

Interaktiv grafikk II

INF101 forelesning 28. mars 2022

Torstein Strømme

Stikkord: Museklikk, observable, tic-tac-toe

Denne uken

- Semesteroppgave 2
 - Åpen oppgave
 - Blob Wars
- Interaktiv grafikk i Java Swing, del II
 - Museklikk
 - Layout manager
 - Frikoble kontroller fra view: Observable -mønsteret
 - Eksempel: Tic tac toe
- Semesteroppgave 1: Refleksjon

Semesteroppgave 2: Åpen oppgave

- Gjør **hva du vil**, så lenge det er Java
- Du *må* ha ukentlig progressjon
- Eksempler
 - Arkade-spill
 - En applikasjon som gjør noe nyttig, kult eller morsomt
 - GUI-baserte applikasjoner anbefales

Semesteroppgave 2: Åpen oppgave

- Dokumentasjon (3 poeng)
- Funksjonalitet (3 poeng)
- Testing (3 poeng)
- Kodestil (3 poeng)
- Arkitektur (3 poeng)

Åpen oppgave eller Blob Wars?

Åpen oppgave

- Lav terskel for å skrive første linje
- Blir mer komplisert etter hvert (særlig hvis du velger en rar arkitektur)
- Hjelpen du får av oss er på et høyere/mer abstrakt nivå

Blob wars

- *Mye* å lese og forstå før første linje kan skrives
- Ikke *egentlig* så mye å skrive selv
- Gruppeledere kan hjelpe på et mer konkret nivå

Swing: et hierarki

```
public class SubPanel extends JPanel {
    private Color bgColor;

    public SubPanel(Color bgColor) { this.bgColor = bgColor; }

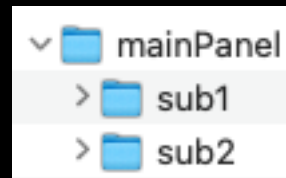
    @Override
    protected void paintComponent(Graphics g) {
        super.paintComponent(g);
        g.setColor(this.bgColor);
        g.fillRect(0, 0, getWidth(), getHeight());
    }

    @Override
    public Dimension preferredSize() { return new Dimension(300, 100); }
}
```

- JPanel -objekter kan organiseres i et hierarki

```
public class MainPanel extends JPanel {
    JPanel sub1 = new SubPanel(Color.BLUE);
    JPanel sub2 = new SubPanel(Color.YELLOW);

    public MainPanel() {
        this.add(sub1);
        this.add(sub2);
    }
}
```

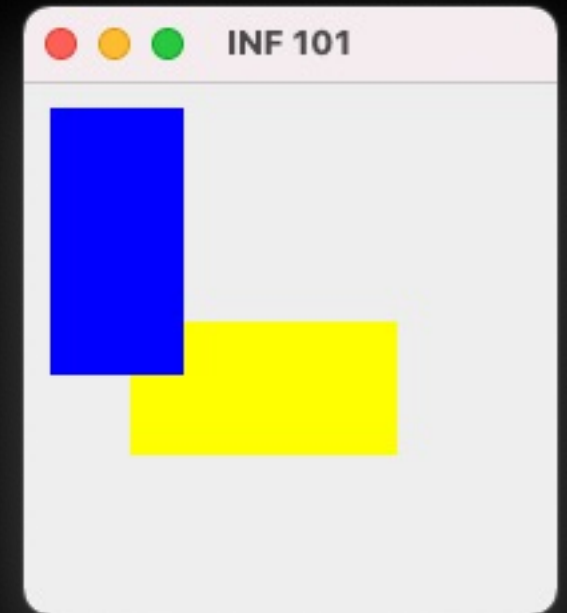


Swing: et hierarki

- JPanel -objekter kan organiseres i et hierarki
- Hvert panel kan plasseres manuelt

```
@Override
protected void paintComponent(Graphics g) {
    super.paintComponent(g);
    this.sub1.setBounds(10, 10, 50, 100);
    this.sub2.setBounds(40, 90, getWidth() / 2, 50);
}
```

```
@Override
public Dimension preferredSize() {
    return new Dimension(200, 200);
}
```



