

INF1000 – Uke 1 time 2

Variable, enkle datatyper og tilordning

Oversikt

- Litt repetisjon
 - Datamaskinen
 - Programmeringsspråk
 - Kompilering og kjøring av programmer
- Variabler, deklarasjoner og typer
- Tilordning
- Uttrykk
 - Verdier
 - Evaluering
- Matematiske funksjoner i Java

Repetisjon - Introduksjon

- Forstå problemer og metoder i informatikk
- **Lære å programmere**
- Mye å gjøre.
 - Både ukeoppgaver og obligatoriske oppgaver må gjøres!
- Maskiner og innlogging
- Filer

Repetisjon - Program

- Maskinen har et begrenset sett med funksjoner
 - Legge sammen tall
 - Flytte data
- Et program er en oppskrift for hva maskinen skal gjøre
- Skriver inn programmet i en fil
- Programmer som eksisterer i maskinen oversetter vårt program til instruksjoner ("maskinspråk")
- Kompilering og kjøring

Repetisjon – Eksempel

```

class Repetisjon {
    public static void main(String[] args){
        // Skriver ut en linje.
        System.out.println(
            "I dag snakker vi om variable og uttrykk."
        );
    }
}
    
```

Alt inne i klasser

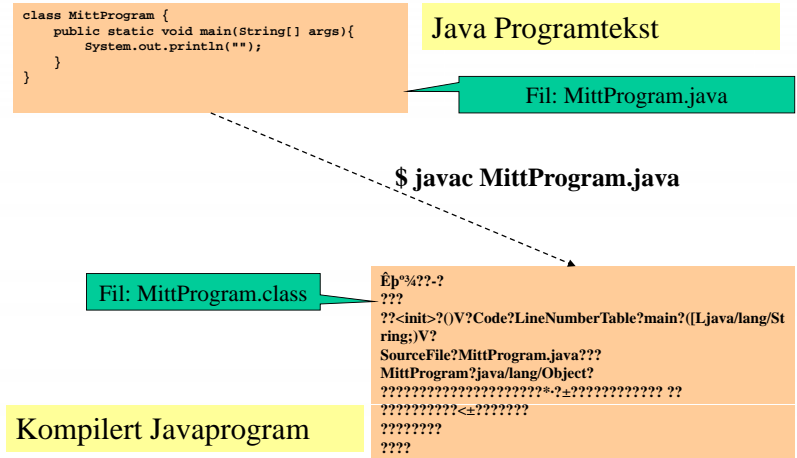
Prosedyren "main"

En kommentar

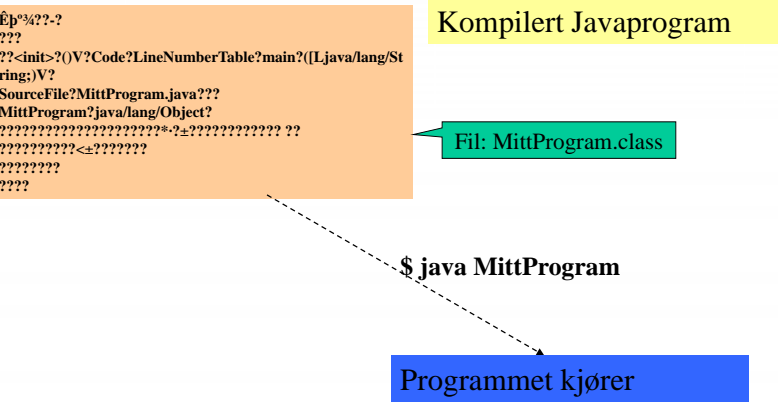
Setninger avsluttes med semikolon

En tekst eller "String"

Kompilering og kjøring



Kompilering og kjøring



Variable – Programmer og data

- Programmet bearbeider data
- Oppgaver som søk, sortering, beregning
 - Summere regning
 - Finne studenten med best gjennomsnittskarakter
 - Finne billigste flybillett
 - Regne ut hvordan været blir i morgen
- Vi må sette av plass til dataene



Variabel – En plass i lageret

- En plass i maskinens lager (minne) kan ses på som
 - en skuff i en kommode, eller som
 - en biloppstillingsplass på en parkeringsplass
- De kan ha forskjellige størrelser avhengig størrelsen på det dataelementet vi skal legge der
- Variable må ha **navn**
 - Slik at vi kan angi i programmet vårt hvilken plass i lageret vi snakker om
- Variable må ha **type**
 - Så vi vet hvor stor plass variabelen tar og hva det er som ligger der



Variable – Deklarasjon

- Deklarasjon
 - Angi navn og type til en variabel
 - Er en setning i programmet

`int radius;`

Type Navn



Variable - Typer

- Velger en type som passer til det vi skal gjøre
- Heltall
 - Alder
 - Antall
- Flyttall
 - Vekt
 - Pris
- Sannhetsverdi
 - Er på?
 - Er tom?
- Tekster
 - Navn
 - Adresse



Variable – Primitive typer

Type	Forklaring	Eksempel
byte	heltall	byte b = 9;
short	heltall	short s = 256;
int	heltall	int i = 76234
long	heltall	long l = 12345690L
float	desimaltall	float f = 2.5F
double	desimaltall	double d = 3.14
char	ett tegn	char c = '9' ;
boolean	sannhetsverdi	boolean b = true;



Variable - String

- Brukes ofte på samme måte som de primitive, men er ikke en primitiv type
- Skrives inne i ""
 - For eksempel: "Fredrik Sørensen", "INF1000"



Variable – Flere i samme setning

```
int lengde, bredde, høyde;  
String navn, adresse;
```

er det samme som

```
int lengde;  
int bredde;  
int høyde;  
  
String navn;  
String adresse;
```



Variable – Tilordning

- Variable har ingen verdi i utgangspunktet
- Verdien settes med en tilordningssetning
- En variabel kan tilordnes flere ganger
- En variabel kan tilordnes hvor som helst etter at den er deklartert



Variable – Tilordning

```
// Deklarasjon  
int lengde;  
String navn;  
  
// Tilordning  
lengde = 34;  
navn = "Fredrik Sørensen";  
  
// Ny tilordning, endrer verdien  
lengde = 38;  
  
// Bruk (avlesing)  
lengde = lengde + 2;
```

Betyr "settes lik". Det er ikke en ligning.



