

A graphic consisting of four horizontal bars of increasing height, colored purple, orange, yellow, and blue from top to bottom. The text is overlaid on the bars.

Agencia de  
Aprendizaje  
a lo largo  
de la vida

# **CODO A CODO INICIAL**

## **Clase 13**

Python 1

# Introducción al Lenguaje Python



# Les damos la bienvenida

Vamos a comenzar a grabar la clase



## Clases 11/12

### Algoritmia 11/12 - Síntesis y repaso

- Integración de temas.
- Repaso de conceptos básicos.
- Entrada y salida de datos.
- Operadores aritméticos, lógicos y relacionales.
- Estructuras de control.

## Clase 13

### Python 1 - Introducción al Lenguaje Python

- Historia y contexto
- Descarga e instalación del entorno de desarrollo
- "Hola, Mundo"

## Clase 14

### Python 2 - Tipos de Datos, Entrada y Salida de Información

- Exploración de tipos de datos
- Cadena como un "contenedor" de caracteres
- Entrada y salida de información por consola
- Ejemplos prácticos de entrada y salida

# Python

Python es un lenguaje de programación de alto nivel. Sus principales características son:

- **Multiparadigma:** Soporta programación ilimperativa, orientada a objetos y funcional.
- **Multiplataforma:** El código que escribimos funciona en Windows, Linux, Mac OS, Android, entre otros
- **Dinámica y fuertemente tipado:** El tipo de las variables se decide en tiempo de ejecución.
- **Interpretado:** El código se ejecuta a medida que se lee.

# Sobre Python

- Fácil de aprender.
- Es un lenguaje de alto nivel (sintaxis sencilla, claro, muy legible).
- Tipado dinámico y fuertemente tipado.
- Lenguaje interpretado.
- Orientado a objetos.
- Creado a comienzos de los '90 por Guido Van Rossum, holandés que trabajaba en el Centro para las Matemáticas y la Informática de los Países Bajos.

# Sobre Python

- Es multiplataforma, es decir, el código Python se puede correr en diferentes sistemas operativos.
- Es Open source, es completamente libre para ser utilizado y redistribuido.
- Librería estándar muy amplia.
- Versátil: permite desarrollar aplicaciones de escritorio, aplicaciones de servidor y aplicaciones web.
- Página web oficial: [www.python.org](http://www.python.org)

# Lenguajes interpretados y lenguajes compilados

- Los lenguajes **compilados** utilizan un compilador que procesa el programa completo en una sola vez, generando un código intermedio o *código máquina*. Estos lenguajes, como C, C++, y C#, ofrecen una eficiencia notable, ya que los errores se identifican después de la verificación completa del programa, asegurando una mayor integridad del código.
- Los lenguajes **interpretados** adoptan un enfoque más lineal mediante el uso de un intérprete que ejecuta el programa línea por línea. Lenguajes como JavaScript y **Python** pertenecen a este grupo. Aunque son menos eficientes que los lenguajes compilados, la facilidad de depuración y la interpretación línea por línea hacen que sean una opción atractiva.



# Historia de Python

## 1. Orígenes (1989-1991):

- El desarrollo comenzó en diciembre de 1989, por Guido van Rossum.
- La primera versión, Python 0.9.0, se lanzó en febrero de 1991.

## 2. Python 2.x (2000-2008):

- Python 2.0 lanzado en 2000, introdujo características como recolección de basura.
- Durante esta época, Python ganó popularidad en la comunidad de desarrollo.

## 3. Python 3.x (2008 en adelante):

- Python 3.0 (también conocido como "Python 3000" y "Py3k") fue lanzado en 2008.
- Incluyó cambios significativos para mejorar la consistencia y claridad del lenguaje.
- La transición de Python 2 a Python 3 llevó tiempo debido a incompatibilidades.

# Historia de Python

## 4. Versiones Recientes (2020-2024):

- Desarrollo continuo con lanzamientos regulares.
- Cada versión agrega nuevas características y mejoras de rendimiento.
- La versión actual es la 3.12.

## 5. Evolución Futura:

- La comunidad y los desarrolladores continúan trabajando día a día para mejorar Python y adaptarlo a las necesidades cambiantes de la tecnología.

# Características de Python

- Comunidad y Ecosistema:
  - Python ha ganado una gran comunidad de desarrolladores y una rica colección de bibliotecas.
  - La Python Software Foundation (PSF) fue creada para promover, proteger y avanzar en Python.
- Uso en Diversos Campos:
  - Ampliamente utilizado en desarrollo web (Django, Flask), ciencia de datos (NumPy, Pandas), inteligencia artificial (TensorFlow, PyTorch), automatización, scripting, etc.
- Impacto en la Educación:
  - Python es frecuentemente elegido como un lenguaje de programación introductorio debido a su sintaxis clara y legible.

# ¿Por qué estudiamos Python?

- Sintaxis Clara y Legible
  - La sintaxis de Python es simple y fácil de leer.
  - La indentación obligatoria fomentan la escritura de código estructurado y legible.
- Facilidad de Aprendizaje
  - Python está diseñado para ser fácil de aprender y usar, lo que reduce la barrera de entrada para los nuevos programadores.
- Amplia Comunidad y Recursos
  - Cuenta con una gran comunidad de desarrolladores y abundancia de recursos educativos, tutoriales y documentación.
  - Los estudiantes pueden encontrar fácilmente apoyo y soluciones a sus problemas a través de foros, comunidades en línea y recursos educativos.

