



INFINITY SCHOOL

V I S U A L A R T C R E A T I V E C E N T E R

Desenvolvido por:
Marcus Azevedo
Rafael Puyau

■ Objetivos da aula:

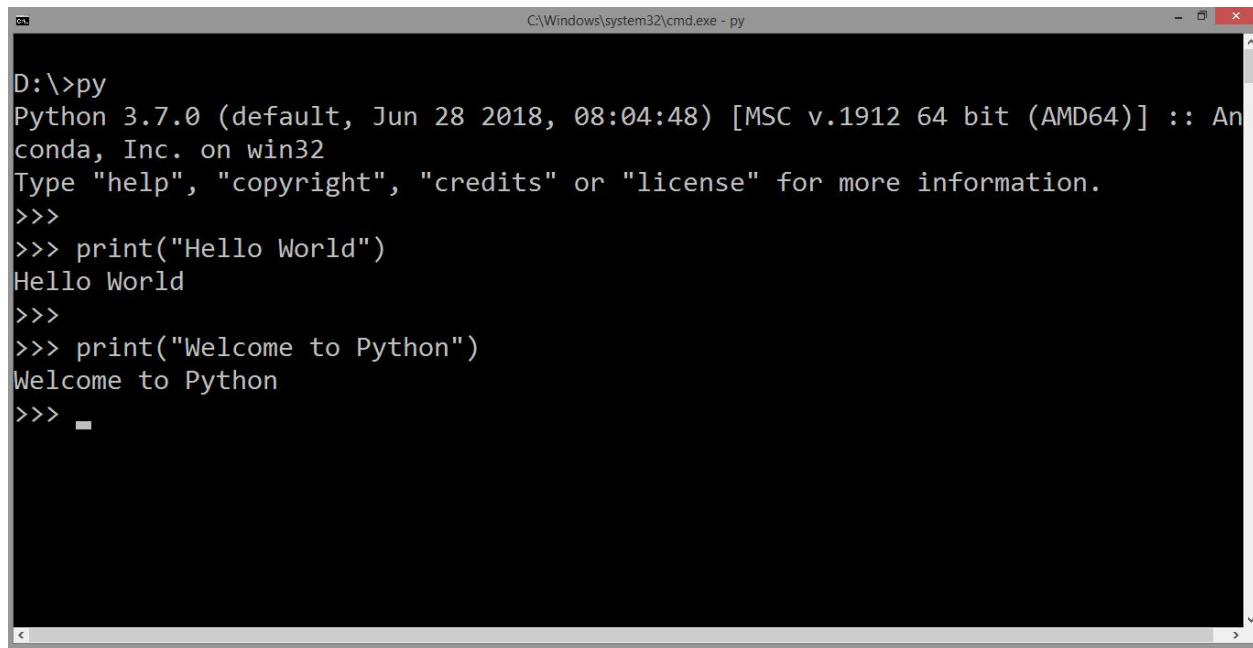
1. Fundamentos (ambientes virtuais e shell).
2. Tipos de variáveis.
3. Print.
4. Input.
5. Type.
6. Casting.
7. Escopo de variáveis.
8. Operadores aritméticos.
9. Estruturas condicionais (decisão).



Python - Aula 01

Fundamentos (Ambientes virtuais e shell)

Shell do Python:



```
C:\Windows\system32\cmd.exe - py
D:\>py
Python 3.7.0 (default, Jun 28 2018, 08:04:48) [MSC v.1912 64 bit (AMD64)] :: Anaconda, Inc. on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
>>> print("Hello World")
Hello World
>>>
>>> print("Welcome to Python")
Welcome to Python
>>> █
```

■ Shell do Python:

- A Python Shell é o interpretador que executa os seus programas Python, outras peças do código Python ou comandos simples.
- Neste interpretador você pode executar qualquer comando ou expressão Python.
- O Shell é uma ótima solução para todo programador, e por isso é uma boa ideia adquirir o hábito de usá-lo.
- Se existir alguma dúvida sobre como algo funciona no Python, o Shell pode esclarecer para você.



■ Criando seu ambiente virtual no shell:

Todo programador python deve saber isolar seu ambiente de trabalho pois em muitos casos somos alocados em projetos diferentes onde usamos, em alguns casos, um determinado pacote em 2 ou mais projetos, porém em versões distintas a depender de cada cliente.

