



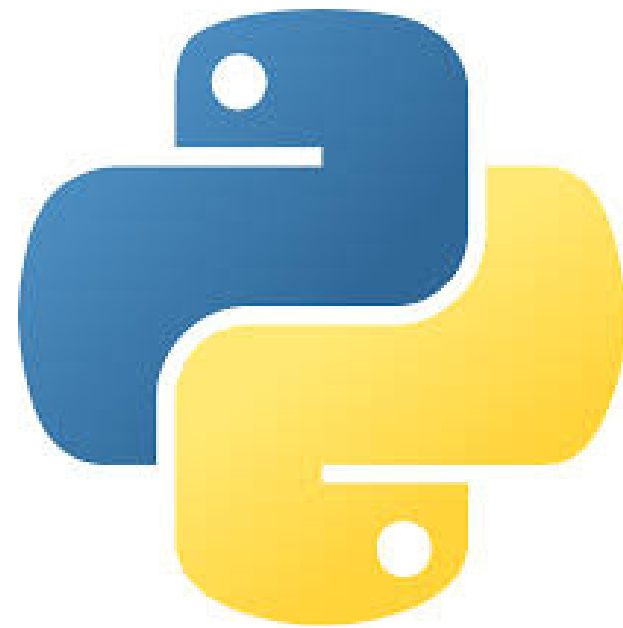
# PROGRAMACIÓN 2

Guía 2

**DAVID VALENCIA SANDOVAL**  
INGENIERO EN SISTEMAS  
[david.valencia@ciaf.edu.co](mailto:david.valencia@ciaf.edu.co)

# ¿Qué es PYTHON?

Python es un lenguaje de programación de alto nivel, interpretado y de propósito general, conocido por su simplicidad y legibilidad. Fue creado por Guido van Rossum y lanzado por primera vez en 1991



# Características

**Sintaxis Sencilla y Legible:** Python tiene una sintaxis clara y sencilla que hace que el código sea fácil de leer y escribir. Esto lo convierte en una excelente opción tanto para principiantes como para desarrolladores experimentados.

**Interpretado:** Python es un lenguaje interpretado, lo que significa que el código se ejecuta línea por línea. Esto facilita la depuración y prueba de código.

**Multiparadigma:** Python admite múltiples paradigmas de programación, incluyendo la programación orientada a objetos, la programación imperativa, la programación funcional y la programación procedimental.

# Características

**Gran Biblioteca Estándar:** Python viene con una amplia biblioteca estándar que proporciona módulos y paquetes para muchas tareas comunes, como trabajar con archivos, realizar operaciones matemáticas, interactuar con servicios web y mucho más.

**Portabilidad:** Python es multiplataforma, lo que significa que el mismo código puede ejecutarse en diferentes sistemas operativos sin modificación.

**Amplia Comunidad y Recursos:** Python tiene una gran comunidad de usuarios y desarrolladores, lo que se traduce en una abundancia de recursos, documentación, tutoriales y soporte en línea.

# Características

**Uso Extendido:** Python se utiliza en una variedad de campos, incluyendo desarrollo web, ciencia de datos, aprendizaje automático, automatización, análisis de datos, desarrollo de software, y más.