# ¿Por qué estudiamos Python?

- Rápido Prototipado y Desarrollo
  - Python facilita el desarrollo rápido de prototipos, permitiendo a los estudiantes experimentar y visualizar resultados rápidamente.
  - Esto fomenta la creatividad y la iteración en el proceso de resolución de problemas.
- Gran Soporte para Estructuras de Datos y Algoritmos
  - Python ofrece una amplia gama de estructuras de datos y bibliotecas que facilitan la implementación de algoritmos.
  - La sintaxis clara permite a los estudiantes concentrarse en la lógica del algoritmo en lugar de en detalles de bajo nivel.

# ¿Por qué estudiamos Python?

- Interactividad y Consola Interactiva
  - La consola interactiva de Python (REPL) permite a los estudiantes probar y experimentar con código de manera interactiva, facilitando el aprendizaje y la depuración.
- Creciente Demanda Laboral
  - Python es uno de los lenguajes de programación más populares y tiene una creciente demanda en la industria.
  - Enseñar Python proporciona a los estudiantes habilidades relevantes para el mercado laboral actual.

# Sintaxis en Python

La **sintaxis** en un lenguaje de programación se refiere a las reglas que definen la estructura correcta de las secuencias de símbolos y palabras clave que son legalmente correctas en ese lenguaje. Es análoga a la gramática en los idiomas humanos.

Por ejemplo, la correcta indentación, el uso de dos puntos (:) para iniciar bloques de código y la sintaxis específica para definir funciones.

Si un programador no sigue estas reglas sintácticas, el código no se ejecutará correctamente. En el caso de Python, los errores de sintaxis se pueden detectar durante la escritura del código gracias a las ayudas del editor.

# Semántica en Python

La **semántica** se refiere al significado de las construcciones sintácticas. No es suficiente que un programa esté sintácticamente correcto; las instrucciones también deben tener sentido lógico para realizar la tarea deseada. La semántica define cómo las estructuras sintácticas se traducen en operaciones que se van a ejecutar en la máquina.

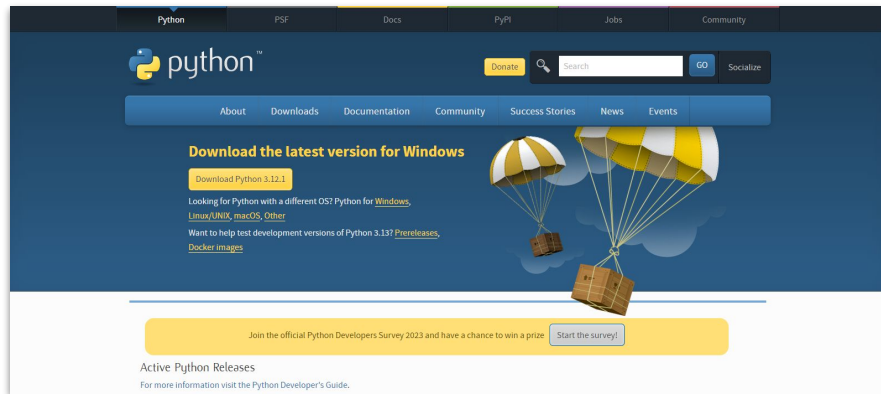
La semántica de Python describe cómo la ejecución del código resulta en un comportamiento específico.

Un programa bien escrito debe ser tanto sintáctica como semánticamente correcto para funcionar como se espera.



# Instalación de Python

Necesitamos instalar el **intérprete Python** desde su [página oficial](#) \*.  
Ingresamos a **Downloads** y haremos clic en el botón **Download Python**.  
Asegúrate de instalar la última versión (o como mínimo la versión 3.8.x).



\* al momento de la preparación de este material la última versión es la 3.12.1

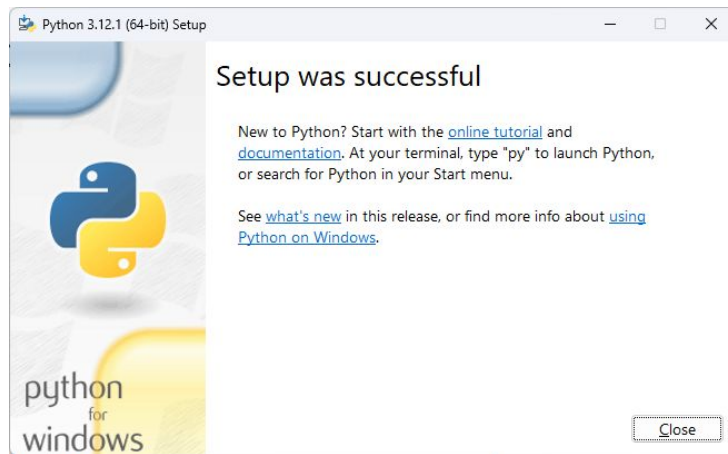
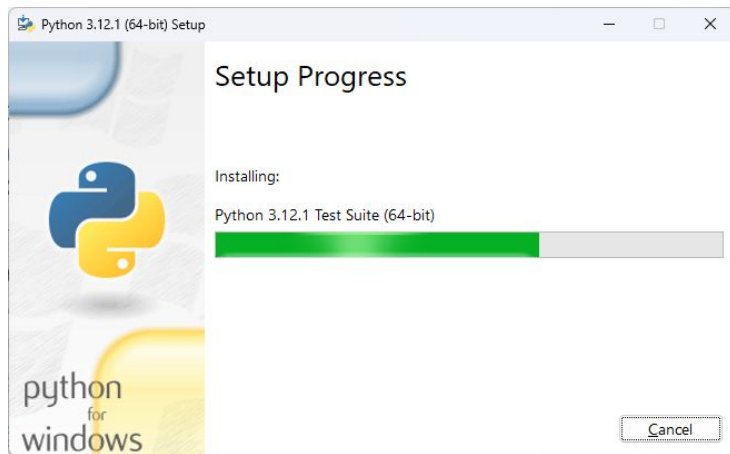
# Instalación de Python

