Name	Name	Last commit date
..		
exam	Updated (Desarrollo Backend con Pyt...	2 years ago
scripts	Updated (Desarrollo Backend con Pyt...	2 years ago
shared	Updated (Desarrollo Backend con Pyt...	2 years ago
README.md	Add (Desarrollo Backend con Python)...	last year

README.md

Curso de Python: PIP y Entornos Virtuales

⭐ Comandos Basicos

Comandos básicos en la terminal, con esto iniciamos el proyecto:

- **pwd:** Indica en qué ubicación estamos
- **mkdir:** Crear una nueva carpeta
- **ll:** Lista de archivos
- **cd:** Nos permite abrir una carpeta
- **clear:** Nos permite despejar la terminal
- **git init:** Inicializar
- **touch:** Crear archivos
- **CTRL+L:** Limpia la terminal

Otros comandos

- **rm:** sirve para borrar archivos
- **rmdir:** sirve para borrar directorios
- **mv:** sirve para mover directorios
- **df:** indica el espacio en disco, el disponible, usado y total

🍎 Instalacion WSL en Mac

Comandos utilizados

- python o python3
- exit()

Normalmente viene instalado en Mac, en caso de que no lo tenga continuar con estos comandosHerramientas de codigo

- sudo xcode-select --install
- sudo xcode-select --reset

Instalación de python

- brew install python3

Verificar la Instalación

- python3

```
sudo xcode-select --install
sudo xcode-select --reset

brew install python3

python3
```

Python con VSCode

Descargamos VSCode de microsoft store e instalamos las extensiones

- Python
- WSL

Luego podemos abrir la terminal de linux y ejecutar VSCode con el codigo `code .` , creamos un archivo con la extension `.py` y ya podremos ejecutarlo desde la terminal con el codigo `python3 [nombre.py](http://nombre.py)`

¿Que es pip?

PIP es el gestor de paquetes de python, es posible buscar librerías en la pagina pypi.org.

- Ver la versión de pip `pip3 -V` .
- Instalación de paquetes `pip3 install <libreria>` .
- Listar las librerías que se tienen en el entorno de python global `pip3 list` .
- Listar todas las librerías de python instaladas por el usuario `pip3 freeze` .

⚠ If you get this error, try `sudo apt-get update` and then install the package

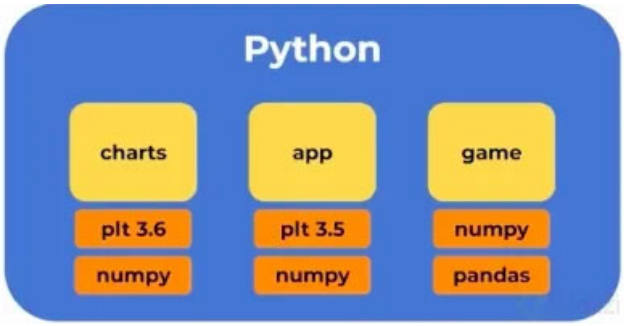
- E: Failed to fetch http://security.ubuntu.com/ubuntu/pool/main/e/expat/libexpat1-dev_2.2.9-1ubuntu0.4_amd64.deb 404 Not Found [IP: 185.125.190.36 80]

- E: Unable to fetch some archives, maybe run apt-get update or try with --fix-missing?

Ambientes virtuales

Instalar a nivel global puede causar distintos problemas al momento de manejar diferentes proyectos, por ejemplo para algunos proyectos necesitaras otro tipo de version, libreria o modulos y para solucionar esto se creo un ambiente virtual en python el cual encapsula cada proyecto y no lo deja de forma compartida.

💡 Para instalar una version diferente de una libreria usamos el Código `pip3 install ==`



Pasos para crear un ambiente virtual

Verificar donde esta python y pip

- `which python3`
- `which pip3`

Si estas en linus o wsl debes instalar

- `sudo apt install -y python3-venv`

Crear ambiente virtual (Poner cada proyecto en su propio ambiente, entrar en cada carpeta)

- `python3 -m venv env`

Activar el ambiente

- `source env/bin/activate`
- `source venv/Scripts/activate` for Windows

Salir del ambiente virtual

- `deactivate`

Podemos instalar las librerias necesarias en el ambiente virtual como por ejemplo

- `pip3 install matplotlib==3.5.0`

Verificar las instalaciones

- `pip3 freeze`

Automatizacion de instalacion de modulos o librerias en ambiente virtual

Requirements.txt = Archivo que gestiona todas las dependencias y en que versiones se necesitan.

Generar el archivo con el siguiente comando

- `pip3 freeze > requirements.txt`

💡 **ESTO ES IMPORTANTE**, debemos mantener el archivo de requirements.txt con todos los modulos utilizados
Revisar lo que hay dentro del archivo

- `cat requirements.txt`

Instalar las dependencias necesarias para contribuir más rápido en proyectos

- `pip3 install -r requirements.txt`

Preparar archivo para contribución

Solicitudes HTTP con Requests

Podemos consultar informacion de una URL con el modulo `requests` se instala con `pip3 install requests` .

```
import requests

def get_categories():
    r = requests.get('https://api.escuelajs.co/api/v1/categories')
    print(r.status_code)
    print(r.text)
    print(type(r.text))
    categories = r.json()
    for category in categories:
        print(category['name'])
```

```
import store

def run():
    store.get_categories()

if __name__ == '__main__':
    run()
```

