

# Python - Básico

- HELLO WORLD EM PYTHON
- VARIÁVEIS E TIPOS
- OPERAÇÕES E OPERAÇÕES COM OUTROS TIPOS
- CONVERSÃO DE TIPOS

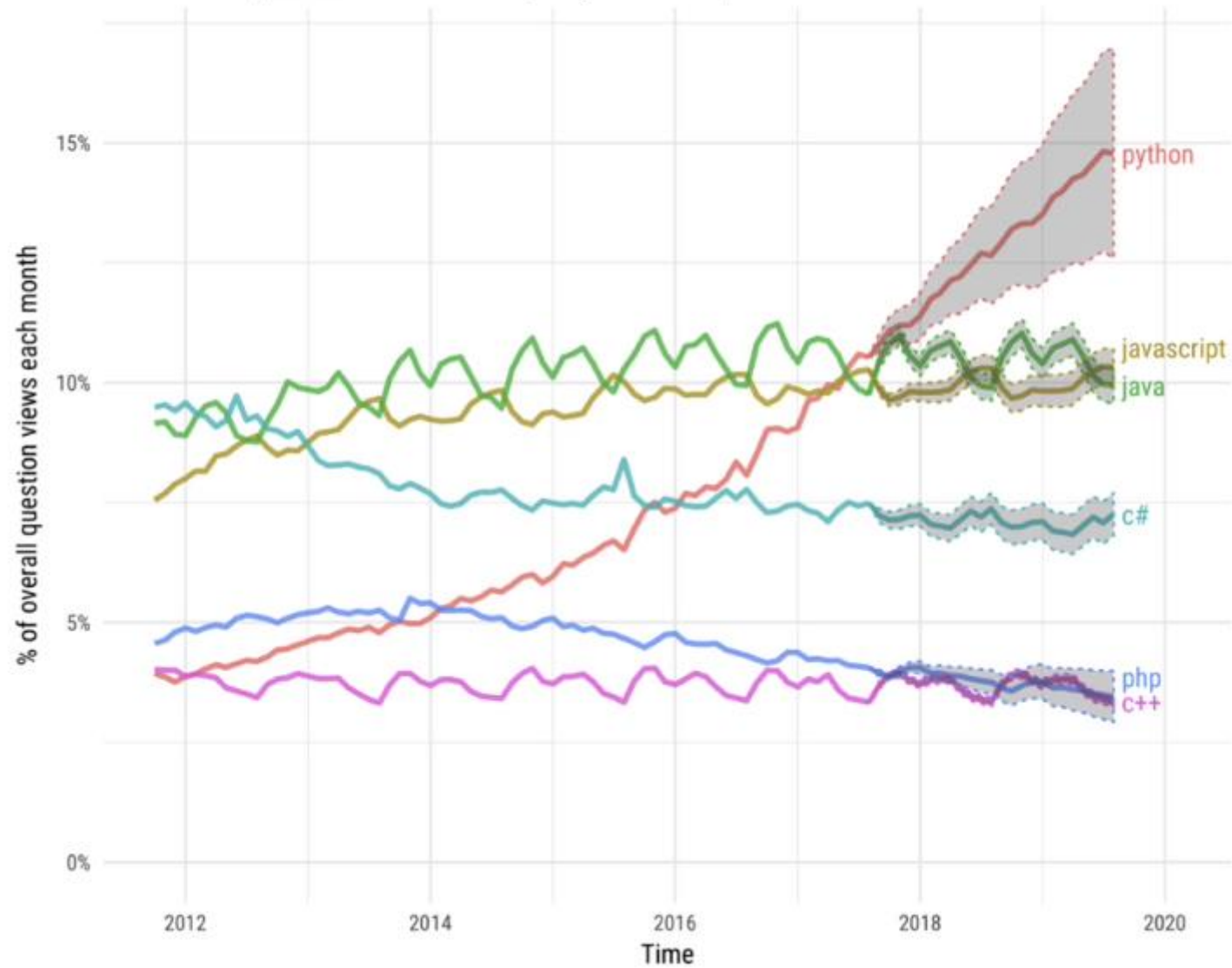
## 1- HISTÓRIA

- Primeiro release em 1991
- Criado por Guido Van Rossum
- Matemático e programador holandês
- Fã do Monty Python
- <https://youtu.be/ghwaliE3Nd8>