Variable – Tilordning og avlesing

- Opprette en lagerplass i maskinen til et heltall:

```
int lengde;
int svar;
```

- Fylle plassen med en verdi (et heltall):

```
lengde = 38;
```

Tilordner verdien til variabelen

- Avlese (eller bruke) verdien:

```
svar = lengde * 2;
```



Variable – Hva skjedde der?

```
svar = lengde + 2;
```

1. Verdien som ligger i variabelen lengde hentes fram (f.eks 7)
 2. En ny verdi regnes ut, som beskrevet av uttrykket "lengde + 2" (f.eks 7+2=9)
 3. Variabelen svar settes til denne nye verdien (f.eks 9)
- For eksempel vil lengde være 7 og svar 9 etter at setningen er utført



Variable – Må tilordnes før avlesing

- En variabel som ikke er tilordnet kan ikke avleses
- Gir feil når vi forsøker å compilere programmet

```
int lengde;
lengde = lengde * 2;
```

Forsøker å lese en variable som ikke er tilordnet

```
$ javac TilordningAvlesing.java
```

Kompilerer programmet

```
TilordningAvlesing.java:7: variable lengde might not have been initialized
```

```
lengde = lengde + 1;
```

Vi får en feilmelding

```
1 error
```



Variable – Kombinert deklarasjon og tilordning

- Vi kan tilordne en verdi direkte ved deklarasjonen.

```
int lengde = 36;
```

```
String navn = "Fredrik Sørensen";
```

Typer ved tilordning

- Ved tilordning må typen til verdien være den samme som typen til variabelen.

```
int verdi1 = 12345; // OK
int verdi2 = 2.0;   // Gir kompileringsfeil

double verdi3 = 2.0; // OK
double verdi4 = true; // Gir kompileringsfeil
```

Variable – Regne ut areal

```
public class Areal {
    public static void main(String[] args){
        double pi = 3.14;
        double radius = 2.0;

        // Regner ut arealet og skriver ut resultatet
        double areal = pi * radius * radius;
        System.out.println("Arealet av en sirkel med radius " + radius + " er " + areal);

        // Øker arealet med 2 og gjentar oppgaven.
        radius = radius + 2;
        areal = pi * radius * radius;
        System.out.println("Arealet av en sirkel med radius " + radius + " er " + areal);
    }
}
```

Variable – Bruk av String

```
System.out.println("2" + "3");
// Resultat: 23
System.out.println("2" + 3);
// Resultat: 23
System.out.println("2 + 3");
// Resultat: 2 + 3
System.out.println(2 + 3);
// Resultat: 5
```

Når + brukes mellom strenger, betyr det streng-sammenslåing

Når et argument til + er streng, vil de andre konverteres til strenger

Variable – Flere tilordningsoperasjoner

- Det finnes flere tilordningsoperasjoner
- Forkortelser for noe vi skriver ofte

```
lengde = lengde + 2;
```

kan også skrives

```
lengde += 2;
```

Variable – Flere tilordningsoperasjoner

- Det finnes tilsvarende for de andre regneoperasjonene

```
lengde -= 2; // Trekke fra 2
```

```
lengde *= 2; // Multiplisere med 2
```

```
lengde /= 2; // Dele på 2
```

Eksempel – Bytte om variabler

- Anta at vi har disse instruksjonene:

```
int første, andre;  
første = 65;  
andre = 77;
```

- Hvordan kan vi bytte om verdiene i de to variablene?
- Vi forsøker dette:

```
første = andre;  
andre = første;
```

- Vil dette virke?
Se boksen til høyre

Når vi har utført ...	så er verdien til:
<code>første = 65;</code> <code>andre = 77;</code>	<code>første: 65</code> <code>andre : 77</code>
<code>første = andre;</code>	<code>første: 77</code> <code>andre : 77</code>
<code>andre = første;</code>	<code>første: 77</code> <code>andre : 77</code>

Eksempel – Bytte om variabler

- Problemet var at vi mistet den opprinnelige verdien til første når vi utførte

```
første = andre;
```

- Vi kan løse problemet ved å ta vare på den opprinnelige verdien i en tredje variabel. Alle instruksjonene:

```
int første, andre, hjelpevar;
```

```
første = 65;
```

```
andre = 77;
```

```
hjelpevar = første;
```

```
første = andre;
```

```
andre = hjelpevar;
```

- Vi sjekker at det virker

Når vi har utført ...	så er verdien til:
<code>første = 65;</code> <code>andre = 77;</code>	<code>første: 65</code> <code>andre : 77</code> <code>hjelpevar : --</code>
<code>hjelpevar = første;</code> <code>første = andre;</code>	<code>første: 77</code> <code>andre : 77</code> <code>hjelpevar : 65</code>
<code>andre = hjelpevar ;</code>	<code>første: 77</code> <code>andre : 65</code> <code>hjelpevar : 65</code>