CodeInActionLab

Python-00

Summary: *This document is the subject for the Python-00 module of the CodeInActionLab @ aba2020.*

Version:

Contents

Chapter I

Instructions

- Only this page will serve as reference: do not trust rumors.

- Watch out! This document could potentially change up before submission.

- Make sure you have the appropriate permissions on your files and directories.

- You have to follow the submission procedures for all your exercises.

- Your exercises will be checked and graded by your fellow classmates.

- On top of that, your exercises will be checked and graded by a program called Moulinette.

- Moulinette is very meticulous and strict in its evaluation of your work. It is entirely automated and there is no way to negotiate with it. So if you want to avoid bad surprises, be as thorough as possible.

- Moulinette is not very open-minded. It won't try and understand your code if it doesn't respect the Norm. Moulinette relies on a program called `norminette` to check if your files respect the norm. TL;DR: it would be idiotic to submit a piece of work that doesn't pass `norminette`'s check.

- These exercises are carefully laid out by order of difficulty - from easiest to hardest. We `will not` take into account a successfully completed harder exercise if an easier one is not perfectly functional.

- Using a forbidden function is considered cheating. Cheaters get -42, and this grade is non-negotiable.

- You'll only have to submit a main() function if we ask for a program.

- Moulinette compiles with these flags: -Wall -Wextra -Werror, and uses `cc`.

- If your program doesn't compile, you'll get 0.

- You cannot leave any additional file in your directory than those specified in the subject.

- Got a question? Ask your peer on the right. Otherwise, try your peer on the left.

- Your reference guide is called `Google / man / the Internet / ...`.

- Check out the "Python-00" part of the forum on the intranet, or the slack Piscine.

- Examine the examples thoroughly. They could very well call for details that are not explicitly mentioned in the subject...

- By Odin, by Thor ! Use your brain !!!

Norminette must be launched with the *-R CheckForbiddenSourceHeader* flag. Moulinette will use it too.

Chapter II

Foreword

Here is a discuss extract from the `Silicon Valley` serie:

- I mean, why not just use Vim over Emacs? (CHUCKLES)
- I do use Vim over Emac.
- Oh, God, help us! Okay, uh you know what? I just don't think this is
  going to work. I'm so sorry. Uh, I mean like, what, we're going to bring
  kids into this world with that over their heads? That's not really fair
  to them, don't you think?
- Kids? We haven't even slept together.
- And guess what, it's never going to happen now, because there is no
  way I'm going to be with someone who uses spaces over tabs.
- Richard! (PRESS SPACE BAR MANY TIMES)
- Wow. Okay. Goodbye.
- One tab saves you eight spaces! - (DOOR SLAMS) - (BANGING)

. . .

(RICHARD MOANS)

- Oh, my God! Richard, what happened?
- I just tried to go down the stairs eight steps at a time. I'm okay, though.
- See you around, Richard.
- Just making a point.

Hopefully, you are not forced to use emacs and your space bar to complete the following exercices.

Chapter III

Exercise 00 : ft__str__print

	Exercise 00
	My first time

Turn-in directory : *ex*00/
Files to turn in : `ft_str_print.py`
Allowed functions : `None`

- Create a program that takes in input a string and print it.

```
42~ > python 3 ft_str_print.py
Insert your string: Hello
Your string is: Hello
42~ >
```

Chapter IV

Exercise 01 : ft__sum

	Exercise 01
	ft_sum
Turn-in directory : *ex01/*	
Files to turn in : `ft_sum.py`	
Allowed functions : `None`	

- Create a program that takes two input numbers. Sum it and print the result.

```
42~ > python 3 ft_sum.py
Insert your first number: 2
Insert your second number: 4
Result: 6
42~ >
```

```
42~ > python 3 ft_sum.py
Insert your first number: 17
Insert your second number: 25
Result: 42
42~ >
```

Chapter V

Exercise 02 : ft__sum__time

	Exercise 02
	ft__sum__time
Turn-in directory : *ex02/*	
Files to turn in : `ft_sum_time.py`	
Allowed functions : `None`	

- Write a program that show you the time in seconds. You have to get in input the numbers of **hours**, **minutes** and **seconds**. Sum it and print the result!

```
42~ > python3 ft_sum_time.py
Insert hours number: 1
Insert minutes number: 35
Insert seconds number: 4
Result in seconds: 5704
42~ >
```

Chapter VI

Exercise 03 : ft_print

	Exercise 03
	ft_print
Turn-in directory : *ex03/*	
Files to turn in : `ft_print.py`	
Allowed functions : `None`	

- In this file, you will write a program that get in input a string **s** and a number **n**. Print all the characters from the index **n** to the index **size-n**.

```
42~ > python3 ft_print.py
Insert your phrase: My name is Marvin
Insert print range: 4
$> nr $
42~ >
```

```
42~ > python3 ft_print.py
Insert your phrase: I live in Rome
Insert print range: 42
Error: index out of range
42~ >
```

Chapter VII

Exercise 04 : ft_print

	Exercise 04
	ft_print_v2
Turn-in directory : *ex04/*	
Files to turn in : `ft_print.py`	
Allowed functions : `None`	

- In this file, you will write a program that get in input a string **s** and a number **n**. Print all the characters from the index **n** to the index **size-n**.

```
42~ > python3 ft_print.py
Insert your phrase: My name is Marvin
Insert print range: 4
$> name is Mar $
42~ >
```

```
42~ > python3 ft_print.py
Insert your phrase: I live in Rome
Insert print range: 42
Error: index out of range
42~ >
```

Chapter VIII

Exercise 05 : ft__space__print

	Exercise 05		
	ft__space__print		
Turn-in directory : *ex05/*			
Files to turn in : `ft_space_print.py`			
Allowed functions : `None`			

- Write a program that print your phrase at the 20th columns of the console.

- Here's how it should be prototyped :

```
42~ > python3 ft_space_print.py
Insert your phrase: Hello
                    Hello
42~ >
```

Chapter IX

Submission and peer-evaluation

Turn in your assignment in your `Git` repository as usual. Only the work inside your repository will be evaluated during the defense. Don't hesitate to double check the names of your files to ensure they are correct.

> ⚠️ You need to return only the files requested by the subject of this project.