



## Laboration1

Utgå ifrån Cirkel exemplet som gicks igenom på F1 (Se G-drive – Java material- F1).  
Programmet bestod av två klasser klassen Cirkel samt en main- klass.

```
package cirkel;

public class Cirkel {

    private final double PI = 3.14;
    private double radie;

    public Cirkel(double radie) {
        this.radie = radie;
    }

    public double getRadie() {
        return radie;
    }

    public void setRadie(double radie) {
        this.radie = radie;
    }

    public double area() {
        double area = radie * radie * PI;
        return area;
    }
}
```

Konstruktör

```
import cirkel.Cirkel;

class MyMain {

    public static void main(String[] args) {
        Cirkel c = new Cirkel(5);

        System.out.println("Cirkelns area med radien " + c.getRadie() + " är " + c.area());
    }
}
```

Se powerpointen F1-Java Standard Edition om något i koden ovan är oklart.

### Uppgift 1:

1. Komplettera klassen Cirkel med ytterligare en metod. Metoden ska beräkna omkretsen på en cirkel.
2. I main-metoden skapa en cirkel som du beräknar area och omkrets på. Skriv sedan ut dessa värden.

### Uppgift 2:

1. Skapa klasser för nya figurer exempelvis triangel och rektangel. Klasserna rektangel och triangel ska ha attribut som är väsentliga för respektive figur. Motsvarande för en cirkel var exempelvis attributet radie.
2. I klassen rektangel och triangel ska du skapa metoder för beräkning av area respektive omkrets samt konstruktörer. Det är bra om du även har privata attribut och get och set metoder för att kunna komma åt dessa attribut.
3. I main-metoden testa nu dina nyskapade klasser. Skapa en triangel respektive rektangel som du beräknar area respektive omkrets på. Skriv ut resultatet av beräkningen.

### Uppgift3:

1. Gör om main-metoden så att programmet startar med att låta användaren välja mellan figurerna.
2. Sedan ska användaren frågas om värdet på respektive attribut för den valda figuren. Slutligen får användaren välja emellan att beräkna area, omkrets eller båda.
3. Beroende på om användaren väljer Cirkel, Rektangel eller Triangel ska respektive figur skapas. Med andra ord ska du skapa instanser av klasserna i main metoden. Sedan ska relevanta beräkningar ske och resultatet skrivs ut.

### Tips

För att kunna läsa in inmatning från en användare använd dig av ett Scanner objekt. Du får nedan en kort presentation av hur man skapar och använder Scanner objektet. Du hittar mer information på powerpoint F1-JJava Standard Edition.

```
Scanner sc= new Scanner(System.in);
```

Nu kan vi använda **sc** som vår scanner och läsa in datan som användaren matar in genom att anropa dess metod `nextLine()` eller `nextInt()`.

```
System.out.print(" vad heter du?");
```

```
String namn= sc.nextLine( );
```

Läser in det användaren matar in och spar det i variabeln **namn** som är av typen String dvs text.

```
System.out.print(" Hur många koppar kaffe dricker du om dagen?");
```

```
int antalKoppar= sc.nextInt( );
```

Läser in det användaren matar in och spar det i variabeln antalKoppar som är av typen int dvs heltal.

För att kunna använda scanner objektet behöver du importera **Import java.util.Scanner;** högst upp i klassen.

Du kommer även att behöva använda dig av en if eller switch sats efter inläsningen.

<http://coursepress.lnu.se/pub/education/course/1DV402/ht13/doc/artiklar/hur-fungerar-switch-satsen.pdf>

```
if (villkor)
{
    sats;
    sats;
}
```

Om vi vill att något ska hända om villkorssatsen är *falsk* så bygger vi på med en **else**-gren: Så här:

```
if (villkor)
{
    sats;
    sats;
}
else
{
    sats;
    sats;
}
```

**I icke programmeringsspråk enligt nedan logik:**

```
If (det_användaren_matar_in= rektangel ){
```

Fråga efter höjd och bas, läs in värdena.

Skapa en rektangel objekt med ett anrop med new operatör till konstruktorn i klassen rektangel. Konstruktorn ska kunna ta emot 2 värden, höjd och bas.

Fråga vad för beräkning som användaren vill ska utföras, area, omkrets eller båda.

Utför beräkningen och presentera resultatet.

```
}
```

```
else if (det_användaren_matar_in=triangel){
```

Fråga efter höjd och bas, läs in värdena, osv.

```
}
```

else {fråga efter radien och skapa en cirkel gör beräkningarna m.m.}

Kom ihåg att Instansering sker genom anrop till respektive figurs konstruktör. Du skapar i och med detta objekt från från figur klasserna. Det är först då som vi har ett objekt som vi kan utföra beräkningar för och använda metoderna och attributen.

### **Exceptions:**

OBS. Vid fel inmatning från användaren kommer du att märka att programmet kommer att krascha. Detta kan inträffa om vi exempelvis antar att användaren kommer att mata ett tal men får som inmatning en sträng. Vi kommer i nästa vecka lära oss hantera dessa typer av problem. För övningen idag se till att mata in rätt typ av indata.

### **Uppgift 4: Bonus uppgift**

Skriv om din main klass så att användaren efter varje beräkning får frågan om den vill fortsätta göra beräkningar på en ny figur eller avsluta programmet. Annars måste man ju starta om programmet för varje beräkning.

Tips: använd while eller do while loop.

Lösningsförslag läggs upp fredag kväll eller lördag dag.