Guess what!

version 1.0.0

COBRA

CODING CLUB

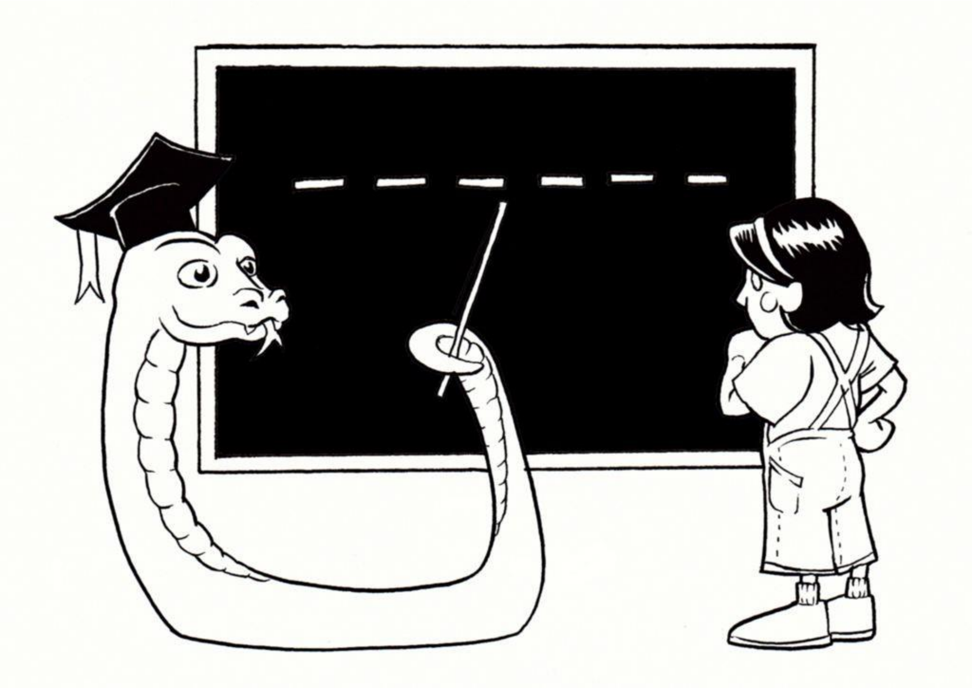
DIVERSITY

by EPITECH

I. Introduction

Audrey is a young schoolgirl who would like to play the hangman game. This famous game consists in finding a word by guessing the letters that compose it.

Usually, this game is played by two or more people, one person chooses a word and the second must guess it. But Audrey is currently at home for lockdown and has no one to play with. So, she decides to design a program to choose a word that she must guess.



Audrey chose Master Python to teach her the game

II. Instructions

- For the installation, please follow the tutorial "Python installation and tools".
- For this project, you will be asked to choose as repository name : cc_hangman.
- Do not forget to push regularly.
- If you have any question, think to ask for help to your neighbor on the right. Then the one on the left. Finally, ask a Cobra (those do not bite) if you are still stuck.
- It is okay to use the internet to find answers or to get information.
- Feel free to make bonuses and add features when your project is finished and validated.

III. The basics of Python

Python is a programming language that appeared in the early 90's, it is characterized by its object-oriented structure like Java but it is much more affordable in its syntax and it is easier to use to do algorithms!

Audrey has never done Python, so she learns about it here. To start, she creates a file that she names pendu.py. In this file, she decides to start her first program to display the content of a variable in the terminal.

IV. Creation of game variables

To start, Audrey will have to choose the word that will have to be guessed and will put it in a variable called "solution" so as not to lose it. It can be changed later. Then she will have to implement a second variable called "turns" to define the possible number of tries for the player. This second variable will define when the game is over.

You will also need to create two empty text variables that will be used to display the word with the letters found and the list of letters found.

```
display = ""
letters_found = ""
```

Now that the variables are created, Audrey can start coding her game!

V. What can this word be?

In the hangman game, at the end of each round, the player is shown the word with the letters that have been found and dashes for the letters that remain to be guessed.

At first, you must loop over the number of letters of the word to guess and fill in the display variable with dashes for each letter. At the end, we will have a result equivalent to this one: " _ _ _ _ _ _ " for a word of six letters.

VI. Let's play a little game together

a. And let us roll!

If Audrey has not used up all her attempts, the game continues. So, she needs a loop to manage the actions to be performed at each turn.

```
while ...                      # The loop needs a condition
    print(...)
    proposal = input(...).lower() #The "lower" is to transform
                                     #the letter into lower case
```

The principle is the following: If the game is not finished, the program displays the word to be guessed with the letters already found and then waits for Audrey to enter one.

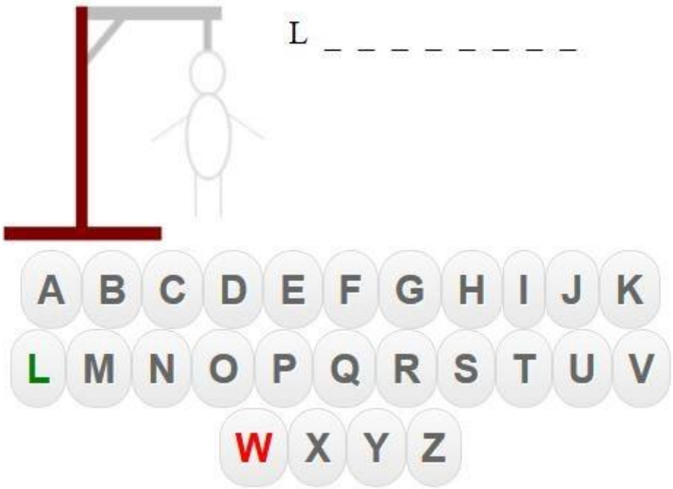
b. Did Audrey find the right letter?

If Audrey's proposal is in the word to be found, it will have to be added to the list of letters found and tell Audrey that the letter found is in the word searched.

```
    if ...          # "If" condition
        letters_found = letters_found + ...
        ...
```

And if the letter is not in the word, you must remove a trial from Audrey and tell her how many are left.

```
    else # Condition "Or"
```



Example of a Hanged Man

c. Is this a victory?

Now we must redefine the display variable because letters could be found in the tour. To do this, we must look at each letter of the solution, if it is in the list of letters found, insert it in the display variable or otherwise display "_" if the letter has not been found.

Finally, if there is no "_" left in the display variable, then you should indicate this by a victory message to Audrey and then exit the game loop.

```
    if "_" not in ...
        ...
```

If the game is over, maybe you should inform Audrey!

VII. Conclusion

Congratulations, you helped Audrey create her game!

To go further, here are some ideas:

- Take a random word from a defined list.
- Have several words separated by a space.
- Draw the hangman according to the number of remaining rounds.
- Display the letters already tested.