1. Ejecutamos el archivo descargado haciendo doble clic en él y seguimos los pasos de la instalación. Recuerda instalar seleccionando la opción “Add python.exe to PATH”:



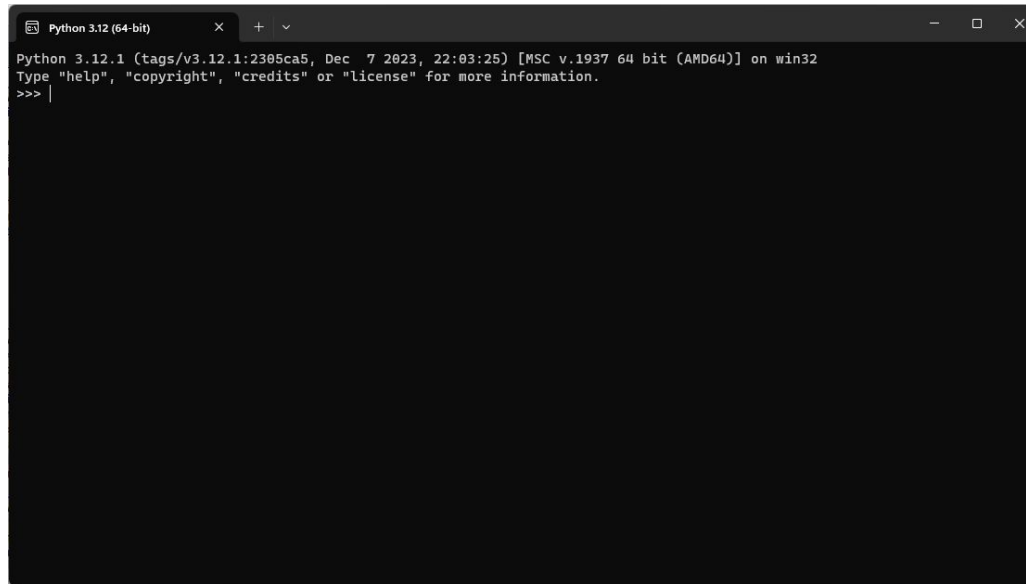
# Instalación de Python

2. Esperamos a que finalice la instalación y confirmamos la instalación:



# Instalación de Python

Al ejecutar el programa aparece una pantalla similar a la siguiente:

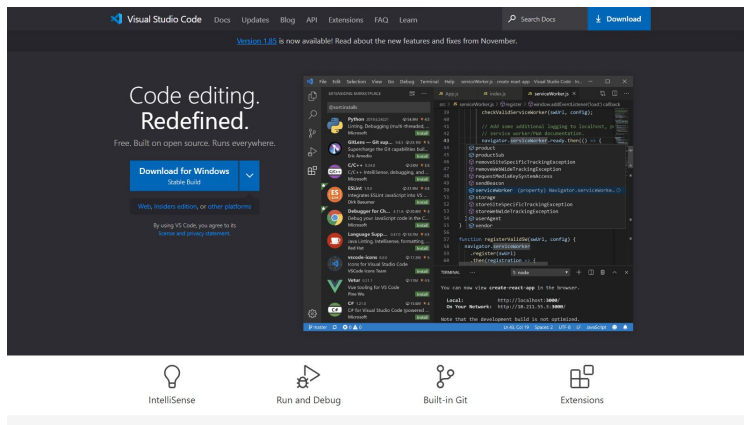
A screenshot of a Windows command prompt window titled "Python 3.12 (64-bit)". The window has a dark background and a light-colored text area. The text inside the window reads: "Python 3.12.1 (tags/v3.12.1:2305ca5, Dec 7 2023, 22:03:25) [MSC v.1937 64 bit (AMD64)] on win32", followed by "Type 'help', 'copyright', 'credits' or 'license' for more information.", and then a prompt ">>> |" with a cursor. The window has standard Windows window controls (minimize, maximize, close) in the top right corner.

```
Python 3.12 (64-bit)
Python 3.12.1 (tags/v3.12.1:2305ca5, Dec 7 2023, 22:03:25) [MSC v.1937 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license" for more information.
>>> |
```