Swing: layout manager

- JPanel -objekter kan organiseres i et hierarki
- Hvert panel kan plasseres automatisk

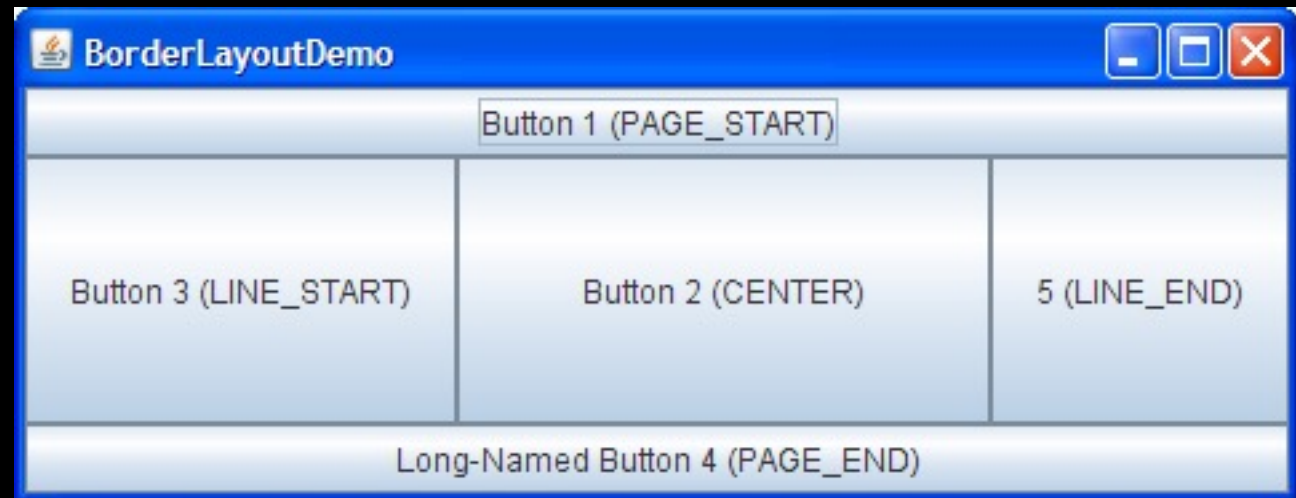


```
JPanel sub1 = new JPanel(Color.BLUE);
JPanel sub2 = new JPanel(Color.YELLOW);

public JPanel() {
    this.setLayout(new BorderLayout(this, BorderLayout.PAGE_AXIS));
    this.add(sub1);
    this.add(sub2);
}
```

