**Como instalar Python, ejecutar un proyecto y generar un exe.**

**Contenido**

[PASO 1: 3](#_Toc173143786)

[PASO 2 (Instalar PIP): 3](#_Toc173143787)

[PASO 3 (Descargar un IDE): 4](#_Toc173143788)

[PASO 4 (Instalar dependencias de un proyecto): 4](#_Toc173143789)

[PASO 5 (Ejecuta el programa): 5](#_Toc173143790)

[Conceptos: 6](#_Toc173143791)

**Aclaraciones:**

En el caso que únicamente desees generar el exe, puedes saltarte los pasos 3 y 4

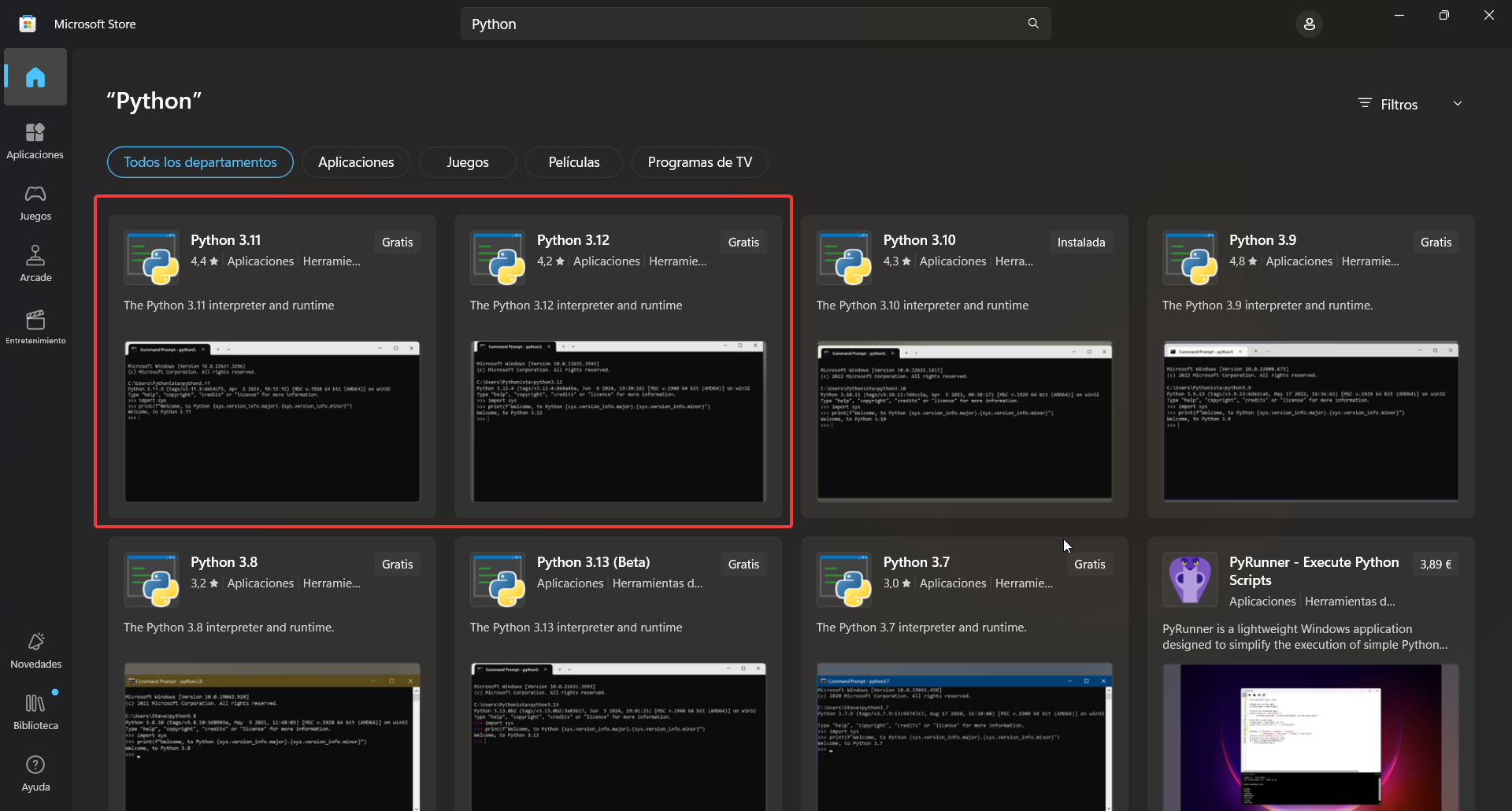
**Objetivo:**

Esta documentación ha sido escrita para demostrar paso por paso como ejecutar, modificar un proyecto escrito en Python (“.py”) y como generar un ejecutable del mismo. Por otro lado también se detallan procesos a seguir para no tener errores durante el proceso.

# PASO 1:

**Opción 1 (Mas rápida):**

Instala Python desde la Microsoft Store.

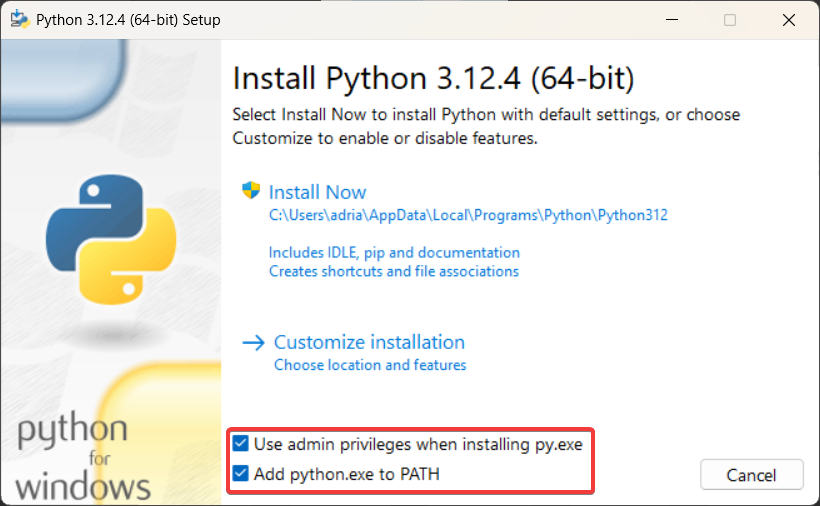


(Las dos opciones son igual de validas).

**Opción 2:**

Instala Python desde la página oficial (Python.org).

Link: <https://www.python.org/ftp/python/3.12.4/python-3.12.4-amd64.exe>



**MUY IMPORTANTE:**

Para que funcione PIP y Python en Windows se deben marcar las opciones que salen marcadas en rojo.

