

FEL!!

- Kompileringsfel
- Logiskt fel
- exekveringsfel

Kontrollera att indata är ett korrekt heltal

```
// Ett avsiktligt felaktigt program, version 1
import javax.swing.*;

public class DemoAvFell {
    public static void main (String[] arg) {
        String t = JOptionPane.showInputDialog("Ett tal?");
        for (int i=1; i<=t.length(); i++)
            if (t.charAt(i) >= '0' && <= '9')
                JOptionPane.showMessageDialog(null,"Talet är OK");
            else
                JOptionPane.showMessageDialog(null,"Inget tal");
        }
    }
}
```

Går det att kompilera?

```
DemoAvFel1.java:7: error: illegal start of expression
    if (t.charAt(i) >= '0' &&   <= '9')
```

Rätta kompileringsfel.

Ändra i if-satsen:

```
// Ett avsiktligt felaktigt program, version 2
import javax.swing.*;

public class DemoAvFel2 {
    public static void main (String[] arg) {
        String t = JOptionPane.showInputDialog("Ett tal?");
        for (int i=1; i<=t.length(); i++)
            if (t.charAt(i) >= '0' && t.charAt(i) <= '9')
                JOptionPane.showMessageDialog(null, "Talet är OK");
            else
                JOptionPane.showMessageDialog(null, "Inget tal");
    }
}
```

Vad händer när man kör programmet?

Rätta logiska fel:

```
// Ett avsiktligt felaktigt program, version 3
import javax.swing.*;

public class DemoAvFel3 {
    public static void main (String[] arg) {
        String t = JOptionPane.showInputDialog("Ett tal?");
        boolean ok = true;    // korrekt, än så länge
        for (int i=1; i<=t.length(); i++)
            if (t.charAt(i) < '0' || t.charAt(i) > '9') {
                ok = false;    // inte längre korrekt
                break;
            }
        if (ok)
            JOptionPane.showMessageDialog(null,"Talet är OK");
        else
            JOptionPane.showMessageDialog(null,"Inget tal");
    }
}
```

Varför måste man krångla till det med variabeln **OK**?

Räcker det inte med att lägga en **break**-sats i **else**-delen av **if**-satsen i version 2 av programmet?

Kan man skriva version 3 av programmet utan att använda en **break**-sats?

Tips: Utöka villkors-delen i **for**-satsen.

```
for (int i=1; i<=t.length() && ok; i++)  
    if (t.charAt(i) < '0' || t.charAt(i) > '9')  
        ok = false;           // inte längre korrekt
```

Vad händer när man provkör version 3 med ett korrekt tal som indata?

```
Exception in thread "main"  
java.lang.StringIndexOutOfBoundsException:  
    String index out of range: 3  
    at java.lang.String.charAt(String.java:658)  
    at DemoAvFel3.main(DemoAvFel3.java:8)
```

Rätta exekveringsfel.

Ändra till:

```
for (int i=0; i<t.length(); i++)
```

Ta hand om exekveringsfel

Ex.

```
Thread.sleep(10000);    // vänta 10 sek
```

Kompileringsfel:

```
unreported exception java.lang.InterruptedException;  
must be caught or declared to be thrown
```

Specificera exceptions

```
public class SignalDemo1 {  
    public static void main (String[] arg)  
        throws InterruptedException {  
        ...  
        Thread.sleep(10000); // vänta 10 sek  
        ...  
    }  
}
```

Fånga exceptions

```
import javax.swing.*;
public class SignalDemo2 {
    public static void main (String[] arg) {
        ...
        try {
            Thread.sleep(10000);    // "farlig" sats
            // Hit kommer man om det inte blev fel
        }
        catch (InterruptedException e) {
            JOptionPane.showMessageDialog(null, "Det blev fel");
        }
        // Hit kommer man till slut i båda fallen
        ...
    }
}
```

Generera exceptions

```
throw new ArithmeticException("negative area");
```

```
throw new IllegalArgumentException("size < 0");
```

Egna exception-klasser:

```
public class CommunicationException extends Exception {  
    public CommunicationException() {  
        super();  
    }  
    public CommunicationException(String s) {  
        super(s);  
    }  
}
```

```
throw new CommunicationException("Timeout in reader");  
throw new CommunicationException();
```

Try-with-resources

```
// Med resurshantering
public static void copyFile(String f1, String f2)
                                throws IOException {
    try (
        BufferedReader in = new BufferedReader(new FileReader(f1));
        PrintWriter out = new PrintWriter(new BufferedWriter
            (new FileWriter(f2)));
    ) {
        while (true) {
            String line = in.readLine();
            if (line == null)
                return;
            out.println(line);
        }
    }
}
```

Krav:

Skall implementera gränssnittet `Auto-Closeable` vilket som enda metod har metoden `close`.