

Java exercises

Every single exercise gives you one point including the subsections. Thus, the total of points is 26.

1. Write code to calculate the distance between two points in a two-dimensional space. The formula for calculating the distance is $\sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2}$.
2. Write code to display all numbers between n and m which are not divisible by 7 and/or 17.
3. Write code that displays all divisors for a given number.
4. We have the following banknotes: 100, 50, 20, 10 and 5. Make a program to break an amount of money into smallest possible number of banknotes. You can assume the amount is an integer number.
5. Write a program where the user tries to guess what number (between 1 and 100) is stored in the program. The program must give a hint to the user whenever he/she/non-binary guesses wrong. The program stops after the guess is correct.
6. Seconds is given as an integer.
 - a) Write a program to display it in years, days, hours, minutes, and seconds.
 - b) Make a clever testing program for point a).
7. Code a program to display a square of given size (height and width). Only the frame of the square is to be displayed.
8. Code a program to display an isosceles triangle. The height of the triangle is given. If the given height is 3, then the triangle should look like:


```

      *
     **
    ***
   ****
      
```
9. Reverse a given integer number. Example: 2385 becomes 5832.
10. Define a two-dimensional String-array. Fill the array with strings representing the indexes. Example: Index [0][0] includes the string "00" and index [7][4] includes the string "74".
11. Draw 100 random numbers between 1 and 100 and save them in an array.
 - a) Traverse through the array and display the positions where the value is 5 or under. How many such positions should there be?
 - b) Make point a) to a subroutine which gets the array and the value to search as parameters.
12. Draw 100 random numbers between -50 and 50.
 - a) Save the negative ones in one array and the positive ones in another array. How long should the arrays be?
 - b) The length of the arrays must be exactly the same as the numbers stored in them. This means you have to make a dynamic array *by hand*. You may not use an ArrayList or any other data structure present in Java.
13. Reversing a string. Input: "Pig is an animal", Output: "lamina na si giP".
 - a) Write the code in the main method.
 - b) Write a subroutine which gets the input as a parameter and returns the reversed string.
 - c) Write a class to hold a String. Make a method which reverses the String saved in the class. The method changes the inner state of the class.
14. Counting characters.
 - a) Write code to count how many times different characters occurs in a String.
 - b) Make the previous a subroutine. The subroutine gets the String as a parameter and returns the result. What is returned is the most important thing here.
15. Make a class which holds an undefined amount of numbers.

- a) The user must be able to insert and delete numbers from the class.
 - b) The class has a method for returning the average of the numbers.
 - c) The class holds the average of the numbers. This means the class *updates* the average whenever necessary. Jari remembers to explain this.
 - d) After finishing part c) think: why would we ever write such code? It has *side-effects* and as such violates f.ex. the paradigm of functional programming.
16. The formula below can be used to calculate the Fibonacci numbers. We could mathematically prove that the formula is correct, but that is quite difficult, especially if you're not a mathematician. Instead, we can prove it works by coding a program to test the formula. Do that. The formula is copied from:
https://fi.wikipedia.org/wiki/Fibonacci_lukujono.

$$F(n) = \frac{1}{\sqrt{5}} \left(\left(\frac{1 + \sqrt{5}}{2} \right)^n - \left(\frac{1 - \sqrt{5}}{2} \right)^n \right)$$

17. A Fibonacci number is calculated as the sum of the two previous Fibonacci numbers, i.e. $F_i = F_{i-1} + F_{i-2}$. The two first numbers are 0 and 1 and after that the series continues as explained. Make a class to calculate any "nacci" series. See https://fi.wikipedia.org/wiki/Fibonacci_lukujono, for details.