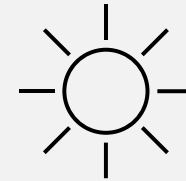


# GRUPPE II IN1010 – UKE 2

Sivert Fjeldstad Madsen

# I DAG

- Arrayer m.m
- Løkker
- I/O
- Typekonvertering
- Datastrukturtegning
- Oppgaver



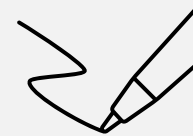
# ARRAY

- Java sin enkleste «liste»-type
- Kan kun inneholde én type data
- Må oppgi størrelse når det lages
- Kan ikke endre størrelsen etter opprettelse



# ARRAY - SYNTAKS

- `<type>[] <variabelnavn>`
- `String[] args;`
- `int[] tallArray = {1, 2, 3, 4};`
- `Person[] personArray = new Person[10];`
  - `Person testPerson = personArray[5];`



# MULTIDIMENSJONALE ARRAYER

- Man kan ha et array av arrayer!
  - Og et array av arrayer av arrayer.
  - Osv...
- `String[][] tabell = new String[10][12];`
- Få verdien til en bestemt rute i tabellen med «`tabell[y][x];`»
- Kan itereres gjennom med doble for-løkker

o		x
	x	
o		o

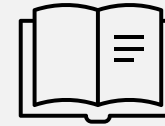
# ARRAYLIST

- Minner mye mer om Python sine lister
  - Kan endre størrelse under kjøring
  - ...men kan fortsatt bare holde på én type objekter.
- Må importeres:
  - `import java.util.ArrayList;`
- `ArrayList<Person> personListe = new ArrayList<>();`
- `personListe.add(new Person(«Julian»));`



# HASHMAP

- Minner om ordbok i Python
  - ...men man må type både nøklene og innholdet
- Må importeres:
  - `import java.util.HashMap;`
- `HashMap<String, int> personnummer = new HashMap<>();`
- `personnummer.put(«Siri», 81549300);`
- `personnummer.get(«Siri»);`



# LØKKER

- while (<uttrykk>) { <kode> }
  - while (teller < 100) { teller ++ }
- for (<variabel>;<uttrykk>;<variabelforandring>) { <kode> }
  - for (int i = 0; i < 100; i++) { System.out.println(i) }
- for (<variabel> : <beholder>) { <kode> }
  - for (int tall : liste) { System.out.println(tall) }





# LESE FRA TERMINAL

- `import java.util.Scanner;`
- `Scanner sc = new Scanner(System.in);`
- `String input = sc.nextLine();`
- Må lukkes etter bruk!
  - `sc.close();`

1010  
1010

# LESE FRA FIL

- Må importere to ting til:
  - `import java.io.File;`
  - `import java.io.FileNotFoundException;`
- `File fil = new File(«filnavn.txt»);`
- `Scanner sc = new Scanner(fil);`
  - ...men her vil vi få en feilmelding!
- Vi må håndtere et unntak



# UNNTAKSHÅNTERING

- try/catch
- try {
  - Scanner sc = new Scanner(fil);
  - // Mer kode du vil utføre
- } catch (FileNotFoundException e) {
  - System.out.println(«Error: kunne ikke finne fil!»);
- }



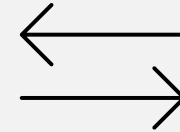
# SKRIVE TIL FIL

- `import java.io.PrintWriter;`
- `PrintWriter fil = new PrintWriter(«filnavn.txt»);`
  - Må også være i en try/catch-blokk!
- `fil.println(«test»);`
- Må også lukkes:
  - `fil.close();`



# KONVERTERING

- Å endre fra en datatype til en annen er litt mer tungvint i Java enn i Python
- Må benytte seg av hjelpeklasser:
  - Integer, Boolean, Character, Double, osv.
- Fra String til int:
  - `int tall = Integer.parseInt(«24»);`
- Fra int til String:
  - `String nummer = Integer.toString(24);`



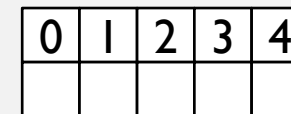
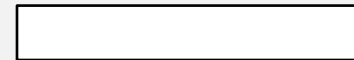
# DATASTRUKTURTEGNING

- Egentlig ganske enkelt
- Kan gjøres for hånd eller på data
  - [draw.io](https://draw.io)
- Pass på å vise de viktigste tingene
  - Trenger nesten aldri å tegne inn metoder eller andre kodelinjer
- Er du usikker på hvordan noe skal tegnes kan du ta egne valg så lenge du også forklarer dem!



# DATASTRUKTURTEGNING - EKSEMPLER

- Objekter – bokser med avrundede hjørner
- Variabler i rektangler
  - Skriv både variabelnavn og type!
- Piler fra variabler til objekter
- Arrayer tegnes som en rekke bokser
  - Gjerne også med indeksnummer
- Statiske variabler i stiplet objektboks
  - Kalles «Klassedatastruktur <objekttype>»



# OPPSUMMERING

- Nå har dere vært gjennom alt dere lærte i Python!
- Pass på at dette sitter ganske godt
- Neste uke går vi over på nytt stoff
  - Vil ikke være like mye nytt hver uke
- Ukas tips:
  - \*-symbolet kan brukes som «hva som helst» i mange sammenhenger
  - `import java.util.*;`
  - `javac *.java;`

