

Teema: Tällä kertaa näkökulmana on valmiin koodin ymmärtäminen ja muokkaaminen

Kertaus toistaiseksi käsitellyistä asioista (Tiedostoon kirjoitus ja tiedostosta lukeminen, poikkeukset, poikkeusten käsittely try catch, merkkijonot ja merkit, taulukot, metodit, toistorakenteet, tulostaminen, valintarakenteet, ohjelmien kirjoittaminen, ohjelmien testaus)

Apua löytyy esim.

<https://www.w3schools.com/java/>

ja <https://docs.oracle.com/javase/tutorial>

1. Luo Eclipseen projekti, lisää sinne luokka ja kirjoita oheinen koodi. Mitä koodi tekee? Analysoi ohjelman toiminta lause lauseelta.

```
public class WhileLoop {  
    public static void main(String[] args) {  
        int i = 1;  
        while (i <= 10){  
            int doubled = i * 2;  
            System.out.println(i + " times two equals " + doubled);  
            i++;  
        }  
        System.out.println("End of program");  
    }  
}
```

2. Korvaa edellisen tehtävän while-rakenne for-rakenteella. Ohjelman toiminnan tulee säilyä ennallaan. Analysoi ohjelman toiminta lause lauseelta.
3. Korvaa edellisen tehtävän 1 while-rakenne do while-rakenteella. Ohjelman toiminnan tulee säilyä ennallaan. Analysoi ohjelman toiminta lause lauseelta.

4.

Suppose you run a small business with three employees. You store the hours worked by each employee in a matrix like the one shown in Table 5-3.

TABLE 5-3: Weekly Hours Worked by Employee

EMPLOYEE	MONDAY	TUESDAY	WEDNESDAY	THURSDAY	FRIDAY
Chris	3	2	8	2	3
Danielle	4	4	4	4	4
Michael	5	5	0	5	5

By using nested `for` loops, you can iterate through all the entries of this matrix, where the position (0,0), first row and first column, refers to the hours worked by Chris on Monday, and position (1,3), second row and fourth column, refers to the hours Danielle worked on Thursday. One `for` loop will refer to the column and the other will refer to the row.

```
int[] [] hoursWorked = {{3,2,8,2,3},{4,4,4,4,4},{5,5,0,5,5}};
String[] employees = {"Chris", "Danielle", "Michael"};
double wage = 8.30;

for (int x = 0; x < hoursWorked.length; x++){ //outer for loop
    System.out.print(employees[x] + " worked ");
    int weeklyHours = 0;

    for (int y = 0; y < hoursWorked[0].length; y++){ //inner for loop
        weeklyHours += hoursWorked[x][y];
    } //close inner for loop

    System.out.println(weeklyHours + " hours at " + wage + " per hour.");
    double weeklyPay = weeklyHours * wage;
    System.out.println("Weekly Pay: " + weeklyPay);
} //close outer for loop
```

The two-dimensional `int` array `hoursWorked` represents the matrix shown in Table 5.3. Each row is a one-dimensional `int` array with five elements. A string array `employees` stores the names of the three employees. A `double` represents the hourly wage paid to employees. Here all employees receive the same pay, but if they are different, there could be a `double` array set up similarly to the array for names.

Mitä ohjelma tekee? Testaa ohjelman kulkua kynällä ja paperilla siten, että kirjaat ylös muuttujien arvojen muutokset ja mahdolliset tulosteet. Sisäkkäiset toistorakenteet saattavat tuntua haastavilta, mutta kynä ja paperi on mainio apuväline niiden ymmärtämiseksi.

5. Ohessa on koodia, jossa on turhaa toistoa. Muokkaa koodi oleellisesti lyhyempään muotoon toiminnan muuttumatta. Mitä ohjelma tekee?

```
int[] sales2014 = {500,720,515,377,400,435,510,1010,894,765,992,1125};
int[] staff2014 = {7,5,5,5,5,6,6,7,7,8,9,9};
int[] salesPerStaff = new int[12];
int totalSales2014 = 0;

salesPerStaff[0] = sales2014[0]/staff2014[0];
salesPerStaff[1] = sales2014[1]/staff2014[1];
salesPerStaff[2] = sales2014[2]/staff2014[2];
salesPerStaff[3] = sales2014[3]/staff2014[3];
salesPerStaff[4] = sales2014[4]/staff2014[4];
salesPerStaff[5] = sales2014[5]/staff2014[5];
salesPerStaff[6] = sales2014[6]/staff2014[6];
salesPerStaff[7] = sales2014[7]/staff2014[7];
salesPerStaff[8] = sales2014[8]/staff2014[8];
salesPerStaff[9] = sales2014[9]/staff2014[9];
salesPerStaff[10] = sales2014[10]/staff2014[10];
salesPerStaff[11] = sales2014[11]/staff2014[11];

totalSales2014 = sales2014[0]+sales2014[1]+sales2014[2]+sales2014[3]
+sales2014[4]+sales2014[5]+sales2014[6]+sales2014[7]+sales2014[8]+sales2014[9]
+sales2014[10]+sales2014[11];
```

6. Tunnilla käytiin läpi esimerkki Tervehdi, jossa käytettiin switch-rakennetta. Tee vastaava ohjelma if-lauseella.
7. Math-luokassa on luokkakohtainen metodi double random(), joka palauttaa satunnaisen reaaliluvun, joka on nolla tai suurempi, mutta pienempi kuin yksi. Tee ohjelma, joka tulostaa näytölle satunnaisen kokonaisluvun suljetulta väliltä 0-100. Käytä metodeja.
8. Lisää edelliseen ohjelmaan osio, jossa satunnaislukuja tuotetaan 120 kappaletta. Miten satunnaisluvut jakautuvat väleille 0-25, 26-50, 51-75, 76-100? Voisiko tässä käyttää switchiä? Bonushaaste: Käytä tehtävässä kaksiulotteista taulukkoa, ihan vaan harjoituksen vuoksi (joutunet vähän säätämään).
9. For-lauseesta on myös lyhyempi muoto. Mitä seuraava ohjelman osa tekee? Tee myös vastaava esimerkki, jossa on arvot 14.90, 129 ja 49.90.

```
String[] nameList = {"Adam Brown","Betsy Dudley","Carl Frank"};

for (String name: nameList){
    System.out.println(name);
}
```

10. Sovella edellistä laskemalla tähän mennessä tekemiesi kotitehtävien määrä (int) ja prosenttiosuus. Tehtävien kokonaismäärä on ollut 10 / kierros.