# Instalación de VSCode

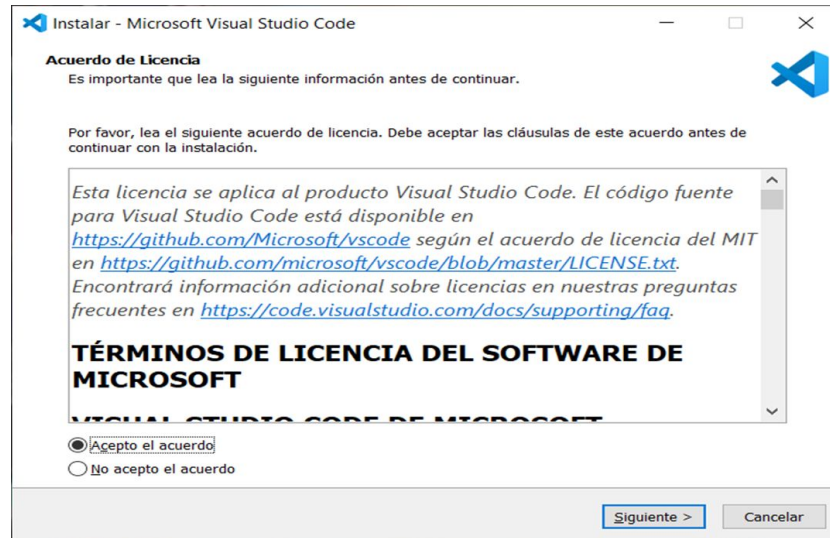
Además utilizaremos un editor de texto llamado **Visual Studio Code**.

Lo puedes descargar desde su [página oficial](#) y seguir los pasos de instalación, similares a los realizados al instalar el intérprete de Python.



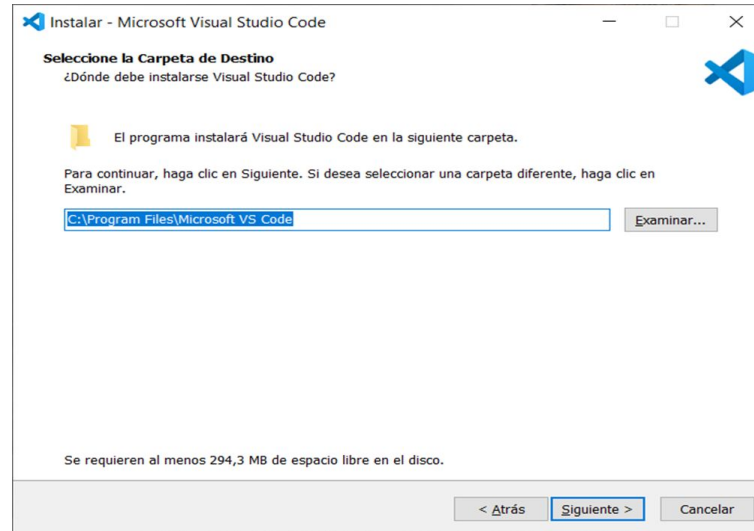
# Instalación de VSCode

1. Una vez descargado el instalador, ejecutarlo haciendo doble clic en el archivo y aceptar el Acuerdo de licencia:



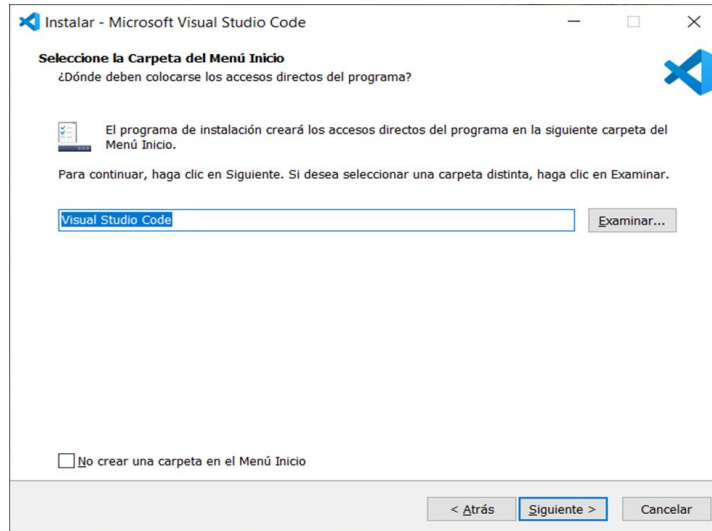
# Instalación de VSCode

2. Es posible mantener la ruta de instalación o cambiarla. De esta manera especificaremos en qué carpeta instalaremos el programa:



# Instalación de VSCode

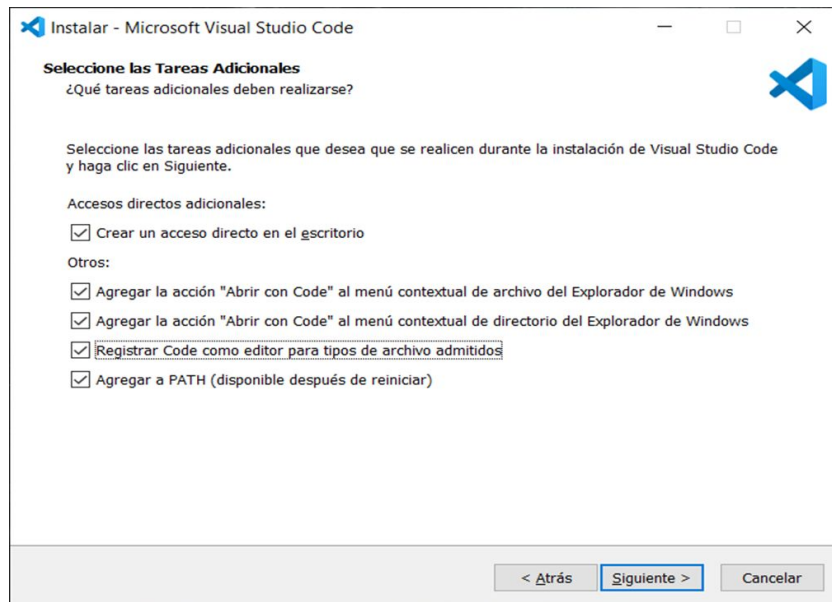
3. Además es posible mantener la carpeta del Menú Inicio donde se creará el acceso directo al programa:





