Kotitehtävä 5 Deadline 12.10. klo 11:45

Maksimipisteet 10

Valitse tehtäviä niin, että saat niistä maksimissaan 10 pistettä

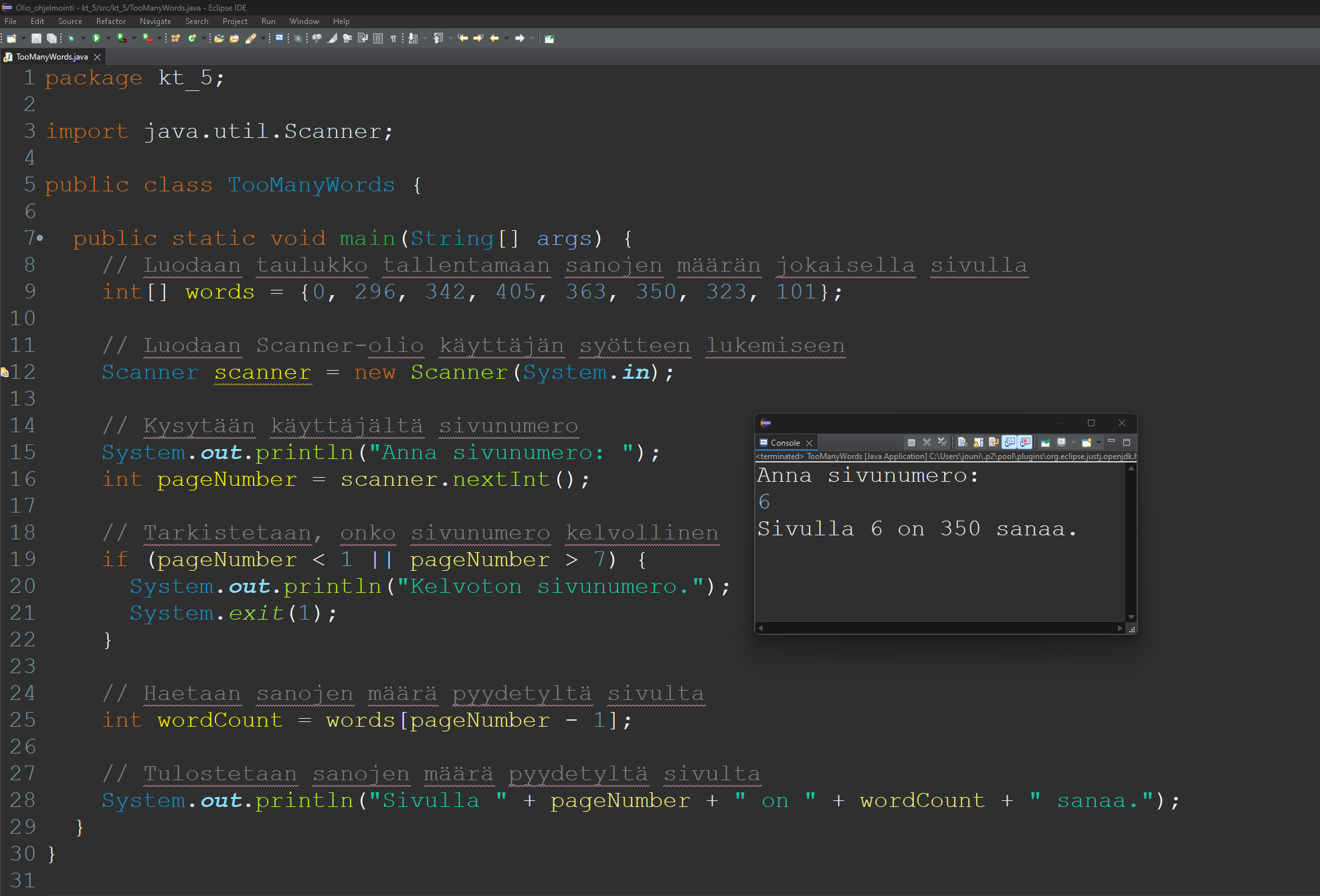
**Tehtävä 1:** Arrayn komponenttien tulostus (1p)

Too Many Words (Try It Out p. 303)

In a fit of shameless narcissism, I made a rough count of the number of words in pages 1 to 7 of this book’s previous edition.

int[] words = {0, 296, 342, 405, 363, 350, 323, 101};

Write a program that prompts the user for a page number and responds with the count of words on that particular page.



**Tehtävä 2:** modify thefor loop (1p)

Spell It Backward (Try It Out p. 303)

Modify the code in Listing 11-1 so that it displays the rooms in reverse room number order (starting with Room 9 and working down to Room 0).

