



GABRIEL THIAGO

Python: **a jornada para se tornar um jedi**

PUG-SE

Primeiramente...



Quem sou eu?



Gabriel Thiago da Silva Ferreira

Estudante de Engenharia de Computação (UFS)

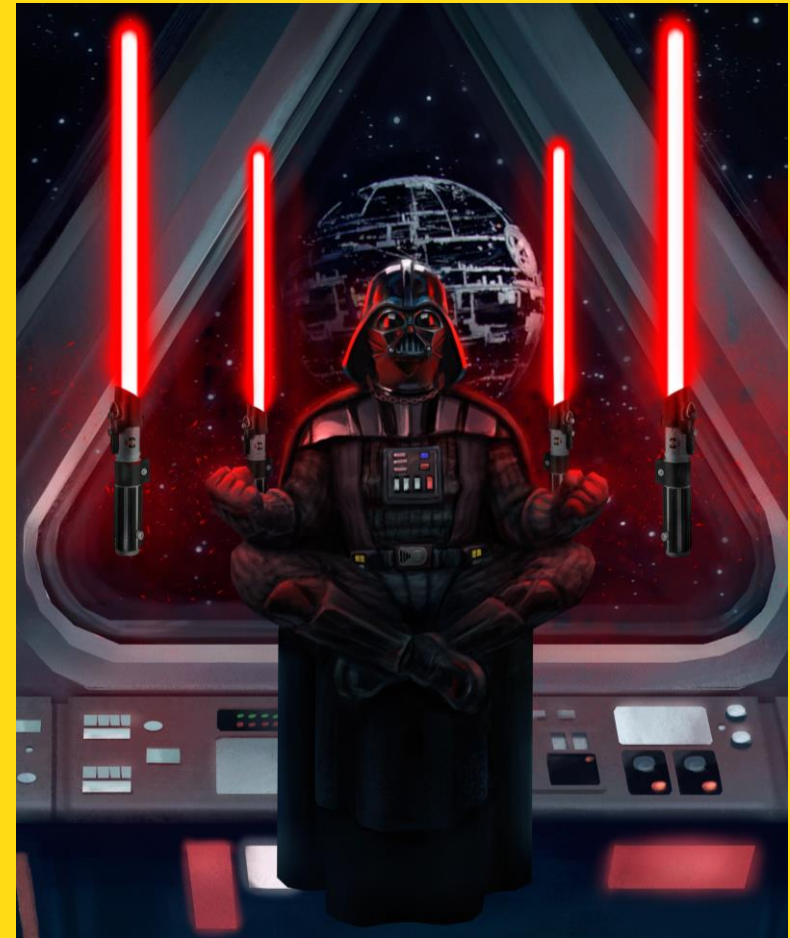
- Apaixonado por Python e por tudo que a linguagem pode fazer
- Um pé no hardware e outro no software
- Web e Mobile

Membro do PUG-SE desde 2017

Primeiramente...



Python e Jedis?





PRIMEIRO PASSO

OS CONCEITOS PRINCIPAIS DA PROGRAMAÇÃO

Operações Aritméticas



$$12 + 7$$

$$100 / 2$$

$$30 * 2$$

$$23 // 2$$

$$14 - 4$$

$$2 ** 4$$

$$32 \% 2$$

Variáveis



Condições



```
se (caixa == 'conteúdo') então:  
    print('Conteúdo Existe!')  
senão:  
    print('Conteúdo não Existe!')
```



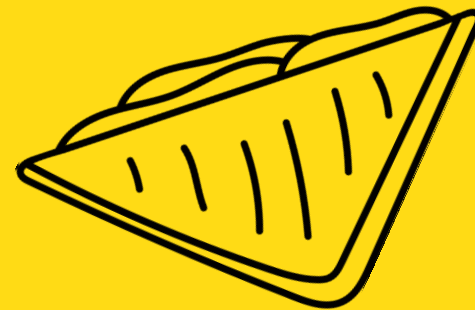
```
se (caixa == 'conteúdo') então:  
    print('Conteúdo Existe!')  
senão:  
    print('Conteúdo não Existe!')
```



Operações Condicionais



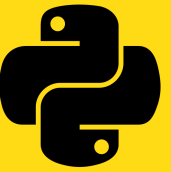
```
se (pão e (queijo e presunto)) então:  
    print('Misto Quente')  
senão se (cuscuz ou não sopa) então:  
    print('Jantar')
```



```
se (não pão e (queijo e presunto)) então:  
    print('Misto Quente')  
senão se (cuscuz ou não sopa) então:  
    print('Jantar')
```



Operações Condicionais



```
se (5 >= 10 ou 20 < 100) então:  
    print('20 menor que 100')  
senão se (-2 <= 5 e 3 != 0) então:  
    print('-2 <= 5 e 3 != 0')
```

20

```
se (5 >= 10 e 20 < 100) então:  
    print('20 menor que 100')  
senão se (-2 <= 5 e 3 != 0) então:  
    print('-2 <= 5 e 3 != 0')
```

