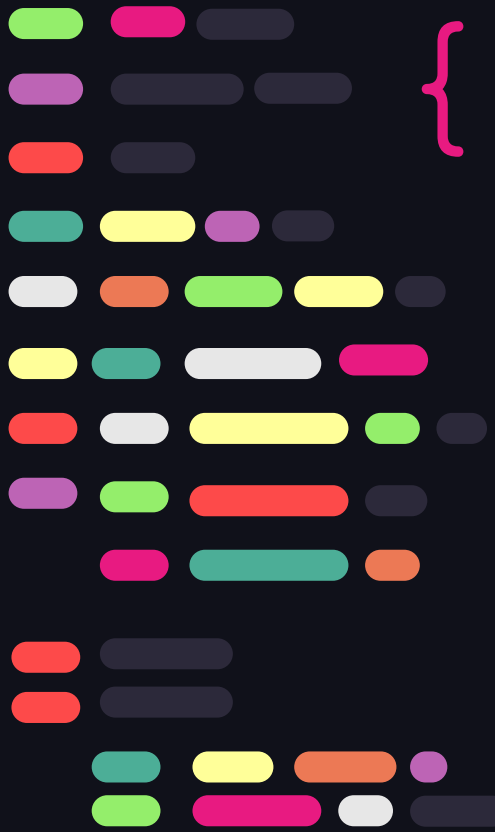
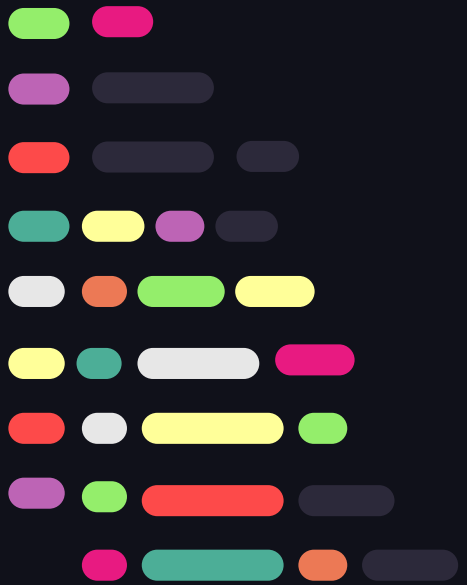


Fundamentos de Python

UFCD 10793 < 50 horas>

Formador: Ricardo Mourão

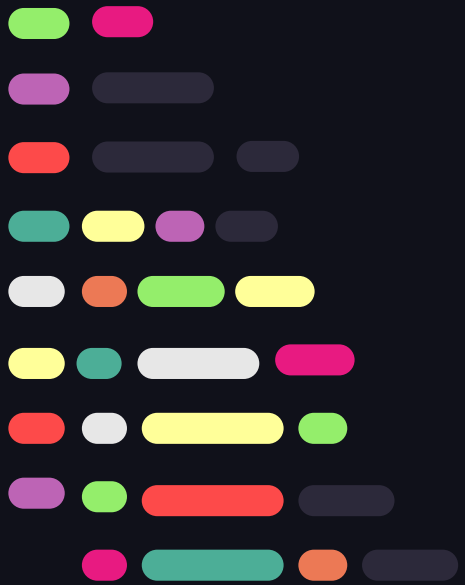
Objetivos Gerais



- Instalar e organizar o ambiente de desenvolvimento.
- Elaborar pequenos scripts em Python.
- Utilizar módulos e bibliotecas.
- Implementar testes unitários.

