

Aula prática 6



Escola
Politécnica

Lógica de Programação e Algoritmos

1
27

Prof. Vinicius Pozzobon Borin

Pré-requisitos desta aula

- **PyCharm instalado e testado**
- **Até a aula teórica 6 assistida e material em texto lido**

- Os exercícios resolvidos ao longo desta aula não terão seu código disponibilizado para vocês
- Por quê? Para que vocês mesmos digitem e pratiquem por conta própria

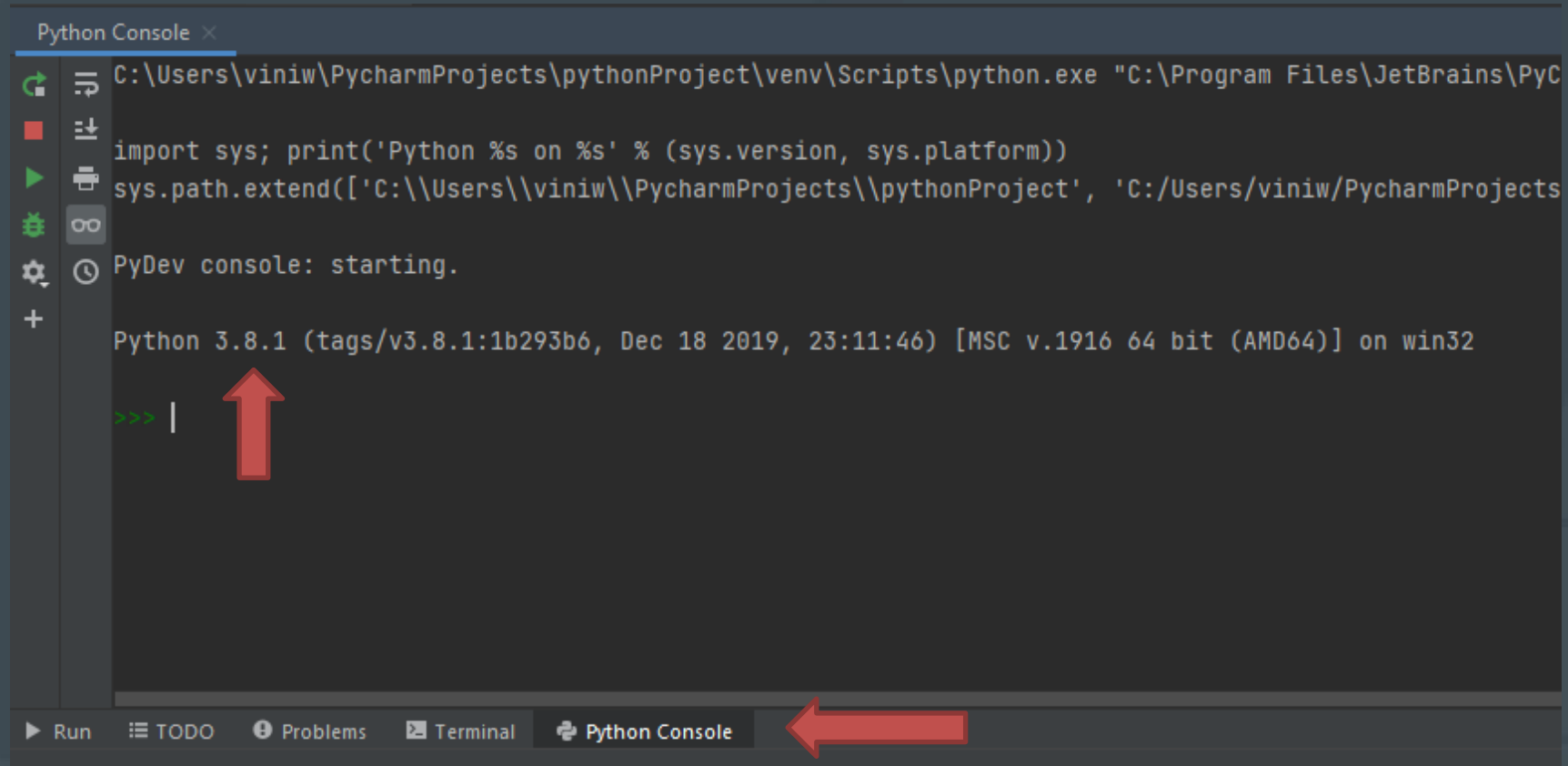


Importando bibliotecas

Importando bibliotecas

- **Importa ao seu programa novas funções**
- **Elas podem ser pré-definidas da linguagem, ou criadas por outra pessoa ou por você mesmo**

1. Encontre a versão do seu Python:



The screenshot shows the Python Console window in PyCharm. The console output displays the Python version and platform information. A red arrow points to the version number '3.8.1' in the output. Another red arrow points to the 'Python Console' tab in the bottom toolbar.