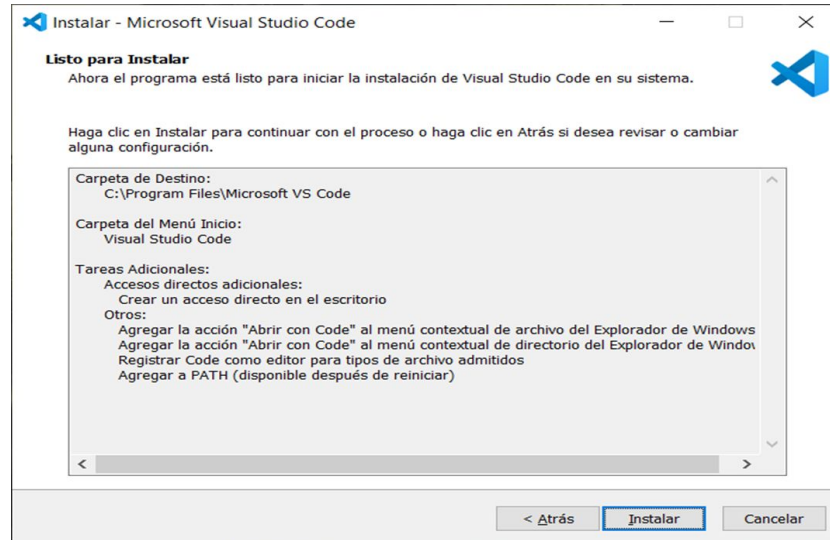
# Instalación de VSCode

4. Recomendamos dejar seleccionadas todas las “*Tareas Adicionales*”:



# Instalación de VSCode

5. Finalmente, presionar “*Instalar*”. El asistente para la instalación finalizará en algunos segundos:



# Extensión Python para VS Code

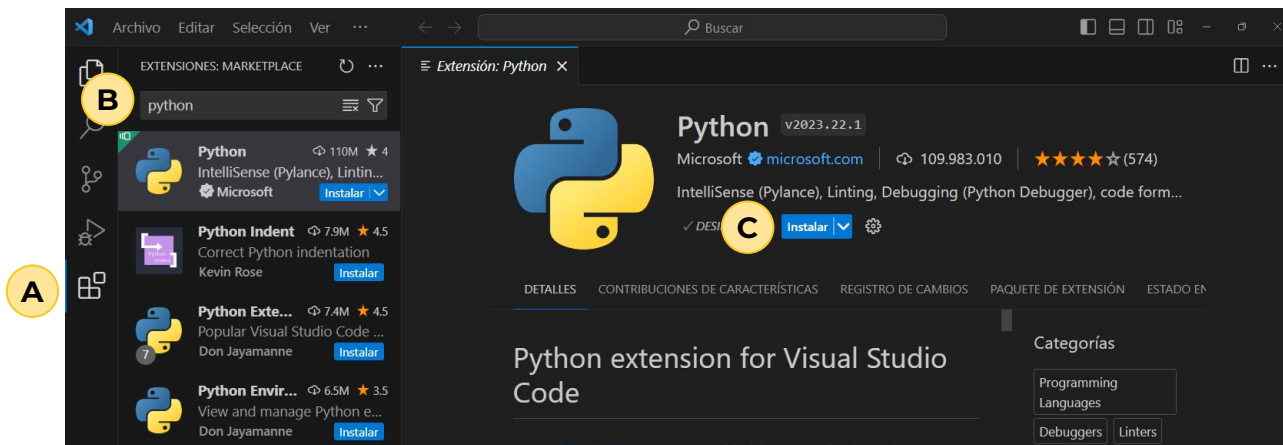
En Visual Studio Code, una **extensión** es un módulo de software que agrega funcionalidades adicionales al entorno de desarrollo. Las extensiones pueden personalizar y mejorar la experiencia del usuario, proporcionando características específicas para diferentes lenguajes de programación, frameworks, servicios y herramientas.

La extensión de **Python** para Visual Studio Code mejora la experiencia de desarrollo al proporcionar herramientas específicas para el lenguaje y facilita la adopción de buenas prácticas de programación.

Es posible instalarla directamente desde el administrador de extensiones de VS Code.

# Extensión Python para VS Code

Es posible descargarla desde el administrador de extensiones de Visual Studio Code **(A)**, buscando la extensión Python de Microsoft **(B)** y haciendo clic en Instalar **(C)**.

