09

Do as I did: automated testing

In this class, we learned about Dominique's story and became familiar with the project she had developed so far. We then created our first automated test, learning a little about what tests are and why they are so important for project development.

So, have you gotten your hands dirty yet?

It's time for you to create your first test!

If you have any questions, check the progress of your project by clicking on **Instructor's Opinion.**

Instructor's opinion

1) Initially, we started from the project that Dominique had already developed. To download the complete code, just click **here** (https://github.com/alura-cursos/2622-python-tdd/archive/57de629597dabe71ad125067e9dbedde5babe577.zip) ;

2) When opening the project in the Pycharm code editor, we begin creating a virtual environment by using the following code in the terminal/prompt:

```
python3 -m venv venv
```

COPY CODE

3) But creating a virtual environment is not enough, we also have to activate it, which we can do using the following code in the terminal:

```
source venv/bin/activate
```

COPY CODE

ATTENTION! If you are using the Windows operating system, creating and activating the virtual environment will be a little different. I recommend that you take a look at this **article** (https://www.alura.com.br/artigos/ambientes-virtuais-em-python) .

4) Soon after, we create a new file called `main.py` in the main directory of the project where we will explore the code made by Dominique and better understand how the method works `idade()` .

```
 from codigo.bytebank import Funcionario

lucas = Funcionario('Lucas Carvalho', '13/03/2000', 1000)

print(lucas.idade())
```

COPY CODE

5) We realized that there is a misunderstanding in the business rule applied in the class `Funcionario` . The date of birth must be received containing day, month and year, as opposed to just year.

6) From this conclusion, we had to correct the logic of the method `idade()` .

```
def idade(self):
        data_nasciemnto_quebrada = self._data_nascimento.split
        ano_nascimento = data_nasciemnto_quebrada[-1]
        ano_atual = date.today().year
        return ano_atual - int(ano_nascimento)
```

COPY CODE

7) After finding this *bug* , Dominique's boss thought it would be interesting to have an alternative way to **test** the code. So we created our first automated test in the `main.py` .

```
from codigo.bytebank import Funcionario

def teste_idade():
    funcionario_teste = Funcionario('Teste', '13/03/2000', 111
    print(f'Teste = {funcionario_teste.idade()}')

    funcionario_teste1 = Funcionario('Teste', '13/03/1999', 11
    print(f'Teste = {funcionario_teste1.idade()}')

    funcionario_teste2 = Funcionario('Teste', '01/12/1999', 11
    print(f'Teste = {funcionario_teste2.idade()}')

teste_idade()
```

COPY CODE

8) We finished the class, having learned a little more about what tests are and what types of tests exist.