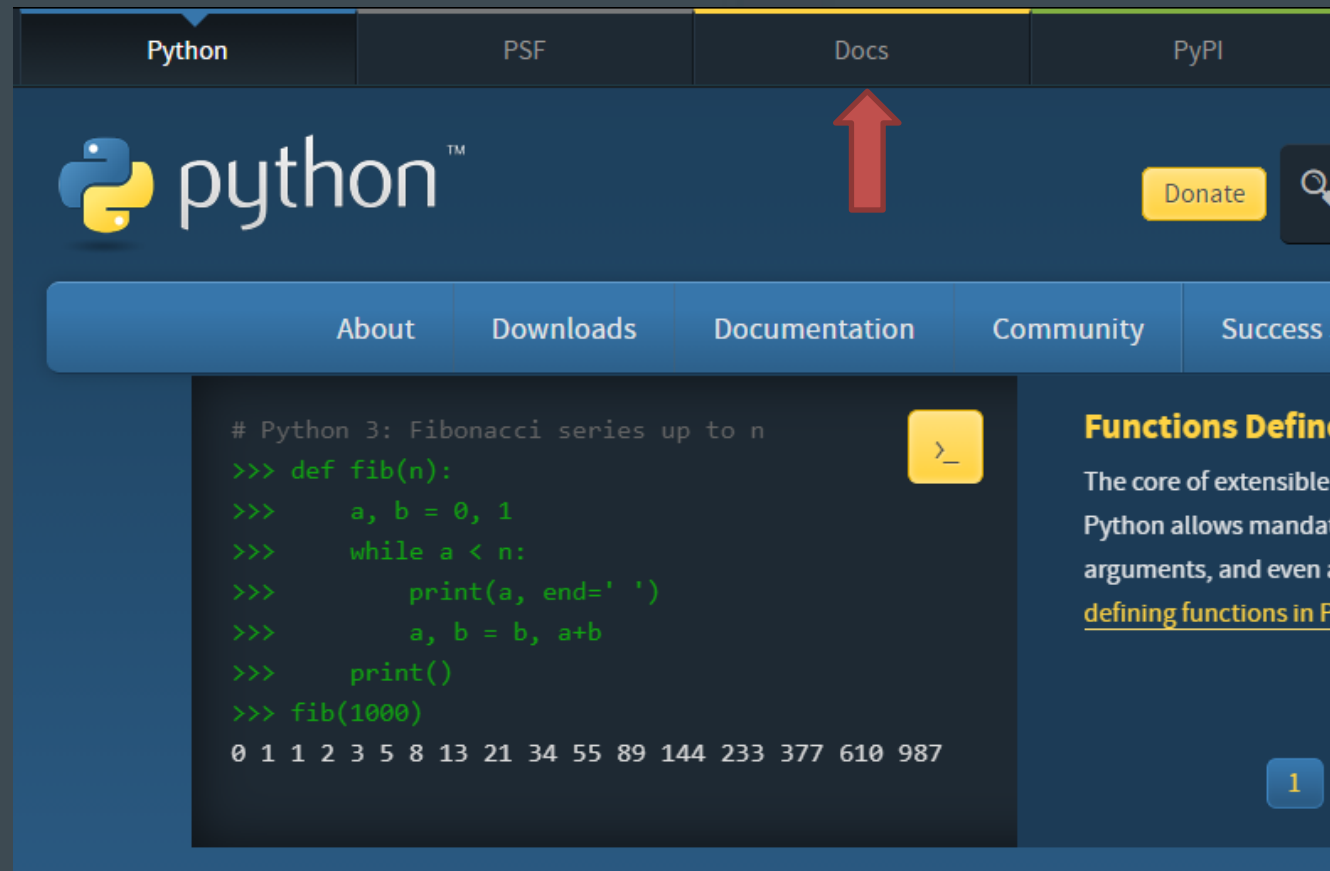
```
Python Console x
C:\Users\viniw\PycharmProjects\pythonProject\venv\Scripts\python.exe "C:\Program Files\JetBrains\PyC
import sys; print('Python %s on %s' % (sys.version, sys.platform))