# PASO 2 (Instalar PIP):

PIP es una dependencia de Python la cual se utiliza para instalar paquetes hechos por la comunidad los cuales brindan funcionalidades y herramientas al lenguaje.

1. Dirígete a la carpeta **PIP** y copia el archivo “get-pip.py” en tu carpeta local.
2. Usando CMD dirígete a la carpeta donde tienes el archivo y ejecuta el siguiente comando:

**py get-pip.py**

1. **Como verificar que se ha instalado correctamente:**

En CMD:

**pip –version**

Si no detecta el comando como propio del sistema, significa que no se instaló correctamente. En caso contrario significa que se instaló correctamente.

# PASO 3 (Descargar un IDE):

Este paso es únicamente necesario si quieres modificar el código. Hay una gran variedad de IDEs, pero el mas popular es Visual Studio Code (“vscode”).

Link: <https://code.visualstudio.com/sha/download?build=stable&os=win32-x64-user>

**En la instalación selecciona la opción para asignarlo en el path.**

Para abrir una carpeta rápidamente con vscode, dirígete a la misma desde CMD y escribe el siguiente comando:

**code .** (“Abre la carpeta directamente”)

# PASO 4 (Instalar dependencias de un proyecto):

1. **Dirígete a la carpeta del proyecto.**
2. **RECOMENDADO (Da menos errores).**

