Uke 13 – GUI del 1

in1010 - gruppe 2

29. april 2025

Beskjeder

- Ingen innlevering til GUI, men bør forvente at det kommer på eksamen siden det utgjør 15% av pensum
- 5-7 mai blir det repetisjonstimer

this week>

- Hva er GUI, og hvorfor er det nyttig.
- JFrame, JButton, JPanel, JLabel
- Hendelsestyrt programmering
- Seminaroppgaver

Hva er GUI?

- Graphical user interface
- Et grafisk brukersnitt i motsetning til terminalen (tekstbasert)
- Brukes til å forenkle interaksjon mellom mennesker og maskin.
  - F.eks kan det være enklere å trykke på en knapp fremfor å skrive i terminalen
- I kurset skal vi lage GUI applikasjoner som kjører på windows og MacOS uten behov for nettleser

Relaterte GUI objekter

- **JFrame**: Selve hovedvinduet til det grafiske brukergrensesnittet
- **JPanel**: en del av vinduet vi fyller med grafisk og/eller interaktivt innhold
- **JLabel**: et objekt med tekst, som vi oftest legger inn i et panel
- **JButton**: en klikkbar knapp forbundet med en handling/hendelse

```
try {
    UIManager.setLookAndFeel(UIManager.getCrossPlatformLookAndFeelClassName());
} catch (Exception e) { System.exit(status:1); }

JFrame vindu = new JFrame(title:"Dette er en JFrame");
vindu.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);

JPanel panel = new JPanel();
JLabel tekst = new JLabel(text:"Dette er en JLabel");
JButton knapp = new JButton(text:"Dette er en JButton");

panel.add(tekst); panel.add(knapp);
vindu.add(panel);

vindu.pack(); vindu.setLocationRelativeTo(c:null);
vindu.setVisible(b:true);
```

Hendelsesstyrt programmering med GUI

- **Hendelsesstyrt programmering** lar vi bruker av programmet **ta kontroll over** hva som skal skje. I motsetning til *imperativ programmering* (at utvikleren bestemmer flyten fra start til slutt).
- **Event dispatch Thread (EDT)/hendelsestråden**
  - Hovedtråden som holder styr på eventer av tråder i GUI.
  - Den holder styr på køen av hendelser, slik at det blir utført i riktig rekkefølge.
  - For små enkle programmer vil Swing starte EDT automatisk når swing-obj blir laget.
  - Ikke det samme som main-tråd.

Hendelsesstyrt programmering med GUI

- **ActionListener**
  - Lytter til bruker, og venter på at bruker skal gjøre noe
  - f.eks trykke på en knapp
- **ActionPerformed**
  - Metoden som kjører automatisk når handling skjer.
  - f.eks skrive ut tekst når knapp er trykt på.

```java
public class ToKnapper {
    Run | Debug
    public static void main(String[] args) {
        try {
            UIManager.setLookAndFeel(UIManager.getCrossPlatformLookAndFeelClassName());
        } catch (Exception e) { System.exit(status:1); }

        JFrame vindu = new JFrame(title:"To knapper");
        vindu.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        JPanel panel = new JPanel();
        JButton knapp1 = new JButton(text:"Knapp 1"), knapp2 = new JButton(text:"Knapp 2");
        panel.add(knapp1); panel.add(knapp2);

        knapp1.addActionListener(new ActionListener() {
            @Override public void actionPerformed(ActionEvent e) {
                System.out.println(x:"Du trykket på Knapp 1");
            }
        });
        knapp2.addActionListener(new ActionListener() {
            @Override public void actionPerformed(ActionEvent e) {
                System.out.println(x:"Du trykket på Knapp 2");
            }
        });

        vindu.add(panel); vindu.pack();
        vindu.setLocationRelativeTo(c:null); vindu.setVisible(b:true);
    }
}
```

Eksempel: Program der vi endrer bakgrunnsfargen til JFrame med enter

- Fange opp tastetrykk hendelser er ikke pensum, blir ikke tatt opp i forelesningene, men en kult tin å gjøre

```
Color[] farger = {Color.BLUE, Color.ORANGE, Color.YELLOW};
Random tilfeldig = new Random();
JFrame vindu = new JFrame(title:"Endre farge paa JFrame med entertasten");
vindu.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
vindu.setSize(width:500, height:300);

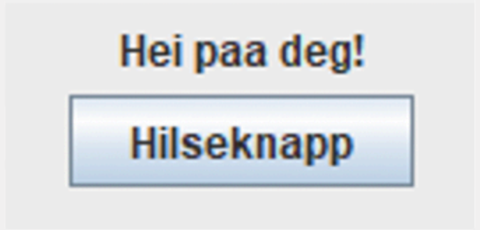

//Bruker addKeyListener og KeyAdapter-interfacet ved tastetrykk-hendelser
vindu.addKeyListener(new KeyAdapter() {
    @Override public void keyPressed(KeyEvent e) {
        if (e.getKeyCode() == KeyEvent.VK_ENTER) {
            vindu.getContentPane()
                .setBackground(farger[tilfeldig.nextInt(farger.length)]);
        }
    }
});

vindu.setFocusable(focusable:true); //Anbefalt for aa la vinduet motta tastetrykk
vindu.setLocationRelativeTo(c:null);
vindu.setVisible(b:true);
```

Oppgave 1: Hilseknapp

a) Lag et JPanel med en JLabel og en JButton. La JLabel-en være tom og la knappen ha teksten “Hilseknapp”.

b) Når man trykker på knappen skal JLabelen vise “Hei paa deg!”. Skriv en ActionListener for knappen.



c) Endre koden; hvis man trykker på knappen en gang til, skal “Ha det!” vises.

seminaroppgaver