# Importancia de Aprender PYTHON

**Versatilidad**

**Demanda en  
el Mercado  
Laboral**

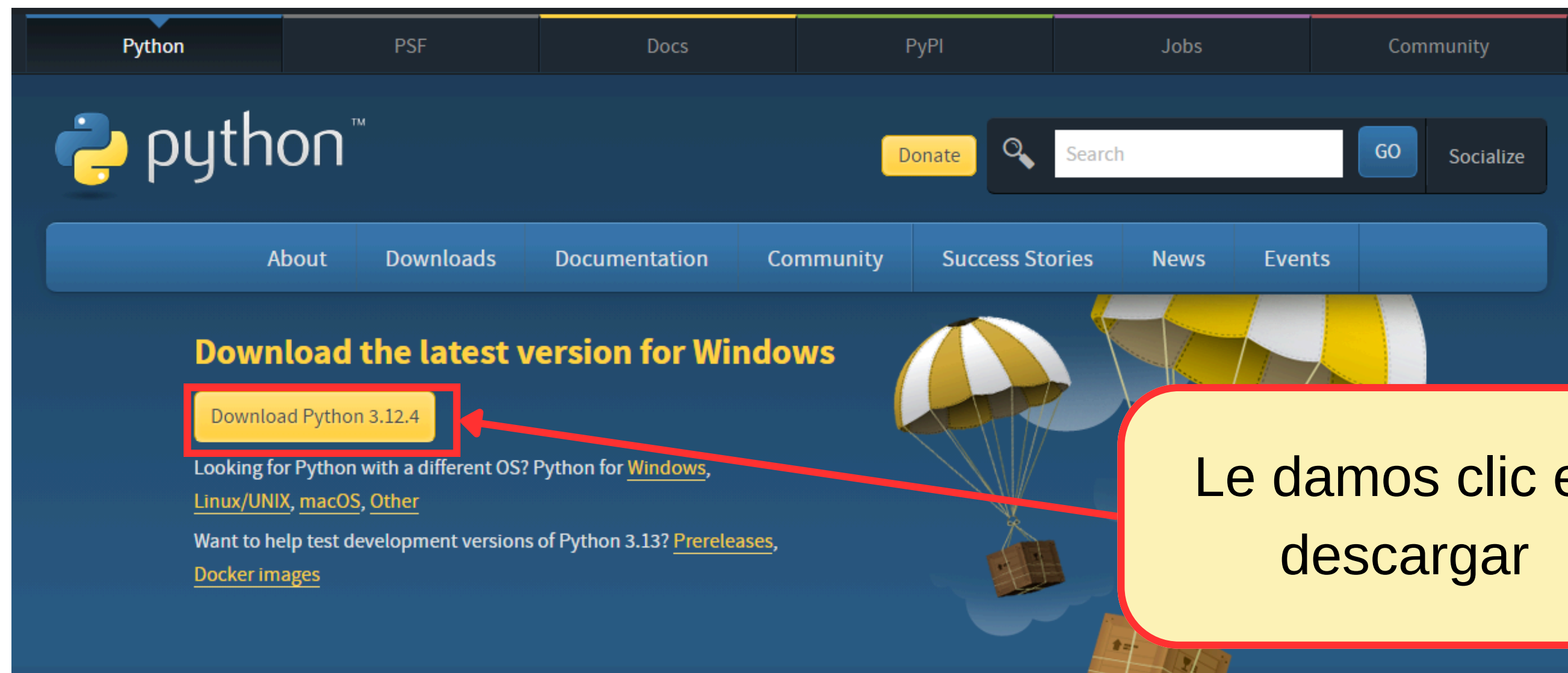
**Gran  
Comunidad y  
Soporte**

**Facilidad de  
Aprendizaje**