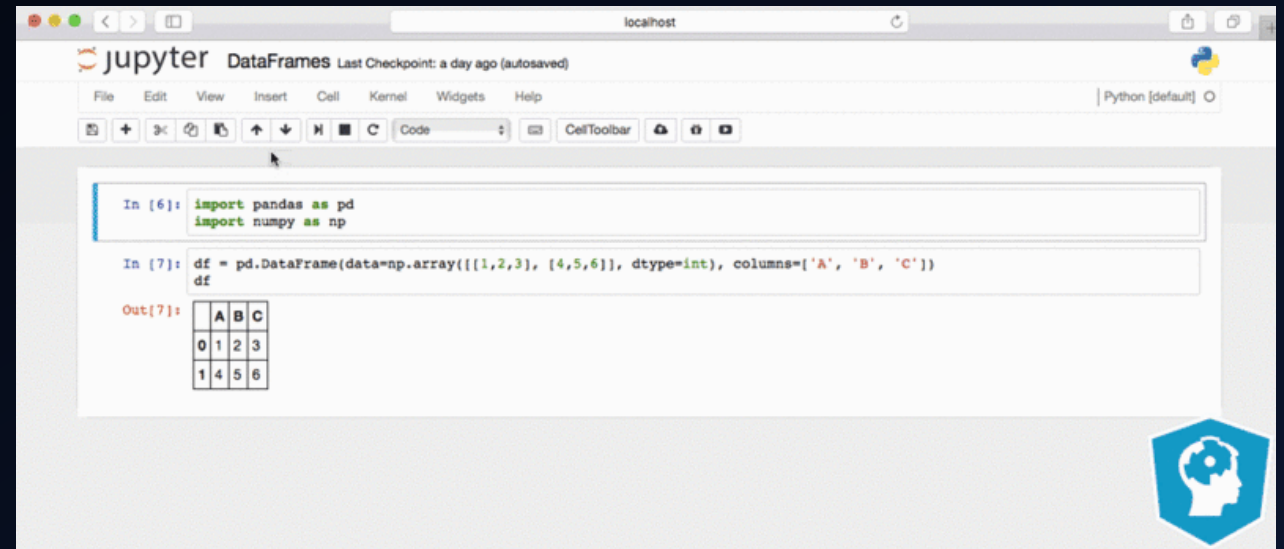
## Projections of future traffic for major programming languages

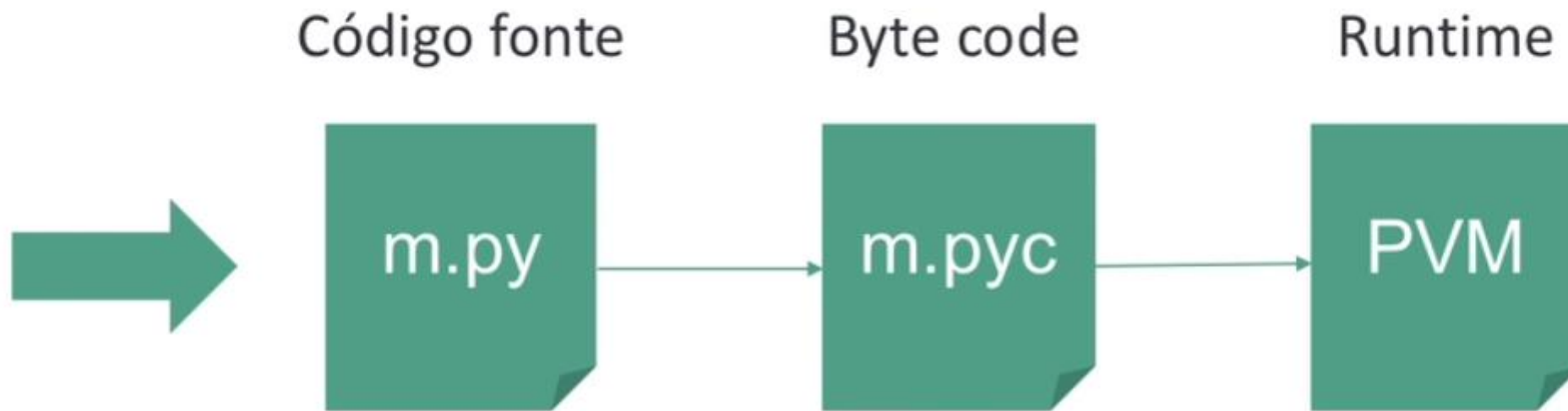
Future traffic is predicted with an STL model, along with an 80% prediction interval.