**Tehtävä 3:** komponenttien lisääminen arrayhin ja komponenttien arvojen käyttö toimintoon (2p)

Make Your Marks (Try It Out p. 303-304)

First the user types five numbers on the keyboard. Then the program displays five lines of characters, each line having the same number of asterisks as one of the user’s numbers. A run of the program might look like this:

>> 5

>> 3

>> 9

>> 0

>> 10

A close-up of a number

Description automatically generated

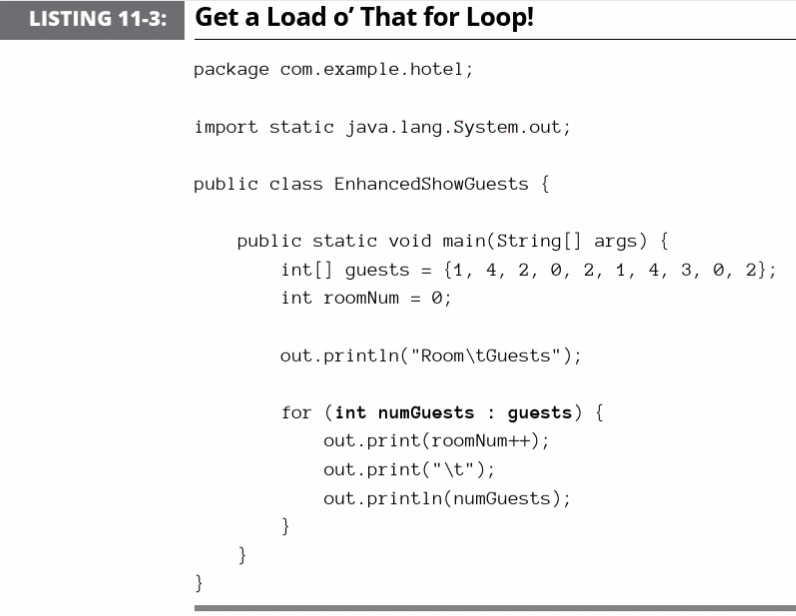
Kuva, joka sisältää kohteen teksti, kuvakaappaus, Multimediaohjelmisto, ohjelmisto

Kuvaus luotu automaattisesti

**Tehtävä 4:** Selitä kuvan avulla (2p)

**Piirrä kuva**

Katso listausta 11-3. Piirrä kuva siitä, miten for-luuppi toimii. Voit myös yrittää selittää loopin omin sanoin. Mitä tapahtuu missäkin luupin vaiheessa? Selitä myös miten enhanced for-loop (parannettu silmukka tai for each toistolause) eroaa tavanomaisesta for loopista.



**Tehtävä 5:** toistolause ja ehtolause (1p)

Find the Biggest (Try It Out p. 306)

Add code to the end of Listing 11-1 so that the program finds the room(s) with the largest number of guests. The program’s new output may look something like this:

4 guests in Room 1

4 guests in Room 6

To do this, have a variable named largestSoFar. Examine the array’s elements one by one and then update the value of largestSoFar whenever you find a new, record-high number.

Kuva, joka sisältää kohteen teksti, kuvakaappaus, ohjelmisto, Multimediaohjelmisto

Kuvaus luotu automaattisesti

**Tehtävä 6:** toistolause (1p)

Find the Total (Try It Out p. 306)

Add code to the end of Listing 11-1 so that the program reports the total number of people staying at the motel. To do this, have a variable named runningTotal. Examine the array’s values one by one, adding each such value to the big runningTotal value.

Kuva, joka sisältää kohteen teksti, kuvakaappaus, ohjelmisto, Multimediaohjelmisto

Kuvaus luotu automaattisesti

**Tehtävä 7:** Selitä (3p)

Kirjassa on sivulla seuraava kommentti: ”In Listing 11-4, the condition roomNum < 10 && guests[roomNum] != 0 can be tricky. If you move things around and write guests[roomNum] != 0 && roomNum < 10 , you can get yourself into lots of trouble.

Lue dokumentti osoitteessa: <https://users.drew.edu/bburd/JavaForDummies6/shortCircuitEval.pdf> ja selitä, miksi saatat joutua vaikeuksiin, jos tarkastelet ensin ehtoa guests[roomNum] != 0.

**Dokumentin mukaan Javassa käytetään lyhytpiiriarviointia (short-circuit evaluation), jossa toisen ehdon testaamista voidaan välttää, jos ensimmäisen ehdon totuusarvo on jo tiedossa. Tässä tapauksessa ensimmäinen ehto on guests[roomNum] != 0, joka tarkistaa, onko annetun huoneen numeron vierasmäärä nollaa suurempi. Toinen ehto on roomNum < 10, joka tarkistaa, onko huonenumero alle 10.**

**Jos ehto guests[roomNum] != 0 on tosi, ehto roomNum < 10 ei koskaan testata. Tämä johtuu siitä, että toisen ehdon totuusarvo on jo tiedossa, ja se on tosi.**

**Esimerkiksi**

**Oletetaan, että guests-taulukossa on seuraavat arvot:**

**guests[0] = 1**

**guests[1] = 2**

**guests[2] = 3**

**guests[3] = 4**

**guests[4] = 5**

**Seuraava koodi tarkistaa, onko huoneen numero 5:ssä vieraita**

boolean hasGuests = guests[5] != 0 && roomNum < 10;

Tämä koodi palauttaa tosi arvon, koska guests[5] != 0 on tosi. Huonenumeroa ei koskaan testata, joten on mahdollista, että roomNum on suurempi tai yhtä suuri kuin 10.

