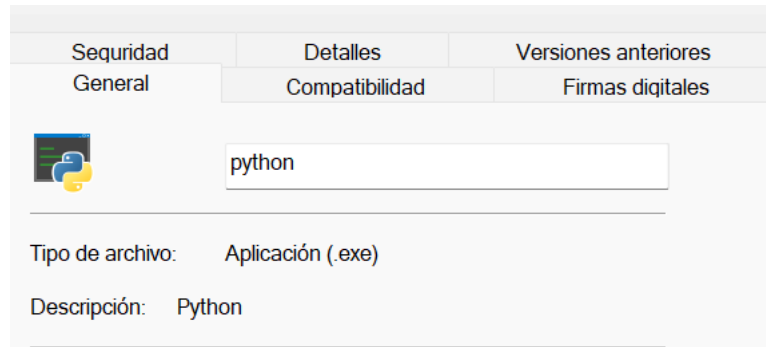


# **Informe del entorno de Python en la computadora: Explorando el intérprete, bibliotecas y módulos**

**Nombre del Archivo del Intérprete de Python:** El archivo principal del intérprete de Python en mi sistema Windows se denomina **python.exe**.



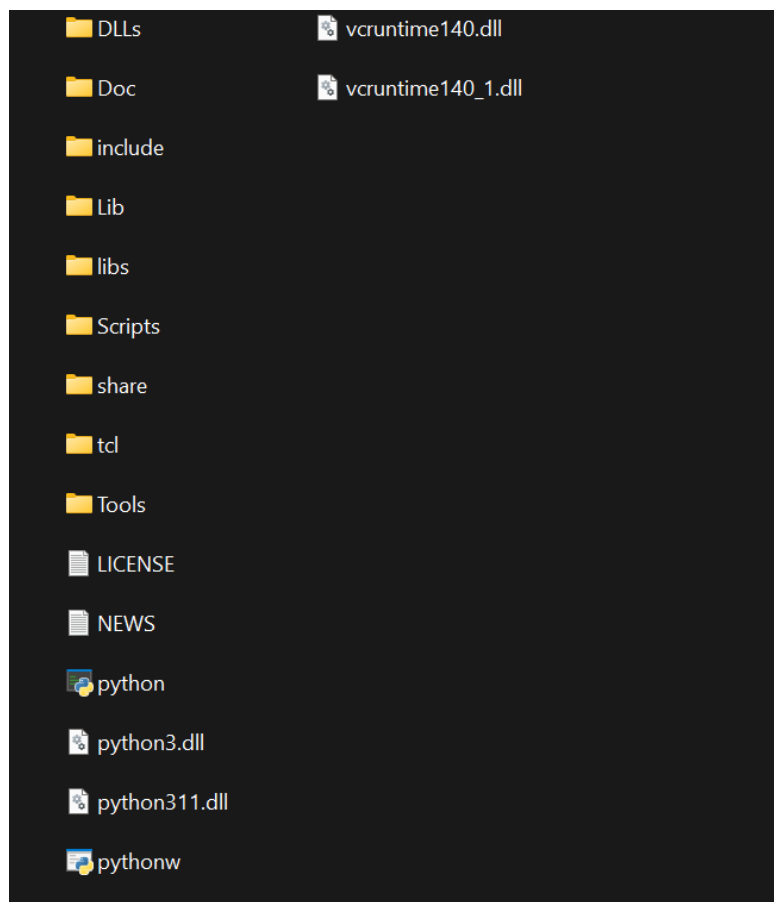
**Extensión del Archivo:** El archivo del intérprete de Python en mi sistema Windows tiene la extensión **.exe**.

**Ubicación de la Carpeta:** En mi sistema operativo Windows, el intérprete de Python se instaló en la siguiente ruta:

***C:\Users\braia\AppData\Local\Programs\Python\Python311***

**Subcarpetas en la Carpeta de Instalación de Python:** En la carpeta de instalación de Python, además de la carpeta principal que contiene el intérprete de Python, también pueden encontrarse subcarpetas como "Scripts", "include", "DLLs", "tcl", entre otras, que albergan archivos y recursos importantes para el funcionamiento de Python y sus módulos.

