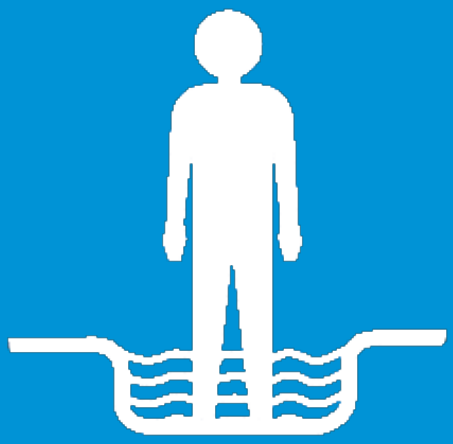
{EPITECH}

PRE-POOL

DAY 04



PRE-POOL

OH MY GIT!
In addition to the tasks below, you must go as far as possible in this game.
Work on it as soon as you have a bit of time, or whenever you need a break in you day!

Conditionals

Task 1.1

Evaluate and explain the following lines:

- ✓ $(42 > 12)$
- ✓ $(12 = 12)$
- ✓ $(12 == 12)$
- ✓ (“hello” == “world”)
- ✓ $(218 >= 118)$
- ✓ (“a”.upper() == “A”)
- ✓ $(1 * 2 * 3 * 4 <= 9)$
- ✓ (“z” in “azerty”)

Task 1.2

Ask an integer to the user:

- ✓ if it’s equal to 42, display “Correct answer”.

Task 1.3

Ask an integer to the user:

- ✓ if it's odd, display "This integer is odd" ;
- ✓ if it's even, display "This integer is even".

From now on, if ever you are stuck on a task, we warmly recommend you to try a no-code tool, such as Flowgorithm.
It generates Python code from which you could get inspiration.

Task 1.4

Ask the user to input a string:

- ✓ if it's "open sesame", display "access granted" ;
- ✓ if it's "will you open, you goddamn !¤*@¡', display"access fucking granted" ;
- ✓ else, display "permission denied".

Flowgorithm?

Task 1.5

Ask the user to input an integer:

- ✓ if it's 42, display "OK" ;
- ✓ if it's smaller or equal than 21, display "OK" ;
- ✓ if it's even, display "OK" ;
- ✓ if this integer divided by 2 is smaller than 21 (excluded), display "OK" ;
- ✓ finally, if it's is odd and greater or equal than 45, display "OK" ;
- ✓ in any other cases, display "You got wrong my poor friend!".

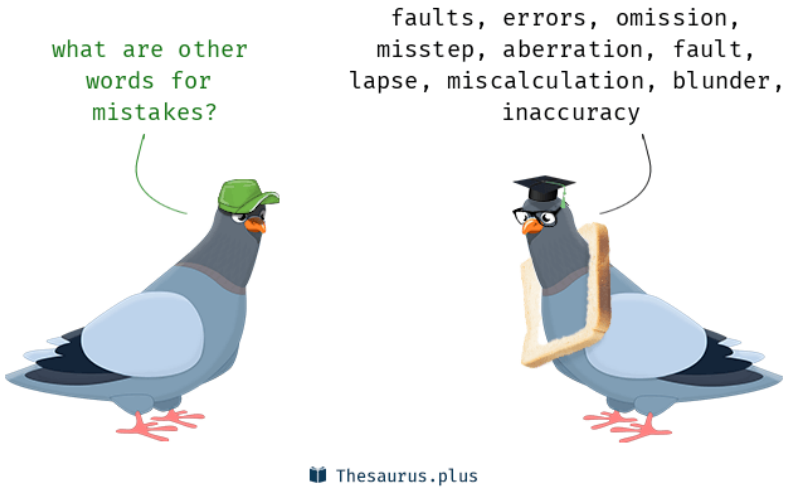
Flowgorithm?!

Task 1.6

Execute and fix the following code:

```
a == 42
b == 41
if a = b
    print("A and B are the sames")
if b =< a
    print("B is equal or lower as A")
if b =! a
    print("B his different from A")
```

As you are at it, also correct the grammar and orthographic mistakes…



Loops

Task 2.1

Display all integers from 1 to 1000.

