

# Introduksjon, python til java

IN1010 - Uke 1

---

Tobias Paulsen

Vår 2022

# Hvem er jeg?

- Tobias Paulsen
- Master robotikk
- Har vært gruppelærer i IN1010 to ganger tidligere + IN1000 og IN2010
- Ta kontakt ved spørsmål eller tilbakemelding rundt gruppetimene
- Mail: **tobiasrp@uio.no**

- Digitale gruppetimer første to uker, seminarrom C etter dette
- Følg med på emnesiden
- Seminar:
  - Hyggelig om folk har på kamera
  - Repetisjon og livekoding + oppgaver
  - Breakout rooms
  - Still spørsmål på chat (enten felles eller direkte til meg) evt. muntlig
  - Flott om dere korrigerer hvis noe er feil:)
  - Legger ut ressurser under "Gruppe 6"
- Oblig 0 er lagt ut, frist 24.01 kl 23.59
- Lab: 10 min med en-til-en hjelp (Lag G Suite)
- Pensumbok ligger gratis i Leganto

- Skriv så mye kode som mulig i starten
- Kan bli veldig vanskelig om man ikke har grunnlag fra de første ukene
- Kodestil, gode vaner fra start
- Les feilmeldinger nøye, alltid prøv og forstå hva de betyr
- Må kompilere før kjøring
  - `javac filnavn.java` for å kompilere fil
  - `javac *.java` for å kompilere alle java filer i current directory
- En klasse per fil, filnavnet skal være det samme som klassenavnet. F.eks *Person.java*
- Bruk javadoc

- Semikolon etter hver linje
- Krøllparanteser for å dele inn i kodeblokker
- Alltid paranteser rundt uttrykk
- Hva menes med **static**?
- Hva med **String[] args**?

```
class Eks {  
    public static void main(String[] args) {  
        int a = 5;  
        if (a < 5) {  
            System.out.println("positiv");  
        }  
        else if (a == 0) {  
            System.out.println("null");  
        }  
        else {  
            System.out.println("negativ");  
        }  
    }  
}
```

# Løkker

- For-løkker, enten med condition eller for-each
- While-løkker på samme måte som i python
- Vær obs på definering av variabelen *i* nedenfor

```
for (int i = 0; i < 100; i++) {  
    System.out.println(i);  
}
```

```
int[] tall = {2, 4, 6, 8, 10};  
for (int i : tall) {  
    System.out.println(i);  
}
```

```
int i = 0;  
while (i < 100) {  
    System.out.println(i);  
    i++;  
}
```

# Beholdere

- I java brukes array, ikke dynamisk som fører til fast lengde
- Bruker ofte ArrayList om man vil ha en dynamisk beholder som ligner mer på python sine lister (må importeres)
- Har også HashMap, LinkedList, Queue...

```
import java.util.ArrayList;

class Eksempel {
    public static void main(String[] args) {
        char[] liste = {'a', 'b', 'c', 'd', 'e'};

        String[] ny = new String[10];
        ny[3] = "Hei";

        ArrayList<Integer> arrayListe = new ArrayList<>();
        arrayListe.add(10);
    }
}
```

- Konstruktør har samme navn som klassen (ikke alltid man trenger)
- Bruk public/private for å si noe om tilgjengeligheten til variabler og metoder
- Metoder må ha returtype

```
public class Dog {  
    private int age, weight;  
    private String name;  
  
    public Dog(String name, int weight) {  
        this.name = name;  
        this.weight = weight;  
        this.age = 0;  
    }  
  
    public String getName() {  
        return this.name;  
    }  
  
    public void birthday() {  
        this.age++;  
    }  
}
```