Para nos ajudar a resolver essa situação, o python disponibiliza/permite a criação de ambientes virtuais onde conseguimos isolar um determinado ambiente em nossa máquina.

Há diversos pacotes, mas aqui no curso utilizaremos o módulo **venv**, que já vem por padrão no python.



■ Criando seu ambiente virtual no shell:

Para ativar o ambiente virtual, primeiro devemos entrar em nossa pasta de trabalho.

Dentro dela, digite o comando a seguir: **python -m venv <nome do ambiente virtual>**

Ex: python -m venv IN

ATENÇÃO: nenhuma mensagem aparecerá no terminal, o que significa que o comando rodou com sucesso. Quando o sistema operacional devolver o prompt de comando pra você sem alguma mensagem, significa que o comando rodou com sucesso.



■ Criando seu ambiente virtual no shell:

Agora é hora de ativar seu ambiente virtual.

Em máquinas **Windows** digite o comando: **IN \ Scripts \ activate**

Em máquinas **Linux** digite o comando: **source IN/bin/activate**

Repare que o nome do seu ambiente virtual aparecerá entre parêntesis informando que o mesmo está ativo.

Agora poderá instalar os pacotes que irá trabalhar em cada projeto.

Para isso, utilize o gerenciador de pacotes do python: pip



■ Criando seu ambiente virtual no shell:

Para instalar um pacote no seu ambiente virtual ativo, digite o comando **pip install** e o nome do pacote

Obs: para instalar uma versão específica: `pip install pacote==0.0.0`
(substitua pacote pelo nome do pacote desejado e 0.0.0 pela versão a ser instalar. Se não informar a versão, a mais atual será instalada)

Para desinstalar: `pip uninstall pacote`

Os pacotes são instalados do repositório www.pipy.org

Para desativar o ambiente virtual, basta digitar o comando `deactivate` em ambos os sistemas operacionais.



■ Criando seu ambiente virtual no shell:

O comando **pip freeze** listará todos os pacotes com suas versões instaladas no seu ambiente virtual ativado.

Para reproduzir este ambiente de trabalho em outra máquina ou para enviar a um outro desenvolvedor, execute os seguintes passos:

1. Com seu ambiente virtual ativado, digite o comando: `pip freeze > requirements.txt`

Isso fará com que a saída do `pip freeze` vá para o arquivo `requirements.txt`. o nome desse arquivo é uma convenção mas você pode utilizar o nome que desejar.



■ Criando seu ambiente virtual no shell:

2. Compartilhe o arquivo com outro desenvolvedor ou leve-o para outra máquina.

3. Na máquina nova, crie um novo ambiente virtual, ative-o e rode o comando: `pip install -r requirements.txt`

Isso instalará todos os pacotes instalados nas mesmas versões.

Os outros devs que receberem esse arquivo deverão executar os mesmos passos descritos acima.



Python - Aula 01

Tipos de variáveis

■ **int**

Guardam números inteiros

Exemplo: 2, 5, 2001, -2001.

■ **float**

Guardam números de ponto flutuante

Exemplo: 2.5, -60.9, 7.0.

■ **str**

Guardam informações de texto

Exemplo: "Bem vindo ao módulo de python."

■ **bool**

True, False

São **variáveis** capazes de conter apenas 1 de 2 valores: verdadeiro ou falso.



Python - Aula 01

Print()

A função para imprimir dados em Python é a função `print()`.

Ela é responsável por mostrar valores em seu terminal.

Exemplo:

```
print('Olá Mundo')
```

No terminal, aparecerá :

```
>>> Olá Mundo
```



Python - Aula 01

Input()

Essa função é muito comum na programação. Quando precisamos que o usuário passe ao programa algum tipo de dado, para preencher uma variável, nós usamos o comando `input()`, que é literalmente 'entrada' em inglês.