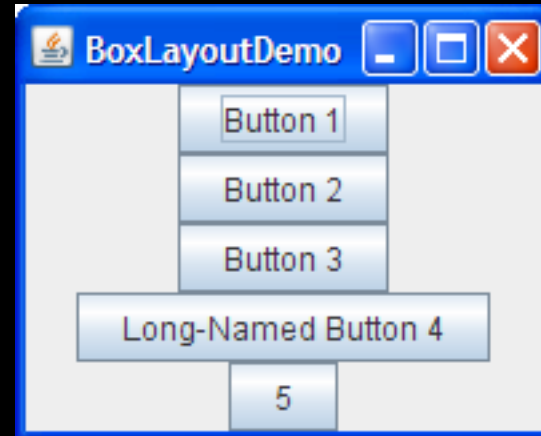

Swing: layout manager

- Flere ulike layout-managere å velge mellom:
- **BorderLayout**



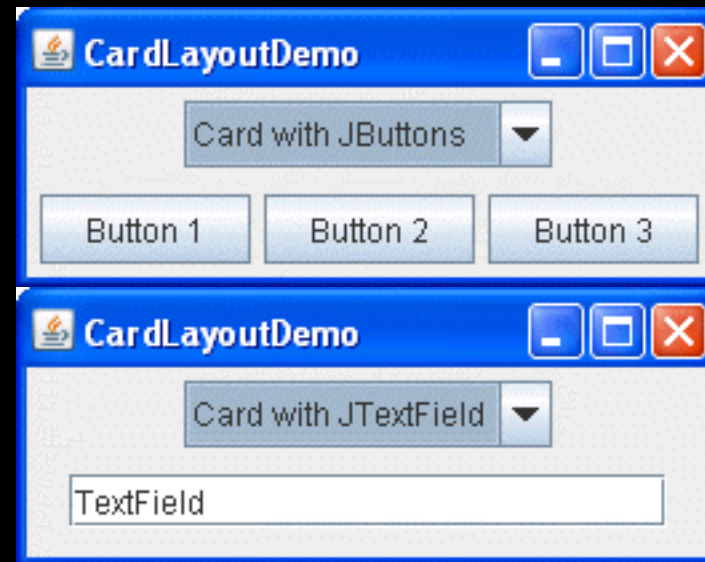
Swing: layout manager

- Flere ulike layout-managere å velge mellom:
- BorderLayout
- **BoxLayout**



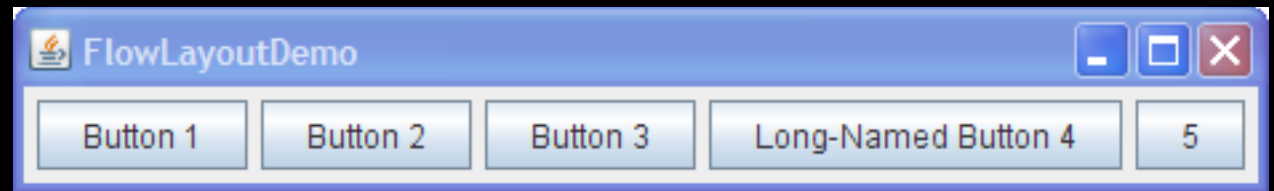
Swing: layout manager

- Flere ulike layout-managere å velge mellom:
- BorderLayout
- BoxLayout
- **CardLayout**



Swing: layout manager

- Flere ulike layout-managere å velge mellom:
- BorderLayout
- BoxLayout
- CardLayout
- **FlowLayout**



Swing: layout manager

- Flere ulike layout-managere å velge mellom:
- BorderLayout
- BoxLayout
- CardLayout
- FlowLayout
- **GridLayout**



Swing: layout manager

- Flere ulike layout-managere å velge mellom:
- BorderLayout
- BoxLayout
- CardLayout
- FlowLayout
- GridLayout
- og flere eksempler:

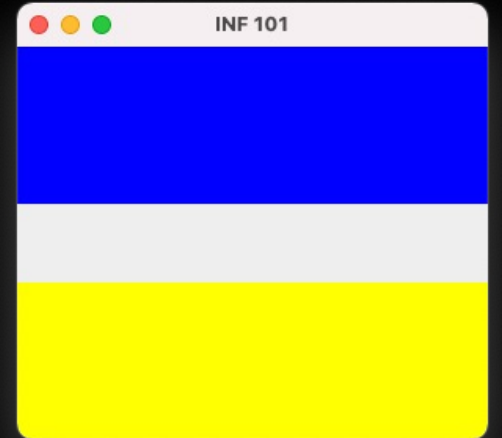
<https://docs.oracle.com/javase/tutorial/uiswing/layout/visual.html>

