

2 Primitive datatyper

Det er nemt at se forskel på primitive og ikke primitive¹ datatyper i Java. Primitive datatyper staves med lille begyndelsesbogstav, ikke-primitive datatyper staves med stort begyndelses bogstav. Primitive datatyper kan repræsenteres i en eller flere bytes og kan sammenlignes med en operator '=='. Ikke primitive datatyper er klasser, som har tilknyttede funktioner. Hver gang man møder en ikke-primitiv datatype, bør man se i dokumentationen, om der er metoder, som man kan bruge. Ikke primitive datatyper kan ikke sammenlignes direkte, men kun ved hjælp af en funktion, fx equals().

Tabel 1: Liste over datatyper.

Ikke primitive	Primitive
String	integer
Array	float
Klasser	char
Interfaces	boolean
	byte
	short
	long
	double

Denne opgave handler om at forstå de forskellige datatyper.

2.1 Opgave

Undersøg for hver primitiv datatype, hvor meget plads (i bytes) hver datatype bruger. Det kan du finde her: <https://data-flair.training/blogs/java-data-types/>. Eksempel: Datatypen integer fylder 4 bytes, og den maksimale værdi er: $2^{31} - 1 = 2.147.483.648$. Noter dine resultater for hver af de 8 primitive datatyper.

2.2 Opgave

Skriv et program, som beviser, hvilke minimums- og maksimumsværdier de primitive datatyper kan indeholde. Find evt. inspiration i programmet i figur 5 på side: 6. Noter dine resultater for hver af de 8 primitive datatyper.

2.3 Opgave

De to datatyper float og double er ikke lige nøjagtige. Det kan de se ved følgende opgave: Bereg kvadratroden af 2 ganget med kvadratroden af 2?

$$\sqrt{2} * \sqrt{2}$$

Lav først et program med sqrt(), som returnerer en float, og herefter med Math.sqrt(), som returnerer en double.

¹<https://data-flair.training/blogs/java-data-types/>

- Slå funktionen `sqrt()` op i Processings dokumentation. Deklarer en variabel `f` af type `float` og initier den med værdien af `sqrt(2)`. Brug kommandoen `println(f)`; til at udskrive værdien af `f`.
- Du kan ikke slå funktionen `Math.sqrt()` op i Processings dokumentation, men den fungerer på helt samme måde som `sqrt()`. Deklarer en variabel `d` af type `double` og initier den med værdien af `Math.sqrt(2)`. Brug kommandoen `println(d)`; til at udskrive værdien af `d`.

Forklar forskellen på de to resultater.

3 Ikke-primitive datatyper

Vi skal i dette kapitel arbejde med ikke-primitive datatyper. Det gælder for de primitive datatyper, at de kan repræsenteres i en byte. Ikke-primitive datatyper kræver meget mere plads og kan ikke repræsenteres i kun en byte. Primitive datatyper er foruddefineret i Java, det er ikke primitive datatyper ikke, med nogle få undtagelser. `String`, `Array` mm. Ikke primitive datatyper skrives med stort begyndelsesbogstav. Ikke primitive datatyper kan indeholde attributter og metoder.

Den ikke-primitive datatype `String`, er et array af karakterer (`chars`). Der er til datatypen knyttet en række metoder:

- `toUpperCase()` Gør alle karakterer til store bogstaver
- `toLowerCase()` Gør alle karakterer til små bogstaver
- `substring()` Returnerer en ny streng, som er en del af den originale streng.
- `length()` Returnerer en heltalsværdi, som repræsenterer antallet af karakterer i strengen.
- `indexOf()` Returnerer indeks af den første forekomst af substreng i strengen.
- `equals()` Sammenligner to strenge.
- `charAt()` Returnerer karakteren på den af indeks angivne plads.

Eksempler på andre ikke primitive datatyper: `Strings`, `Arrays`, `Classes`, `Interface`, etc.

3.1 Opgave

Lav et program som klipper i vores streng. Den skal tage fra position 83 og til slut. Udskriv den nye streng.

3.2 Opgave

Lav et program som klipper i vores streng. Den skal tage fra position 83 og til 85. Udskriv den nye streng.

3.3 Opgave

Lav hele sætningen om til store bogstaver og udskrive den.

3.4 Opgave

Lav hele sætningen om til små bogstaver og udskriv den.

3.5 Opgave

Lav opgaver på codingbat.com: String 1