sys.path.extend(['C:\\Users\\viniw\\PycharmProjects\\pythonProject', 'C:/Users/viniw/PycharmProjects
PyDev console: starting.

Python 3.8.1 (tags/v3.8.1:1b293b6, Dec 18 2019, 23:11:46) [MSC v.1916 64 bit (AMD64)] on win32

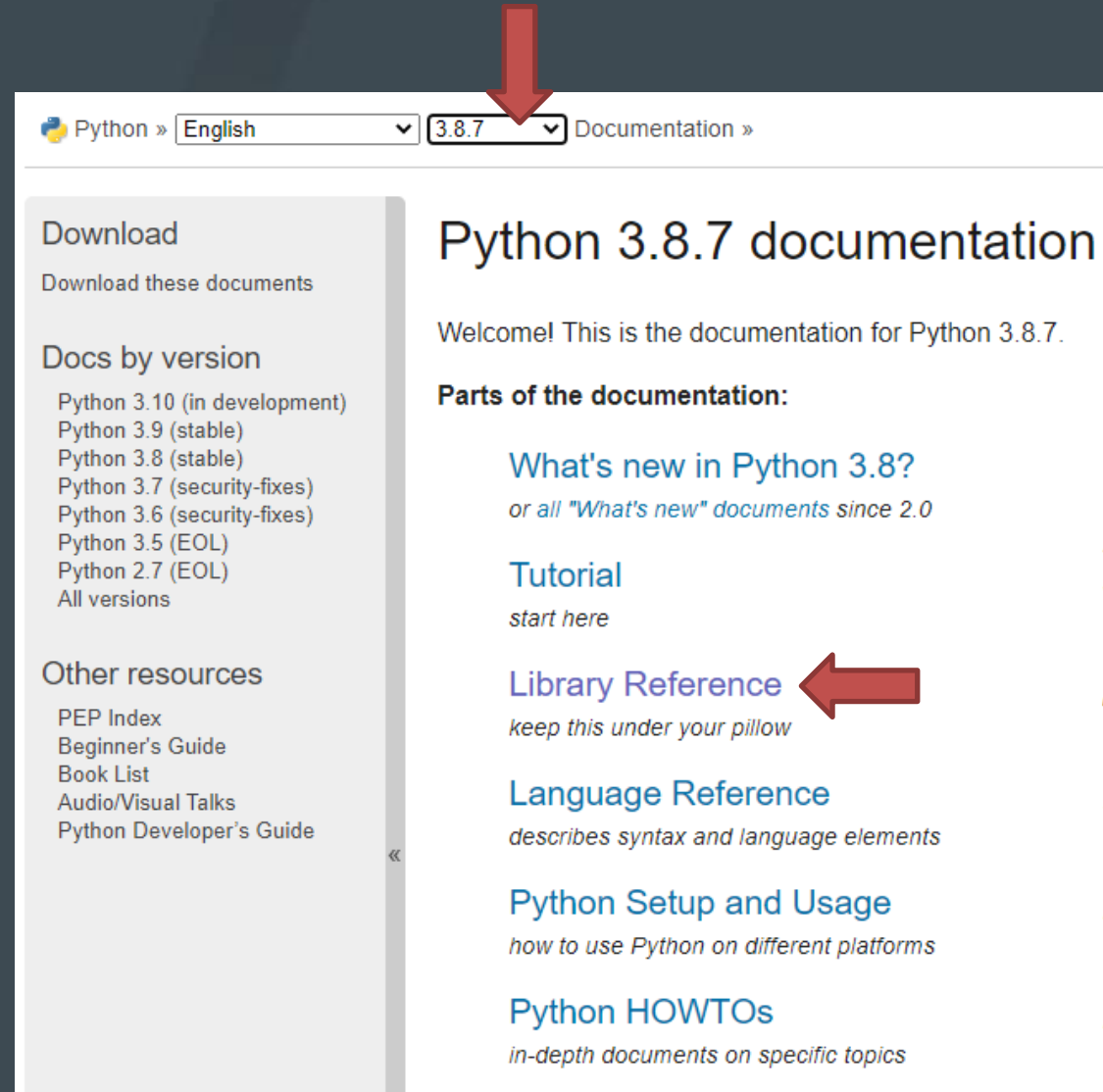
>>> |
```