Python para Backend: web server con FastAPI

FastAPI nos permite crear un servidor con el que podemos interactuar, para instalarlo seguimos estos pasos:

- `pip3 install fastapi`
- `pip3 install "uvicorn[standard]"`

En el archivo `[main.py](http://main.py)` escribimos el siguiente codigo para enviar una peticion

```
from fastapi import FastAPI

app = FastAPI()

@app.get('/')
def get_list():
    return [1,2,3,4]
```

Luego para inicializar el servidor en la terminal escribimos:

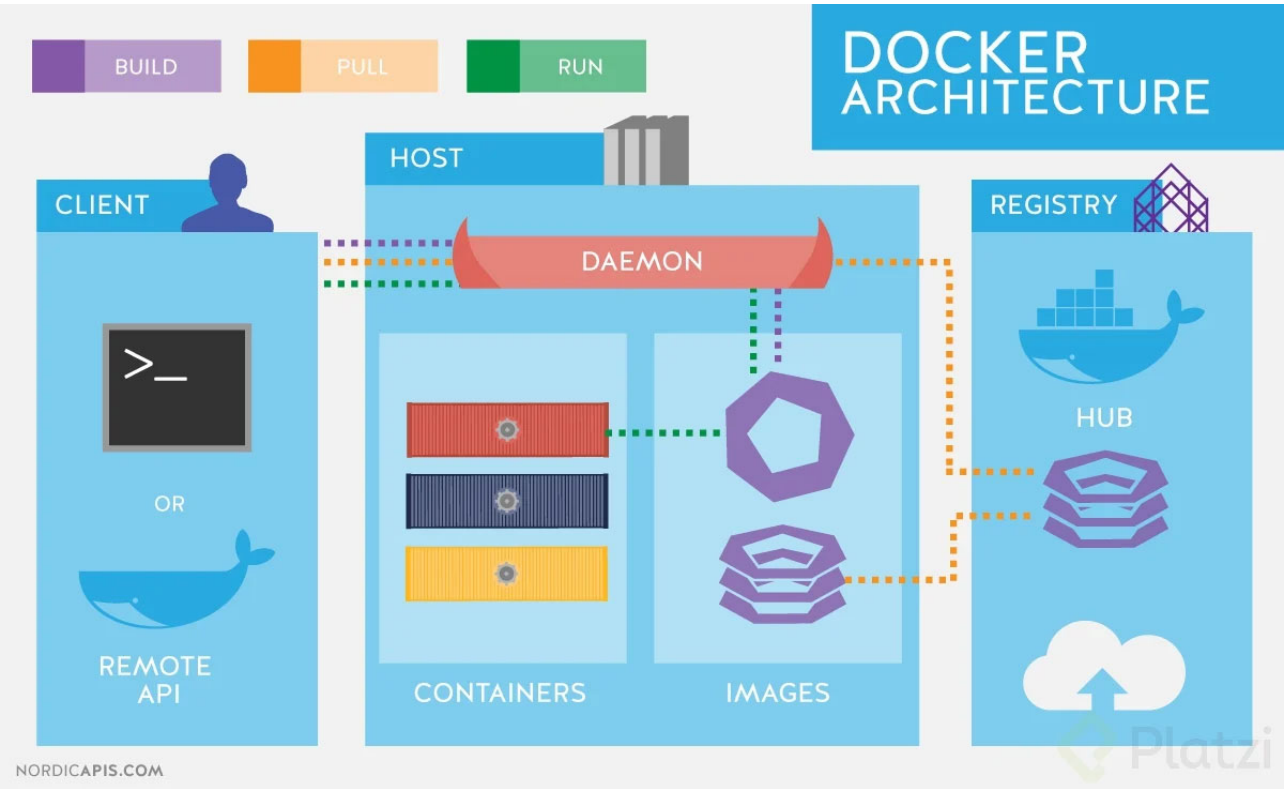
- `uvicorn main:app --reload`

¿Qué es Docker?

Es una herramienta que nos sirve para aislar entornos de ejecucion, eso lo hace con contenedores que tiene docker por detras.

Instalacion de Docker

Instalación de Docker - Platzi



Dockerizando scripts de Python

```
FROM python:3.8

WORKDIR /app
COPY requirementes.txt /app/requirementes.txt

RUN pip install --no-cache-dir --upgrade -r /app/requirementes.txt

COPY . /app/

CMD bash -c "while true; do sleep 1; done"
```

archivo *docker-compose.yml*

```
services:
  app-csv:
    build:
      context: .
      dockerfile: Dockerfile
```

💡 Debemos tener inicializado el docker
Para construir el contenedor

```
docker-compose build
```

Para lanzar digitamos el comando

```
docker-compose up -d
```

Para ver el estado del contenedor

```
docker-compose ps
```

Para ejecutar el contenedor

```
docker-compose exec app-csv bash
```

Docker para el día a día: automatizando la vinculación de archivos

Esta automatizacion nos permite reflejar los cambios en el docker sin necesidad de salir y recrear el mismo, para ello solo debemos agregar a nuestro archivo `docker-compose.yml` lo siguiente:

```
volumes:
      - .:/app
```

Y ya con esto todos los cambios que apliquemos en nuestro contenedor se veran reflejados en tiempo real