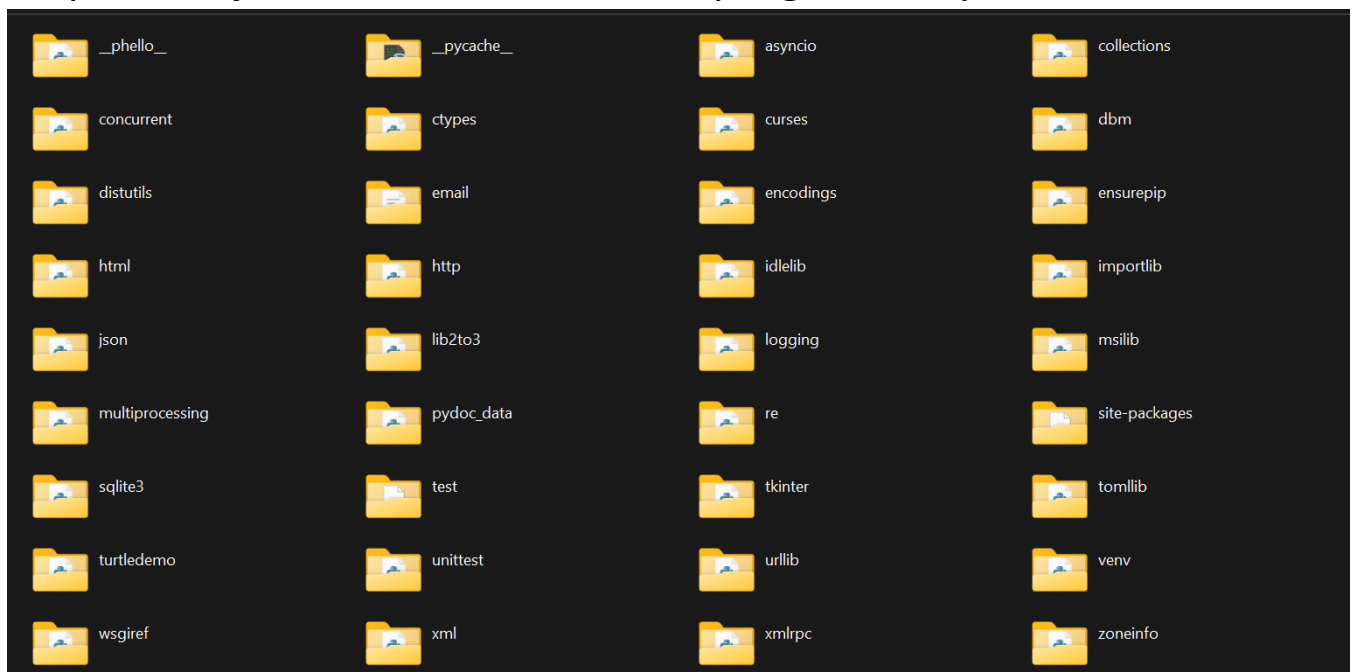
Aquí dejo una imagen para entenderlo aun mejor.



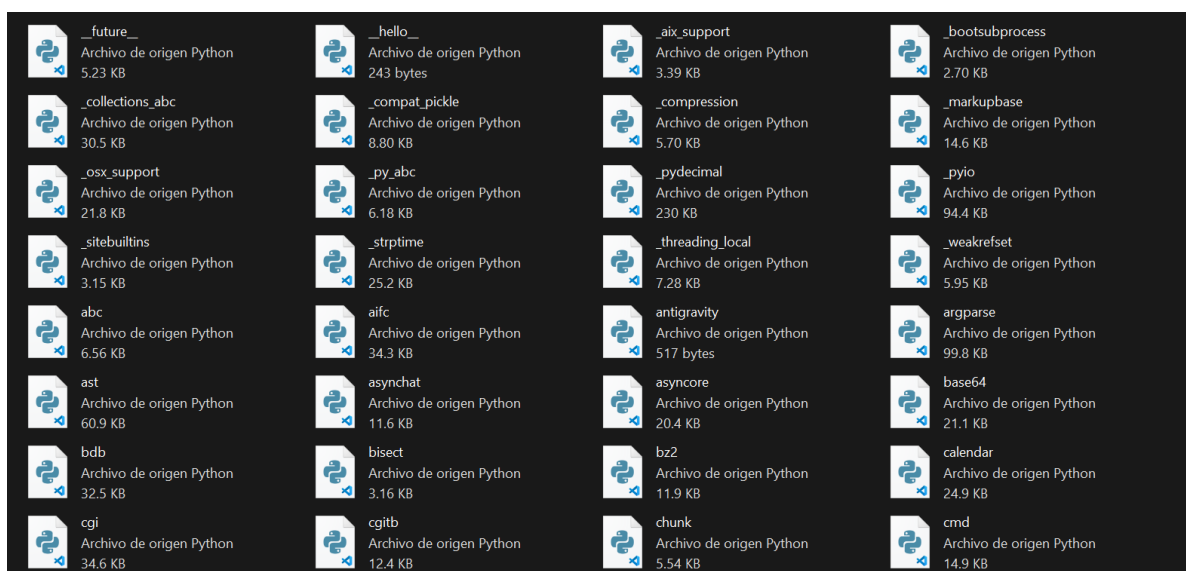
**Las subcarpetas son esenciales para el funcionamiento de Python, contienen scripts, bibliotecas y elementos fundamentales.**

