



Missão

Módulo 9 - Python

Preâmbulo: Neste Módulo, veremos como manipular dicionários. Como arrays, mas mais convenientes!

Versão: 3.1

Sumário

I	Uma palavra sobre esta Missão	2
II	Introdução	3
III	Instruções gerais	4
IV	Exercício 00: seu_livro_de_nomes	5
V	Exercício 01: assuntos_de_família	7
VI	Exercício 02: ajude_seu_professor	9
VII	Exercício 03: pessoas_de_interesse	10
VIII	Entrega e avaliação por pares	12

Capítulo I

Uma palavra sobre esta Missão

Bem-vindo(a)!

Você começará o primeiro Módulo desta Missão de programação de computadores. Nosso objetivo é apresentar a você o código por trás do software que você usa diariamente e mergulhar no aprendizado entre pares, o modelo educacional da 42.

Programação é sobre lógica, não matemática. Ela fornece blocos de construção básicos que você pode montar de inúmeras maneiras. Não existe uma única solução “correta” para um problema — sua solução será única, assim como as soluções de seus colegas.

Rápido ou lento, elegante ou bagunçado, contanto que funcione, isso é o que importa! Esses blocos de construção formarão uma sequência de instruções (para cálculos, exibições, etc.) que o computador executará na ordem que você projetar.

Em vez de fornecer um curso onde cada problema tem apenas uma solução, nós o(a) colocamos em um ambiente de aprendizado entre pares. Você procurará elementos que possam ajudá-lo(a) a enfrentar seu desafio, refiná-los através de testes e experimentação e, finalmente, criar seu próprio programa. Discuta com os outros, compartilhe suas perspectivas, apresente novas ideias juntos, e teste tudo você mesmo(a) para garantir que funcione.

A avaliação por pares é uma oportunidade fundamental para descobrir abordagens alternativas e identificar possíveis problemas em seu programa que você possa ter perdido (considere o quão frustrante uma falha de programa pode ser). Cada revisor abordará seu trabalho de forma diferente — como clientes com diferentes expectativas — dando a você novas perspectivas. Você pode até formar conexões para futuras colaborações.

Ao final desta Missão, sua jornada será única. Você terá enfrentado desafios diferentes, validado projetos diferentes e escolhido caminhos diferentes dos outros — e isso é perfeitamente normal! Esta é uma experiência coletiva e individual, e todos ganharão algo com ela.

Boa sorte a todos; esperamos que você aproveite esta jornada de descoberta.

Capítulo II

Introdução

O que este Módulo mostrará a você:

- Você aprenderá como manipular arrays e suas funções associadas.

Capítulo III


Instruções gerais

A menos que especificado de outra forma, as seguintes regras se aplicam todos os dias desta Piscina.

- Este documento é a única fonte confiável. Não confie em boatos.
- Este documento pode ser atualizado até uma hora antes do prazo de entrega.
- As tarefas devem ser concluídas na ordem especificada. Tarefas posteriores não serão pontuadas, a menos que todas as anteriores sejam concluídas corretamente.
- Preste muita atenção aos direitos de acesso de seus arquivos e pastas.
- Suas tarefas serão avaliadas por seus colegas da Missão.
- Todas as tarefas de shell devem ser executadas usando `/bin/bash`.
- Você não deve deixar nenhum arquivo em seu espaço de trabalho de envio além daqueles explicitamente solicitados pelas tarefas.
- Tem uma pergunta? Pergunte ao seu vizinho à sua esquerda. Caso contrário, tente seu vizinho à sua direita.
- Todas as respostas técnicas de que você precisa podem ser encontradas nas páginas de `man` ou online.
- Lembre-se de usar o servidor do Discord!
- Leia os exemplos completamente, pois eles podem revelar requisitos que não são imediatamente óbvios na atribuição descrição.
- Por Thor, por Odin! Use seu cérebro!!!

Capítulo IV

Exercício 00: seu_livro_de_nomes

	Exercício : 00
Organize esta lista de nomes para mim!	
Pasta de entrega : <i>ex00/</i>	
Arquivos para entregar : <code>seu_livro_de_nomes.py</code>	
Funções ou bibliotecas autorizadas : All	

- Crie um script chamado `seu_livro_de_nomes.py`.
- Este script deve conter um método chamado `array_de_nomes`.
- Este método deve receber um dicionário que associa primeiros nomes com sobrenomes como seu parâmetro.
- Ele deve construir um array contendo os nomes completos das pessoas, com a primeira letra de cada nome capitalizada. O método deve então retornar este array. Consulte o exemplo.
- Por exemplo, o seguinte script:

```
# sua definição de método aqui

pessoas = {
    "jean": "valjean",
    "grace": "hopper",
    "xavier": "niel",
    "fifi": "brindacier"
}

print(array_of_names(persons))
```

Produzirá a seguinte saída:


```
?> ./your_namebook.py
['Jean Valjean', 'Grace Hopper', 'Xavier Niel', 'Fifi Brindacier']
?>
```



Você pode usar o método `capitalize` em Python.