Formador: Ricardo Mourão

Tabela de Conteúdos



01 Introdução ao Python

< Definição e vantagens sobre outras linguagens >

< Diferenças entre versões >

02 Anaconda e ambientes de desenvolvimento

< IDE's (Spyder e VS Code) >

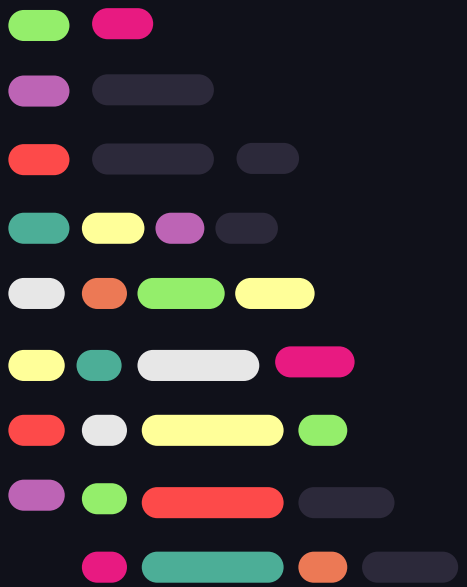
< Introdução ao Jupyter Notebook >

{ Primeiro programa em Python }

{ Python crash course }

Formador: Ricardo Mourão

Tabela de Conteúdos



03 Utilizações de Python

< Tratamento de dados de várias fontes: TEXTO, CSV, SQL, XLS >

< Listas, variáveis e dicionários >

< Controlo do programa (for, while,if) >

< Ficheiros e iteradores >

< Benchmark >

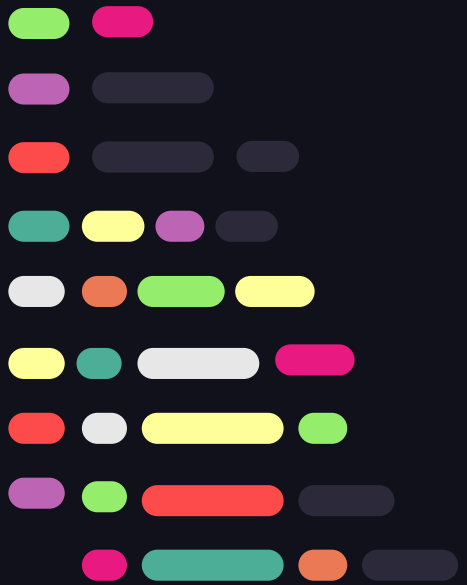
< Profilers de memória e CPU >

< Widgets >

< Geradores >

Formador: Ricardo Mourão

Tabela de Conteúdos



04 Conceitos genéricos de programação em Pyhton

< Tipos de dados >

< Programação condicional >

< Funções >

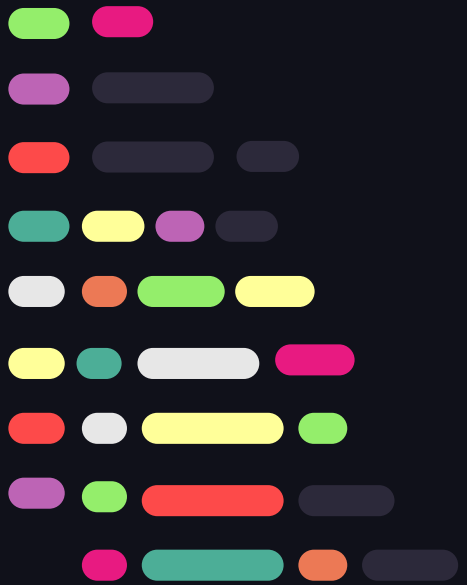
< Iterações >

< Classes >

- { Construtores }
- { Métodos e atributos }
- { Herança }
- { Decoradores }

Formador: Ricardo Mourão

Tabela de Conteúdos



05 Bibliotecas

< Introdução ao Pandas e NumPy >

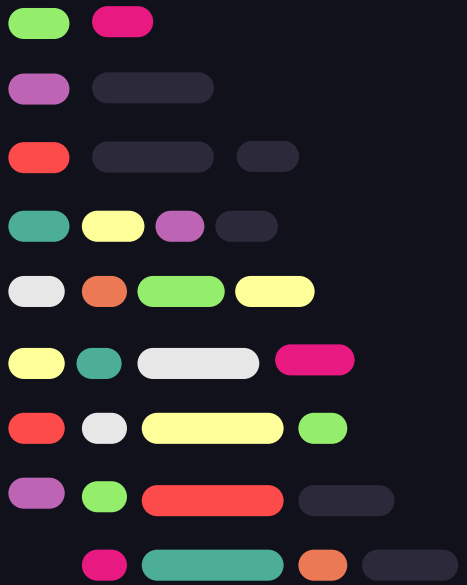
< Análise gráfica com Mapplotlib >

< Importação de SQL, CSV >

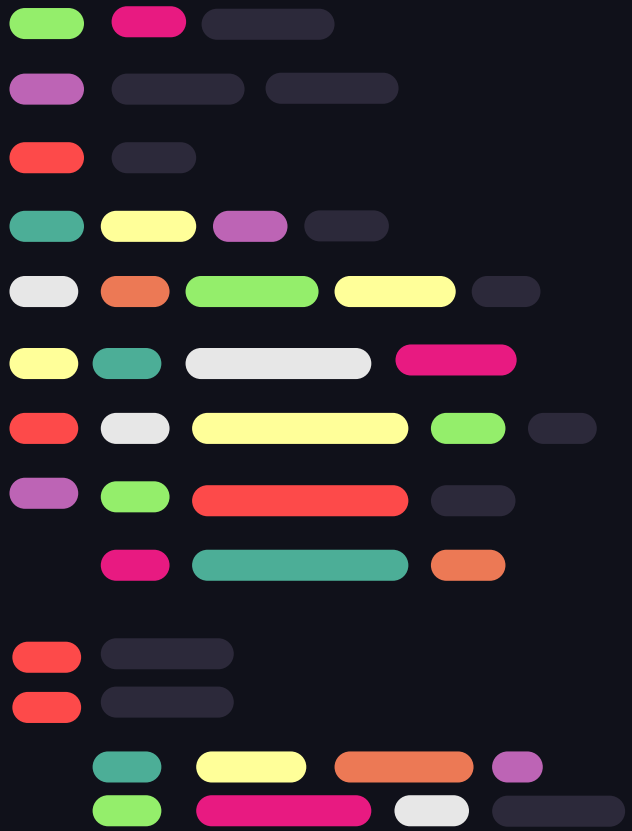
< Testes unitários >

Formador: Ricardo Mourão

Tabela de Conteúdos



Formador: Ricardo Mourão

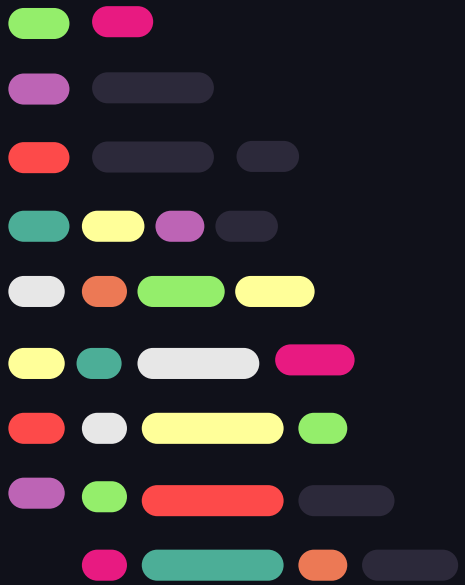


Python}

< Introdução à Computação >

Formador: Ricardo Mourão

Introdução à Computação



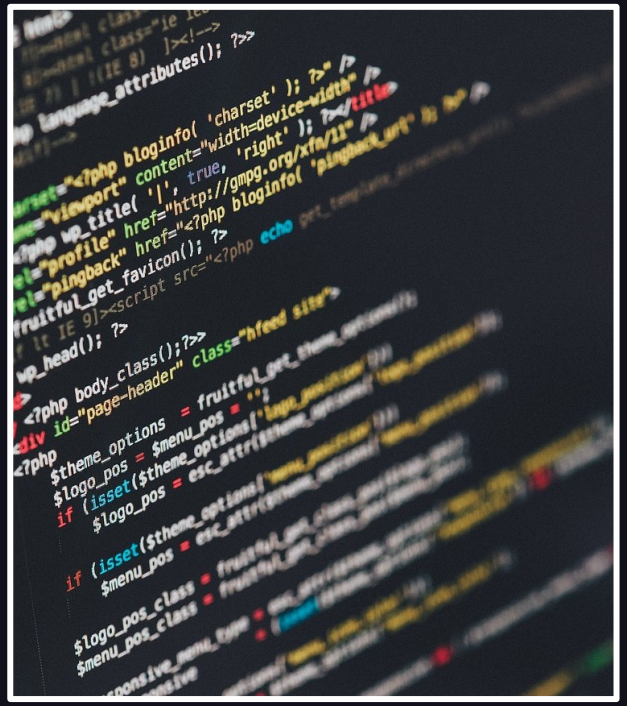
Formador: Ricardo Mourão

Introdução à Computação

O que é a ‘Computação’?{

< A computação pode ser definida como a procura de uma solução para um problema a partir de entradas (inputs), trabalhados através de um algoritmo que apresenta os seus resultados através de saídas (outputs) >

}



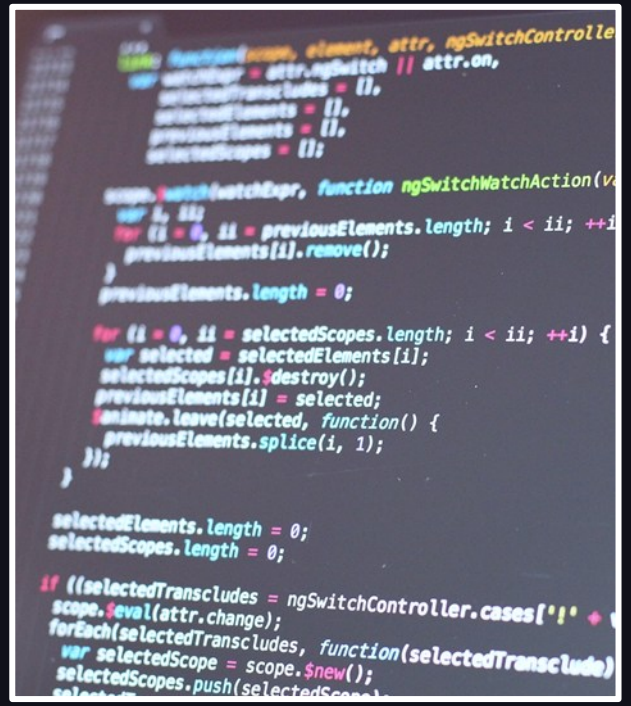
Formador: Ricardo Mourão

Introdução à Computação

O que é um ‘Algoritmo’?{

< Um algoritmo é uma sequência finita e bem definida de passos lógicos e instruções, desenvolvida para resolver um problema específico ou realizar uma tarefa de maneira eficiente e sistemática. >

}



Formador: Ricardo Mourão

Introdução à Computação

O que é ‘Programar’?{

< É o processo pelo qual uma pessoa (programador) escreve, numa linguagem de programação, o código-fonte de um software. >

< Esse código indicará ao programa informático o que fazer, quando fazer e de que forma fazer. >

< O programador encarrega-se de escrever, verificar, averiguar e manter o código-fonte. >

}

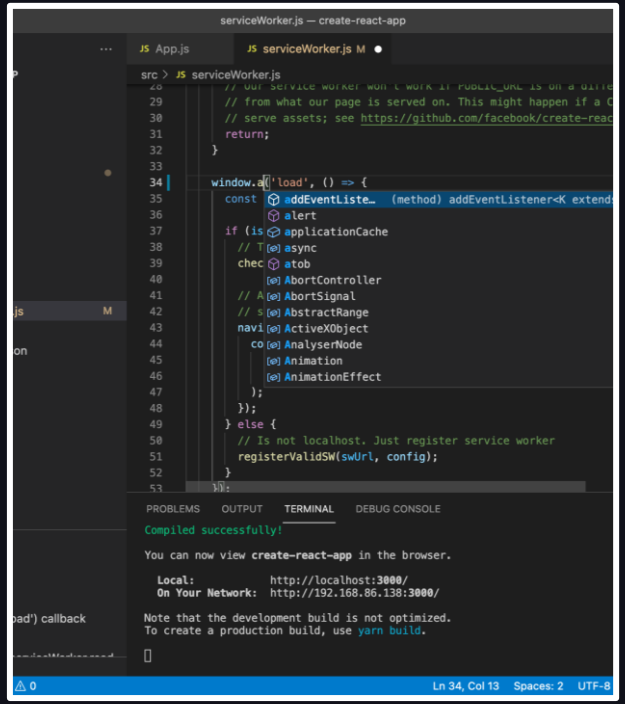
Formador: Ricardo Mourão

Introdução à Computação

O que é uma 'IDE'?{

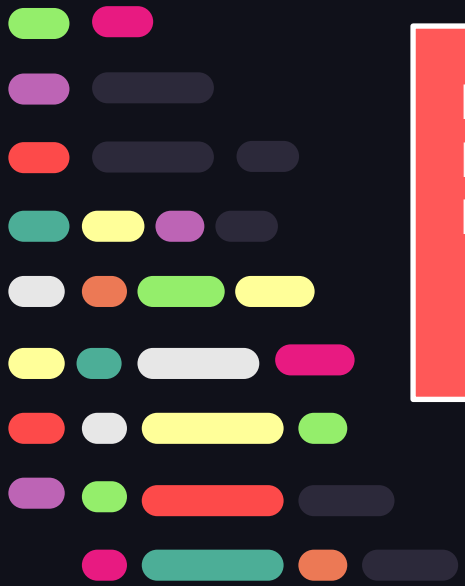
< Uma IDE (Integrated Development Environment, ou Ambiente de Desenvolvimento Integrado) é um software que oferece um conjunto de ferramentas e recursos facilitadores para programadores, permitindo escrever, editar, compilar, depurar e executar código numa única interface. >

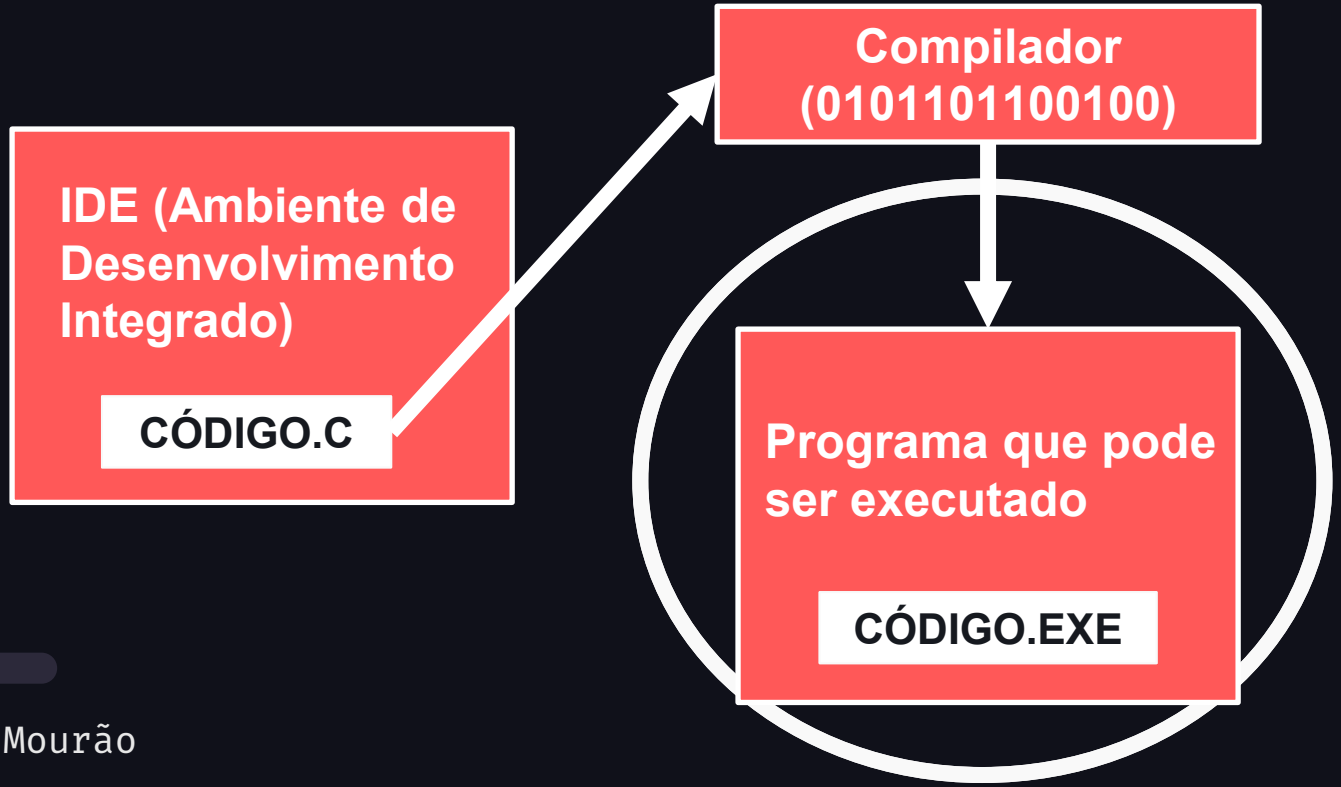
}



Formador: Ricardo Mourão

Introdução à Computação

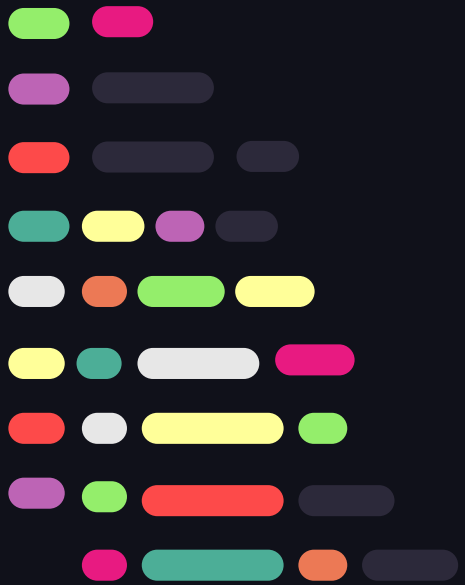




Formador: Ricardo Mourão

Introdução à Computação

No entanto, em Python:



Também podemos criar o código numa IDE, mas o guardamos como codigo.py.

Não é necessário compilar. O Python faz isso 'por trás das cortinas', transformando o código em algo chamado "bytecode".

Não é criado um ficheiro .exe. Pois é possível rodar o script diretamente com o interpretador Python.

OU SEJA

No Python, escreve-se, guarda-se e já se pode rodar. Não tem aquela etapa de compilação para gerar um arquivo executável como no C. O interpretador Python trata de tudo.

Formador: Ricardo Mourão

01 {..

Introdução ao Python

..}

Formador: Ricardo Mourão

O que é o Python?

História do **Python**

Python foi criado por Guido van Rossum em 1989, enquanto ele trabalhava no Centrum Wiskunde & Informatica (CWI) na Holanda.

A linguagem foi inspirada noutras linguagens como ABC e Modula-3, mas com foco em ser mais fácil de ler e escrever.

Ao longo dos anos, Python foi adotada por várias organizações e programadores, crescendo em popularidade e utilidade.

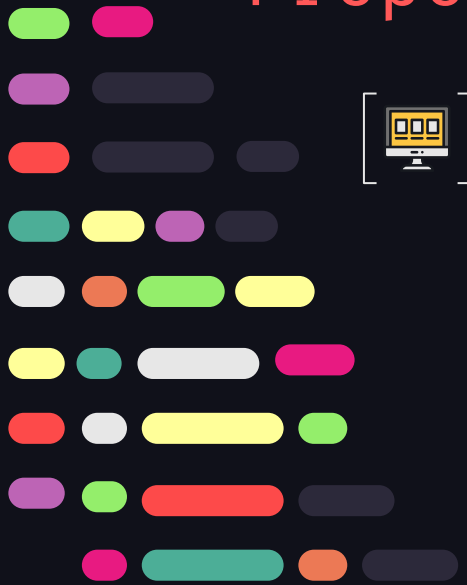
Formador: Ricardo Mourão

O que é o Python?

Guido van Rossum

Formador: Ricardo Mourão

O que é o Python?



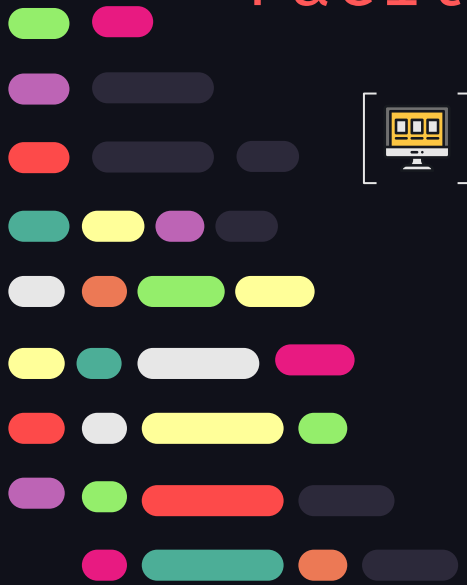
Propósito **Geral**

Ao contrário de linguagens que foram criadas com um fim específico, como R para estatísticas ou HTML para marcação de texto em páginas web, Python é uma linguagem de propósito geral.

Isso significa que ela pode ser usada numa ampla variedade de aplicações, desde o desenvolvimento web até a análise de dados, machine learning e automação.

Formador: Ricardo Mourão

O que é o Python?



Facilidade de **Uso**

Um dos principais atrativos de Python é a sua simplicidade.

A linguagem foi projetada com legibilidade em mente, utilizando uma sintaxe clara e concisa.

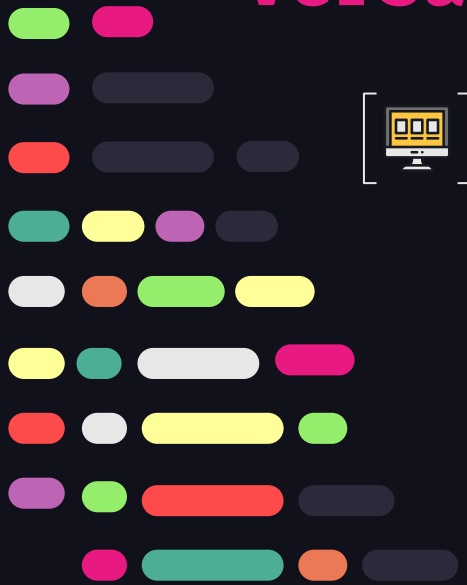
Isso facilita tanto para novatos aprenderem programação quanto para equipas de desenvolvimento colaborarem em projetos complexos.

A ideia é que o código Python é quase como inglês legível, o que torna fácil entender o que um programa está a fazer apenas olhando para o código.

Formador: Ricardo Mourão

O que é o Python?

Versatilidade



Python é notável pela sua versatilidade.

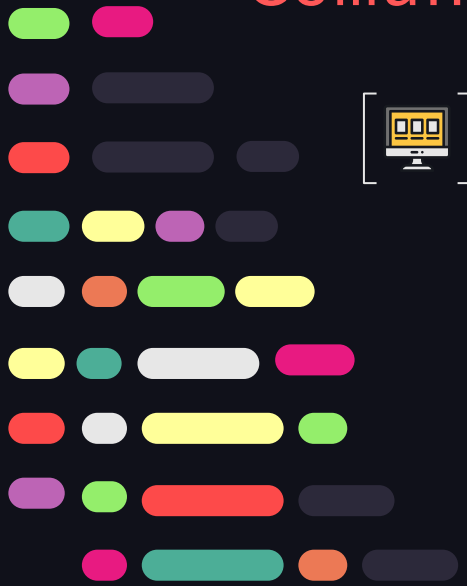
Ele tem uma extensa biblioteca padrão, bem como um ecossistema de bibliotecas de terceiros para praticamente qualquer coisa.

Desde desenvolvimento web (Django, Flask) até ciência de dados (Pandas, NumPy) e machine learning (TensorFlow, scikit-learn).

Essa vasta gama de bibliotecas torna Python uma "linguagem versátil", que pode unir diferentes sistemas e disciplinas.

Formador: Ricardo Mourão

O que é o Python?



Comunidade **Ativa**

A comunidade Python é uma das mais ativas e envolvidas em todo o mundo da programação.

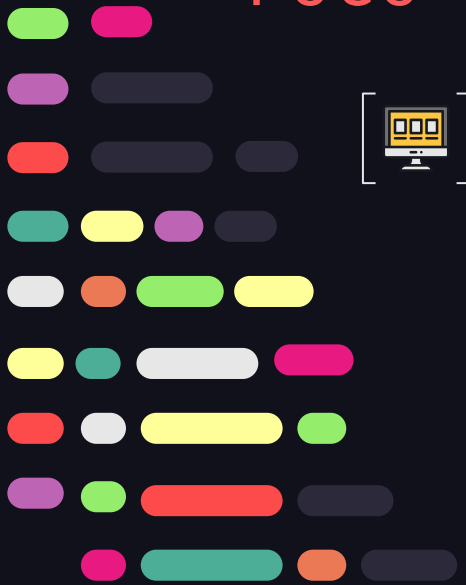
Isso reflete a vasta quantidade de recursos disponíveis, como bibliotecas, frameworks e tutoriais.

A comunidade também é responsável pela organização de eventos como a PyCon, uma conferência anual que reúne entusiastas de Python de todo o mundo.

Formador: Ricardo Mourão

O que é o Python?

Foco **Educativo**



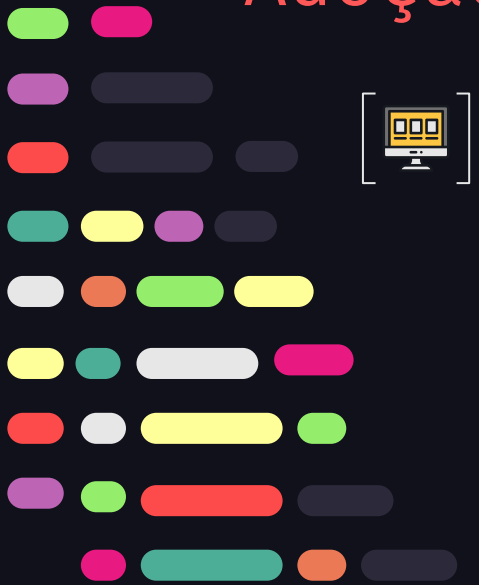
Devido à sua simplicidade e legibilidade, Python é frequentemente escolhida como a primeira linguagem de programação para ensinar conceitos de ciência da computação.

Isso fez com que ela fosse adotada em muitos cursos e bootcamps de programação, tornando-a uma das linguagens mais populares para educação em tecnologia.

Formador: Ricardo Mourão

O que é o Python?

Adoção por **Grandes Empresas**



Python também ganhou a validação por ser adotada por várias grandes empresas e organizações.