# Preparar Nuestro Entorno de Desarrollo

## 1. Descargar Python:

- Ingresamos a la pagina <https://www.python.org/downloads/>

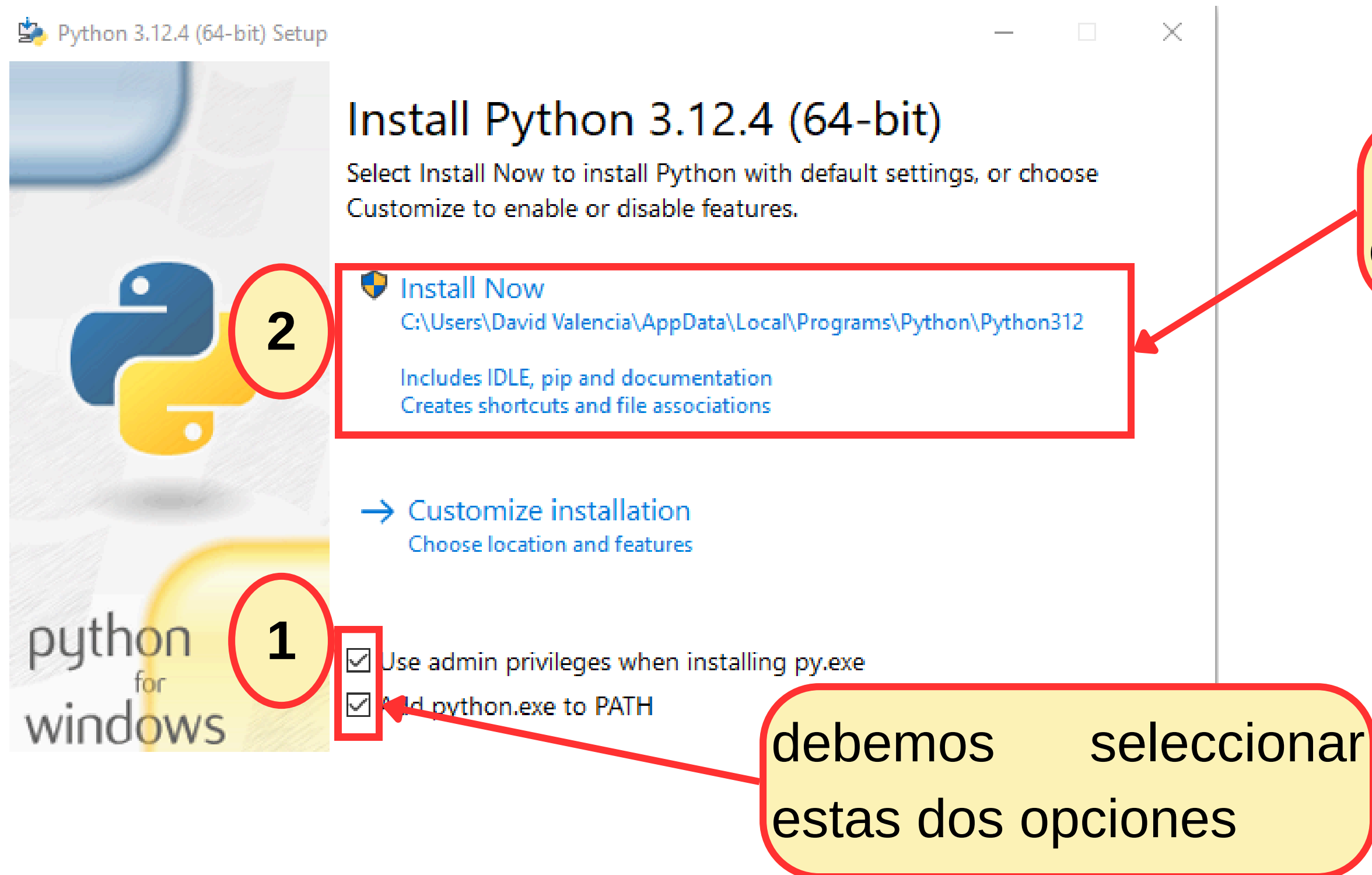


Le damos clic en  
descargar

