

Lógica de Programação

Linguagem Python

Profa. Ma. Samara Souza
040601858@prof.unama.br

Relembrando...

Etapas de um algoritmo

- ❑ **Entrada:** São as informações que iniciam o algoritmo.
- ❑ **Processamento:** São os passos necessários para atingir a meta.
- ❑ **Saída:** São os resultados do processamento.

O que vamos ver na aula de hoje?

- ☐ Linguagem Python
- ☐ Variáveis
- ☐ Tipos de Dados
- ☐ Entrada e Saída
- ☐ Operadores Aritméticos
- ☐ Exercícios

O que é Python?

- ❑ Python é uma linguagem de programação de alto nível, interpretada e orientada a objetos.
- ❑ Foi criada por Guido Van Rossum e lançada em 1991.
- ❑ É conhecida por sua sintaxe limpa e legível, o que a torna uma excelente escolha para iniciantes.









Características do Python

- Linguagem de alto nível
- Orientada a objetos
- Tipagem dinâmica e forte
- Fácil aprendizagem
- Comunidade ativa
- Muitas bibliotecas/recursos
- Linguagem interpretada

- ❑ Apesar de sua simplicidade, o Python pode ser aplicado em projetos robustos.
- ❑ Back-end de sistemas web, ciência de dados, machine learning e simulações são alguns exemplos de projetos em que o Python pode marcar presença.
- ❑ Os setores de tecnologia de empresas, como Google, YouTube, Disney e IBM estão repletos de funcionários que dominam o Python.



- ❑ Python é a primeira linguagem de programação mais utilizada no mundo.

Feb 2024	Feb 2023	Change	Programming Language		Ratings	Change
1	1			Python	15.16%	-0.32%
2	2			C	10.97%	-4.41%
3	3			C++	10.53%	-3.40%
4	4			Java	8.88%	-4.33%
5	5			C#	7.53%	+1.15%
6	7	^		JavaScript	3.17%	+0.64%

<https://www.tiobe.com/tiobe-index//>



Configurando o ambiente para instalação

- ❑ Acesse o site <https://www.python.org/downloads/> e, após o download, localize o arquivo no computador e dê um duplo clique.
- ❑ Ao finalizar a instalação, vamos nos certificar de que tudo está em ordem, executando os seguintes passos:
 - Na Barra de pesquisa, localizada ao lado do botão Windows, digite “cmd”.
 - Localize o ícone do aplicativo Prompt de Comando.
 - No Prompt de Comando digite a palavra “python”. Para finalizar, pressione Enter.
- ❑ Observe que, no terminal, além de ser exibida a versão instalada, o cursor permanece piscando, aguardando uma instrução.



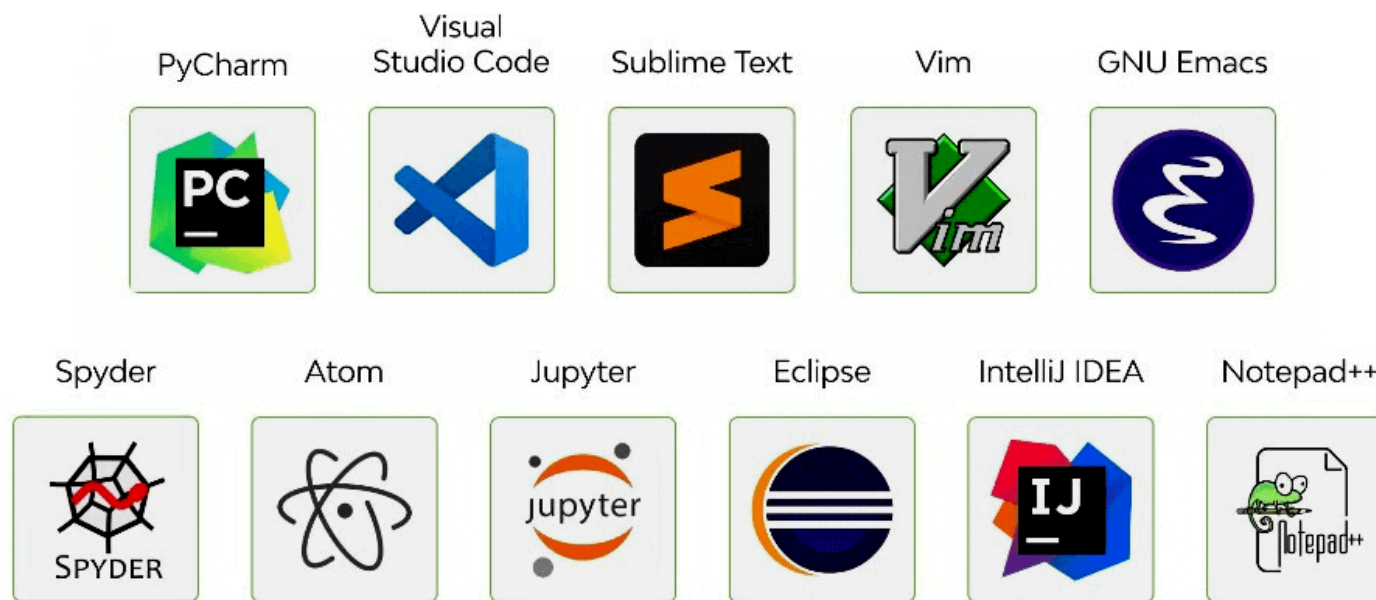
Integrated Development Environment (IDE)