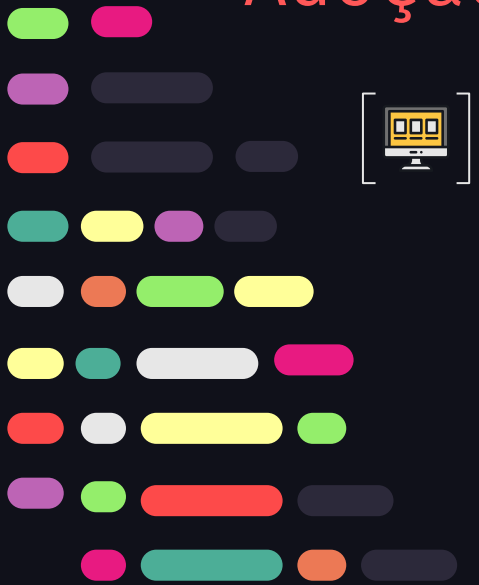
Guido van Rossum, por exemplo, trabalhou no Google e posteriormente no Dropbox, empresas que fizeram uso extensivo de Python para diversas aplicações.

Essa adoção por grandes players do mercado tecnológico serve como um testemunho da robustez e utilidade da linguagem.

Formador: Ricardo Mourão

O que é o Python?

Adoção por **Grandes Empresas**



Python também ganhou a validação por ser adotada por várias grandes empresas e organizações.

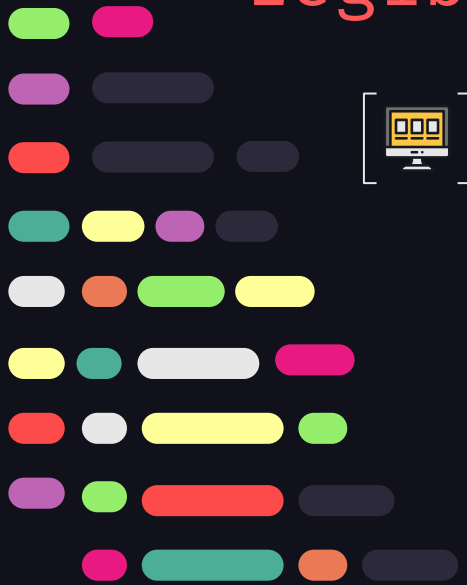
Guido van Rossum, por exemplo, trabalhou no Google e posteriormente no Dropbox, empresas que fizeram uso extensivo de Python para diversas aplicações.

Essa adoção por grandes players do mercado tecnológico serve como um testemunho da robustez e utilidade da linguagem.

Formador: Ricardo Mourão

Vantagens sobre outras linguagens

Legibilidade e Sintaxe **Clara**



Uma das características mais notáveis de Python é sua legibilidade.

A linguagem foi projetada com a clareza em mente, e isso é evidente na sua sintaxe.

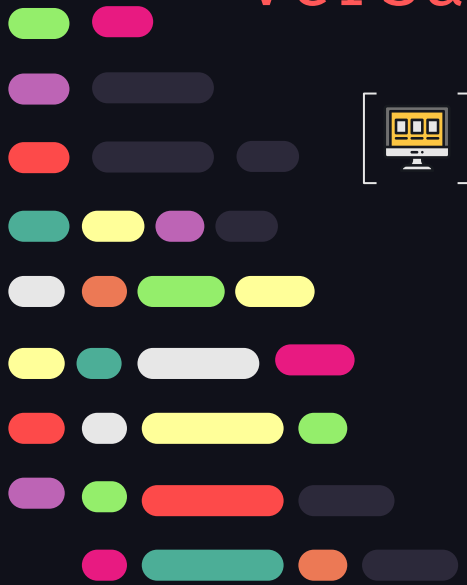
Os Blocos de código são definidos por indentação em vez de chaves, tornando o código mais limpo e fácil de ler.

Essa característica torna Python uma excelente escolha para programadores iniciantes e também reduz o custo de manutenção do software.

Formador: Ricardo Mourão

Vantagens sobre outras linguagens

Versatilidade e Ecosistema **Rico**



Python é incrivelmente versátil.

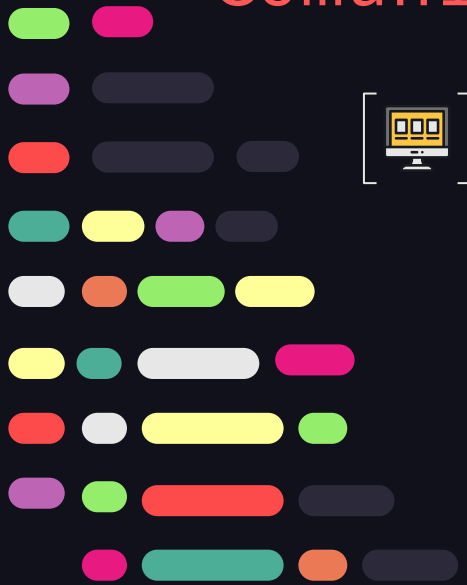
É usado numa variedade de domínios, incluindo, mas não limitando-se a, desenvolvimento web (Django, Flask), análise de dados (Pandas, NumPy), IA e Machine Learning (TensorFlow, scikit-learn).

Este ecossistema rico torna mais fácil encontrar bibliotecas e frameworks para quase qualquer projeto, economizando tempo e esforço.

Formador: Ricardo Mourão

Vantagens sobre outras linguagens

Comunidade **Ativa** e Suporte **Extensivo**



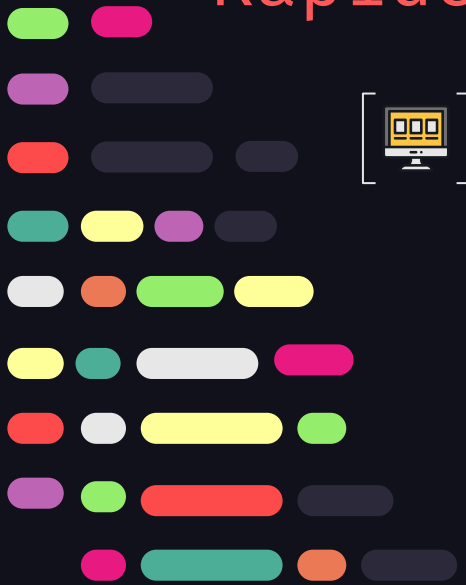
Python goza de uma comunidade muito ativa, o que significa que existe uma grande quantidade de documentação, tutoriais e mais importante, uma grande quantidade de bibliotecas e frameworks de terceiros disponíveis.

Isso não apenas torna mais fácil resolver problemas, mas também significa que é mais provável encontrar bibliotecas altamente otimizadas para tarefas específicas, algo que pode não ser verdade para linguagens mais novas ou menos populares.

Formador: Ricardo Mourão

Vantagens sobre outras linguagens

Rapidez no **Desenvolvimento**



Python permite um desenvolvimento mais rápido em comparação com linguagens como Java e C++.

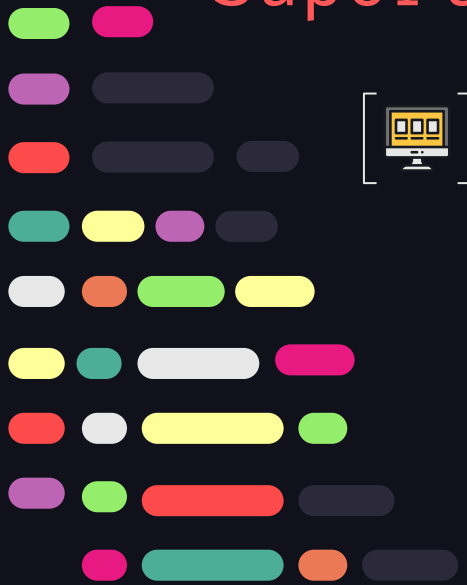
Isso é em parte devido à sua sintaxe clara e simples, e também por causa de sua extensa biblioteca padrão que suporta muitas tarefas comuns de programação.

Essa rapidez no desenvolvimento torna Python uma escolha popular para startups e outros ambientes onde o tempo até o mercado é uma consideração crítica.

Formador: Ricardo Mourão

Vantagens sobre outras linguagens

Suporte a **Múltiplos Paradigmas**



Python suporta múltiplos paradigmas de programação, incluindo programação procedural, orientada a objetos e funcional.

Isso dá aos programadores a flexibilidade para escolher o paradigma que melhor se adapta ao problema que estão tentando resolver, em vez de se adaptarem às limitações da linguagem.

Formador: Ricardo Mourão

Estrutura de um Python

Embora nenhuma linguagem seja perfeita ou adequada para todas as situações, Python oferece um conjunto único de vantagens que o tornam uma escolha atraente para muitos projetos. A sua sintaxe clara e legível, comunidade de suporte extensa, e versatilidade em aplicação fazem dela uma das linguagens de programação mais populares e em crescimento no mundo atual. É uma linguagem que consegue equilibrar facilidade de uso para novatos com a profundidade e complexidade necessárias para desenvolvimento de software em grande escala, tornando-a uma escolha sólida para uma ampla gama de projetos.

