



ORGANISMO EMISOR:

ASUNTO: Guía instalación Python-Streamlit

REFERENCIA:

FECHA: 14/03/2025

REV.: 1

ÍNDICE

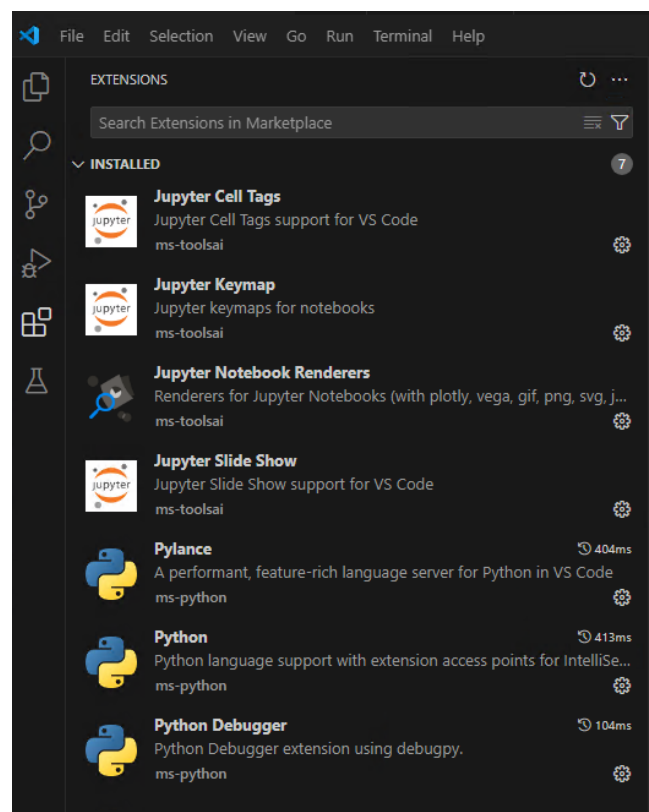
1.	Instalación python – visualstudiocode	2
2.	Ejecución de código sin firmar.....	3
3.	Creación del entorno virtual.....	3
4.	Kernel de jupyter Notebook	4

1. INSTALACIÓN PYTHON – VISUALSTUDIOCODE


Para la ejecución de del código sería suficiente con instalar Python (actualmente se está trabajando con la 3.11). No obstante, para poder editar el código y revisarlo si aparece algún error, es recomendable instalar el Visual Studio Code.

- <https://code.visualstudio.com/docs/?dv=win>
- <https://www.python.org/downloads/>

Hay varias extensiones que permiten ejecutar el código desde visual studio y facilitan la edición. Las recomendaciones son:



*Además, podrían instalarse paquetes de idiomas, temas de visualización...

Para los ordenadores corporativos, la instalación es bastante sencilla. Dentro de visual studio, en la barra lateral izquierda hay un icono  que permite realizar las instalaciones.

2. EJECUCIÓN DE CÓDIGO SIN FIRMAR

En general, la primera vez que se instala Python en un equipo tiene restringido la ejecución de código sin firmar. Sin embargo, todos los desarrollos no van a estar firmados y será necesario eliminar esta restricción. Para ello, se puede acceder a powershell para cambiarlo o ejecutar directamente en un terminal la siguiente instrucción:

```
powershell -Command "Set-ExecutionPolicy Unrestricted -Scope CurrentUser -Force"
```

3. CREACIÓN DEL ENTORNO VIRTUAL

Para crear un entorno virtual, es necesario tener instalada la librería virtualenv. En la carpeta de instalación están los archivos necesarios y bastaría con ejecutar:

```
pip install -r requirements_virtual_env.txt
```

A continuación, hay que generar el entorno virtual con el nombre deseado. Por ejemplo, con el nombre “desarrollo” podría ejecutarse:

```
python -m venv env_desarrollo
```

Se ha denominado “env_desarrollo” para que quede claro que es un entorno y no una aplicación o un kernel.

Una vez creado, debe activarse el entorno:

```
env_desarrollo\Scripts\activate
```

Y por último instalar las librerías necesarias. En la carpeta de instalación están los archivos necesarios y bastaría con ejecutar:

```
pip install -r requirements.txt
```

4. KERNEL DE JUPYTER NOTEBOOK

En caso de utilizarse Jupyter Notebook como plataforma de desarrollo, es necesario crear un kernel asociado al entorno virtual. Para ello hay que ejecutar:

```
python -m ipykernel install --user --name env_desarrollo --display-name "Kernel desarrollo"
```