Capítulo V

Exercício 01: assuntos_de_família

	Exercício : 01
Assuntos de Família	
Pasta de entrega : <i>ex01/</i>	
Arquivos para entregar : <i>family_affairs.py</i>	
Funções ou bibliotecas autorizadas : <i>All</i>	

- Crie um script chamado `family_affairs.py`.
- Este script deve conter um método chamado `find_the_redheads`.
- Este método receberá um dicionário como parâmetro, onde as chaves são os primeiros nomes dos membros da família e os valores são suas cores de cabelo.
- O método deve usar a função `filter` para coletar os primeiros nomes de indivíduos com cabelo ruivo em uma nova lista, que ele retornará.
- O resultado deve ser convertido em uma lista antes de ser retornado usando a função `list()`.
- Por exemplo, o seguinte script:

```
# sua definição de método aqui
familia_dupont = {
    "florian": "red",
    "marie": "blond",
    "virginie": "brunette",
    "david": "red",
    "franck": "red"
}

print(find_the_redheads(dupont_family))
```

Produzirá a seguinte saída:


```
?> ./family_affairs.py
['florian', 'david', 'franck']
?>
```




Google Python dictionary `filter`, `keys()`, `list()`

Capítulo VI

Exercício 02: ajude_seu_professor

	Exercício : 02
Dê uma Mão ao Seu Professor	
Pasta de entrega : <i>ex02/</i>	
Arquivos para entregar : <i>help_your_professor.py</i>	
Funções ou bibliotecas autorizadas : <i>All</i>	

- Crie um script chamado `help_your_professor.py`.
- O script deve conter um método chamado `average`.
- Este método deve receber um dicionário como parâmetro, onde as chaves são os primeiros nomes dos alunos e os valores suas pontuações em uma tarefa. O método deve então calcular a média da classe para essa tarefa.
- Por exemplo, o seguinte script:

```
# sua definição de método aqui
class_3B = {
    "marine": 18,
    "jean": 15,
    "coline": 8,
    "luc": 9
}
class_3C = {
    "quentin": 17,
    "julie": 15,
    "marc": 8,
    "stephanie": 13
}


print(f"Average for class 3B: {average(class_3B)}.")
print(f"Average for class 3C: {average(class_3C)}.")
```

Produzirá a seguinte saída:

```
?> ./help_your_professor.py
Average for class 3B: 12.5.
Average for class 3C: 13.25.
?>
```

Capítulo VII

Exercício 03: pessoas_de_interesse

	Exercício : 03
Pessoas que Vale a Pena Conhecer	
Pasta de entrega : ex03/	
Arquivos para entregar : <code>persons_of_interest.py</code>	
Funções ou bibliotecas autorizadas : All	

- Crie um script chamado `persons_of_interest.py`.
- O script deve conter um método chamado `famous_births`.
- Este método deve receber um dicionário como parâmetro, representando figuras históricas. Cada entrada no dicionário é em si um dicionário com as chaves: `:name` e `:date_of_birth`.
- O método deve classificar o dicionário por datas de nascimento e, em seguida, exibir cada entrada (consulte o exemplo abaixo).
- Por exemplo, o seguinte script:

```
# sua definição de método aqui

women_scientists = {
    "ada": { "name": "Ada Lovelace", "date_of_birth": "1815" },
    "cecilia": { "name": "Cecila Payne", "date_of_birth": "1900" },
    "lise": { "name": "Lise Meitner", "date_of_birth": "1878" },
    "grace": { "name": "Grace Hopper", "date_of_birth": "1906" }
}

famous_births(women_scientists)
```

produzirá a seguinte saída:

```
?> ./persons_of_interest.py
Ada Lovelace is a great scientist born in 1815.
Lise Meitner is a great scientist born in 1878.
Cecila Payne is a great scientist born in 1900.
Grace Hopper is a great scientist born in 1906.
?>
```



Google python dictionary & sorted



Você também pode pesquisar os nomes mencionados acima no Google, vale a pena aprender sobre eles!

Capítulo VIII

Entrega e avaliação por pares

- Você deve ter a pasta `missao` na raiz do seu diretório home.
- Dentro da pasta `missao`, você deve ter uma pasta chamada `módulo9`.
- Dentro da pasta `módulo9`, você deve ter uma pasta para cada exercício.
- O exercício 00 deve estar na pasta `ex00`, o exercício 01 na pasta `ex01`, etc.
- Cada pasta de exercício deve conter os arquivos solicitados na tarefa.



Observe que, durante sua defesa, qualquer coisa que não esteja presente na pasta do módulo não será verificada.