

7 Løkker og Funktioner

Læs om while-løkker i kapitel 3.3 og om for-løkker i kapitel 3.4. Læs om funktioner i kapitel 3.5

Du kender nu til følgende datatyper og instruktioner:

Datatyper:

- float: kommatatal.
- double: mere præcis end float.
- char: en enkelt karakter. Char er en unsigned byte og kan derfor ikke indeholde minustal.
- int: heltal.
- long: 2 gange så stor som en integer.
- boolean: sand eller falsk.

Instruktioner:

- `size();` // sætter størrelsen for canvas
- `line();` // tegner en linje
- `stroke();` // farven på en streg
- `strokeWeight();` // tykkelse af streg
- `rect();` // tegner en rektangel
- `circle();` // Tegner en cirkel
- `arc();` // Tegner en bue
- `fill();` // udfylder en figur med farve. ²

²Find rgb farver her: www.rapidtables.com/web/color/RGB_Color.html

Vi skal også arbejde med funktioner. Du kender dem fra matematikken $f(x)$, hvor en funktion beregner/returnerer en værdi. Vi bruger funktioner når vi skal udføre den samme sekvens flere gange, med forskellige variabler. Funktioner i Java, består af 4 dele.

Navn: Funktioner skal navngives med et ikke reserveret ord (ord som er brugt i forvejen), men hvor navnet er sigende for funktionens funktion. I mit eksempel er navnet "minFunktion". Vi bruger navnet når vi skal bruge funktionen.

Returdatatype: Funktioner skal deklareres efter returdatatype. Funktionen kunne returnere en int, float, string eller char, men altid kun én ting. Hvis funktionen ikke returnerer noget, skal den defineres som void.

Parameter: Funktionen kan modtage en lang række forskellige parameter. En funktions paramenter skal angives i parantesen efternavnet samtidig med at vi deklarerer datatypen. Man kan deklarere mange parametre til sin funktion. Parameter er kun gyldige lokalt. Det vil sige at du kan ikke bruge en variabel du har deklareret i andre funktioner uden at skulle deklarere dem igen.

Kode: En funktions kode skal skrives imellem de to tuborgparanteser .

7.1 Opgave

Skriv et program hvor dit canvas er 800x800. Opret en funktion cirkel, som tegner en cirkel på dit canvas. Cirklen skal være rød. Ryd op efter dig. Navngiv din funktion så den fortæller hvad funktionen gør. Opret en funktion som tegner en firkant. Firkanten skal være blå. Navngiv din funktion så den fortæller hvad funktionen gør.0

7.2 Opgave

Tilpas din funktion så den tegner 8 cirkler. Brug et for loop og brug dit index i til at gange din x koordinat i cirklen, således at din cirkel ikke bliver tegnet det samme sted.

```
for (del 1; del 2; del 3) {  
    // Den sekvens af kode som løkken skal udføre  
}
```

Et for-loop består af tre dele. Del 1 er en deklarering og initiering af vores index. Vi kalder index for i. Har vi brug for flere indlejret løkker følger vi

alfabetet og kalder den næste j så k etc. Del 2 er den betingelse som skal være opfyldt for at løkken bliver udført. Betingelsen knytter sig typisk til vores index i. For eksempel: $i_j=10$. Del tre beskriver den handling som skal ske med i efter hver iteration. Vi kan tælle i op med 1; $i++$. Eller trække en fra $i--$.

```
for (int i = 0; i < 10; i++){  
    System.out.println(i);  
}
```

Figur 8: For-løkke

7.3 Opgave

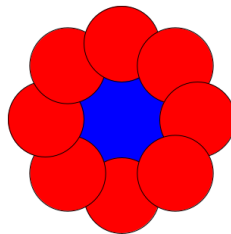
Tilpas din funktion firkant så den tegner et antal firkanter vertikalt. Hvor mange firkanter kan du få plads til?

7.4 Opgave

Omskriv din for-løkke til en while-løkke.

7.5 Opgave

Lave en blomst af cirkler. Placer en cirkel i midten og lav kronbladene af 8 cirkler. Du kan måske bruge funktionerne `pushMatrix()`, `popMatrix()`, `translate()` og `rotate()`?



Figur 9: Blomst