Android Apps – Kotlin introductory exercises

I. The first (and last ☺) reference point: https://kotlinlang.org/docs/home.html

II. To follow the exercises, you may just add the main function in any of your kotlin files. (You may also use one of the online REPLs like: https://play.kotlinlang.org/ but please keep in mind that they are typically some limitations like working with user inputs (but you may always set the variable value directly in the code instead of providing it in run-time, so it is up to you).

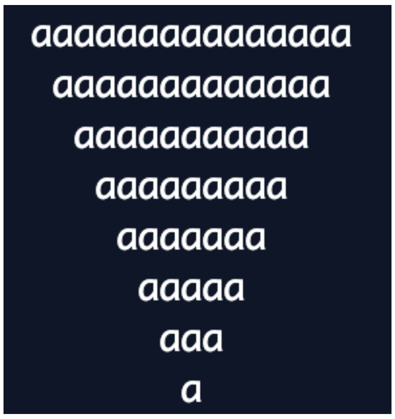
Variable declaration

- In Kotlin we differentiate the variable (mutable value) from the constant declaration.
  - Variables are declared using the keyword var.
  - Constants are declared using the keyword val .
- Please declare and initialize two numerical variables a and b.
- Next print out (function print()/println()) their sum
- Declare the numerical variable c. Assign the sum of a and b to c.
- Print out the string like: The result of 5 + 6 = 11. Please di it:
  - Using the concatenation operator ("+")
  - Using the "inlining" operator ("$)
- Print out the string like: The value of incremented c is 12. Please do it:
  - Using concatenation operator
  - Using "inlining" operator
  - Play with doing it using + and ++ operators.
- Do the same but declare c as a constant.
- Declare the numerical variable "u" of Int type. Set the value of u to the value provided by the user. Do it using readln() and readLine() functions.
  - Run the code providing as the user 5 (as the number) and five as the string. Is there any difference? What kind of?

Control Flow

As the flow control we may use any of the "classic" instructions like: If, switch, for (-in), repeat, while and do-while

- Implement the calculator with providing +-*/ operations.
- Extend the calculator with ^ (power) and ! (factorial) operations
- Implement the power function in three variants with:
  - for-in,
  - while
  - repeat instructions.
- Make your calculator working in a loop as long as the user does not decide to quit.
- Implement the function fib(n:Int) printing out the consecutive Fibonacci numbers in range {1..n}
- Implement the function printing out the up-side-down pyramid with the height of N. like the one presented below for N=8:



```
aaaaaaaaaaaaaaa
 aaaaaaaaaaaaa
  aaaaaaaaaaa
   aaaaaaaaa
    aaaaaaa
     aaaaa
      aaa
       a
```

Implement the function reversing the letters/signs in the string.

Arrays

Create the sample array of Names. Print out all the names in the array.

Extend the before this way that you print out also the position of the name in the array.

Classes and objects

Create Animal class containing Name, Age and Species as the attributes and Move() method

Create a sample animal and make him moving.

Add toString() method to your Animal class printing something like: “Hi. I’m a spider. I’m 6 years old and my name puffy”.

Create classes Fish and Dog both inheriting from Animal class. Override the Move() method thereto inform that the fish is swimming and the dog is running.

Create sample fish and dog, print out their attributes and make them moving.

Create a new fish but declare it as of the Animal type and let them moving. Is it actually moving or swimming?