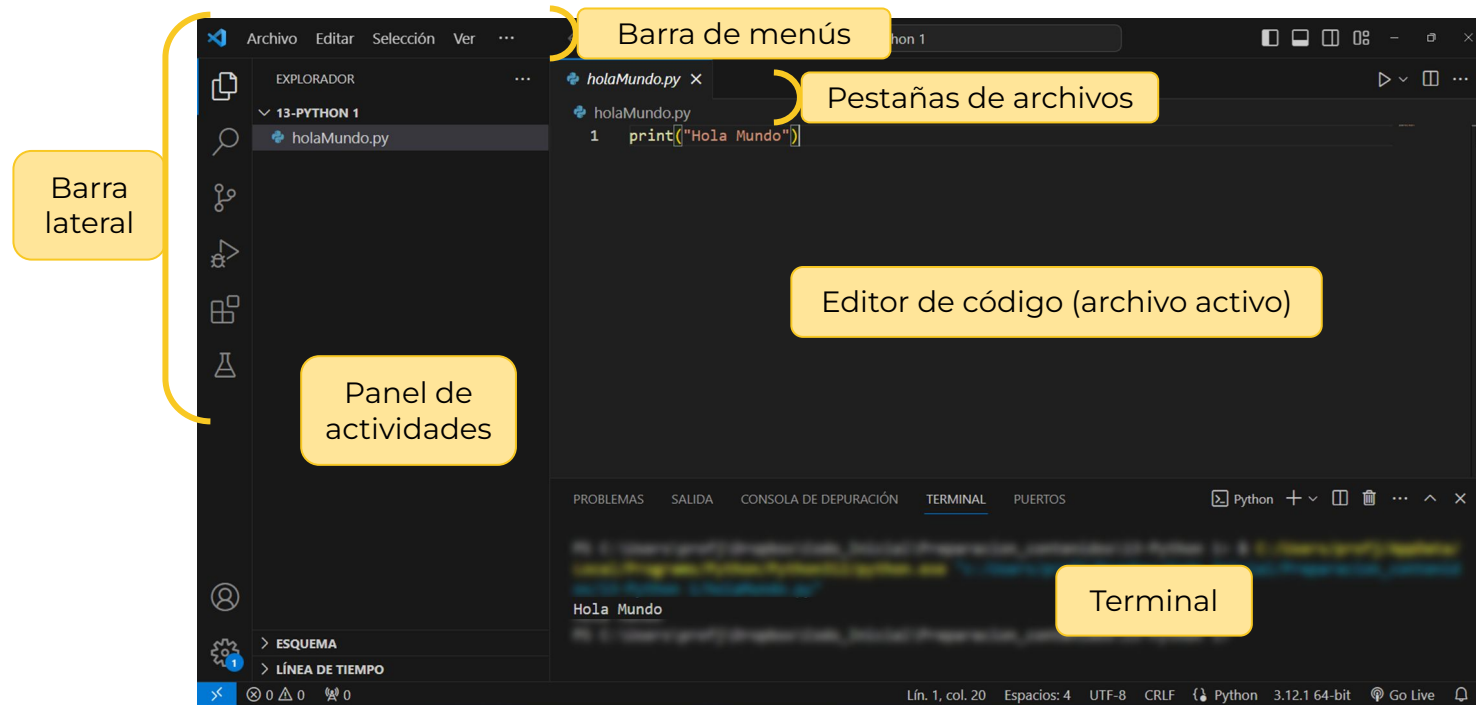


# Extensión Spanish Language Pack

Por defecto, Visual Studio Code está en inglés. Para cambiar el idioma a español podemos descargar la extensión “*Spanish Language Pack for Visual Studio Code*” de Microsoft de la misma manera que hicimos con la extensión anterior:



# El entorno de desarrollo de VS Code



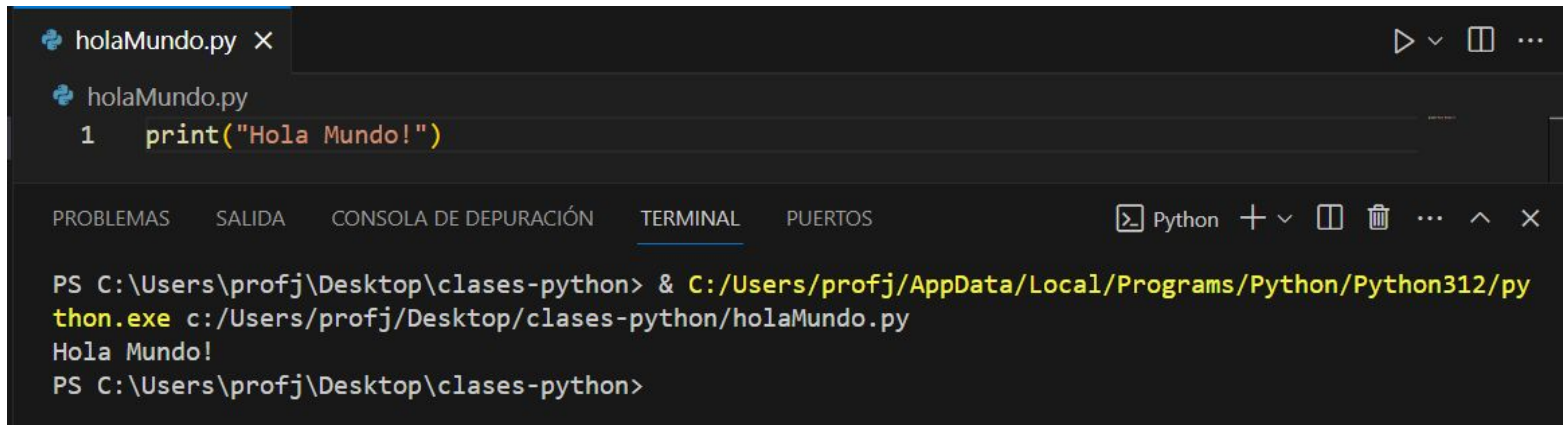
# La consola de Visual Studio Code

En Visual Studio Code, la sección donde se muestra el resultado de la ejecución de un programa se llama **"Terminal"** o, coloquialmente **"la consola"** debido a su función similar a una consola de comandos o terminal. En nuestro curso utilizaremos ambos términos como sinónimos.