Run | TODO | Problems | Terminal | Python Console

2. Acesse Python.org e vá em Docs



3. Selecione a versão do seu Python e vá em Library Reference



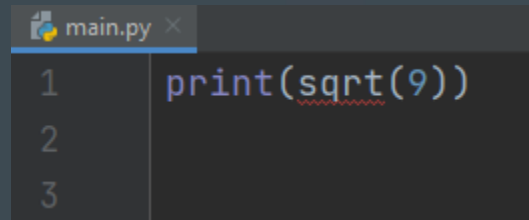
The screenshot shows the Python 3.8.7 documentation page. At the top, there is a navigation bar with "Python »", a language dropdown set to "English", a version dropdown set to "3.8.7" (highlighted by a red arrow), and "Documentation »". The main heading is "Python 3.8.7 documentation". Below it, a welcome message says "Welcome! This is the documentation for Python 3.8.7." A section titled "Parts of the documentation:" lists several links: "What's new in Python 3.8?" (with a subtitle "or all 'What's new' documents since 2.0"), "Tutorial" (with a subtitle "start here"), "Library Reference" (with a subtitle "keep this under your pillow" and a red arrow pointing to it), "Language Reference" (with a subtitle "describes syntax and language elements"), "Python Setup and Usage" (with a subtitle "how to use Python on different platforms"), and "Python HOWTOs" (with a subtitle "in-depth documents on specific topics"). On the left side, there is a sidebar with sections: "Download" (with a link "Download these documents"), "Docs by version" (listing Python 3.10 (in development), Python 3.9 (stable), Python 3.8 (stable), Python 3.7 (security-fixes), Python 3.6 (security-fixes), Python 3.5 (EOL), Python 2.7 (EOL), and All versions), and "Other resources" (listing PEP Index, Beginner's Guide, Book List, Audio/Visual Talks, and Python Developer's Guide).

4. Selecione a biblioteca desejada para aprender o que tem nela

- weakref — Weak references
- types — Dynamic type creation and names for built-in types
- copy — Shallow and deep copy operations
- pprint — Data pretty printer
- reprlib — Alternate repr() implementation
- enum — Support for enumerations
- Numeric and Mathematical Modules
 - numbers — Numeric abstract base classes
 - math — Mathematical functions
 - cmath — Mathematical functions for complex numbers
 - decimal — Decimal fixed point and floating point arithmetic
 - fractions — Rational numbers
 - random — Generate pseudo-random numbers
 - statistics — Mathematical statistics functions
- Functional Programming Modules
 - itertools — Functions creating iterators for efficient looping
 - functools — Higher-order functions and operations on callable objects
 - operator — Standard operators as functions
- File and Directory Access
 - pathlib — Object-oriented filesystem paths
 - os.path — Common pathname manipulations
 - fileinput — Iterate over lines from multiple input streams
 - stat — Interpreting stat() results
 - filecmp — File and Directory Comparisons
 - tempfile — Generate temporary files and directories
 - glob — Unix style pathname pattern expansion

Importando bibliotecas

- Tente realizar o *print* da raiz quadrada de 9:



```
main.py x
1 print(sqrt(9))
2
3
```

- O que acontece? Erro:

```
C:\Users\viniw\PycharmProjects\pythonProject\venv\Scripts\python.exe C:/Users/viniw/PycharmProjects/pythonProject/main.py
Traceback (most recent call last):
  File "C:/Users/viniw/PycharmProjects/pythonProject/main.py", line 1, in <module>
    print(sqrt(9))
NameError: name 'sqrt' is not defined

Process finished with exit code 1
```

Importando bibliotecas

■ **Correto:**

```
1 import math
2
3 print(math.sqrt(9))
4
5
```

■ **Saída:**

```
C:\Users\viniw\PycharmProjects\pythonProject\venv\Scripts>python
3.0

Process finished with exit code 0
```

Importando bibliotecas

- Dando um apelido à biblioteca:

```
1 import math as m
2
3 print(m.sqrt(9))
4
5
```