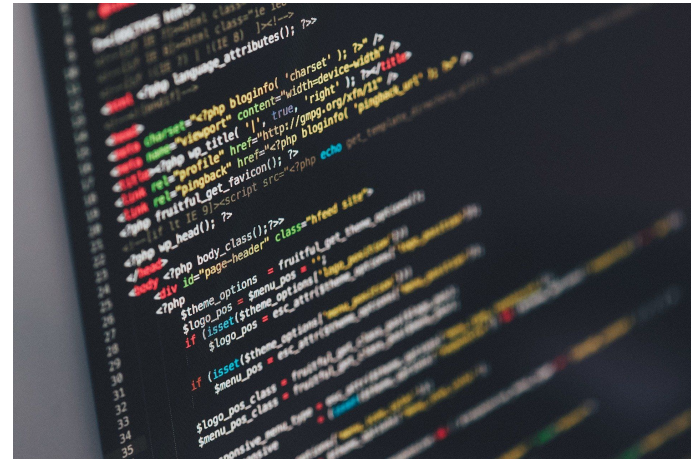
A função `input()` recebe como parâmetro uma string que será mostrada como auxílio ao usuário, geralmente o informando que tipo de dado o programa está aguardando receber.

Exemplo:

```
nome: input("Escreva seu nome: ")
```

No terminal, aparecerá :

```
>>> Escreva seu nome:
```



Python - Aula 01

■ Type()

Para que serve o Type no Python?

A função **type()** mostra o tipo do dado.

Dentro desta função, você deve passar o nome da variável ou o próprio valor. E ele retornará o tipo de dados variável.

Exemplo:

```
var1 = 155
```

```
type(var1)
```

```
print(type(var1))
```



Python - Aula 01

■ Casting

Casting é o método para converter o tipo de dados da variável em um certo tipo de dados para que a operação seja executada pelos usuários.

Em poucas palavras, é o processo de indicar o tipo que será declarado na variável.

Por exemplo: Transformar int em float ou float em int.

Desta forma, ao transformar float em int, será desconsiderada a parte fracionária do número.

```
float()  
int()  
str()
```

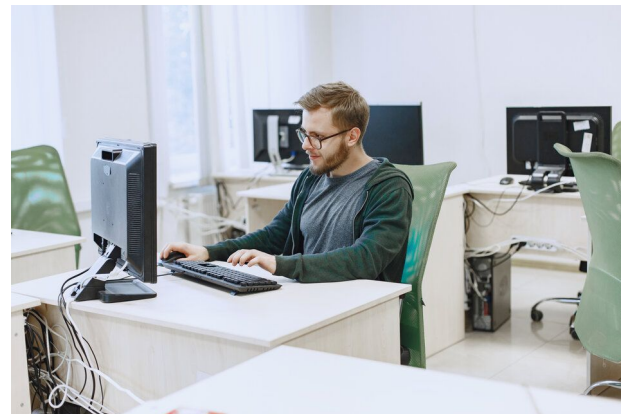


■ Escopo de variáveis

O escopo de uma variável é todo trecho de execução do código onde aquela variável pode ser encontrada, lida e alterada. Uma vez criada a variável fica disponível dentro desses trechos e só será apagada quando nenhuma referência é feita a ela. Fora de seu escopo a variável não existe e, por isso, o mesmo nome pode ser atribuído à outra variável, sem conflito.

Por exemplo:

Uma variável que declaramos dentro de um bloco de for, while ou até mesmo uma função.



Python - Aula 01

Operadores aritméticos

- + soma
- - subtração
- * multiplicação
- / divisão
- ** potência
- // divisão inteira
- % resto da divisão (módulo)

+	soma	$3+2=5$
-	subtração	$3-2=1$
*	multiplicação	$3*2=6$
/	divisão	$3/2=1.5$
%	módulo	$3\%2=1$
**	exponenciação	$3**2=9$
//	floor division	$3//2=1$

Operadores relacionais

- == igual
- != diferente
- < menor que
- > maior que
- >= maior igual a
- <= menor igual a

Condicionais if:

Uma estrutura condicional, como o próprio nome já diz é uma estrutura que vai analisar uma condição e baseado no resultado dessa condição é que vamos executar uma determinada ação.

A função IF, traduzindo ela significa SE. Dessa forma, facilitamos o entendimento.

Um exemplo simples:

SE 10 for maior que 5, então, execute uma ação. Caso contrário não faça nada.

