

Fönsterapplikation Java – Java Swing

För att skapa ett grafiskt gränssnitt kan man använda biblioteket Java Swing.

- Börja med att skapa ett nytt projekt och en ny klass för Main-metoden.

Ett första fönster

Börja med att skapa en **JFrame** som är ett fönster som du är van vid från Windows.

För att det ska fungera ska vi första importera ett bibliotek:

```
import javax.swing.JFrame;
```

Sedan ska vi skapa själva fönstret. Det är ett objekt som det mesta i Java och i konstruktorn sätter vi titeln på fönstret. Vi måste också tala om att när vi stänger på krysset ska hela programmet stängas, inte bara fönstret. Annars fortsätter applikationen att köra i bakgrunden. Vi sätter också storleken på fönstret.

```
JFrame frame = new JFrame("Java Swing");
```

```
//För att programmet och inte bara fönstret ska stängas vid tryck på kryss.  
frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
```

```
//Gör fönstret synligt.  
frame.setVisible(true);
```

Skapa separat klass

För att vi ska ha mer kontroll över vårt nya fönster ska vi skapa en separat klass. Det blir då en subklass till vårt fönster som vi normalt använder.

Skapa en ny klass som heter MainFrame. Denna blir alltså en subklass till vårt huvudfönster.

I main kan vi skapa ett nytt objekt och det gör vi på ett lite speciellt sätt så här:

```
new MainFrame();
```

Resten ska vi kopiera till vår nya klass.

Först ska vi tala om av vi vill ärva från vårt huvudfönster så här:

```
public class MainFrame extends JFrame implements ActionListener
```

Det vi kopierade ska vi sedan lägga i konstruktorn i vår nya klass.

Nu behöver vi inte sätta *frame* framför.

```
public MainFrame(){ //Konstruktör
```

```
//Namn på fönster  
super("Hello world!");
```

```
setLayout(null);
```

```
//Sätter storlek på fönstret.  
setSize(1200, 1000);
```

```
//Hindra att fönstret ändrar storlek  
setResizable(false);
```

```
//För att programmet och inte bara fönstret ska stängas vid tryck på kryss.  
setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
```

```
//Gör fönstret synligt.  
setVisible(true);
```

```
getContentPane().setBackground(Color.CYAN);
```

Det finns olika layouter på hur saker ska ordnas i ett fönster. Googla på **visual guide layout swing** och titta på översta resultatet.

Förs importerar vi biblioteket:

Sedan skriver vi (under `super("Hello World")`):

Sedan:

```
add(btn, BorderLayout.SOUTH);
```

Vi ska dock börja med att inte ha någon layout alls.

```
setLayout(null);
```

Vi deklarerar dom ovanför konstruktorn.

```

/*****
*
*      Detta är de ramar man kan prova för stor textrutan
*
* *****/

blackline = BorderFactory.createLineBorder(Color.black);
raisedetched = BorderFactory.createEtchedBorder(EtchedBorder.RAISED); loweredetched =
BorderFactory.createEtchedBorder(EtchedBorder.LOWERED);
raisedbevel = BorderFactory.createRaisedBevelBorder();
loweredbevel = BorderFactory.createLoweredBevelBorder();
empty = BorderFactory.createEmptyBorder();
*****/

```

I vår konstruktor skapar de komponenter vi behöver

```
alist = new ArrayList<book>(); //Skapar en lyssnar som lyssnar efter händelser

//Knappar
btn = new JButton("Lägg in bok");
btn.setBounds(120,70,200,20);
add(btn);
btn.addActionListener(this); //Skapar en lyssnar som lyssnar efter händelser

btn2 = new JButton("Visa böcker");
btn2.setBounds(220,470,200,20);
add(btn2);
btn2.addActionListener(this); //Skapar en lyssnar som lyssnar efter händelser

//Textfält
txtFieldTitle = new JTextField();
txtFieldTitle.setBounds(570,70,200,20);
add(txtFieldTitle);

txtFieldAuthor = new JTextField();
txtFieldAuthor.setBounds(570,110,200,20);
add(txtFieldAuthor);

txtFieldPrice = new JTextField();
txtFieldPrice.setBounds(570,150,200,20);
add(txtFieldPrice);

txtFieldLanguage = new JTextField();
txtFieldLanguage.setBounds(570,190,200,20);
add(txtFieldLanguage);

//Textareor (större textfält)
ta1 = new JTextArea();
ta1.setBounds(720,470,400,420);
ta1.setBorder(BorderFactory.createRedBorder());
add(ta1);

//RadioKnappar start
proRadio = new JRadioButton("Bok");
amateurRadio = new JRadioButton("Programmeringsbok");
proRadio.setSelected(true);
playersBtnGroup = new ButtonGroup();
playersBtnGroup.add(proRadio);
playersBtnGroup.add(amateurRadio);

proRadio.setBounds(420,300,200,20);
amateurRadio.setBounds(620,300,200,20);

add(proRadio);
add(amateurRadio);
//RadioKnappar end

//Labels
txtTextArea = new JLabel("Böcker");
txtTextArea.setBounds(720,430,200,20);
add(txtTextArea);

txtTitle = new JLabel("Titel");
txtTitle.setBounds(500,70,200,20);
add(txtTitle);
```

ActionPerformed

När man skapat en lyssnare som vi gjort ska man ha en metod som tar emot händelsen (att man trycker på en knapp t.ex.)

I denna metoden talar vi om vad som ska hända när vi har tryckt på knappen t.e.x.

```
public void actionPerformed(ActionEvent e)
{
    if(e.getSource() == btn) //Lägg in bok
    {
        String title = txtFieldTitle.getText();
        String anAuthor = txtFieldAuthor.getText();
        String anPrice = txtFieldPrice.getText();
        String aLanguage = txtFieldLanguage.getText();

        int thePrice = Integer.parseInt(anPrice);

        book enBok = new book(title, anAuthor , thePrice);
        alist = new ArrayList<book>();
        alist.add(enBok);

        if (proRadio.isSelected() == true){
        }

        if (amateurRadio.isSelected() == true){
        }

    }

    //if getSource

    if(e.getSource() == btn2) //Visa böcker
    {
        //Foreach JavaWay :)
        //For each book b IN alist

        for (book b : alist) {
            ta1.append("Titel: " + b.getTitle() + "\n");
            ta1.append("Författare: " + b.getAuthor() + "\n");
            ta1.append("Pris " + Integer.toString(b.getPrice()) + "\n");
            ta1.append("\n");
        }
    }
}
```

Uppgifter

Programmet är gjort för att vara en grund som ni kan bygga vidare på:

Att göra:

- Skapa de klasser ni behöver.
 - Skapa minst en klass för att öva.
 - Ni kan skapa en klass till och/eller göra övningen med golf.
- Snygga till programmet.
 - Lägg till de labels som behövs.

- Snygga till på andra sätt.
 - Testa och prova olika saker.
 - Kanske använda en färdig layout istället.
- Det finns inga felhanterare och inget som talar om att du har sparat en bok.
 - Lägg till något som talar om att du har lagt in en bok och att det fungerade.