width,applet_height);
        JLabel etykieta = new JLabel("Mój aplet");
        setLayout(new FlowLayout());
        add(wykr);
        etykieta.setBorder(BorderFactory.createLineBorder(Color.orange));
        JButton przycisk = new JButton("START");
        przycisk.addActionListener(this);
        przycisk.setBorder(BorderFactory.createLineBorder(Color.green));
        add(etykieta);
        add(przycisk);
    }
    public void actionPerformed(ActionEvent e){
        wyk.repaint();
        wyk.rysuj=true;
    }
}
class Wykres1 extends JPanel{
    int wielkosc=200;
    boolean rysuj;
}

```

```

public Wykres1(){setBorder(BorderFactory.createLineBorder(Color.blue));
rysuj=false;
}

public Dimension getPreferredSize()
{return new Dimension(wielkosc,wielkosc);}

public void paintComponent(Graphics graf){
super.paintComponent(graf);
if(rysuj){
graf.drawOval((int)(wielkosc*Math.random()),
(int)(wielkosc*Math.random()), 10, 10);
try {
Thread.sleep(100); // sleep for 10 msec
} catch (InterruptedException t){}
repaint();
}
}
}

```

Przycisk „Start” zamień na „Start/Stop” (ponowne wciśnięcie zatrzymuje animację).
Dodaj drugi komplet Panel+Przycisk.