- ❑ *Integrated Development Environment* (IDE) ou **Ambiente de Desenvolvimento Integrado** é um software que combina ferramentas comuns para desenvolvimento em uma única interface gráfica. Isso facilita o desenvolvimento de aplicativos!



Integrated Development Environment (IDE)

- ❑ Em vez de usarmos a ferramenta do Python para criar e executar programas, a utilização de uma **IDE** torna o desenvolvimento mais **fácil** e **agradável** com um aspecto moderno, o que vai facilitar bastante no **desenvolvimento** das atividades.
- ❑ Atualmente, temos disponível no mercado várias IDEs.



- ❑ Agora que já conhecemos um pouco sobre python...
- ❑ Vamos fazer um algoritmo para nos livrarmos da maldição?

Vamos invocar o poder do Python e quebrar a maldição começando com o sagrado 'Hello, World!' - o primeiro passo na jornada para desvendar os mistérios da programação!



Comando print()

- ❑ O comando **print()** é utilizado quando desejamos imprimir alguma mensagem de texto na tela.
- ❑ Equivalente ao comando de saída **escreva()** do VisuAlg.

Comando print()

- ❑ A sintaxe da função **print()** requer que se utilize os parênteses com o texto que será exibido.
- ❑ Esse texto deve ser escrito, também, **entre aspas**, podendo ser aspas simples ' ou dupla ''.
- ❑ Veja um exemplo:

```
print("Hello, World!")
```



Importante!

Case Sensitive

- ❑ A linguagem Python é Case Sensitive. Você sabe o que isso significa?
 - Significa que ela faz distinção entre letras maiúsculas e minúsculas.
- ❑ Portanto, é importante estar atento à escrita dos comandos.
- ❑ Por exemplo: a função **print()** deve ser escrita em letra minúscula.



Case Sensitive

Agora que já sabemos que a linguagem Python é Case Sensitive, vamos visualizar alguns erros de sintaxe que podem ocorrer durante a digitação da linguagem:

Erros de digitação referente à caixa alta (letras maiúsculas) e à caixa baixa (letras minúsculas):

Print ("Hello World")

PRINT ('Hello World')

Erros de falta de parênteses ou de aspas:

print (Hello World

print (Hello World")



Declaração de Variáveis

- ❑ Na linguagem Python **não é necessário** declarar as variáveis **no início do código**. Isso pode ser feito a qualquer momento, durante a codificação.
- ❑ Sobre o tipo da variável, não é necessário defini-lo em sua declaração. A **linguagem interpreta** o tipo de dado que está sendo armazenado.

Caso seja necessário forçar o reconhecimento de algum tipo específico, basta realizar as conversões necessárias.

Variável

❑ Sintaxe de **atribuição** em python:

<variável> = <valor>

nome = "samara"

Idade = 25



Variável

❑ Diferente da maioria das linguagens, uma variável em Python não tem um tipo fixo, apenas o tipo do conteúdo.