# CARACTERÍSTICAS

- Fácil aprendizagem e fácil leitura
- Multiplataforma
- Linguagem simples baseada em scripts
- Python é uma linguagem interpretada
- Orientação a objetos
- Interpretação em três níveis
- Shell
- Script
- Interativo (Jupyter)

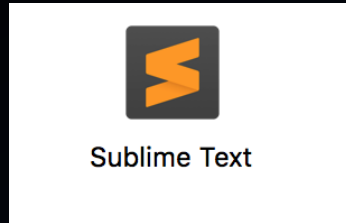




# INSTALAÇÃO

- Já está na maioria das distribuições Linux
- Instalador para Windows e Mac
- Possibilidade de ambiente virtual
- Versão 2 logo será descontinuada
- Estamos na versão 3.8

# INSTALAÇÃO



- Suportado por várias IDEs
- IDE, do inglês Integrated Development Environment ou Ambiente de Desenvolvimento Integrado, é um programa de computador que reúne características e ferramentas de apoio ao desenvolvimento de software com o objetivo de agilizar este processo. É normalmente onde o programador realiza a codificação.



# INSTALAÇÃO



Bibliotecas para Análise de dados

- Python possui uma riquíssima coleção de bibliotecas, que cresce a cada dia, para resolver inúmeros problemas.
- A maior parte dos projetos e programas feitos em Python se utilizam de uma série de bibliotecas.
- As distribuições mais populares de Python, como o Anaconda, já trazem as bibliotecas mais utilizadas



# INSTALAÇÃO



- O Anaconda é uma distribuição gratuita e de código aberto das linguagens de programação Python e R para Computação Científica e Ciência de Dados, que visa simplificar o gerenciamento e a implantação de pacotes.
- <https://www.anaconda.org>

## 1- HELLO WORLD

- Em outras Linguagens:

### #JAVA

```
public class PrimeiraClasse {  
    public static void  
    main(String[] args) {  
        System.out.println("Hello  
world" );  
    }  
}
```

### #C

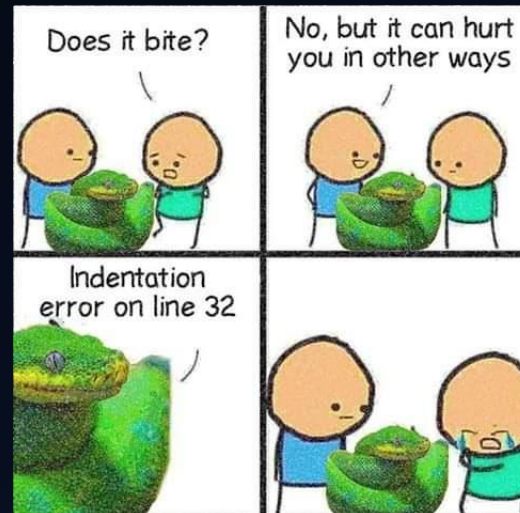
```
int main(){  
    printf("Hello World");  
    return 0;  
}
```