Então a função IF vai executar uma ação somente se a condição testada for verdadeira, nesse caso vai executar tudo que estiver dentro dela (com a indentação, espaçamento que temos abaixo do IF para mostrar que as informações fazem parte dele).

if 10 > 5:

```
print('10 é maior que 5')
```

Operadores Lógicos

```
variavel =  
  
if nome:  
    print('A variável nome não é vazia')  
  
if nome == #Igual  
  
if nome != #Diferente  
  
if nome >= #Maior ou Igual  
  
if nome <= #Menor ou Igual  
  
if nome > #Maior  
  
if nome < #Menor  
  
if a in nome #Está dentro  
  
if a in lista  
  
if is True #É verdadeiro  
  
if True
```



■ Condicionais elif: e else:

Aqui temos a função ELSE (que seria o senão). Primeiramente iremos testar a informação do IF e se ela não for verdadeira nós vamos para opção else.

Agora teremos 2 resultados para essa nossa comparação, ou seja, duas alternativas.

Funciona basicamente assim:

Se a primeira opção (condição) for verdadeira: Execute.

Senão, execute a segunda opção (condição).

Só que algumas vezes nós precisamos de mais de 2 resultados (condições), como por exemplo, um semáforo de trânsito.

Se a luz verde estiver acesa, podemos passar, luz amarela precisamos de atenção e no vermelho precisamos parar.

Ao invés de utilizar 3 vezes a função IF ou usar IF ELSE e depois outro IF, nós temos a função ELIF, que seria basicamente a junção de um ELSE + IF.

ELIF

```
cor = "alguma cor"

if cor == 'verde':
    print('Acelerar')

elif cor == 'amarelo':
    print('Atenção')

else:
    print('Parar')
```

Parar



Hora de praticar

- Faça um Programa que peça dois números ao usuário e imprima o maior deles.
- Faça um Programa que peça ao usuário um valor e mostre na tela se o valor é positivo ou negativo.
- Faça um Programa que verifique se uma letra digitada é "F" ou "M". Conforme a letra escrever: F - Feminino, M - Masculino, Sexo Inválido.
- Faça um Programa que verifique se a letra digitada pelo usuário é vogal ou consoante.
- Faça um programa para a leitura de quatro notas de um aluno. O programa deve calcular a média alcançada apresentar:
 - a. A mensagem "Aprovado", se a média alcançada for maior ou igual a sete;
 - b. A mensagem "Reprovado", se a média for menor do que sete;
 - c. A mensagem "Aprovado com Distinção", se a média for igual a dez.
- Faça um Programa que receba 2 números e em seguida pergunte ao usuário qual operação ele deseja realizar. O resultado da operação deve aparecer com uma frase que diga se o número é:
 - a. par ou ímpar;
 - b. positivo ou negativo;
 - c. inteiro ou decimal.



Hora de praticar

- Faça um programa que faça 5 perguntas para uma pessoa sobre um crime. As perguntas são:
 - a. "Telefonou para a vítima?"
 - b. "Esteve no local do crime?"
 - c. "Mora perto da vítima?"
 - d. "Devia para a vítima?"
 - e. "Já trabalhou com a vítima?"O programa deve no final emitir uma classificação sobre a participação da pessoa no crime. Se a pessoa responder positivamente a 2 questões ela deve ser classificada como "Suspeita", entre 3 e 4 como "Cúmplice" e 5 como "Assassino". Caso contrário, ele será classificado como "Inocente".
- Um posto está vendendo combustíveis com a seguinte tabela de descontos:
 - a. Álcool:
 - b. até 20 litros, desconto de 3% por litro
 - c. acima de 20 litros, desconto de 5% por litro
 - d. Gasolina:
 - e. até 20 litros, desconto de 4% por litro
 - f. acima de 20 litros, desconto de 6% por litroEscreva um algoritmo que leia o número de litros vendidos, o tipo de combustível (codificado da seguinte forma: A-álcool, G-gasolina), calcule e imprima o valor a ser pago pelo cliente sabendo-se que o preço do litro da gasolina é R\$ 2,50 o preço do litro do álcool é R\$ 1,90.



Você concluiu a aula 01 do seu módulo de
Python.
Continue praticando e até a próxima aula!



INFINITY SCHOOL

V I S U A L A R T C R E A T I V E C E N T E R