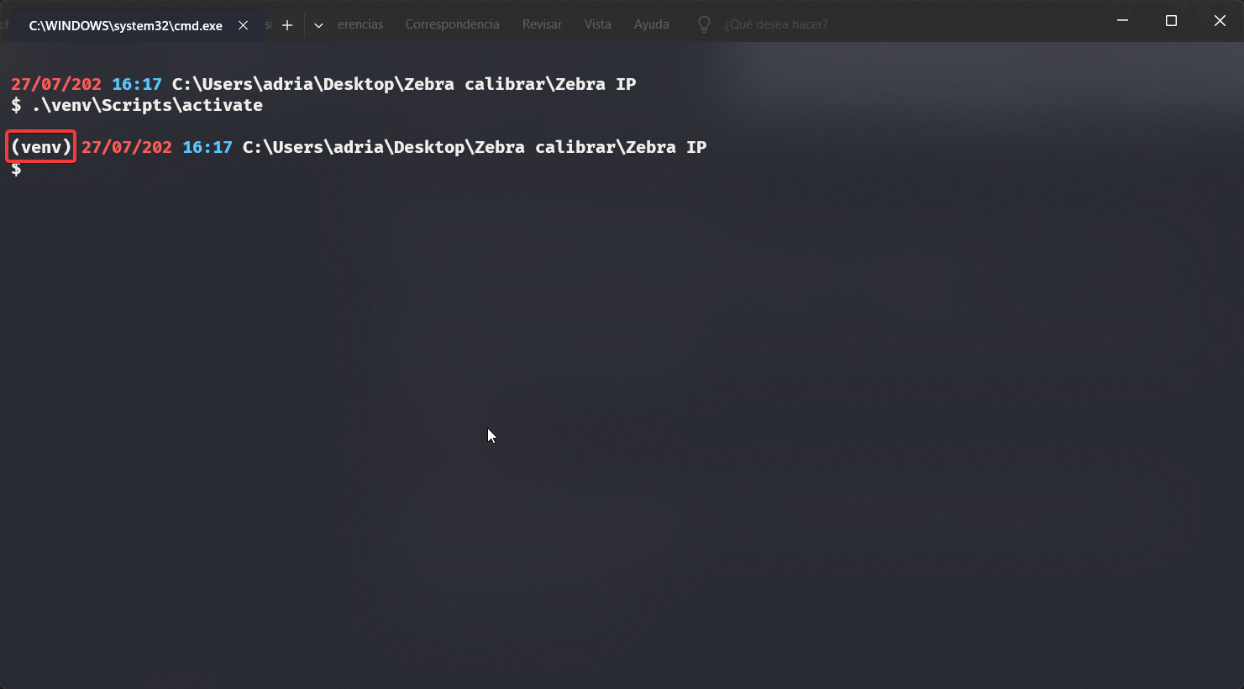
El procedimiento descrito a continuación es **OPCIONAL**,pero te asegura que no haya errores de cara a la ejecución del programa. En el caso que quieras saltarlo dirígete hasta el paso **Instalación de las librerías.**

* 1. **Generaremos un entorno virtual, hay varias formas. Prueba los siguientes comandos de CMD hasta que uno te funcione.**

py venv venv

py -m venv venv

**En el caso que ninguno te funcione (no debería pasar), como alternativa busca información sobre crear entornos virtuales con conda.**



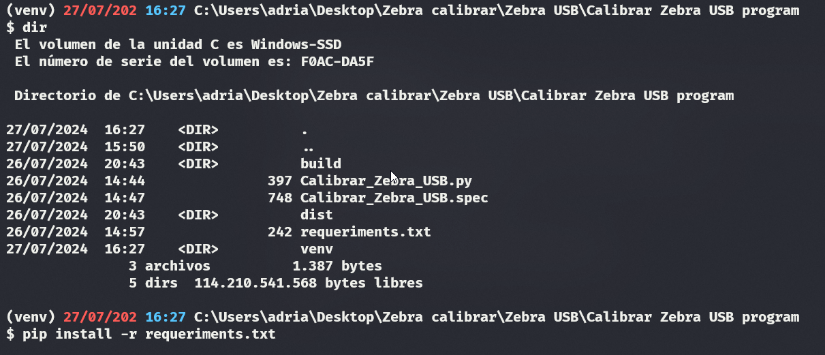
* 1. **Activa el entorno virtual con el siguiente comando.**

Si te sale la palabra **(venv)** como sale en la referencia significa que se ha activado correctamente.

