

Framework

Luiza
<CODE>

O que vamos ver?

- Módulos e pacotes
 - Próprios
 - Atividade 1
 - Nativos
 - Atividade 2
 - Terceiros (PIP)
 - Atividade 2
- FastAPI
 - *Teoria* e prática
 - Atividade 3



Módulos e pacotes

Módulos próprios

- Arquivo `.python`
- Nome do arquivo é o nome do módulo.
- São *importados* por outros *scripts*:
 - `import nome-modulo`
 - `from nome-modulo import`
- Suas *definições* são acessadas pela notação de ponto.

```
# script06_01
def funcao01():
    print("Funcao 01 em ", __name__)

if __name__ == "__main__":
    print("script06_01")
    funcao01()
```

```
import script06_01
# from script06_01 import funcao01

print("script 06_02")
script06_01.funcao01()
```

Módulos nativos

- O Python possui vários módulos e pacotes pré-definidos (bibliotecas).
- <https://docs.python.org/3/py-modindex.html>
 - **json**: Codificação e decodificação de JSON.
 - **decimal**: Números decimais de propósito geral.
 - **os**: Interfaces de comunicação com o sistema operacional.

```
import os

VAR_PATH = os.environ.get('PATH')
existe = os.path.exists("script01.py")

from datetime import date, datetime, timedelta
amanha = date.today() + timedelta(days=1)
agora = datetime.now()
dia_semana = agora.weekday
```

Atividade 1

- Criar um script que converte um arquivo .CSV com cabeçalho para um arquivo JSON.

```
nome,idade
Fulana,17
Beltrana,45
Ciclana,20
```



```
[{
  "nome": "Fulana",
  "idade": "17"
},
{
  "nome": "Beltrana",
  "idade": "45"
},
{
  "nome": "Ciclana",
  "idade": "20"
}]
```


Módulos de terceiros

- Podemos *instalar* pacotes de terceiros e utilizá-los em nossos scripts.
- Ferramenta: **pip**
 - <https://pypi.org/project/pip>

```
C:\Users\ozair_a\treinamento02>c:\opt\python3.9\Scripts\pip install requests
Collecting requests
  Using cached requests-2.25.1-py2.py3-none-any.whl (61 kB)
Collecting certifi<=2017.4.17
  Downloading certifi-2021.5.30-py2.py3-none-any.whl (145 kB)
    | 145 kB 3.3 MB/s
Collecting idna<3,>=2.5
  Using cached idna-2.10-py2.py3-none-any.whl (58 kB)
Collecting chardet<5,>=3.0.2
  Using cached chardet-4.0.0-py2.py3-none-any.whl (178 kB)
Collecting urllib3<1.27,>=1.21.1
  Downloading urllib3-1.26.5-py2.py3-none-any.whl (138 kB)
    | 138 kB 3.3 MB/s
Installing collected packages: urllib3, idna, chardet, certifi, requests
  WARNING: The script chardet.exe is installed in 'c:\opt\python3.9\Scripts' which is not on PATH.
  Consider adding this directory to PATH or, if you prefer to suppress this warning, use --no-warn-script-location.
Successfully installed certifi-2021.5.30 chardet-4.0.0 idna-2.10 requests-2.25.1 urllib3-1.26.5
WARNING: You are using pip version 21.0.1; however, version 21.1.2 is available.
You should consider upgrading via the 'c:\opt\python3.9\python.exe -m pip install --upgrade pip' command.

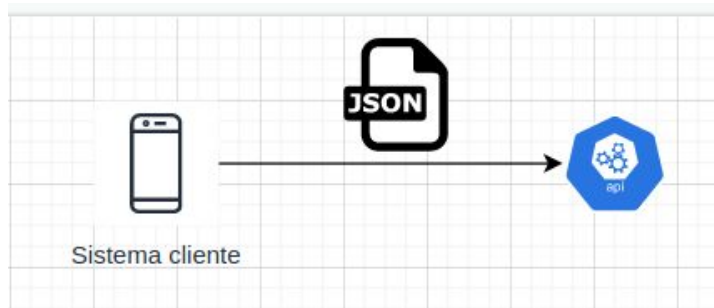
C:\Users\ozair_a\treinamento02>dir c:\opt\python3.9\Lib\site-packages\requests*
O volume na unidade C é OS
O Número de Série do Volume é FC01-8016

Pasta de c:\opt\python3.9\Lib\site-packages

13/06/2021  14:21    <DIR>          requests
13/06/2021  14:22    <DIR>          requests-2.25.1.dist-info
```

