



Missão

Módulo 4 - Python

Preâmbulo: Neste Módulo 4, você aprenderá como manipular objetos simples.

Versão: 2.1

Sumário

I	Uma palavra sobre esta Missão	2
II	Introdução	3
III	Instruções gerais	4
IV	Exercício 00: UPCASE_IT	5
V	Exercício 01: Idade	6
VI	Exercício 02: Calculadora	7
VII	Exercício 03: Decimal	8
VIII	Exercício 04: Eu Não Gosto de Vírgulas	9
IX	Exercício 05: Maiúsculas e Minúsculas	10
X	Entrega e avaliação por pares	11

Capítulo I

Uma palavra sobre esta Missão

Bem-vindo(a)!

Você começará um Módulo desta Missão de programação de computadores. Nossa objetivo é apresentar a você o código por trás do software que você usa diariamente e imersí-lo(a) na aprendizagem entre pares, o modelo educacional da 42.

Programação é sobre lógica, não matemática. Ela lhe dá 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 cada um de seus colegas.

Rápido ou lento, elegante ou confuso, desde que funcione, é isso 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 pesquisará elementos que podem ajudá-lo(a) a enfrentar seu desafio, refiná-los por meio 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 pode ter perdido (considere o quão frustrante pode ser uma falha de programa). Cada avaliador 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 diferentes desafios, validado diferentes projetos e escolhido caminhos diferentes dos outros — e isso é perfeitamente aceitável! 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 objetos simples.

Capítulo III

Instruções gerais

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

- Este documento é a única fonte confiável. Não confie em boatos.
- Este documento pode ser atualizado até uma hora antes do prazo de envio.
- As tarefas devem ser concluídas na ordem especificada. As tarefas posteriores não serão avaliadas, 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.
- Toda resposta técnica de que você precisa pode ser encontrada no páginas `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: UPCASE_IT

	Exercício : 00
	Veja em Maiúsculas
	Pasta de entrega : <i>ex00/</i>
	Arquivos para entregar : <i>upcase_it.py</i>
	Funções ou bibliotecas autorizadas : Todas

- Crie um programa chamado *upcase_it.py*.
- Garanta que este programa seja executável (preste atenção às permissões).
- Este programa deve:
 - Solicitar ao usuário que insira uma palavra.
 - Exibir a palavra em letras maiúsculas.
- Exemplo:

```
?> ./upcase_it.py
Me dê uma palavra: banana
BANANA
?>
```



Pesquise por "upcase" no Google.

Capítulo V

Exercício 01: Idade

	Exercício : 01
Recuperar e Modificar um Número	
Pasta de entrega : <i>ex01/</i>	
Arquivos para entregar : <i>age.py</i>	
Funções ou bibliotecas autorizadas : Todas	

- Crie um programa chamado *age.py*.
- Garanta que este programa seja executável.
- Este programa deve:
 - Solicitar ao usuário que insira sua idade.
 - Exibir a idade atual do usuário e sua idade em 10, 20 e 30 anos.

```
?> ./age.py
Por favor, me diga sua idade: 15
Você tem atualmente 15 anos de idade.
Em 10 anos, você terá 25 anos de idade.
Em 20 anos, você terá 35 anos de idade.
Em 30 anos, você terá 45 anos de idade.
?>
```



Pesquise por "conversão de string para inteiro"em Python.

Capítulo VI

Exercício 02: Calculadora

	Exercício : 02
Realizando Operações Básicas	
Pasta de entrega : <i>ex02/</i>	
Arquivos para entregar : calculator.py	
Funções ou bibliotecas autorizadas : Todas	

- Crie um programa chamado **calculator.py**.
- Garanta que este programa seja executável.
- O programa deve:
 - Solicitar ao usuário que insira dois números.
 - Armazenar esses números como valores numéricos em duas variáveis.
 - Exibir o resultado da adição, subtração, divisão e multiplicação desses números.

```
?> ./calculator.py
Me dê o primeiro número: 10
Me dê o segundo número: 2
Obrigado!
10 + 2 = 12
10 - 2 = 8
10 / 2 = 5
10 * 2 = 20
?>
```

Capítulo VII

Exercício 03: Decimal

	Exercício : 03
Trabalhando com Números de Ponto Flutuante	
Pasta de entrega : <i>ex03/</i>	
Arquivos para entregar : float.py	
Funções ou bibliotecas autorizadas : Todas	

- Crie um programa chamado **float.py**.
- Garanta que este programa seja executável.
- Este programa deve:
 - Solicitar ao usuário que insira um número.
 - Determinar se o número inserido é decimal ou não e exibir o resultado.

```
?> ./float.py
Me dê um número: 42
Este número é um inteiro.
?>

?> ./float.py
Me dê um número: 42.00
Este número é um inteiro.
?>

?> ./float.py
Me dê um número: 42.42
Este número é um decimal.
?>
```



Pesquise por "string to float".

Capítulo VIII

Exercício 04: Eu Não Gosto de Vírgulas

	Exercício : 04
	Eu Não Gosto de Vírgulas
	Pasta de entrega : <i>ex04/</i>
	Arquivos para entregar : <i>round_up.py</i>
	Funções ou bibliotecas autorizadas : Todas

- Crie um programa chamado **round_up.py**.
- Garanta que este programa seja executável.
- Este programa deve:
 - Solicitar ao usuário que insira um número.
 - Exibir o número arredondado para cima.

```
?> ./round_up.py
Me dê um número: 41.42
42
?>

?> ./round_up.py
Me dê um número: 42
42
?>

?> ./round_up.py
Me dê um número: 0.001
1
?>
```

Capítulo IX

Exercício 05: Maiúsculas e Minúsculas

	Exercício : 05
A Fusão de Maiúsculas e Minúsculas	
Pasta de entrega : <i>ex05/</i>	
Arquivos para entregar : <i>up_low.py</i>	
Funções ou bibliotecas autorizadas : Todas	

- Crie um programa chamado *up_low.py*.
- Garanta que este programa seja executável.
- Este programa deve:
 - Solicitar ao usuário que insira uma string.
 - Exibir a string com letras maiúsculas alteradas para minúsculas e vice-versa.

```
?> ./up_low.py
Hello World
hELLO wORLD
?>
?> ./up_low.py
aaaaAAA
AAAAaaa
?>
?> ./up_low.py
hello 42
HELLO 42
?>
```

Capítulo X

Entrega e avaliação por pares

- Você deve ter a pasta `missao` na raiz do seu diretório pessoal.
- Dentro da pasta `missao`, você deve ter uma pasta nomeada `modulo4`.
- Dentro da pasta `modulo4`, 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, tudo o que não estiver presente na pasta do módulo não será verificado.