-2 3

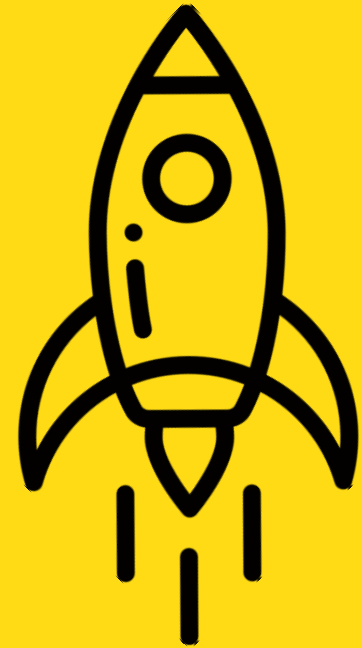
Loops



```
para contador de 1 até 4 faça:  
    print(contador)
```

```
contador = 1  
enquanto contador < 4:  
    print(contador)  
    contador -= 1
```

1
2
3



Funções



```
função escovarDentes(escova, pasta):  
    enquanto pasta na escova:  
        escovar()  
    retorne 'Dentes Escovados!'
```

```
escova = 'nova'  
pasta = 'colgate'  
dentes = escovarDentes(escova, pasta)  
print(dentes)
```



Comentários



```
# Isso aqui é um comentário
```

```
minhaSoma = 2 + 2
```

```
'''
```

```
Isso aqui é um comentário  
de bloco  
'''
```

```
# minhaSoma é uma variável  
print(minhaSoma)
```



SEGUNDO PASSO

PYTHON

O que é Python?



- Criada por Guido Van Rossum em 1989
- Monty Python e a cobra píton
- Linguagem de programação de alto nível
- Propósito Geral
- Motivação principal: SER SIMPLES
- Tipagem dinâmica e forte
- Baseado em Indentação
- Baterias Incluídas
- Organização
- Bibliotecas
- Zen of Python



Zen of Python



- Bonito é melhor que feio
- Simples é melhor que complexo
- Complexo é melhor que complicado
- Legibilidade conta
- Erros nunca devem passar silenciosamente
- Agora é melhor que nunca

Por que Python?



“Por que escolher Python?”

Por que ele é simples

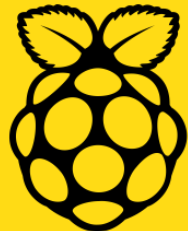
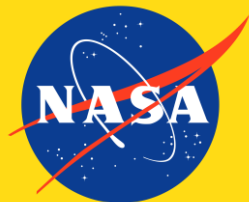
“Pra que serve Python?”

Para o que você quiser

“Qual o tempo de aprendizado?”

Por ter uma curva baixa, o aprendizado é cadenciado e em menos de 1 mês de dedicação já pode fazer alguns projetos bem legais

O que fazer com Python?



- **Como as grandes empresas usam?**
- **Como eu posso usar?**
 - Analisar ações
 - Criar um player de música
 - Inteligência Artificial
 - Jogos
 - Sites
 - Café

Instalando Python



1- Interpretador

- Linux/MAC: Já vem pré-instalado
- Windows: <https://goo.gl/dwmjsP>

2- Editor/IDE

- PyCharm: <https://goo.gl/2geuFy>
- VSCode: <https://goo.gl/PS8wnd>



TERCEIRO PASSO

PROGRAMANDO EM PYTHON

Quebrando a Maldição



```
print('Hello World!!')
```

Conceitos Aplicados



Variáveis

```
varInt = 16  
varFloat = 2.2  
varString = 'Python Jedi'
```

Operações

```
soma = 2 + 2  
mult = varInt * varFloat
```

Conceitos Aplicados



Condições

```
if soma == 4 and mult < 0:  
    print(soma)  
elif mult >= 0:  
    print(mult)  
else:  
    print(not False)
```

Conceitos Aplicados



Loops

```
for i in range(1,4):  
    print(i)
```

```
cont = 1  
while cont < 4:  
    print(cont)  
    cont += 1
```

Conceitos Aplicados



Funções

```
def somar(num1, num2):  
    soma = num1 + num2  
    return soma
```

Comentários

```
'''  
Aqui em cima tem uma função  
'''
```


Recebendo dados



```
nome = input()  
idade = int(input())  
altura = float(input())
```



QUARTO PASSO

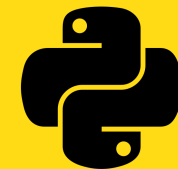
UM POUCO MAIS DE LÓGICA

Estruturas de dados

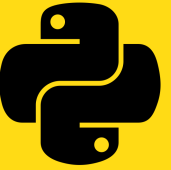


```
lista = ['Gabriel', 'Taisa', 'Lucas']  
dicionario = {'chave': valor}  
tupla = (1920, 1080)
```

Trabalhando com Strings



```
string = str(input())  
string_invertida = string.replace[::-1]  
string_min = string.lower()  
string_mai = string.upper()
```



QUINTO PASSO

O PARADIGMA ORIENTADO A OBJETOS

Classes



```
class Jedi():  
    def __init__(self):  
        self.forca = 100  
    def getForca(self):  
        return self.forca
```

Métodos



```
class Jedi():  
    def __init__(self):  
        self.forca = 100  
    def getForca(self):  
        return self.forca
```

Atributos



```
class Jedi():  
    def __init__(self):  
        self.forca = 100  
    def getForca(self):  
        return self.forca
```




SEXTO PASSO

ONDE TUDO É POSSÍVEL

Bibliotecas



No terminal

```
pip install <módulo>
```

No código

```
import <módulo>
```

ou

```
from <módulo> import <método>
```

Analizando o Twitter



UM EXEMPLO DO QUE FAZER
VAMOS AO CÓDIGO

GABRIEL THIAGO



OBRIGADO!

 @gabriel.f7

 @GTFerreira

 @souogabriel_py