Vaihtoehto

Vaihtoehtoisesti voidaan käyttää seuraavaa koodia:

boolean hasGuests = roomNum < 10 && guests[roomNum] != 0;

Tämä koodi testaa ensin huoneen numeron. Jos huonenumero on alle 10, se testaa sitten vieraiden määrän. Tämä estää ongelman, jossa ehto guests[roomNum] != 0 ei koskaan testata.

Esimerkkikoodi

Seuraava koodi esittelee tämän ongelman ja sen ratkaisun:

Kuva, joka sisältää kohteen teksti, kuvakaappaus, ohjelmisto, Multimediaohjelmisto

Kuvaus luotu automaattisesti

**Tämä koodi tulostaa seuraavan:**

**Kuva, joka sisältää kohteen Fontti, kuvakaappaus, teksti, musta

Kuvaus luotu automaattisesti**

**Yhteenveto**

**Jos tarkastellaan ensin ehtoa guests[roomNum] != 0, voi joutua vaikeuksiin, jos toinen ehto roomNum < 10 ei koskaan testata. Tämä johtuu siitä, että toisen ehdon totuusarvo on jo tiedossa, ja se on tosi.**

**Vaihtoehtoisesti voidaan käyttää toista järjestystä, jossa tarkastellaan ensin huoneen numeron. Tämä estää ongelman, jossa toinen ehto ei koskaan testata.**

**Tehtävä 8:** Selitä (2p)

Käytä kirjan lukua 11 osaa ”When to close a file” hyväksesi ja selitä, miksi tiedostot on hyvä sulkea esim. diskScanner.close() -komennolla mahdollisimman pian lukemisen, kirjoittamisen tai muun tehtävän jälkeen?

Tiedostojen sulkeminen mahdollisimman pian on tärkeää useasta syystä.

* Resurssien vapauttaminen: Kun tiedosto suljetaan, vapautetaan järjestelmän resursseja, kuten muistia ja levytilaa. Tämä voi parantaa suorituskykyä ja vähentää järjestelmän kuormitusta.
* Virheiden välttäminen: Jos tiedostoa ei suljeta, se voi jäädä auki muistiin tai levylle. Tämä voi johtaa virheisiin, kuten tiedostovirheisiin tai muistiongelmiin.
* Turvallisuus: Avatut tiedostot voivat olla haavoittuvaisia ​​hyökkäyksille. Sulkemalla tiedostot varmistat, että niitä ei voi käyttää haitallisesti.

Kirjan luku 11 osa ”When to close a file” antaa seuraavat ohjeet tiedostojen sulkemiseen:

* Sulje tiedosto mahdollisimman pian sen jälkeen, kun se ei enää ole käytössä.
* Sulje tiedostot aina, kun ohjelmasi päättyy.
* Sulje tiedostot, jos ohjelmasi kohtaa virheen.

Seuraavassa on esimerkki siitä, miten tiedosto voidaan sulkea diskScanner.close() -komennolla:

Kuva, joka sisältää kohteen teksti, kuvakaappaus, Multimediaohjelmisto, ohjelmisto

Kuvaus luotu automaattisesti

**Tehtävä 9:** Arrayn komponenttien arvon muuttaminen, toistolause (2p)

Tally Ho! (Try It Out p. 311)

Add code to the end of Listing 11-1 so that the program reports the number of empty rooms, the number of rooms with only one guest, the number of rooms with exactly two guests, and so on. To do this, create an array named howManyRoomsWith. Examine the motel’s rooms one by one. When you encounter an empty room, add 1 to howManyRoomsWith[0]. When you encounter a room with one guest, add 1 to howManyRoomsWith[1]. And so on.

**Tehtävä 10:** array muuttuja objektissa

DOUBLE YOUR PRESSURE (Try it Out p. 318)

In Chapter 9, you create a Student class. Each student has a name and an ID number. For this programming challenge, imagine that each student has five grades — one for each of the five courses the student takes. Each grade is a double value from 0.0 to 4.0 ( 4.0 is the best). A student’s grade point average (GPA) is the average of the student’s five grade values.

In this chapter’s Student class, one of the fields is an array of five double values. Your program finds the student’s GPA and displays it (along with the student’s name and ID number) on the screen.