# 1- HELLO WORLD

O programa mais simples em python:

```
print("Hello World")
```

Atenção à indentação  
1 tab ou 4 spaces



# 1- COMENTÁRIOS

```
# ESTA LINHA É UM COMENTÁRIO
```

```
print('Hey python world')
```

```
"""
```

```
    ESTE BLOCO É UM COMENTÁRIO  
    TEXTO QUE NECESSITAR PARA DOCUMENTAR ALGO
```

```
"""
```

```
# Não confundir com atribuição multilinha para strings
```

```
meu_texto = """
```

```
ESTE BLOCO NAO SERÁ TRATADO COMO UM COMENTÁRIO MAS SIM PARA ATRIBUIR UM VALOR STRING  
AQUI POSSO COLOCAR MAIS LINHAS  
MAIS LINHAS AINDA  
E QUANTAS LINHAS EU QUISER COLOCAR  
"""
```

```
print(meu_texto)
```

## 2- VARIÁVEIS E TIPOS

As variáveis em Python não possuem tipos em sua declaração desta forma os tipos são implícitos.

```
mensagem = 'Exemplo de mensagem!'
```

```
n = 253
```

```
pi = 3.141592653589931
```

# Tipos Built-in (Primitivos)

- int - para números inteiros
- str - para conjunto de caracteres
- bool - armazena True ou False
- list - para agrupar um conjunto de elementos
- tupla - semelhante ao tipo **list**, porém, imutável
- dic - para agrupar elementos que serão recuperados por uma **chave**

## 2- VARIÁVEIS E TIPOS

Python é uma linguagem Case-sensitive

```
mensagem = 'Exemplo de mensagem!'
Mensagem = 'Outra mensagem'
```

```
Print(Mensagem)
```

```
Output:  
'Outra mensagem'
```



## 2- VARIÁVEIS E TIPOS

Descobrimos os tipos atribuídos após a declaração.

```
type (mensagem)
<class 'str'>
type (n)
<class 'int'>
type (pi)
<class 'float'>
```

## 2- VARIÁVEIS E TIPOS

```
<Tipo_a_ser_convertido>(<informação>)
```

```
##Variável atribuída
```

```
x = 20
```

```
type(x)
```

```
##Conversão da Variavel
```

```
x = str(x)
```

```
type(x)
```

```
hex(56)
```

```
bin(286)
```

```
float(10)
```

```
int(23.81)
```

## 2- VARIÁVEIS E TIPOS

### Conversão de variável

```
##Variável atribuída
x = 10.5
type(x)

##Conversão da Variável
x = int(x)
type(x)

### Novo valor
print(x)
```

## 2- VARIÁVEIS E TIPOS

### Operações matemáticas

```
# Adição, Subtração
```

```
print(5 + 5)
```

```
print(5 - 5)
```

```
# Multiplicação, Divisão, Módulo, Potência
```

```
print(3 * 5)
```

```
print(10 / 3)
```

```
print(18 % 7)
```

```
print(4 ** 2)
```

```
Print(10 // 3)
```

# STRINGS

Strings em Python são armazenadas como uma sequência indexada de caracteres

A palavra “Python” na verdade é interpretada pelo Python como :

P – y – t – h – o – n

Cada letra ocupa uma posição indexada na sequência acima.

Exemplo:

```
var = 'PYTHON'
```

```
print(var[2])
```

# STRINGS

O tipo String em Python é imutável, não podemos trocar os elementos por um outro valor

Em tipos string como eles são objetos Podemos utilizar métodos que podem executar tarefas

```
_string = "É simple é o que ?"
```

```
_string.replace(' o que ?', ' Python')
```

```
print(_string)
```

objeto.propriedade

objeto.funcao(<parametros>)

- PEP 8 (Python Enhancement Proposal) ?
- <https://www.python.org/dev/peps/pep-0008/>



## 2- VARIÁVEIS E TIPOS

Exercícios: Calculando com variáveis

```
# Crie a variável 'divida'
divida = 100

# Crie a variável juros_ao_ano. Considere juros_ao_ano 1.1

# Calcule o resultado da divida em 7 anos

# Mostre o resultado com a mensagem: "Meu total é: "
```

## 2- VARIÁVEIS E TIPOS

### Resposta:

```
# Crie a variável 'divida'
divida = 100

# Crie a variável juros_ao_ano. Considere juros_ao_ano 1.1
juros_ao_ano = 1.1

# Calcule o resultado da divida em 7 anos
resultado = divida*juros_ao_ano**7

# Mostre o resultado
print("Meu total é: " + str(resultado))
```