- Importando somente a função desejada:

```
1 from math import sqrt
2
3 print(sqrt(9))
4
5
```



Relembrando

13
27



Tupla

- Estrutura de dados **estática**
- A tupla é **imutável**
- Representada em Python por **parênteses ()**

Listas

- Estrutura de dados **dinâmica**
- Podemos alterar dados e tamanho
- Indexadas por valores numéricos inteiros
- Representadas em Python por **colchetes []**

Alguns métodos para listas

Função/método	
append(item)	Adiciona um item ao final da lista
count(item)	Retorna o número de ocorrências de <i>item</i> na lista
index(item)	Retorna o índice da primeira ocorrência de <i>item</i>
insert(índice, item)	Insere <i>item</i> no respectivo índice
pop()	Remove o último item
remove(item)	Remove a primeira ocorrência de <i>item</i> na lista
reverse()	Inverte a ordem da lista
sort()	Ordena a lista

Dicionários

- Estrutura de dados **dinâmica**
- Podemos alterar dados e tamanho
- Indexados por chaves (palavras-chave)
- Representados em Python por **chaves** `{}`

Métodos para dicionários

- ***values***: obtém somente os dados
- ***keys***: obtém somente as chaves
- ***items***: obtém o par chave:dado

Relação de métodos para uso com *strings*

Função/método	Objetivo
startswith	Verifica se caracteres existem no início da <i>string</i>
endswith	Verifica se caracteres existem no final da <i>string</i>
lower	Converte <i>string</i> para minúscula
upper	Converte <i>string</i> para maiúscula
find	Busca a primeira ocorrência de um padrão de caracteres em uma <i>string</i>
rfind	Idêntico ao <i>find</i> , mas inicia a busca da direita para a esquerda
center	Centraliza uma <i>string</i>
ljust, rjust	Ajustam uma <i>string</i> com alinhamentos à esquerda ou à direita, respectivamente
split	Divide uma <i>string</i>
replace	Substitui caracteres em uma <i>string</i>
lstrip, rstrip	Removem espaços em branco à esquerda ou à direita, respectivamente
strip	Remove espaços em branco das extremidades

Relação de métodos para validação de dados em *strings*

Função/método	Retorna <i>True</i> para uma <i>string</i> com...
isalnum	Somente letras e números; acentos são aceitos
isalpha	Somente letras; acentos são aceitos
isdigit	Somente números
isnumeric	Somente números; aceita também caracteres matemáticos, como frações
isupper	Somente caracteres maiúsculos
islower	Somente caracteres minúsculos
isspace	Somente espaços. Inclui TAB, quebra de linha, retorno etc.
isprintable	Somente caracteres possíveis de serem impressos na tela



Exercícios de fixação

Lista

- Dada uma lista contendo as notas de alunos em uma disciplina, escreva uma expressão para:

`notas = [9, 7, 7, 10, 3, 9, 6, 6, 2]`

- a) Encontrar quantos alunos tiraram nota 7
- b) Alterar a última nota para 4

Lista

notas = [9, 7, 7, 10, 3, 9, 6, 6, 2]

- c) Encontrar a maior nota
- d) Ordenar a lista de notas
- e) A média das notas



Problemas

24
27



Exercício 1

- Escreva um algoritmo que crie uma tupla com 10 palavras. Encontre dentro dessa tupla as vogais de cada palavra. Faça um *print* na tela com o nome da palavra e suas respectivas vogais

Exercício 2

- **Crie um jogo de pedra, papel ou tesoura (Jokenpô). Você deverá jogar contra o computador. Você irá sempre escolher uma das opções: 1- pedra, 2 – papel, 3 – tesoura**
- **O computador irá sempre sortear um número de 1 até 3 para jogar**
- **Armazene todos os resultados em uma lista e no final apresente o vencedor**
- **Encerre o programa ao digitar zero**

Exercício 3

- **Crie um programa para ler o nome, ano de nascimento e sexo de diferentes pessoas**
- **Armazene os dados em um dicionário com listas**
- **Ao encerrar o cadastro, apresente:**
 - **O total de cadastros efetuados**
 - **A média das idades das pessoas**
 - **Uma lista de mulheres com menos de 30 anos**
 - **Uma lista de homens com idade acima da média**





Fechar