Formador: Ricardo Mourão

Programas feitos em Python; {

‘Grandes aplicações foram feitas em python’

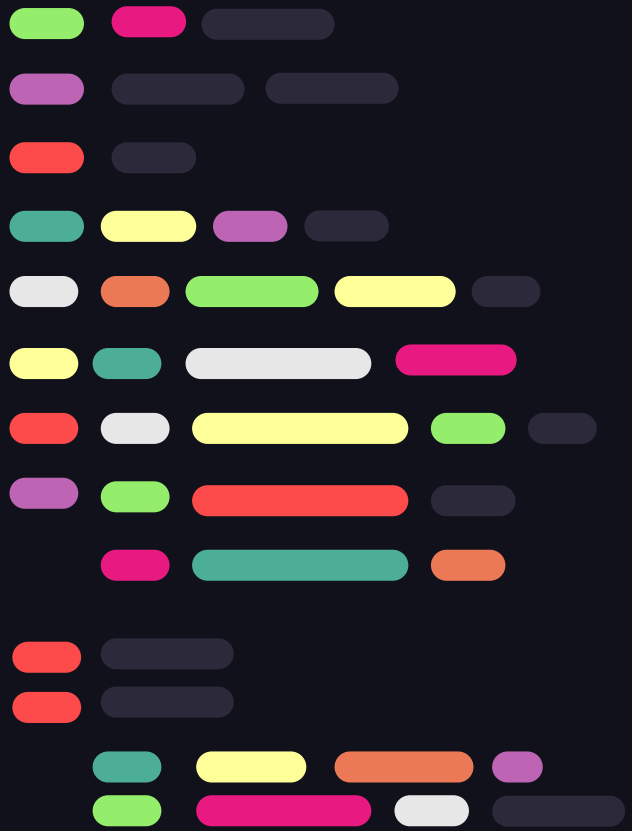
<p Instagram, Spotify, Google, Netflix, Dropbox, Reddit >

‘Mas também é procurado por inúmeras áreas’

<p Previsão e Análise de Dados, Veículos Autónomos, Assistência Médica Personalizada, Reconhecimento de Imagem e Voz, Chatbots >

}

Formador: Ricardo Mourão



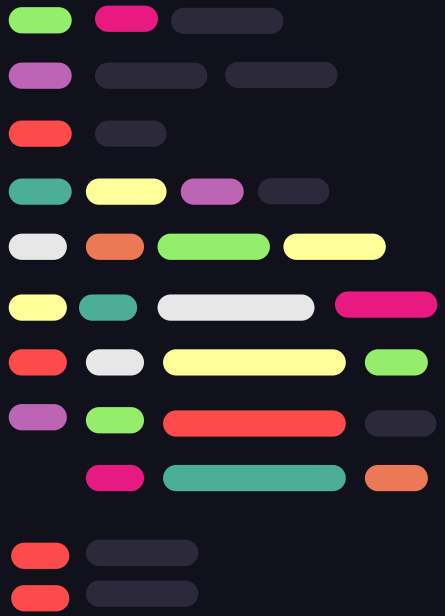
{ . .

Qual a IDE Indicada para desenvolver em Python?

} . .

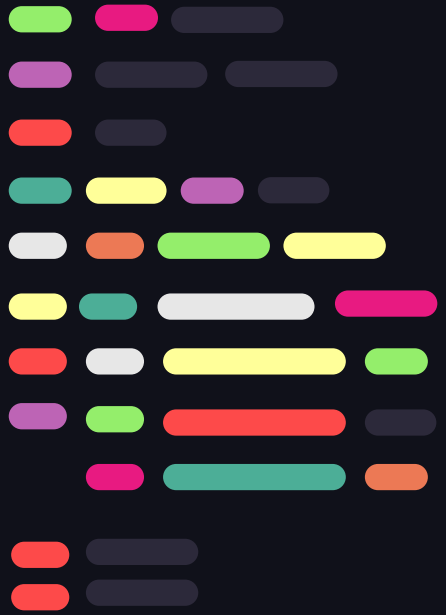
Formador: Ricardo Mourão

Há inúmeras IDEs adequadas



Formador: Ricardo Mourão

Há inúmeras IDEs adequadas



Sublime

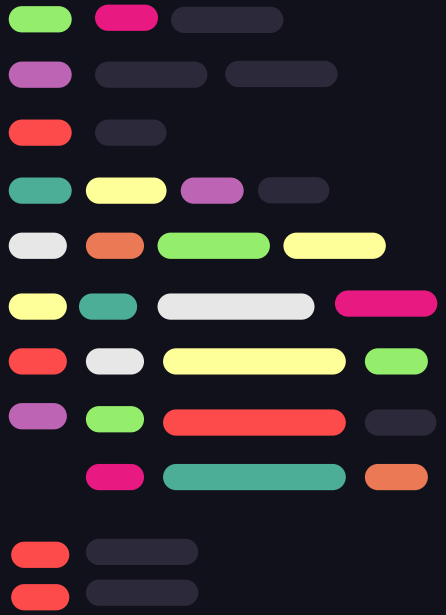
Pycharm

VSCode

Spyder

Formador: Ricardo Mourão

IDE para as aulas



Pycharm

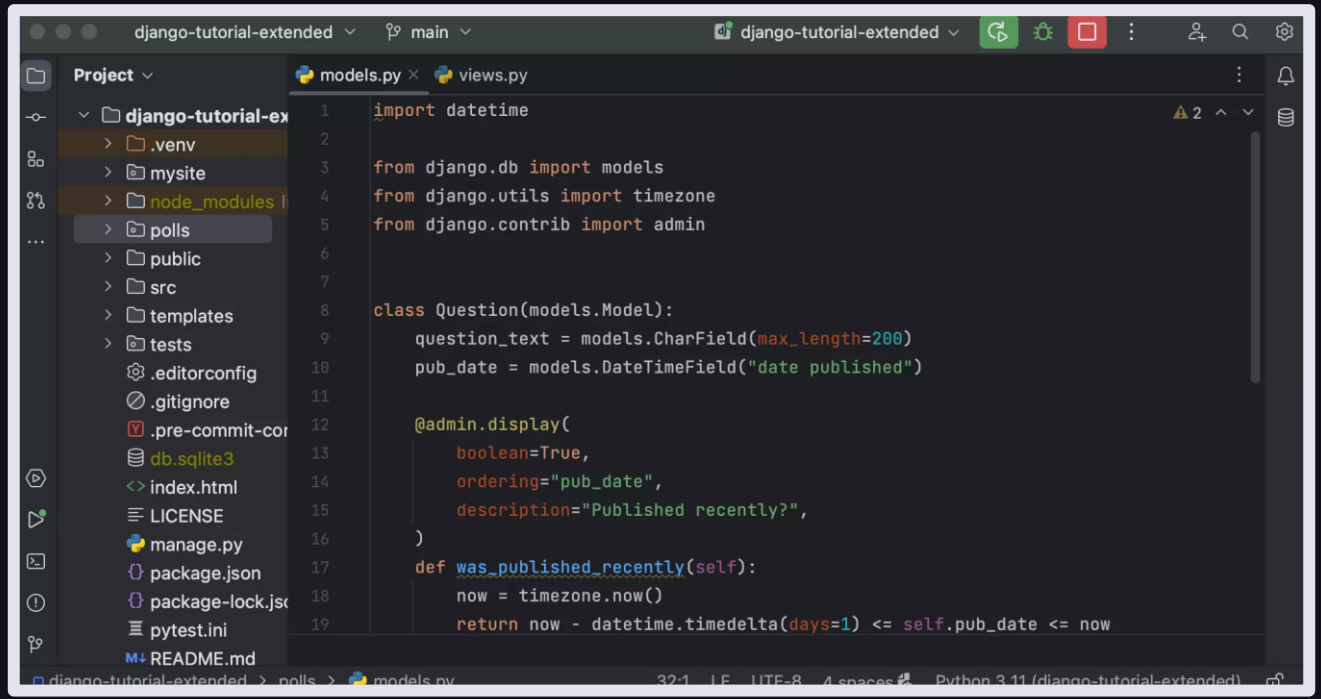
Site: https://www.jetbrains.com/pycharm/download/

Fazer download do **PyCharm Community Edition**.

--- A versão Community Edition é GRATUITA ---

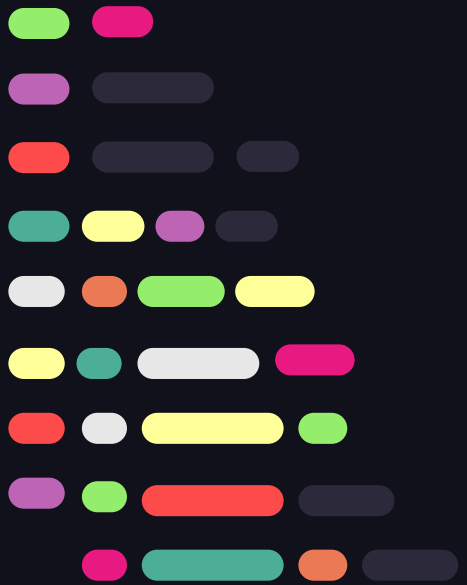
Formador: Ricardo Mourão

IDE para as aulas



Formador: Ricardo Mourão

Dinâmica em Aula



01 Criar uma pasta com o nome “Aulas Fundamentos Python”

02 Criar três pastas internas

2.1- Uma com o nome “Aulas”

2.2- Uma outra com o nome “Exercícios”

2.3- Uma terceira com o nome “Projetos”

Formador: Ricardo Mourão

01 { ..

Primeiros comandos em Python

} . .

Formador: Ricardo Mourão

Primeiros comandos em Python {

Funções

Importação

Comentários

Código Principal

}

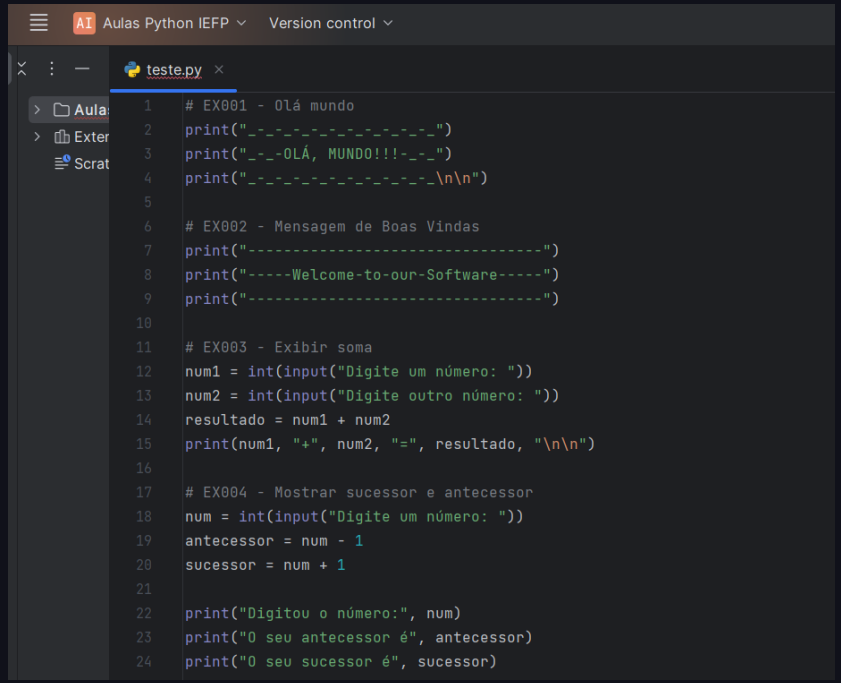
Formador: Ricardo Mourão

Primeiros comandos em Python {

Importação

A importação de bibliotecas refere-se ao ato de incluir código externo no programa. Uma biblioteca é um conjunto de funções e métodos pré-definidos que pode usar para realizar várias tarefas sem ter que escrever o código do zero.