<https://docs.oracle.com/javase/tutorial/uiswing/examples/layout/index.html>

Swing: layout manager

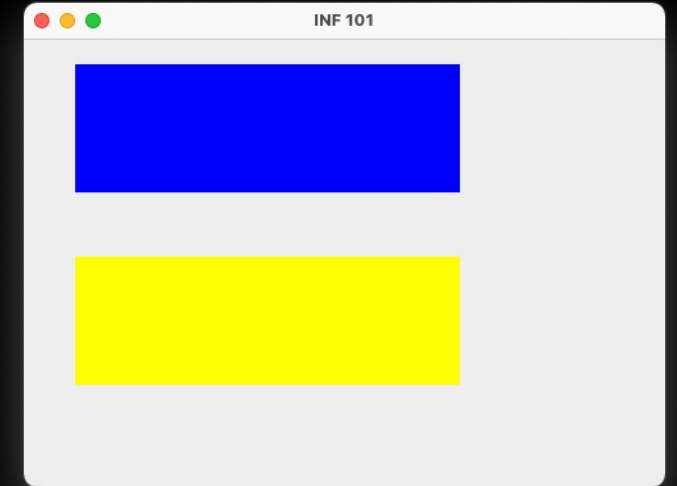
- Legg til ekstra tomrom

```
this.setLayout(new BorderLayout(this, BorderLayout.PAGE_AXIS));  
this.add(sub1);  
this.add(Box.createRigidArea(new Dimension(0, 50)));  
this.add(sub2);
```



Swing: layout manager

- Legg til en ramme

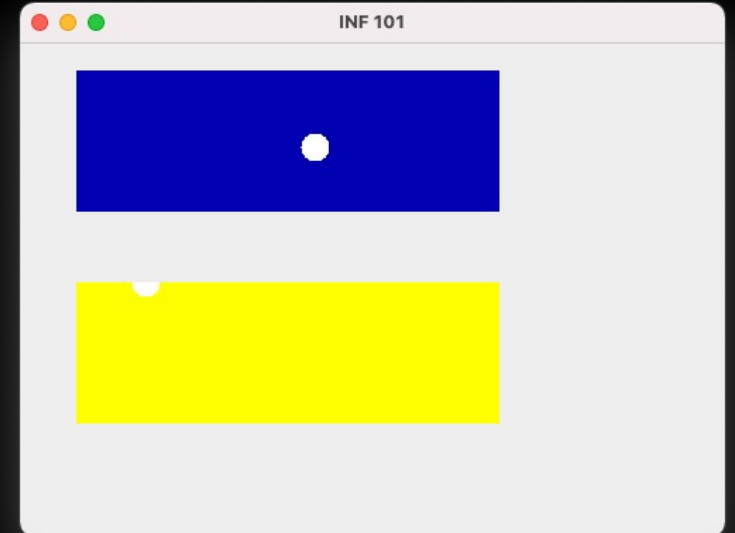


```
this.setLayout(new BorderLayout(this, BorderLayout.PAGE_AXIS));  
this.setBorder(BorderFactory.createEmptyBorder(20, 40, 80, 160));  
this.add(sub1);  
this.add(Box.createRigidArea(new Dimension(0, 50)));  
this.add(sub2);
```


Swing: håndtere musebevegelser og klikk

- Legg til en `MouseListener` eller en

```
public SubPanel(Color backgroundColor) {  
    this.bgColor = backgroundColor;  
  
    MouseListener mouseListener = this;  
    this.addMouseListener(mouseListener);  
  
    MouseMotionListener mouseMotionListener = this;  
    this.addMouseMotionListener(mouseMotionListener);  
}
```



Observable -mønsteret

- Få beskjed når en verdi har endret seg
 - Eksempel: kall repaint på en rute i TicTacToe når en verdien blir endret
 - Eksempel: kall keyPressed når variabelen for om en taste er trykket endrer seg
 - Eksempel: kall mousePressed når variabelen for om musetasten er nedtrykket endrer seg
- Den som er avhengig av en variabel (observer):
 - Installerer seg selv på subjektet (observable)
 - En spesiell metode blir kalt på observer når state endrer seg

Observable -mønsteret