* 1. Seguidamente, puedes proceder con la instalación de las librerías con PIP, explicado más adelante.

1. **Instalación de las librerías.**

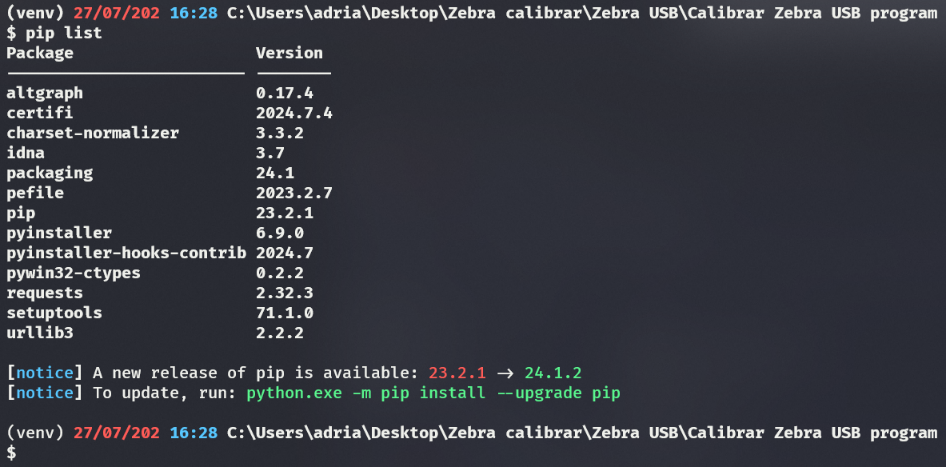
Dentro de cada carpeta podrás ver el archivo “requeriments.txt”. Este archivo te facilitara la instalación de las dependencias para cada proyecto.



Dentro de la carpeta ejecuta el siguiente comando en CMD:

**pip install -r requeriments.txt**

1. **Como verificar la instalación de las librerías:**

En CMD: **pip list**

# PASO 5 (Ejecuta el programa):

Para ejecutar el programa desde CMD ejecuta el siguiente comando:

**py [“nombre del programa”].py**

Remplaza [“nombre del programa”] por el nombre de tu archivo.

# PASO 6 (Como **generar** un exe):

En Python hay varios métodos, pero te voy a enseñar como hacerlo con la librería pyinstaller.

1. Descarga la librería con el siguiente comando en CMD:

**pip install pyinstaller**

1. Seguidamente, generaremos el exe con el siguiente comando:

**pyinstaller [“nombre del programa”].py**

Remplaza **[“nombre del programa”]** por el nombre de tu archivo.

**PARAMETROS OPCIONALES:**

* + --onefile (“Genera un solo archivo”)
  + --noconsole (“No te mostrara la consola al ejecutar el programa”)
  + --icon=[“icono”].ico (“Agrega un icono a tu exe”) Remplaza [“icono”] por el nombre de tu ico.

1. Se te generara dos carpetas y un archivo. El archivo se llama igual que tu programa pero con la extensión “.spec”. Las carpetas son “dist” y “build”. Al generar el exe tu archivo compilado se encuentra dentro de la carpeta “dist”.
2. El archivo con extensión “.spec” es el que contiene todas las configuraciones de tu exe.
3. Para generar otro exe con las configuraciones del “.spec”, pondremos el siguiente comando en cmd:

**pyinstaller [“nombre de tu archivo”].spec**

Remplaza [“nombre de tu archivo”] por tu nombre.

Esto remplazara el exe creado anteriormente con las configuraciones nuevas aplicadas.

# Conceptos:

**¿Qué es un venv?:** Un entrono virtual o virtual environment, es un entorno de programación aislado de tu máquina. Se usa generalmente cuando se quieren utilizar librerías con versiones muy concretas. De esta forma nos aseguramos de utilizar la versión deseada sin tener conflictos con las librerías ya instaladas en tu sistema.