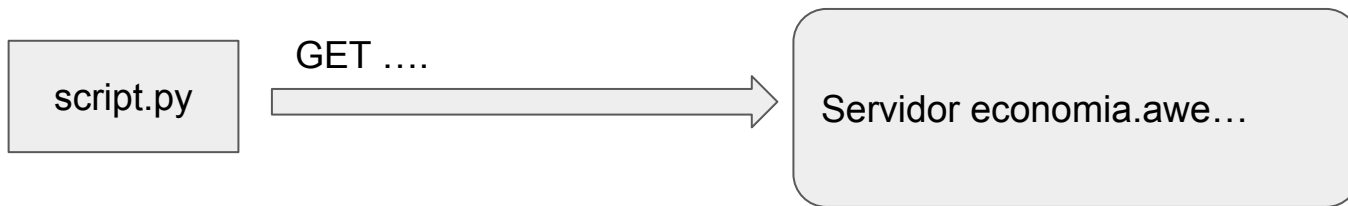
Pré Atividade 2: API Rest

- API (*Application Programming Interface*): Conjunto de normas que possibilita a comunicação entre plataformas através de uma série de padrões e protocolos ([Tchutado](#))
- REST (*Representational State Transfer*): Conjunto de restrições para aplicações Web (*Red Hat*).
- API REST: Conjunto de restrições e normas para comunicação entre sistemas HTTP(s)/Web.



Atividade 2

- Criar um script que lê de uma API o valor da cotação do dólar de hoje.
- Fonte: <https://economia.awesomeapi.com.br/json/last/USD-BRL>
- Biblioteca/Pacote: <https://pypi.org/project/requests/>

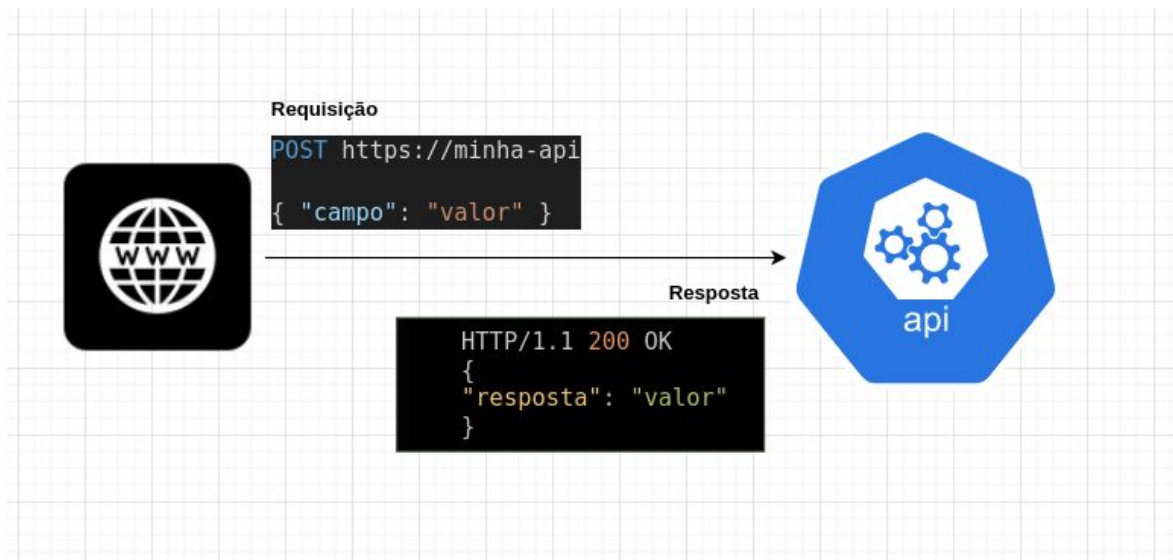




API Rest e FastAPI

API Rest

- Para a comunicação:
 - Métodos (GET, POST)
 - Caminhos: "api/produtos".
 - Parâmetros



Frameworks

- Bibliotecas para *desenvolvimento rápido* para Web (Api Rest).



FastAPI

- FastAPI: *Framework* Web para desenvolvimento de APIs Rest.
- <https://fastapi.tiangolo.com/>

```
bash

$ pip install fastapi
```

```
bash

$ pip install "uvicorn[standard]"
```




Atividade 3

(Fase 1)

Atividade 3 - Fase 1

- Cadastro de músicas para uma playlist.
- Um música possui:
 - `codigo`: Chave do sistema para identificar a música.
 - `nome`: Nome da música.
 - `artista`: Nome do artista/banda.
 - `tempo`: Tempo da música em segundos (opcional).
- Cadastro da música (Fase 1):
 - Registrar uma música.
 - Pesquisar todas as músicas.
 - Pesquisar uma música pelo código.

Atividade 3 - Fase 1

- Script com código único.
- Código organizado e estruturado em:
 - Funções para o controlador.
 - Funções de negócio (caso de uso)
 - Funções de persistência/repositório.
 - Vamos salvar na memória.

Magalu



#VemSerFeliz