}

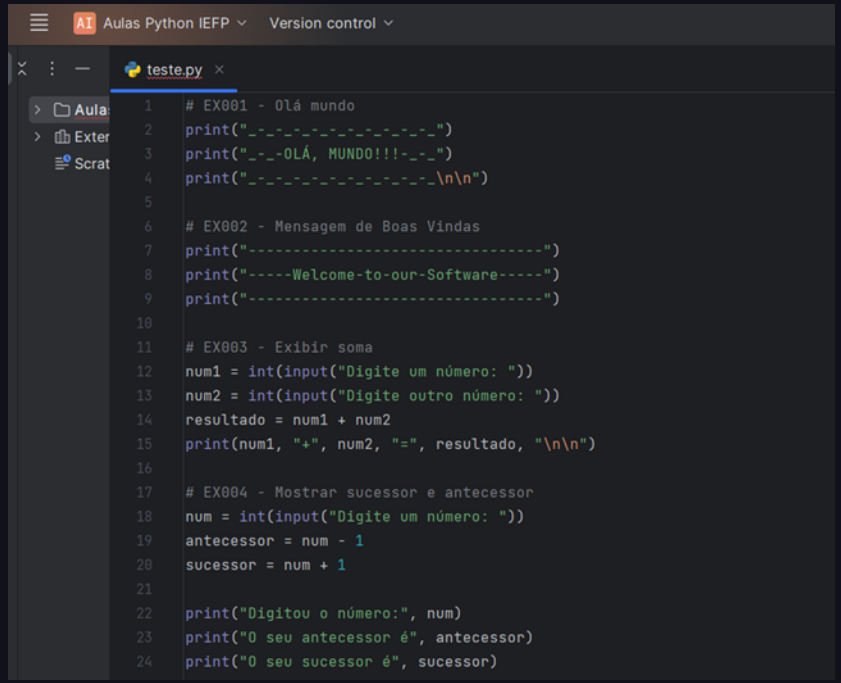


Formador: Ricardo Mourão

Primeiros comandos em Python {

Funções

Uma função em Python é um bloco de código que só é executado quando é chamado. Funções são definidas com a palavra-chave def, seguida do nome da função e parênteses que podem conter parâmetros.



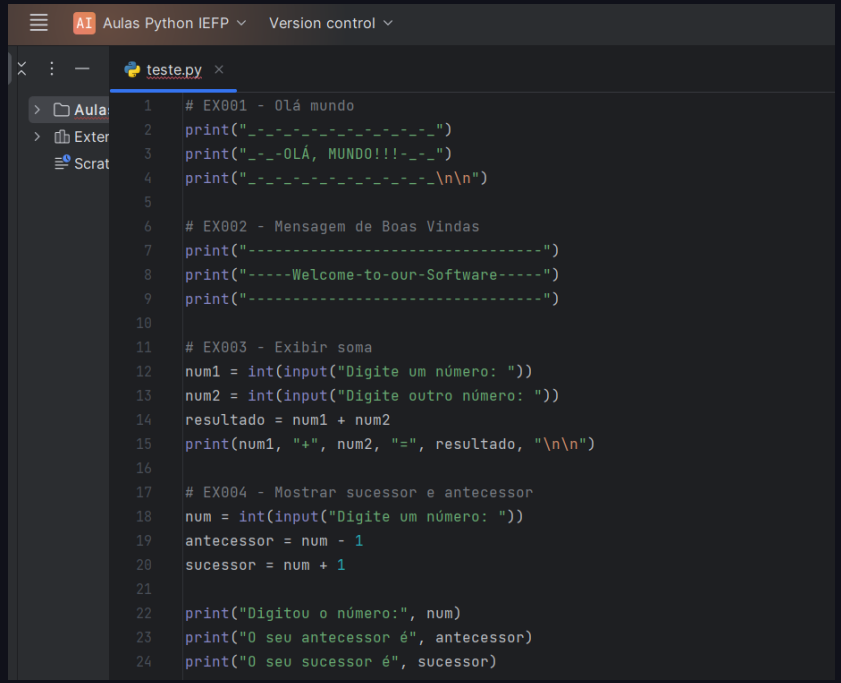
}

Formador: Ricardo Mourão

Primeiros comandos em Python {

Código Principal

O "código principal" refere-se à parte do código que executa as ações centrais do programa. É a sequência de comandos que se escreve para realizar tarefas específicas, que são executadas quando o programa é rodado.



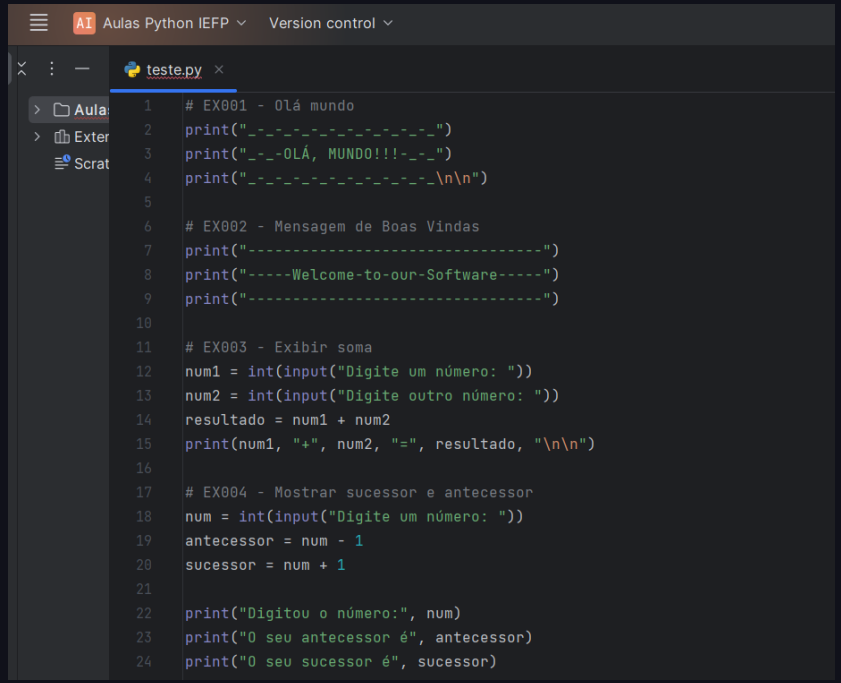
}

Formador: Ricardo Mourão

Primeiros comandos em Python

Comentários

Comentários são linhas no código que não são executadas pelo Python. São usados principalmente para explicar o que o código faz, tornando-o mais legível. Em Python, qualquer texto após o símbolo # numa linha é tratado como um comentário.



Formador: Ricardo Mourão

Estrutura de um programa em Python

```python
#Importação de biblioteca
import math

# Definição de função
def calcular_area_retangulo(largura, altura):
    """Calcula a área de um retângulo."""
    return largura * altura

# Código principal
# Definir as dimensões do retângulo
largura = 5
altura = 3

#Chamada da função para calcular a área
area = calcular_area_retangulo(largura, altura)

#Imprimindo o resultado
print(f"A área do retângulo é {area}")
```

Formador: Ricardo Mourão

Copiem este Código

```python
#Importação de biblioteca
import math

# Definição de função
def calcular_area_retangulo(largura, altura):
    """Calcula a área de um retângulo."""
    return largura * altura

# Código principal
# Definir as dimensões do retângulo
largura = 5
altura = 3

#Chamada da função para calcular a área
area = calcular_area_retangulo(largura, altura)

#Imprimindo o resultado
print(f"A área do retângulo é {area}")
```

Formador: Ricardo Mourão

Estrutura de um programa em Python

Input

Input (entrada), é considerada a entrada de informação ou os dados que o programa recebe para o processamento. A entrada pode vir de várias fontes como um arquivo, uma rede, o teclado, etc.

Output

Output (saída), é a informação ou dados que um programa produz. Os outputs podem ter múltiplos destinos como arquivos, redes, ecrã do utilizador, etc.

Formador: Ricardo Mourão

Estrutura de um programa em Python

Input

Em Python, a forma mais comum de ler a entrada é usar a função input(), que lê os dados inseridos através do teclado.

```
num = input("Digite um número: ")  # Lê uma string do teclado
num = int(num)  # Converte a string para um número inteiro
```

Formador: Ricardo Mourão

Estrutura de um programa em Python

Output

Em Python, o método mais comum para imprimir dados no ecrã do utilizador é a função print().

```
num = 10
print(”O número é:“, num) // Imprime O número é: 10
```

Formador: Ricardo Mourão

Estrutura de um programa em Python

Sequências de escape

Sequências de escape são combinações de caracteres que produzem um determinado efeito numa string impressa no ecrã através da função print(). Eles podem ser utilizados para mudar de linha, colocas aspas únicas e duplas e outros.

Formador: Ricardo Mourão

Estrutura de um programa em Python

Sequências de escape Lista

\a	Ativa um som de alerta
\n	Nova linha
\’	Aspas simples
\”	Aspas duplas
\t	Tab horizontal

Formador: Ricardo Mourão

02 { ..

Tipos de Dados, variáveis e constantes

} ..

Formador: Ricardo Mourão

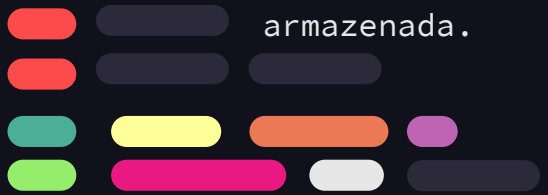
Tipos de Dados, variáveis e constantes

Variáveis

Variáveis são locais na memória onde se armazena valores que podem ser alterados durante a execução de um programa. Elas possuem um nome e um tipo de dado associado que define o tipo de informação que pode ser armazenada.

Constantes

Constantes, por outro lado, são locais na memória onde armazenamos valores que não podem ser alterados após a sua definição inicial. Assim como as variáveis, constantes também possuem um nome e um tipo de dado associado.



