

# Taulukkomuuttujat

- Tavalliseen muuttujaan tallennetaan yksi arvo:  
`String oppilas = "Matt";`
- Taulukkomuuttujaan saadaan monta arvoa:  
`String[] oppilaat = { "Mika", "Petri", "Jouni", "Jarkko", "Mikko" };`
- Yksi taulukon "solu" saadaan hakasulkeiden avulla:
  - `System.out.println(oppilaat[0]);` tulostaa: **Mika**
  - Solut alkavat siis nolasta: 0, 1, 2, ...
- Taulukko on kokoelma avain–arvo-pareja: 0 ja "Mika", 2 ja "Jouni".

# Taulukon esittely

- Vain esittely, hakasuluilla kerrotaan että kyseessä on taulukko. Sen jälkeen pitää varata tila.

```
String[] oppilaat;  
oppilaat = new String[5]
```

– Taulukon koko pitää varata aina, muuten taulukkoa ei voi käyttää.

- Taulukon tilan varaaminen samalla - tarvitaan vain yksi rivi! Käytä tätä!

```
String[] oppilaat = new String[5];
```

- Alustaminen samalla (tai tätä)

```
String[] oppilaat = { "Mika", "Petri", "Jouni", "Jarkko",  
"Mikko" };
```

- Kaikki nämä tekevät viiden solun taulukon, jonka indeksit ovat 0,1,2,3 ja 4
  - voi myös nimittää 5-paikkaiseksi taulukoksi

# Tallentaminen taulukkoon

- Indeksien avulla:

```
oppilaat[0] = "Jukka";
```

```
oppilaat[1] = "Timo";
```

```
jne.
```

```
oppilaat[4] = "Esa";
```

# Moniulotteinen taulukko

a1	a2	a3	a4
b1	b2	b3	b4
c1	c2	c3	c4

- Ylläoleva 4x3-taulukko saadaan määriteltyä seuraavasti:

```
String[][] taulukko =  
{  
    { "a1", "a2", "a3", "a4" },  
    { "b1", "b2", "b3", "b4" },  
    { "c1", "c2", "c3", "c4" },  
};
```

- taulukko[0][0] arvo on "a1"

- taulukko[2][3] arvo on "c4"

- Voidaan myös kirjoittaa erikseen `taulukko[0][0] = "a1"; taulukko[0][1] = "a2";`  
ja niin edelleen... vähän hitaampaa vain noin.

# Assosiatiiviset avaimet (vain PHP, ei Javassa)

- Taulukon avain voi olla myös tekstiä:
  - ```
$opettajat["etunimi"] = "Mika";  
$opettajat["sukunimi"] = "Kähkönen";  
$opettajat["aine"] = "AT";
```
- Array-funktiolla tällaiset määritellään näin:
  - ```
$opettajat = array (  
    "etunimi" => "Mika",  
    "sukunimi" => "Kähkönen",  
    "aine" => "AT" );
```
- Jos halutaan monta opettajaa, voi käyttää moniulotteista taulukkoa:
  - ```
$opettajat[0]["etunimi"] = "Mika";  
$opettajat[0]["sukunimi"] = "Kähkönen";  
$opettajat[0]["aine"] = "AT";  
$opettajat[1]["etunimi"] = "Joku";  
$opettajat[1]["sukunimi"] = "Toinen";  
$opettajat[1]["aine"] = "FY";
```

# Taulukon läpikäynti for- ja foreach-lauseella

- Tulostetaan kaikki oppilaat-taulukon tiedot:

```
for (int i = 0; i < oppilaat.length; i++)  
{  
    System.out.println(oppilaat[$i]);  
}
```

- (PHP:tä, jätä huomiotta) Assosiativisten avainten kanssa for ei kelpaa, tarvitaan foreach:

```
- foreach ( $oppilaat as $avain => $oppilas )  
{  
    echo "$avain: $oppilas";  
}
```

- suomeksi: Käy kaikki ( taulukko AS avain => arvo )

# Taulukon lajitteleminen

- Nopea tapa on käyttää valmista funktiota `sort`
  - `Arrays.sort(oppilaat);`
- Nyt tulostuisi:
  - Jarkko Jouni Mika Mikko Petri
- Taulukon lajittelu omalla koodinpätkällä on harjoitustehtävänä

# Harjoituksia 6

- Ohjelmointitehtäviin: <http://writecodeonline.com/php/> ja USB-Linux
  - Katso myös Taulukko.java
1. Määrittele taulukko, jossa on arvoina lukuja. Tulosta arvot for-lauseen avulla.
  2. Etsi, onko taulukossasi arvo 2. (Vinkki: käy taulukon arvot läpi for-lauseella ja käytä sitten if-lausetta.)
  3. Kirjoita funktio, joka laskee n luvun summan. Tulosta summa. (Vinkki: Lähetä funktiolle taulukkomuuttuja ja käytä hyväksi for- ja count-lauseita.)
  4. Sama kuin 1, mutta ennen tulostamista vaihda taulukon arvojen 1 ja 3 paikkaa keskenään. (Vinkki: tallenna ensin toisen arvo uuteen muuttujaan).
  5. Yksinkertainen tapa järjestää taulukko on käydä jokainen arvo läpi (for-lauseella) ja verrata sitä taulukon seuraaviin arvoihin (toinen, sisäkkäinen for-lause). Jos seuraava arvo on suurempi, vaihdetaan arvojen paikkaa. Järjestä tehtävän 1 taulukko.
  6. Sama kuin 2, mutta avaimena vielä kellonaika. Lisäksi pitää tehdä moniulotteinen taulukko sivun 5 tapaan.