Una vez descargado ejecutamos el instalador

# Preparar Nuestro Entorno de Desarrollo

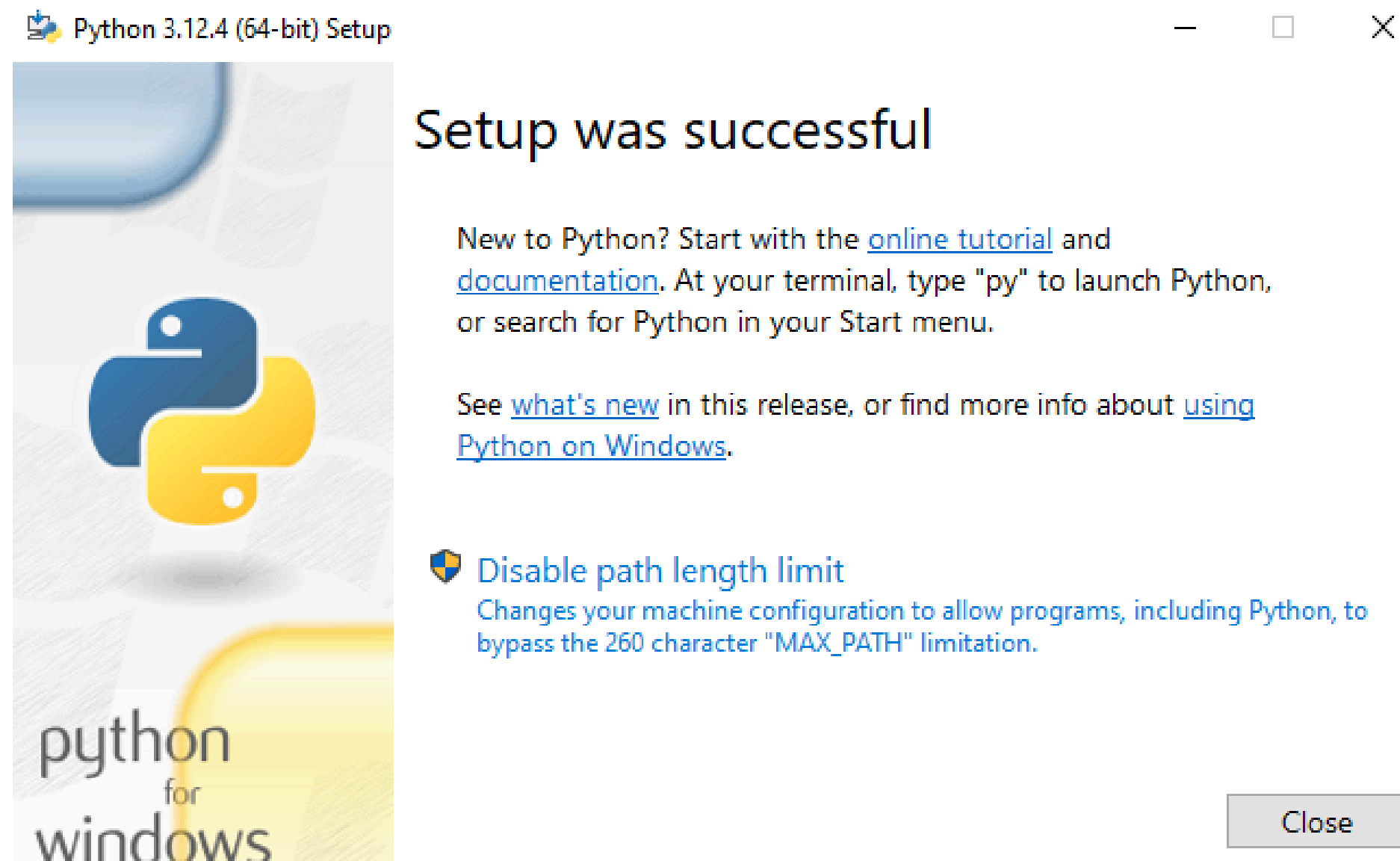
1.1 Luego de ejecutar la instalador nos saldrá la siguiente pantalla:





