

МОНГОЛ УЛСЫН ИХ СУРГУУЛЬ

Хэрэглээний Шинжлэх Ухаан Инженерчлэлийн Сургууль

Док. Д.Энхзол, Маг. Б.Батням, Маг. Ц.Лхамролом

A1) Дараах Test01гэсэн класс дотор өөр нэгэн шинэ дотоод класс үүсгэ (ICSI402) дотоод класс гэж нэг классын дотор талд тодорхойлсон өөр нэгэн класс юм. Жишээ нь PrintRectangle). Тус класс нь Printable-ийг хэрэгжүүлсэн “Triangle” гэж хэвлэдэг функцтай байна.

```
public class Test01{
    public interface Printable{
        public void print();
    }
    public class PrintRectangle implements Printable{
        public void print(){
            System.out.println("Rectangle");
        }
    }
    public class PrintTriangle implements Printable{
        public void print(){
            System.out.println("Triangle");
        }
    }
}
```

A2) Дараах кодыг уншаад асуултад хариулна уу.

```
public class AnonymousInnerClassExercise{
    public AnonymousInnerClassExercise() {
        Frame frame = new Frame();
        frame.addWindowListener(new WindowAdapter() {
            public void windowClosing(WindowEvent e) {
                System.exit(0);
            }
        });
    }
}
```

1. Өмнөх кодоод өгөгдсөн нэргүй дотоод классын (классын нэр нь тодорхой бус) эх класс (удамшуулсан класс) нь юу болохыг бичнэ үү.

WindowAdapter класс

2. Дээрх кодыг нэртэй дотоод класс ашигласан байдлаар болгон өөрчилж дахин бичнэ үү.

```
public class AnonymousInnerClassExercise {
    private class WindowCloser extends WindowAdapter {
        public void windowClosing(WindowEvent e) {
            System.exit(0);
        }
    }

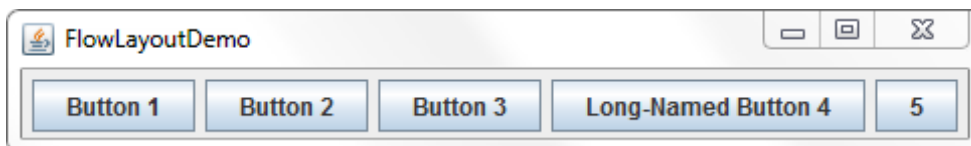
    public AnonymousInnerClassExercise() {
        Frame frame = new Frame();
        frame.addWindowListener(new WindowCloser());
    }
}
```

3. Дээрх кодыг дээд түвшний класс (бие даасан байдлаар классын гадна талд тодорхойлсон класс болгон) ашиглаж байгаагаар дахин бичнэ үү.

```
public class WindowCloser extends WindowAdapter {
    public void windowClosing(WindowEvent e) {
        System.exit(0);
    }
}

public class AnonymousInnerClassExercise {
    public AnonymousInnerClassExercise() {
        Frame frame = new Frame();
        frame.addWindowListener(new WindowCloser());
    }
}
```

A3) Дараах харагдацыг үүсгэх FlowLayoutDemo –классыг үүсгэ. Үүнд JButton, JFrame –ийг хэрэгжүүлту ашиглах ба програмд шаардлагатай импортын хэсгийг гүйцээн бичнэ. Тус даалгаварт зөвхөн чухал шаардлагатай импортыг л зааж өгөхийг анхаараарай.)



```
import java.awt.Container;
import java.awt.FlowLayout;
*****
```

Import –ийг гүйцээ...

```
import java.awt.Button;
import java.awt.Frame;
```

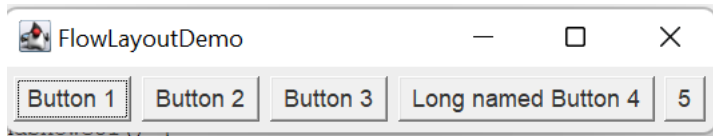
A4) Дээрх харагдацыг үүсгэхийн тулд дараах функцад 5 товчлуурыг нэмэх үйлдлийг гүйцээх хэсэг.

```
public static void addComponentsToPane(Container pane)
    pane.setLayout(new FlowLayout());
    //Энд кодыг бичнэ
    pane.setSize(400, 150);
    pane.setVisible(true);
    Button b1 = new Button("Button 1");
    pane.add(b1);
```

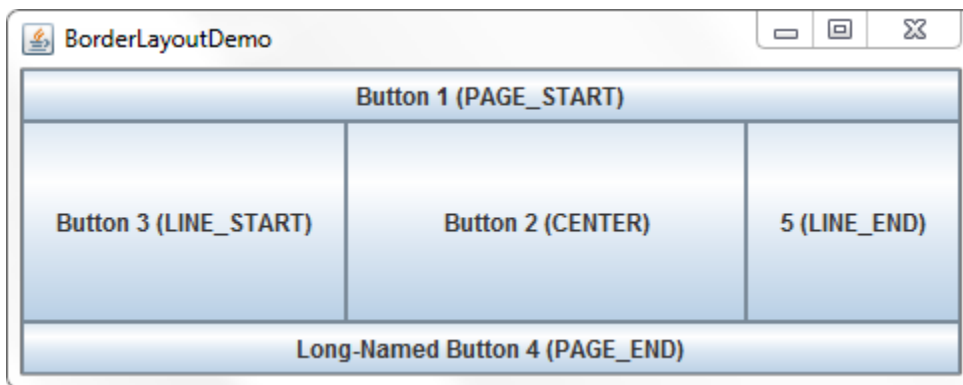
Програм хангамжийн хөгжүүлэлт (ICSI402)

```
Button b2 = new Button("Button 2");
pane.add(b2);
Button b3 = new Button("Button 3");
pane.add(b3);
Button b4 = new Button("Long named Button 4");
pane.add(b4);
Button b5 = new Button("5");
pane.add(b5);
```

```
}
```



A5) [BorderLayoutDemo.java](#) –г хараад дараах харагдацыг үүсгэхэд шаардлагатай `addComponentsToPane` функцийг гүйцээн бичнэ үү.