■ Exemplos:

X = 10

Ao atribuir o valor 10 na variável X, ela passará a ser reconhecida com um tipo **inteiro**.

cidade = 'Belém'

Atribuindo o valor 'Belém' na variável cidade, ela passará a ser reconhecida com um tipo **string**.



Tipos de dados

- ❑ Inteiro (int)
- ❑ Real (float)
- ❑ Booleano/lógico (bool)
 - Verdadeiro (*True*) e Falso (*False*)
- ❑ String (str)

- ❑ Para melhor compreensão dos tipos de variáveis, vamos utilizar a função **type()** para identificar o tipo do valor armazenado em uma variável.

Na Prática

```
nome = "samara"
```

```
idade = 25
```

```
altura = 1.59
```

```
aluno = False
```

```
tipo = type(altura)
```

```
print(altura)
```

```
print(tipo)
```



Concatenando dados na saída padrão

- ❑ Até o momento, visualizamos as informações com a função **print()** exibindo apenas uma informação por vez.
- ❑ No momento da exibição podemos **concatenar (unir)** várias informações em apenas uma linha. Para isso, utilizamos a **vírgula (,)** para separar textos e as variáveis.

Concatenando dados na saída padrão

```
nome = "samara"  
idade = 25  
altura = 1.59  
aluno = False  
  
print("A(0) ", nome, " tem ", idade, " anos!")
```



- ❑ O método **.format()** em Python é usado para formatar strings, permitindo a inserção de valores de variáveis em posições específicas dentro de uma string formatada. Aqui está um exemplo básico de como utilizá-lo:

```
nome = "Samara"
idade = 25

print("Olá, meu nome é {} e tenho {} anos.".format(nome, idade))
```



```
numero = 3.14159  
  
# Formata com duas casas decimais  
print("{:.2f}".format(numero)) # Saída: 3.14
```

- ❑ Até o momento, trabalhamos com a entrada de dados por meio do código, porém, para que possa ter uma interação do usuário tornando o programa dinâmico, temos de trabalhar com a função de entrada de dados, que no Python é conhecida como:

input

Entrada de Dados

```
# Solicita ao usuário que insira um número  
fruta = input("Digite o nome de uma fruta")  
  
# Imprime o valor armazenado  
print(fruta)
```



- ❑ Quando utilizamos a entrada de dados por meio da função **input**, essa será considerada do tipo **string**. Assim, caso seja necessário realizar a entrada de valores numéricos, temos de converter o tipo de dado, de acordo com o tipo que desejamos armazenar na variável.

input

Entrada de Dados

```
nome = input("Digite o seu nome: ")
numero = input("Digite um número: ")
idade = int(input("Digite a sua idade: "))
altura = float(input("Digite a sua altura: "))

print(type(numero))
print(type(idade))
```



Operadores Aritméticos

Símbolo	Operação	Exemplo
+	Adição	$3 + 4 = 7$
-	Subtração	$5 - 2 = 3$
*	Multiplicação	$8 * 5 = 40$
/	Divisão normal	$3 / 2 = 1.5$
//	Divisão inteira	$3 // 2 = 1$
%	Módulo (resto da divisão inteira)	$10 \% 2 = 0$
**	Potenciação	$3 ** 2 = 9$

- 1) Escreva um programa que receba dois números inteiros como entrada e forneça como saída a soma desses números.
- 2) Escreva um programa que receba o nome de uma pessoa, seu ano de nascimento e o ano atual. Como saída, exiba o nome da pessoa e sua idade atual.
- 3) Escreva um programa que leia um valor como saldo, e depois leia um valor de retirada e apresente quanto saldo ainda resta.
- 4) Escreva um programa que leia a idade de uma pessoa expressa em anos, meses e dias e mostre-a expressa apenas em dias.
- 5) Escreva um programa que solicite a entrada da largura e altura de um retângulo e calcule sua área. Em seguida, exiba o resultado.

Lógica de Programação

Linguagem Python

Profa. Ma. Samara Souza
040601858@prof.unama.br