La "Terminal" en Visual Studio Code puede ser utilizada como una interfaz de línea de comandos integrada que permite ejecutar comandos directamente desde el editor o para mostrar la salida del programa, como cualquier mensaje de impresión (por ejemplo, utilizando print en Python) o cualquier error que pueda ocurrir durante la ejecución.

# La consola de Visual Studio Code

Como dijimos, la Terminal o Consola de Visual Studio Code mostrará la salida de tus programas Python, así como cualquier mensaje de error o información relevante durante la ejecución:



The screenshot shows the Visual Studio Code interface. At the top, a file named 'holaMundo.py' is open, containing a single line of Python code: `print("Hola Mundo!")`. Below the editor, the 'TERMINAL' tab is active, displaying the command prompt output. The command executed is `& C:/Users/profj/AppData/Local/Programs/Python/Python312/python.exe c:/Users/profj/Desktop/clases-python/holaMundo.py`, which results in the output `Hola Mundo!`. The terminal prompt is `PS C:\Users\profj\Desktop\clases-python>`.

```
holaMundo.py x
holaMundo.py
1 print("Hola Mundo!")

PROBLEMAS  SALIDA  CONSOLA DE DEPURACIÓN  TERMINAL  PUERTOS
Python + v [icon] [icon] ... ^ x

PS C:\Users\profj\Desktop\clases-python> & C:/Users/profj/AppData/Local/Programs/Python/Python312/py
thon.exe c:/Users/profj/Desktop/clases-python/holaMundo.py
Hola Mundo!
PS C:\Users\profj\Desktop\clases-python>
```



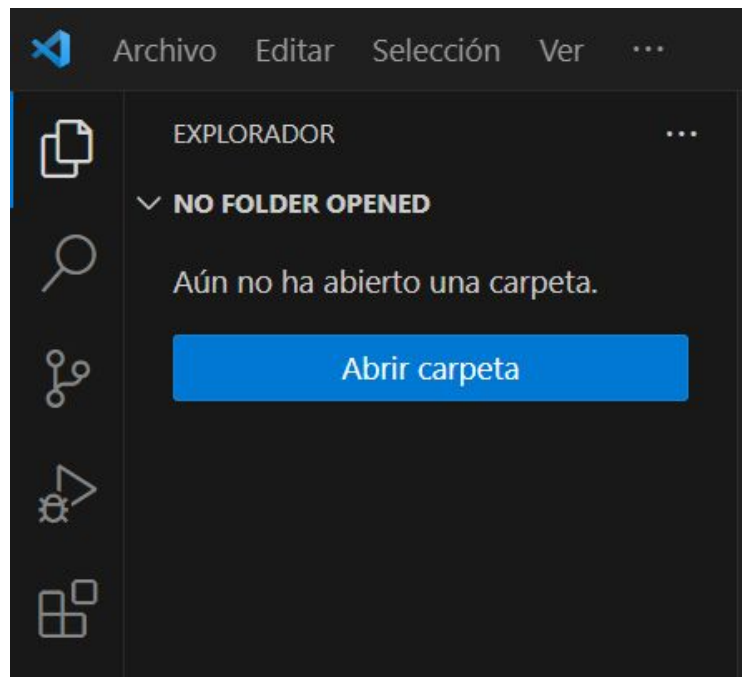
# Hola Mundo

"**Hola Mundo**" es una tradición en la programación y se refiere al programa más simple que puede escribirse en un lenguaje de programación para demostrar su funcionalidad básica. El objetivo es imprimir o mostrar en la pantalla el mensaje "Hola Mundo".

Para escribir y mostrar por pantalla un mensaje en Python utilizamos la instrucción **print()**, similar al Escribir de PSeint.

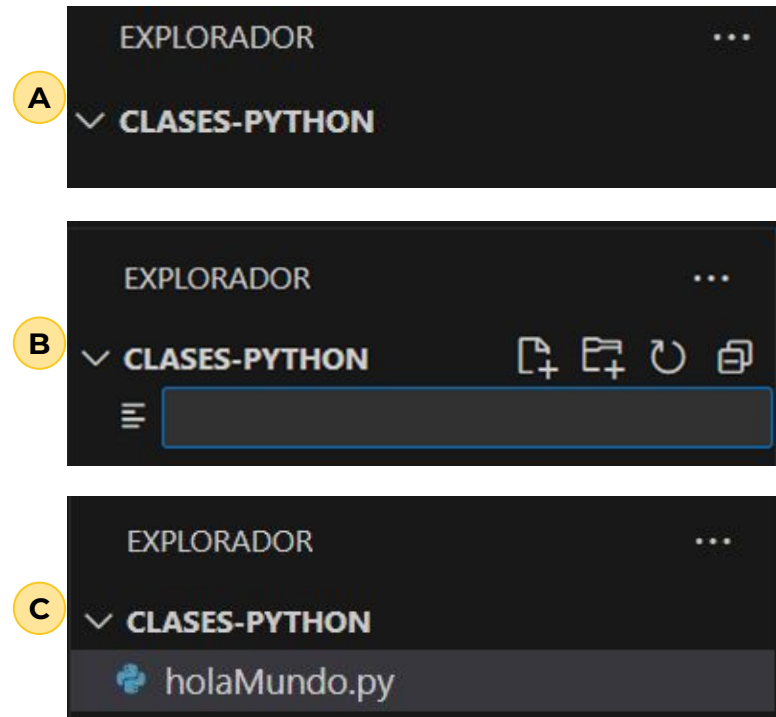
# Creando nuestro primer “Hola Mundo” en Python

1. Comenzamos creando una **carpeta de trabajo** para guardar nuestros archivos. Para ello abrimos Visual Studio Code y seleccionamos en el **panel lateral** el botón *Explorador* (con la forma de dos hojas superpuestas) y allí hacemos clic en el botón **Abrir carpeta**. Seleccionamos nuestra carpeta de trabajo (si existe, sino la creamos) y hacemos clic en **Agregar**.



