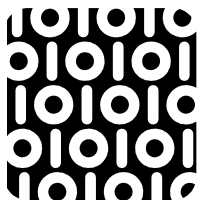
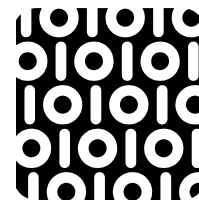


Ohjelmoinnin perusteet

E0007



5 op



Viime viikon kotitehtävät

- Viopen luku1
- Viopen luku 3

B

Muuttujat, tyypit, lauseet ja lausekkeet



Viopen luku 4

Muuttuja (Variable)

- ❑ Muuttujaa voidaan ajatella tallelokerona, johon pian tarvittava tieto laitetaan
- ❑ Ohjelmoinnissa muuttuja talletetaan oikeasti tiettyyn paikkaan tietokoneen muistissa
- ❑ Muuttujaan voidaan tallettaa esim. jokin luku (38) tai merkkijono ("asdf qwerty").

Arvon sijoittaminen muuttujaan

- Javalla ja monilla muilla ohjelmointikielillä muuttujaan x tallennetaan arvo 38 seuraavalla tavalla:

```
X = 38;
```

Huomaa, että '='-merkki tarkoittaa sijoitusoperaatiota Javalla...

- ...kun taas matematiikassa '='-merkki tarkoittaa vertailuoperaatiota (Javalla vertailuoperaatio merkitään "==")

Muuttujista lisää

- Muuttujien nimeäminen
 - Anna muuttujalle aina kunnollinen nimi!
 - Muuttujan nimi ei saa alkaa numerolla
 - Muuttujan nimi ei saa olla varattu sana
 - Tavallinen muuttuja
 - Esim. pallonTilavuus
 - Arvoa tullaan mahdollisesti muuttamaan
 - Alkaa pienellä kirjaimella
 - Sanan vaihtuessa iso kirjain
 - Vrt. luokka PallonTilavuudenLaskija
 - Vakio
 - Esim. MAX_ARVO
 - Arvo asetetaan vain kerran, eikä sitä muuteta enää koskaan
 - Kaikki kirjaimet isoilla
 - Sanojen välissä '_'-merkki eli "*alaviiva*"

Muuttujista lisää

- Java on ns. vahvasti tyypitetty kieli
- Java-kielellä muuttuja on määriteltävä jonkin **tyypiksi** ennen kuin siihen voi tallentaa tietoa
 - "Tähän tallelokeroon saa sijoittaa ainoastaan kokonaisluvun"
 - "Tähän tallelokeroon saa sijoittaa ainoastaan kirjaimen"
 - "Tähän tallelokeroon saa sijoittaa ainoastaan merkkijonon"
- Heikosti tyypitetyillä kielillä, kuten PHP, samaan muuttujaan saa sijoittaa peräjäälkeen eri tyyppisiä arvoja.

Alkeistietotyypit

Tietotyyppi	Selitys	Koko	Arvoalue
byte	tavu	8-bittinen	-128 – 127
short	lyhyt kokonaisluku	16-bittinen	-32768 – 32767
int	tavallinen kokonaisluku	32-bittinen	-2147483648 – 2147483647
long	pitkä kokonaisluku	64-bittinen	$-2^{63} - 2^{63}-1$
float	liukuluku	32-bittinen	IEEE 754 – standardin mukaan
double	tarkempi liukuluku	64-bittinen	IEEE 754 – standardin mukaan
char	Unicode-merkki	16-bittinen	kaikki merkit
boolean	boolean-tietotyyppi	8- tai 16-bittinen	true tai false

Merkkijonotyyppi

- ❑ Alkeistietotyyppien lisäksi oleellinen ohjelmoinnissa tarvittava tyyppi on merkkijonotyyppi
- ❑ Voidaan tallentaa
 - Sana
 - Lause
 - Pitkä teksti
- ❑ Merkkijonotyyppi on oliotyyppi
 - Ymmärretään paremmin vasta myöhemmin