Formador: Ricardo Mourão

Tipos de Dados, variáveis e constantes

Variáveis

As variáveis são essenciais na programação, pois permitem armazenar e manipular dados durante a execução de um programa. Diferentemente de linguagens como C, Python é uma linguagem de *tipagem dinâmica*, o que significa que não é necessário especificar explicitamente o tipo de dado que uma variável vai armazenar. Para declarar uma variável em Python, basta atribuir um valor a um nome.

Formador: Ricardo Mourão

Tipos de Dados, variáveis e constantes

Variáveis

O sinal de = deve ser lido como **recebe**

a é o nome da minha variável

```
a = 19
```

Valor que vai ser atribuído à variável

Formador: Ricardo Mourão

Tipos de Dados, variáveis e constantes

Constantes

A importância de se usar valores que são tratados como constantes em Python reside no princípio de que eles não devem ser alterados, contribuindo para a consistência e segurança do programa. Apesar de Python não ter constantes no sentido estrito como em algumas outras linguagens, a convenção de usar nomes em maiúsculas para valores que não devem ser modificados ajuda a prevenir alterações acidentais.

Formador: Ricardo Mourão

Tipos de Dados, variáveis e constantes

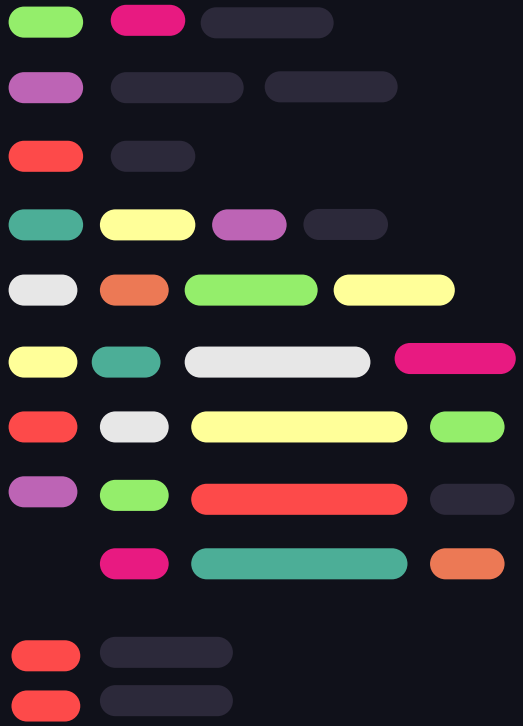
Constantes

O sinal de = deve ser lido como **recebe**

PI é o nome da minha constante ← PI = 3.14 → **Valor** que vai ser atribuído à constante

Formador: Ricardo Mourão

Tipos de Dados, variáveis e constantes

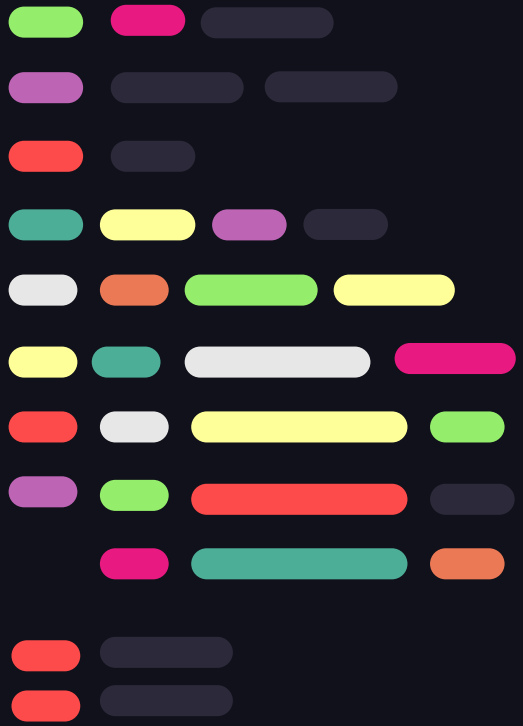


Algumas regras de declaração

- Não começar com números
- Não usar caracteres especiais
- Não utilizar palavras reservadas

Formador: Ricardo Mourão

Tipos de Dados, variáveis e constantes



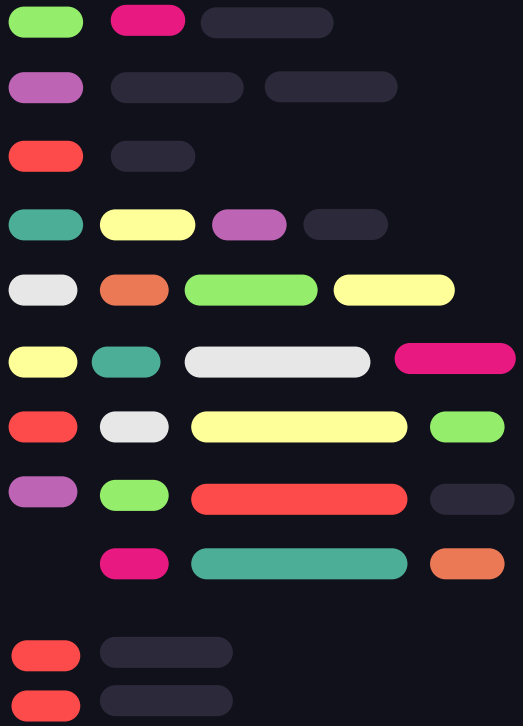
Escolher nomes significativos

Escolher nomes significativos para variáveis e constantes é fundamental para criar um código legível e fácil de manter.

Ao nomear variáveis e constantes, evite nomes genéricos e curtos, como a, x, temp ou count, e opte por nomes mais descritivos que reflitam o propósito da variável, como altura, velocidade, saldo, numeroDeClientes ou idade_utiliador.

Formador: Ricardo Mourão

Tipos de Dados, variáveis e constantes



Escolher nomes significativos

- idadeUtilizador;
- dias_ano;
- nome_utilizador;

Formador: Ricardo Mourão

Tipos de Dados, variáveis e constantes

Escrever no ecrã

Escrever **Olá, Mundo!**
➢ print(“Olá, Mundo!”)

Escrever texto com variável
➢ print(“O valor de a é”, a)

Escrever o valor de uma variável a
➢ print(a)

Escrever e saltar linha
➢ print(“Ola!\nOlá noutra linha”)

Formador: Ricardo Mourão

Tipos de Dados, variáveis e constantes

Tipos de variáveis comuns

Int : números inteiros
- a = 5

Float : números com casas decimal
- pi = 3.14

Char : letras únicas
- c = ‘C’

Escrever valores de variáveis

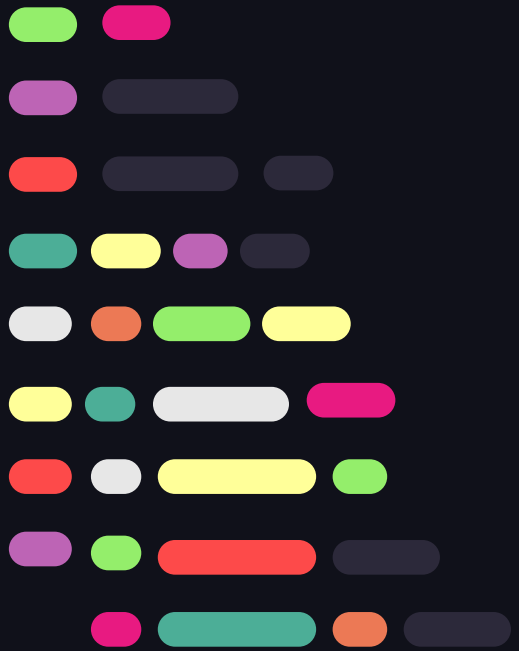
Inteiros
- print(nomeDaVariavel)

Float
- print(nomeDaVariavel)

Char
- print(nomeDaVariavel)

Formador: Ricardo Mourão

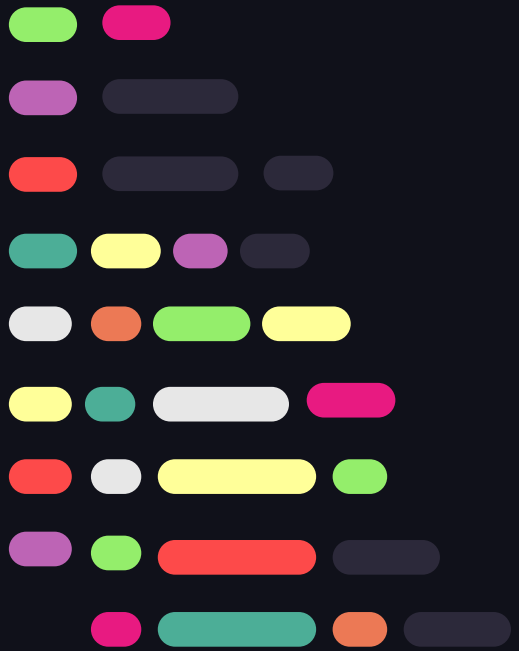
PRÁTICA! Exercicio 01



Crie um programa simples em Python que imprima no ecrã “Olá mundo”

Formador: Ricardo Mourão

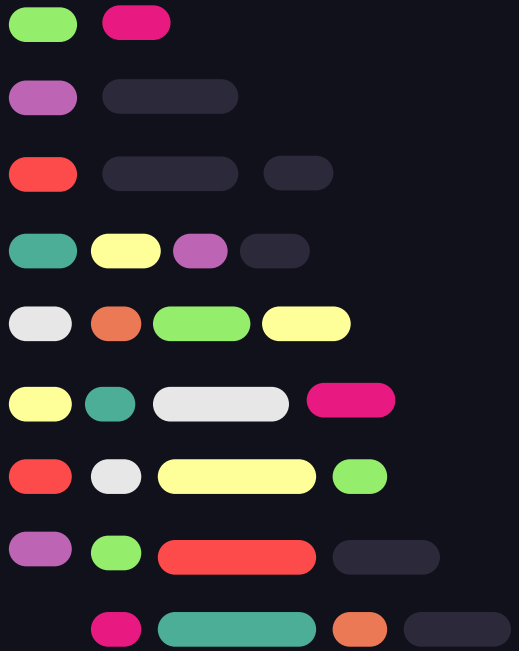
PRÁTICA! Exercicio 02



Cria um programa simples em Python que imprima no ecrã “Estou a aprender Python.”