# Creando nuestro primer “Hola Mundo” en Python

2. Aparece en el **panel de actividades** la carpeta *clases-python* que abrimos para el ejemplo **(A)**. En cualquier parte vacía hacemos clic con el botón derecho y seleccionamos la opción **Nuevo archivo...**
3. Aparece un recuadro **(B)** donde debemos colocar el nombre del archivo y su extensión, en este caso holaMundo.py. El archivo será creado **(C)** y listo para ser utilizado.



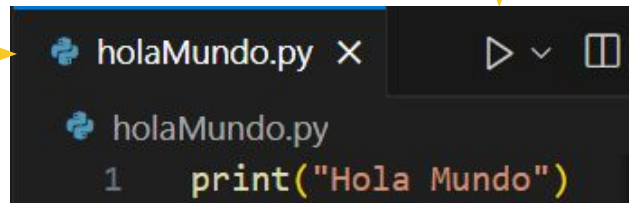
# Creando nuestro primer “Hola Mundo” en Python

Los archivos de Python tienen una extensión **.py**.

Nombre del archivo

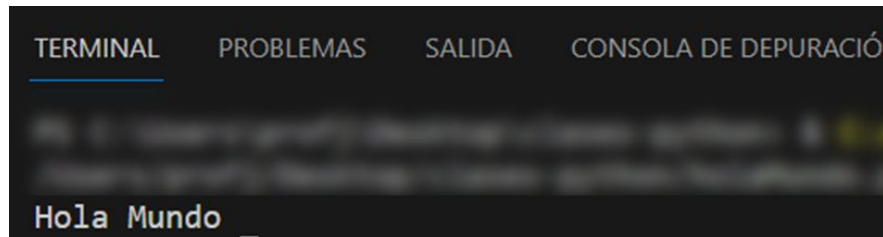
Este ícono **ejecuta** el programa

La terminal muestra la **salida** del programa. Se puede limpiar escribiendo el comando **clear**.



```
holaMundo.py x
holaMundo.py
1 print("Hola Mundo")
```

Código del programa



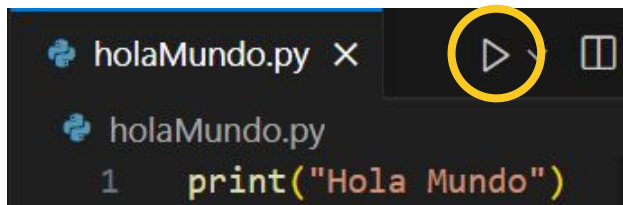
```
TERMINAL  PROBLEMAS  SALIDA  CONSOLA DE DEPURACIÓN
Hola Mundo
```

Salida por pantalla

# Creando nuestro primer “Hola Mundo” en Python

Para **ejecutar nuestro código** tenemos dos opciones:

- Hacer clic en el botón con forma de Play (es la forma más sencilla):



- Desde la terminal (Ver > Terminal) escribir:

```
python holaMundo.py
```

# Configuraciones y atajos de teclado

**Autoguardado:** Permite guardar automáticamente un archivo. *Archivo > Autoguardado.*

**Tema de color:** Sirve para cambiar el tema de color del editor. *Archivo > Preferencias > Tema > Tema de color.*

**Tamaño de fuente:** Permite cambiar el tamaño de la fuente del editor de código. *Archivo > Preferencias > Configuración > Editor de texto > Fuente y editar Font Size*

**ALT + Z:** Ajuste de palabra. Permite ajustar el texto horizontalmente y no usar el scroll.

**CTRL + “+”** o **CTRL + “-”:** Zoom. Acerca o aleja la ventana de VSC.

# Material extra

# Artículos de interés

## Material extra:

- [Visual Studio Code Online](#)
- [Python Online Compiler](#) | Programiz.com
- [¿Qué es Visual Studio Code y cuáles son sus ventajas?](#) | Fernán García de Zúñiga en arsys.es
- [Atajos de teclado en Visual Studio Code](#) | Ma del Carmen Valera en El Blog de Aitana
- [Documentación de Visual Studio Code](#)

## Videos:

- [Cómo instalar Python](#) e [Instalacion de Visual Studio Code](#) | Tutoriales sobre ciencia y tecnología
- [Mi primer programa en Python](#) | Tutoriales sobre ciencia y tecnología
- [Cómo configurar Visual Studio Code para Python](#) | divcode



# No te olvides de dar el presente

# Recordá:

- Revisar la Cartelera de Novedades.
- Hacer tus consultas en el Foro.
- Realizar el Ejercicio de Repaso.

**Todo en el Aula Virtual.**

**Muchas gracias por tu atención.**

**Nos vemos pronto**