String

- ❑ Kokoelma peräkkäisiä merkkejä
- ❑ Pituutta ei ole rajoitettu

Muuttujan määrittely

`int syntymavuosi;`

`String etunimi;`

`char sotunLoppuosanViimeinenMerkki;`

`boolean onSaanutJaykkakouristusrokotteen;`

`<muuttujan tyyppi> <muuttujan nimi>;`

Muuttujan määrittely

- Usean muuttujan määrittely saman tyyppiseksi yhdellä kertaa

```
String etunimi, sukunimi, osoite;
```

- Muuttujan määrittely ja alkuarvon asetus yhdellä kertaa

```
int syntymavuosi = 2008;
```

Muuttujan arvon alustaminen

- Muuttujalle on annettava arvo ennen kuin sitä käytetään
 - Laskutoimituksissa
 - Tulostuksessa
 - Jne.
- Arvo voidaan asettaa heti määrittelyssä tai määrittelyn jälkeen, kunhan se tehdään ennen muuttujan muuta käyttöä

```
String nimi;  
nimi = "Elisa";  
nimi = "Sakari";
```

```
boolean onSaanutJaykkakouristusrokotteen = true;
```

Nimetty vakio

```
// määritellään muuttuja, johon voi sijoittaa arvon vain kerran  
final double PII;
```

```
// sijoitetaan arvo ensimmäisen ja viimeisen kerran  
PII = 3.14;
```

```
/* tämän jälkeen kääntäjän ilmoittaisi virheestä, jos  
muuttujalle pii yritettäisiin sijoittaa uudelleen arvoa */
```

Muuttujan arvon tulostaminen näytölle

```
public class TulostaVuosi {  
    public static void main(String[] args) {  
  
        int tamaVuosi = 2008;  
  
        System.out.print("Nyt on vuosi ");  
        System.out.println(tamaVuosi);  
    }  
}
```

- System.out.println() tulostaa näytölle **parametrina** annetun tekstin ja laittaa rivinvaihdon perään
- System.out.print() ei laita perään rivinvaihtoa

Tärkeimpiä erikoismerkkejä

Koodi	Merkki
\n	Rivinvaihto
\t	Sarkain
\\	Kenoviiva
\'	Heittomerkki
\"	Lainausmerkki

```
System.out.println("javac löytyy hakemistosta:\nC:\\eclipse\\jdk\\bin\\");
```

Merkkijonojen yhdistäminen eli katenoiminen

```
String koulunNimi = "Laurea";  
String koulunTyyppi = "ammattikorkeakoulu";  
  
char valiviiva = '-';  
  
System.out.println(koulunNimi + valiviiva +  
    koulunTyyppi);
```


char-tyyppisten muuttujien määrittely

```
/* Seuraavat muuttujat voivat saada  
arvokseen esim. jonkin merkeistä  
'n', 'M', 'K', ... */
```

```
char sukupuoli;
```

```
char valinta, merkki;
```

int-tyyppisten muuttujien määrittely

```
// muuttuja syntymävuoden tallentamiseen
```

```
int syntymävuosi;
```

```
/* samassa lauseessa voi samantyyppisiä muuttujia  
   määritellä kirjoittamalla ensin tyypin ja sen jälkeen  
   muuttujien nimet pilkuilla eroteltuna */
```

```
int nauloja, ruuveja, pultteja;
```

```
/* mm. rivinvaihdolla voi lähdekoodia selkeyttää  
   (tai päinvastoin) */
```

```
int kissoja,  
   koiria;
```

float ja double

```
float ihmisenPituus = 1.84f;  
double euroja = 1345435.9873;  
double pituus, leveys, korkeus;
```

Tyypimuunnos

```
int iLuku = 100;  
double dLuku;  
  
//implisiittinen tyypimuunnos  
dLuku = iLuku;  
// dLuku saa arvokseen luvun 100 (liukuluvun muodossa)
```

```
int iLuku;  
double dLuku = 12.34;  
  
//eksplisiittinen tyypimuunnos  
iLuku = (int)dLuku;  
// iLuku saa arvokseen luvun 12
```