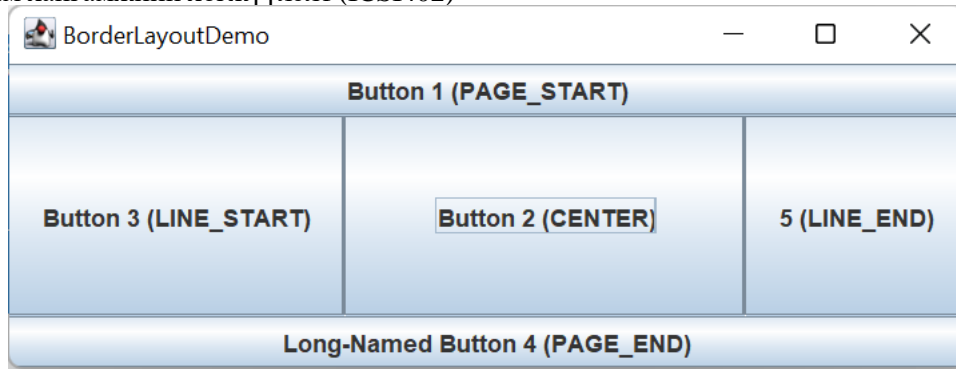
```
public static void addComponentsToPane(Container pane) {
    JButton button = new JButton("Button 1 (PAGE_START)");
    pane.add(button, BorderLayout.PAGE_START);
    // Энд кодыг бичнэ

    button = new JButton("Button 2 (CENTER)");
    button.setPreferredSize(new Dimension(200, 100));
    pane.add(button, BorderLayout.CENTER);

    button = new JButton("Button 3 (LINE_START)");
    pane.add(button, BorderLayout.LINE_START);

    button = new JButton("Long-Named Button 4 (PAGE_END)");
    pane.add(button, BorderLayout.PAGE_END);

    button = new JButton("5 (LINE_END)");
    pane.add(button, BorderLayout.LINE_END)
}
```



GradientsOutlines.java кодыг нээж ажиллуулахад дараах харагдац харагдана.



A6) strokeButton, c1Button болон c2Button товчлуурууд дээр ActionListener нэмж өгнө.

<https://docs.oracle.com/javase/tutorial/uiswing/events/actionlistener.html>

```
@Override
public void actionPerformed(ActionEvent e) {
    strokeButton.setText(textArea.getText().concat("You have clicked the Stroke
Colour button\n"));
    c1Button.setText(textArea.getText().concat("You have clicked the Colour 1
button\n"));
    c2Button.setText(textArea.getText().concat("You have clicked the Colour 2
button\n"));
}
```

A7) Товч тус бүр нь нэг нэг өнгөний (Color) хувьсагчид утга оноодог байхаар JColorChooser-ийг хэрэгжүүлсэн байх хэрэгтэй ба товч дээр дарах үед JColorChooser цонх харуулдаг байхаар програмчилна. <https://docs.oracle.com/javase/tutorial/uiswing/components/colorchooser.html>

```
Color StrokeBColor = JColorChooser.showDialog( drawPanel, "Unguu songo sb",
sColor);
Color c1Color = JColorChooser.showDialog( drawPanel, "Unguu songo c1", c1);
Color c2Color = JColorChooser.showDialog( drawPanel, "Unguu songo c2", c2);
```

A8) JColorChooser –ийг хэрэгжүүлсэн ийн буцаах утгын төрөл юу вэ?

Өнгө буцаана.

A10) Хэрвээ JColorChooser дээр cancel товч дарах үед ямар утга буцах вэ? Үүнийг хэрхэн засах вэ?

Null утга буцаана.

A11) Тойрог зурахад шаардлагатай зүйлийг paint() дотор бичиж өгнө үү.
<https://docs.oracle.com/javase/tutorial/2d/geometry/strokeandfill.html>

```
public void paint(Graphics g) {  
    super.paintComponents(g);  
    Graphics2D g2 = (Graphics2D) g;  
    g2.setStroke(strokes[0]);  
    g2.setPaint(c1);  
    g2.fill(circleShape);  
    g2.setColor(sColor);  
    g2.draw(circleShape);  
}
```

A12) Зурсан тойргыг c1 болон c2) өнгөөр уусгаж будахын тулд юу хийж өгөх шаардлагатай вэ?
Будахдаа g2) –ийг хэрэгжүүлсэнийг ашиглана.

```
GradientPaint gradinetCircle = new GradientPaint(circleShape.getBounds().x,  
circleShape.getBounds().y, c2,  
circleShape.getBounds().x + circleShape.getBounds().width,  
circleShape.getBounds().y + circleShape.getBounds().height,  
c1);
```