Formador: Ricardo Mourão

PRÁTICA! Exercicio 03

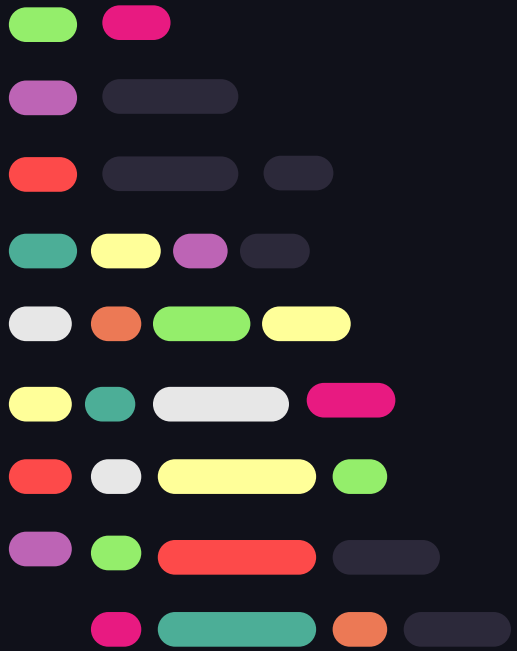


Cria um programa que mostre uma mensagem de boas vindas ao utilizador.

Ex: “Damos as boas vindas ao meu programa”

Formador: Ricardo Mourão

PRÁTICA! Exercicio 04



Cria um programa que mostre:

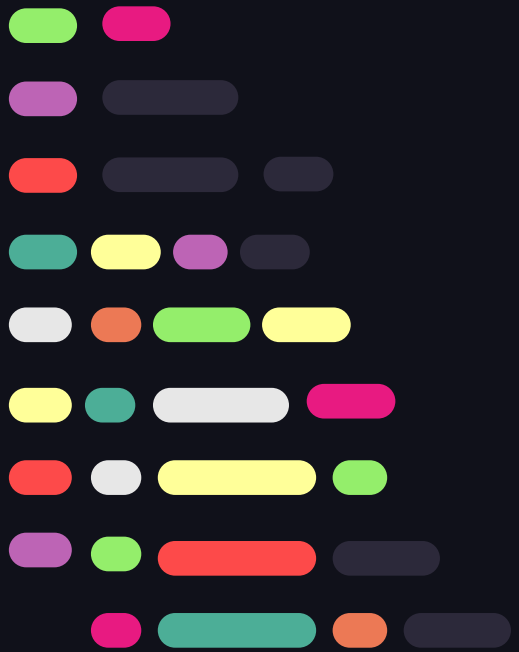
-_-_-_-_-_-_-_-_-

O MEU PROGRAMA

-_-_-_-_-_-_-_-_-

Formador: Ricardo Mourão

PRÁTICA! Exercicio 05



Cria um programa que mostre:

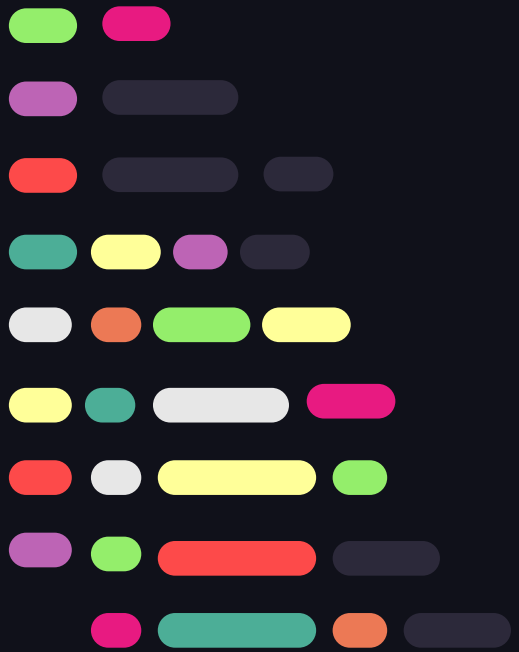
Em linhas de código, a arte se revela,
Python se destaca, de forma singela.
Identação precisa, seu charme discreto,
Torna a programação um caminho direto.

Com loops que iteram, decisões a tomar,
Cada função em Python, há algo a explorar.
Manipula os dados com pandas, sem parar,
E na simplicidade do código, vê-se a lógica brilhar.

Uma lista, um dicionário, tudo bem organizado,
Em Python, cada estrutura tem seu lugar bem marcado.
De strings a números, tudo se conecta,
A linguagem é poderosa, flexível e direta.

Formador: Ricardo Mourão

PRÁTICA! Exercicio 06

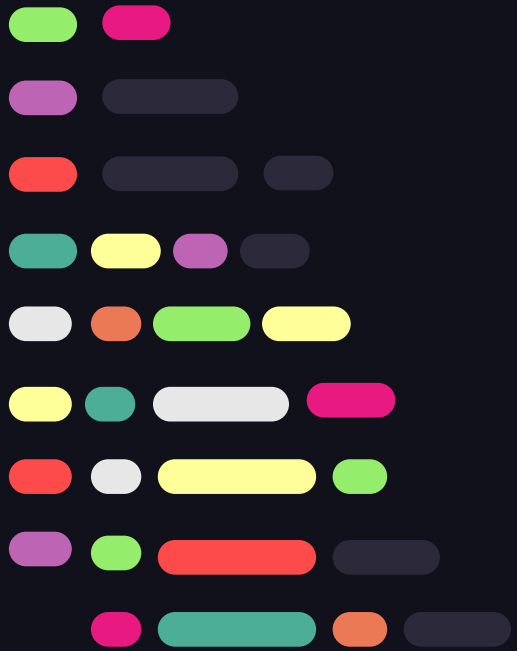


Crie o seguinte menu:

```
--- Calculadora ---
[ 1 ] - Tabuada
[ 2 ] - Calculadora
[ 3 ] - Fatorial
[ 4 ] - Números primos
```

Formador: Ricardo Mourão

PRÁTICA! Exercicio 07

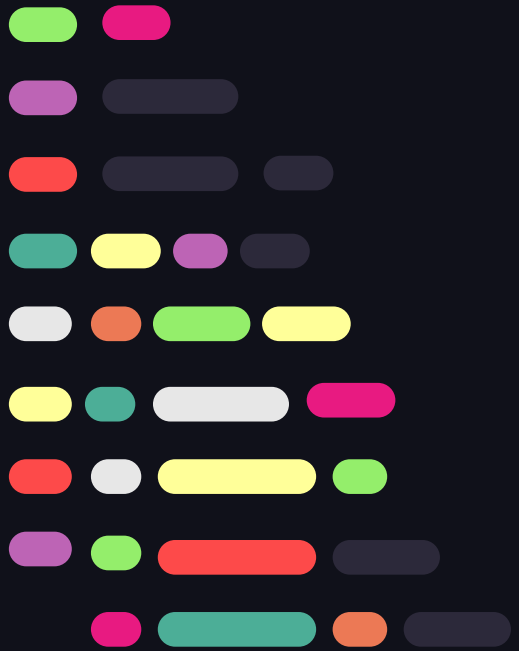


Cria um programa que peça o nome do utilizador e dê uma mensagem de boas vindas com o nome do utilizador.

Ex: “Olá, Ricardo. Damos as boas vindas ao nosso programa.”

Formador: Ricardo Mourão

PRÁTICA! Exercicio 08

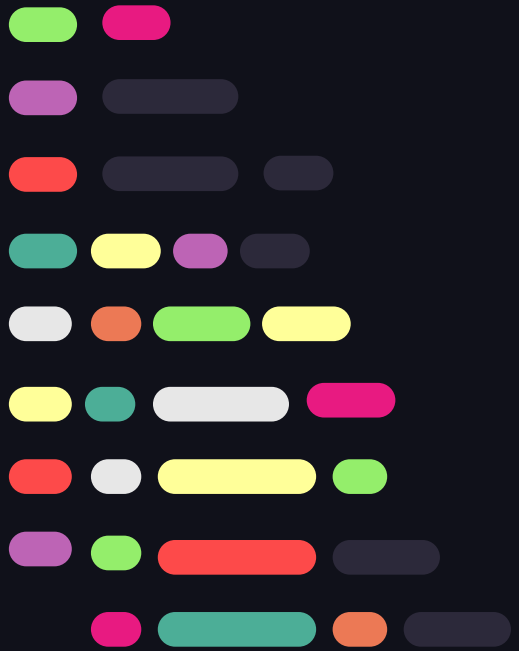


Cria um programa que peça o nome, idade e a cidade onde mora. Depois faça uma apresentação com todos os elementos recebido.

Ex: “Olá, o meu nome é Ricardo, tenho 32 anos e sou do Porto.”

Formador: Ricardo Mourão

PRÁTICA! Exercicio 09



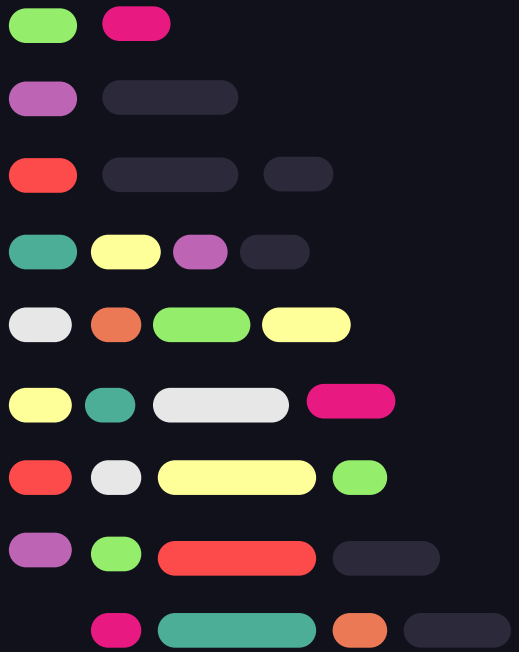
Cria um programa que simule uma conversa:

Ex:

Olá, eu sou um BOT, como te chamas?
Dou-te as boas vindas “nome”
Que idade tens?
Fantástico, já tens “idade” anos. És de onde?
Uau, “cidade”?? Nunca visitei!

Formador: Ricardo Mourão

PRÁTICA! Exercicio 10



Crie o seguinte menu que leia a opção que escolheu:

--- Calculadora ---

[1] - Tabuada

[2] - Calculadora

[3] - Fatorial

[4] - Números primos

Escolheste a opção “opcao”

Formador: Ricardo Mourão