## Tipos de Archivos en la Carpeta Lib de Python:
















Dentro de la carpeta "Lib" de Python, se encuentran una variedad de archivos que son vitales para el funcionamiento de Python. Estos archivos incluyen tanto los códigos fuente en formato ".py" como los archivos de código byte compilado en formato ".pyc", los cuales son fundamentales para la ejecución eficiente de los programas Python.



Carpeta con sus correspondientes archivos ".py" o ".pyc"



Archivos ".py"

|  |                 |                      |        |
|--|-----------------|----------------------|--------|
|  _future_.cpython-311       | 24/7/2023 21:30 | Compiled Python File | 5 KB   |
|  _compat_pickle.cpython-311 | 24/7/2023 21:30 | Compiled Python File | 8 KB   |
|  _compression.cpython-311   | 24/7/2023 21:30 | Compiled Python File | 8 KB   |
|  _markupbase.cpython-311    | 24/7/2023 21:30 | Compiled Python File | 14 KB  |
|  _strptime.cpython-311      | 26/7/2023 17:40 | Compiled Python File | 28 KB  |
|  _weakrefset.cpython-311    | 24/7/2023 21:30 | Compiled Python File | 13 KB  |
|  argparse.cpython-311       | 24/7/2023 21:30 | Compiled Python File | 111 KB |
|  ast.cpython-311            | 24/7/2023 21:30 | Compiled Python File | 107 KB |
|  base64.cpython-311         | 24/7/2023 21:30 | Compiled Python File | 28 KB  |
|  bdb.cpython-311            | 26/8/2023 10:45 | Compiled Python File | 38 KB  |
|  bisect.cpython-311         | 24/7/2023 21:30 | Compiled Python File | 4 KB   |
|  bz2.cpython-311            | 24/7/2023 21:30 | Compiled Python File | 16 KB  |
|  calendar.cpython-311       | 24/7/2023 21:30 | Compiled Python File | 44 KB  |
|  cmd.cpython-311            | 26/8/2023 10:45 | Compiled Python File | 21 KB  |
|  code.cpython-311           | 26/8/2023 10:45 | Compiled Python File | 14 KB  |

## ***Archivos “.pyc”***

## Un fragmento de código del modulo random.py

```
def normalvariate(self, mu=0.0, sigma=1.0):
    """Normal distribution.

    mu is the mean, and sigma is the standard deviation.

    """
    # Uses Kinderman and Monahan method. Reference: Kinderman,
    # A.J. and Monahan, J.F., "Computer generation of random
    # variables using the ratio of uniform deviates", ACM Trans
    # Math Software, 3, (1977), pp257-260.

    random = self.random
    while True:
        u1 = random()
        u2 = 1.0 - random()
        z = NV_MAGICCONST * (u1 - 0.5) / u2
        zz = z * z / 4.0
        if zz <= -_log(u2):
            break
    return mu + z * sigma
```

**Función `normalvariate()` del Módulo `random.py`:** *La función `normalvariate()` en el archivo `random.py` genera números aleatorios distribuidos. Toma dos parámetros opcionales, `mu` y `sigma`, que representan la media y la desviación estándar. La función utiliza números aleatorios distribuidos para calcular variables aleatorias normalmente distribuidas.*

**Fuente:** <https://interactivechaos.com/es/python/function/randomnormalvariate>

## Resumen:

En este informe se examinó el entorno de Python en la computadora, desde la ubicación del intérprete hasta la estructura de la carpeta de instalación y los archivos en "Lib". Se analizó la función **normalvariate()** del módulo **random.py**, vital para generar números aleatorios distribuidos normalmente.