Flowgorithm!

Task 2.2

Ask the user a string.
Display all the characters of this string twice.
For instance: “taxi” will become “ttaaxxii”.

Task 2.3

Print all integers divisible by 7 from 10 000 to 1.

Task 2.4

For all integers from -30 to 30:

- ✓ if it’s a multiple of 3, display “Fizz”
- ✓ if it’s a multiple of 5, display “Buzz”
- ✓ if it’s a multiple of 3 and 5, display “FizzBuzz”
- ✓ if it does not meet any of the previous conditions, just print the integer itself.

Task 2.5

Generate the lyrics of the song "99 bottles of beer".
The songs ends when there is no more bottles on the wall.

1 bottle is singular…

Task 2.6

Write a program that takes an n integer as input and displays, for each integer from 2 to $n/2$, the list of its multiples strictly smaller than n, in descending order.

For example, if $n = 14$, your program should display:

```
12 10 8 6 4 2
12 9 6 3
12 8 4
10 5
12 6
7
```

For example, if $n = 27$, your program should display:

```
26 24 22 20 18 16 14 12 10 8 6 4 2
24 21 28 15 12 9 6 3
24 20 16 12 8 4
25 20 15 10 5
24 18 12 6
21 14 7
24 16 8
18 9
20 10
22 11
24 12
26 13
```

Flowgorithm!!!

CHALLENGE

Write the shortest possible code that realizes the following:

- ✓ ask the user for an integer and a string ;
- ✓ if this integer is 0, then quit ;
- ✓ if the string contains a vowel, display the integer ;
- ✓ if the integer is greater or equal than 42, display the integer ;
- ✓ else display the string.

Encryption

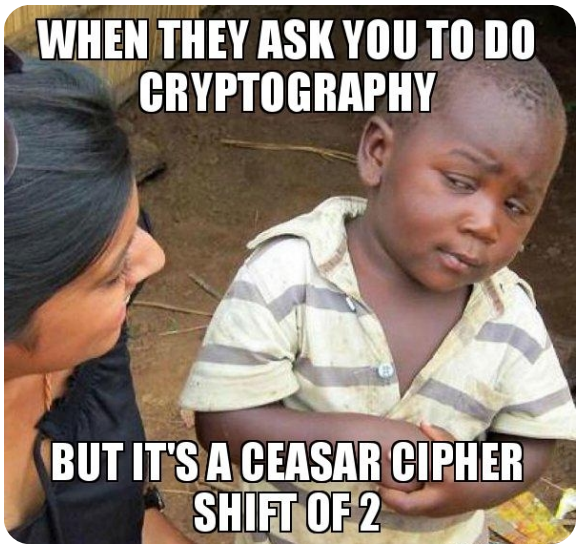
Task 3.1

The Caesar Cipher is a simple encryption technique named after Julius Caesar, who used it to encrypt messages of military significance.
It is a type of substitution cipher in which each letter of a plaintext is replaced by a letter with a fixed number of positions down the alphabet.

The encryption key is an integer between 1 and 25 that determines the shift of the letters. For example, with a key of 3, 'A' would be replaced by 'D', 'B' would be replaced by 'E', and so on. To decrypt the message, the recipient simply uses the same key to shift the letters back.

For example, "Hello world" with a key of 4 gives "Lipps asvph".

Prompt the user for a message to be encrypted and for a key.
Then print the result of the encryption of Caesar Cipher.



Task 3.2

Write a program that can decrypt any Caesar-ciphered text.

Task 3.3

Write a program that can encrypt and decrypt a text using a Vigenere code.

Task 3.4

Write a program that can decrypt an English Vigenere-ciphered text given the length of the key, but without knowing the key.

Day 03, task 3-05

Task 3.5

Write a program that can decrypt an English Vigenere-ciphered text without any information about the key.

v 1.1.1