Aritmetiikkaa

```
/*  
 * laskutoimituksia numeraalisilla alkeistyypeillä int ja double  
 */  
public class Aritmetiikkaa {  
    public static void main(String[] args) {  
  
        int iLuku1 = 4 + 5;  
        int iLuku2 = 2 * 6;  
        double dLuku1 = 3.3;  
        double dLuku2 = dLuku1 * iLuku1;  
  
        System.out.println(iLuku1);  
        System.out.println(iLuku2);  
        System.out.println(dLuku1);  
        System.out.println(dLuku2);  
        System.out.println("- - -");  
        System.out.println(iLuku2 / iLuku1);  
        System.out.println(iLuku2 / iLuku1 * 1.0);  
        System.out.println(1.0 * iLuku2 / iLuku1);  
        System.out.println("- - -");  
        System.out.println(iLuku1 + iLuku2 / dLuku1 * dLuku2);  
    }  
}
```

Katenointi vai yhteenlasku

```
public class KatenointiVaiYhteenlasku {  
    public static void main(String[] args) {  
        String mj1 = "pää", mj2 = "ry", mj3 = "nä";  
        String hedelmä;  
  
        // tavutetaan sana päärynä ja myös ilman tavuviivoja  
        hedelmä = mj1 + "-" + mj2 + "-" + mj3;  
        hedelmä = mj1 + mj2 + mj3;  
  
        // katenointia  
        System.out.println(3 + " " + hedelmä + "ä korissa");  
        System.out.println("Korissa on " + 2 + 1 + " " + hedelmä + "ä");  
  
        // katenointia ja yhteenlaskua  
        System.out.println("Korissa on " + (2 + 1) + " " + hedelmä + "ä");  
        System.out.println(1 + 2 + " " + hedelmä + "ä korissa");  
    }  
}
```

→ 3 päärynää korissa

→ Korissa on 21 päärynää

→ Korissa on 3 päärynää

→ 3 päärynää korissa

Laskutoimitukset

- $\langle \text{operandi1} \rangle \langle \text{aritmeettinen operaattori} \rangle \langle \text{operandi2} \rangle$
- Sulkeiden käyttö matematiikasta tutulla tavalla
- Esimerkkejä
 - $3 + 1$
 - $x - y$
 - $4 / 2$
 - $2 * (x + 1)$

 - $5/2$
 - $5\%2$

Laskutoimitukset

```
int x = 1;
```

```
int y = 2;
```

```
int summanTulos;
```

```
int kertolaskunTulos;
```

```
summanTulos = x+y;
```

```
kertolaskunTulos = x*y;
```

```
System.out.println(summanTulos);
```

```
System.out.println(kertolaskunTulos);
```


Laskutoimitukset

```
int x = 4;
```

```
int y = 3;
```

```
x = x+2; // x saa arvokseen 6
```

```
y = y*3; // y saa arvokseen 9
```

```
int x = 4;
```

```
int y = 3;
```

```
x += 2; // x saa arvokseen 6
```

```
y *= 3; // y saa arvokseen 9
```

Laskutoimitukset

```
int i = 65;
```

```
int j = 22;
```

```
i = i+1; // i saa arvokseen 66
```

```
j = j - 1; // j saa arvokseen 21
```

```
int i = 65;
```

```
int j = 22;
```

```
i++; // i saa arvokseen 66.
```

```
j--; // j saa arvokseen 21
```

Javan aritmeettiset operaattorit

Operaattori	Selitys
++	Unaarinen kasvatusoperaattori
--	Unaarinen vähennysoperaattori
+	Yhteenlasku
-	Vähennyslasku
*	Kertolasku
/	Jakolasku
%	Jakojäännös (modulo)
+=	Lisäysoperaatio
-=	Vähennysoperaatio
*=	Kertolaskuoperaatio
/=	Jakolaskuoperaatio
%=	Jakojäännösoperaatio