

Thank you for your interest in our Python Developer position, here we outline a set of tasks which we would like you to complete. We have attempted to explain as much as necessary on what needs to be done and what the expected result is.

The main purpose of this process is to give a clear understanding to the candidate of the basic knowledge required for this position, and for the candidate to evaluate from their part whether we are a good fit or not for the next step in their career.

As a developer you will have the freedom to layout and code your project as best as you see fit, however, keep in mind that code structure, naming conventions and formatting will play a big role in the evaluation of your submission.

There are different *levels*, you are required to reach up to *Level 3* for a successful submission. Each level assumes the completion of the previous one, they are not self-contained.

You have 5 days upon receiving this document to complete the levels and submit your work.

Technical Specification of Submission

- 1) Your submission should be a zip file containing *only the necessary files* to run your project. You could also just provide a Github repository which we can clone.
- 2) We will run your code on a clean **venv** Python 3.9.0 generated as

```
fotis→ ~ ▷ python3 --version
Python 3.9.0
fotis→ ~ ▷ python3 -m venv virtualenv-[candidatename]
fotis→ ~ ▷ ls virtualenv-candidatename/bin/activate
virtualenv-candidatename/bin/activate
```

- 3) Upon extracting your submission, we will run *pip -r* to install any requirements which should be listed in the root of your project in a *requirements.txt* file.
- 4) After installing your requirements we will run a single executable which should be capable of managing the different tasks using command line arguments. The name of the arguments, and their restrictions are up to you.
- 5) We will ***NOT*** call you, ask you nor fix any bugs in order to run your program. If it doesn't work, we truly apologize for your time and effort, but we will not consider this application.
- 6) You can provide *some* documentation, however it is **not** required, we will mainly run the commands and look at the code.

Level 1

In true geek form, you are tasked with consuming the “The Star Wars API” (<https://www.swapi.tech/>), and allow the user of the program to search based on the name of any character.

- a) If a character is found, your output should look like this

```
(virtualenv) macbook01 → src ▷ python main.py search 'luke sky'
```

```
Name: Luke Skywalker  
Height: 172  
Mass: 77  
Birth Year: 19BBY
```

- b) If a character is not found

```
(virtualenv) macbook01 → src ▷ python main.py search 'what is star wars?'
```

```
The force is not strong within you
```

Level 2

In order to better understand the universe and the worlds in a *galaxy far away*, provide a way in which we can get more information of the home world for the character we searched for, giving some insights into how that world compares with ours here on earth.

To do so, provide a correlation between an earth day, and an earth year for the characters' world

- a) If the world option is provided return the following

```
(virtualenv) macbook01 → src ▷ python main.py search 'luke sky' --world
```

```
Name: Luke Skywalker  
Height: 172  
Mass: 77  
Birth Year: 19BBY
```

```
Homeworld  
-----
```

```
Name: Tatooine  
Population: 200000
```

```
On Tatooine, 1 year on earth is 0.83 years and 1 day 0.96 days
```

Level 3

It seems that searching for Star Wars characters, has raised some eyebrows, and there is some movement within the *force*.

- a) In order to calm the dark side, make sure that if we have already searched for something, you don't request it again, showing the last time of the search.

[Tech Spec: You can use **any** form of persistence which suits your needs, however we can't install binaries in order to support your persistence method]

```
(virtualenv) macbook01 → src ▷ python main.py search 'luke sky'
```

Name: Luke Skywalker
Height: 172
Mass: 77
Birth Year: 19BBY

cached: 2020-12-19 18:29:48.685572

```
(virtualenv) macbook01 → src ▷ python main.py search 'luke sky' --world
```

Name: Luke Skywalker
Height: 172
Mass: 77
Birth Year: 19BBY

Homeworld

Name: Tatooine
Population: 200000

On Tatooine, 1 year on earth is 0.83 years and 1 day 0.96 days

cached: 2020-12-19 18:42:10.792764

- b) Provide a method to clean the cache

```
(virtualenv) macbook01 → src ▷ python main.py cache --clean  
removed cache
```

Level 4

Provide a visualization which includes

- The searches made
- The result
- Time of the search

Provided that the user *has not* cleaned their cache beforehand.

May the force be with you