# Preparar Nuestro Entorno de Desarrollo

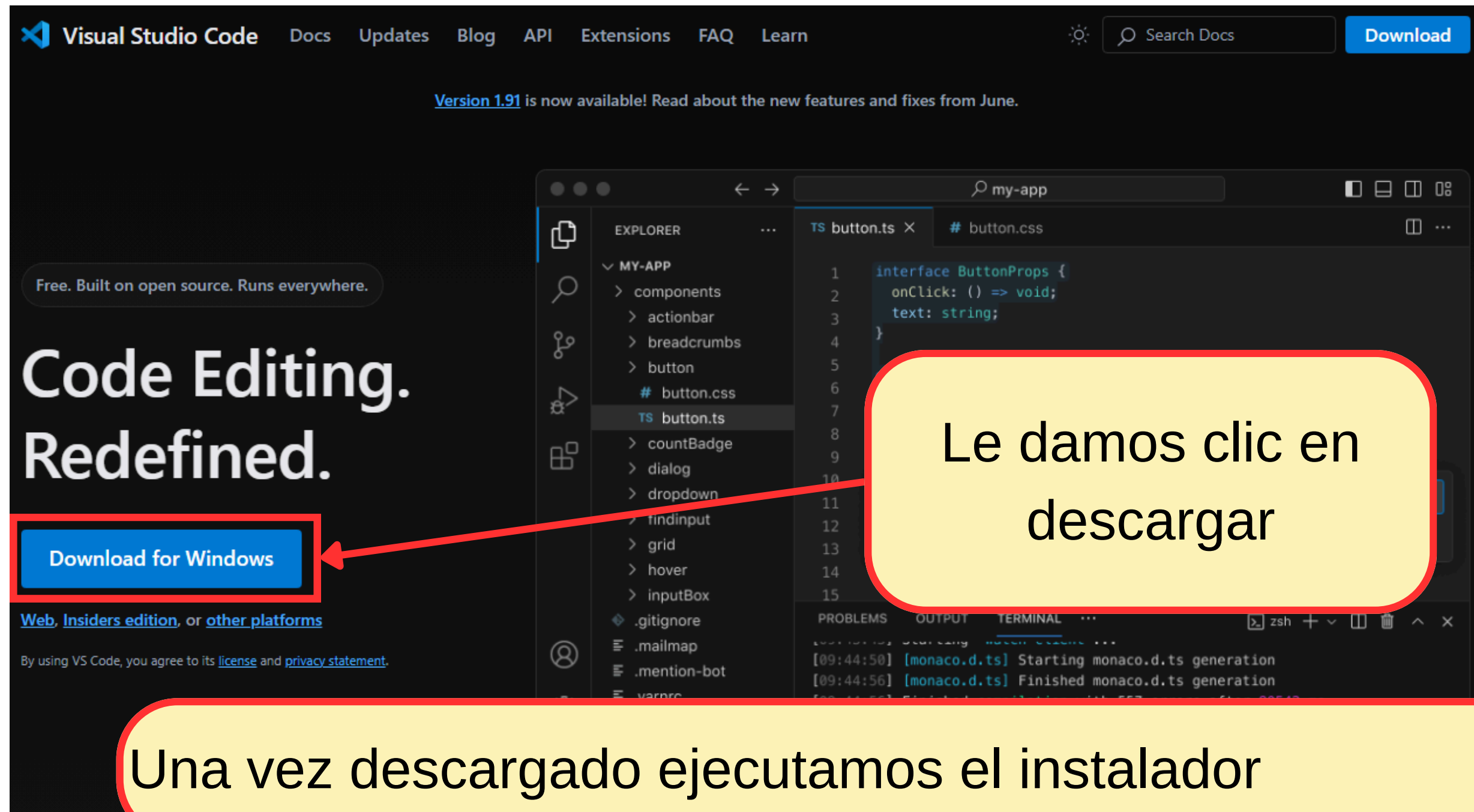
1.1 Luego de darle en la opción nos saldrá la pantalla de carga de la instalación y por último nos saldrá el mensaje de que Python se ha instalado correctamente:



# Preparar Nuestro Entorno de Desarrollo

## 2. Descargar Visual Studio Code

- Ingresamos a la pagina <https://code.visualstudio.com/>



The screenshot shows the Visual Studio Code website. The main heading is "Code Editing. Redefined." Below it, there is a blue button labeled "Download for Windows". A red box highlights this button, and a red arrow points from a yellow callout box to it. The callout box contains the text "Le damos clic en descargar". Another yellow callout box at the bottom of the page contains the text "Una vez descargado ejecutamos el instalador". The background of the screenshot shows a preview of the Visual Studio Code editor interface with a file explorer on the left and a code editor in the center.

Visual Studio Code Docs Updates Blog API Extensions FAQ Learn

Version 1.91 is now available! Read about the new features and fixes from June.

Free. Built on open source. Runs everywhere.

# Code Editing. Redefined.

[Download for Windows](#)

[Web](#), [Insiders edition](#), or [other platforms](#)

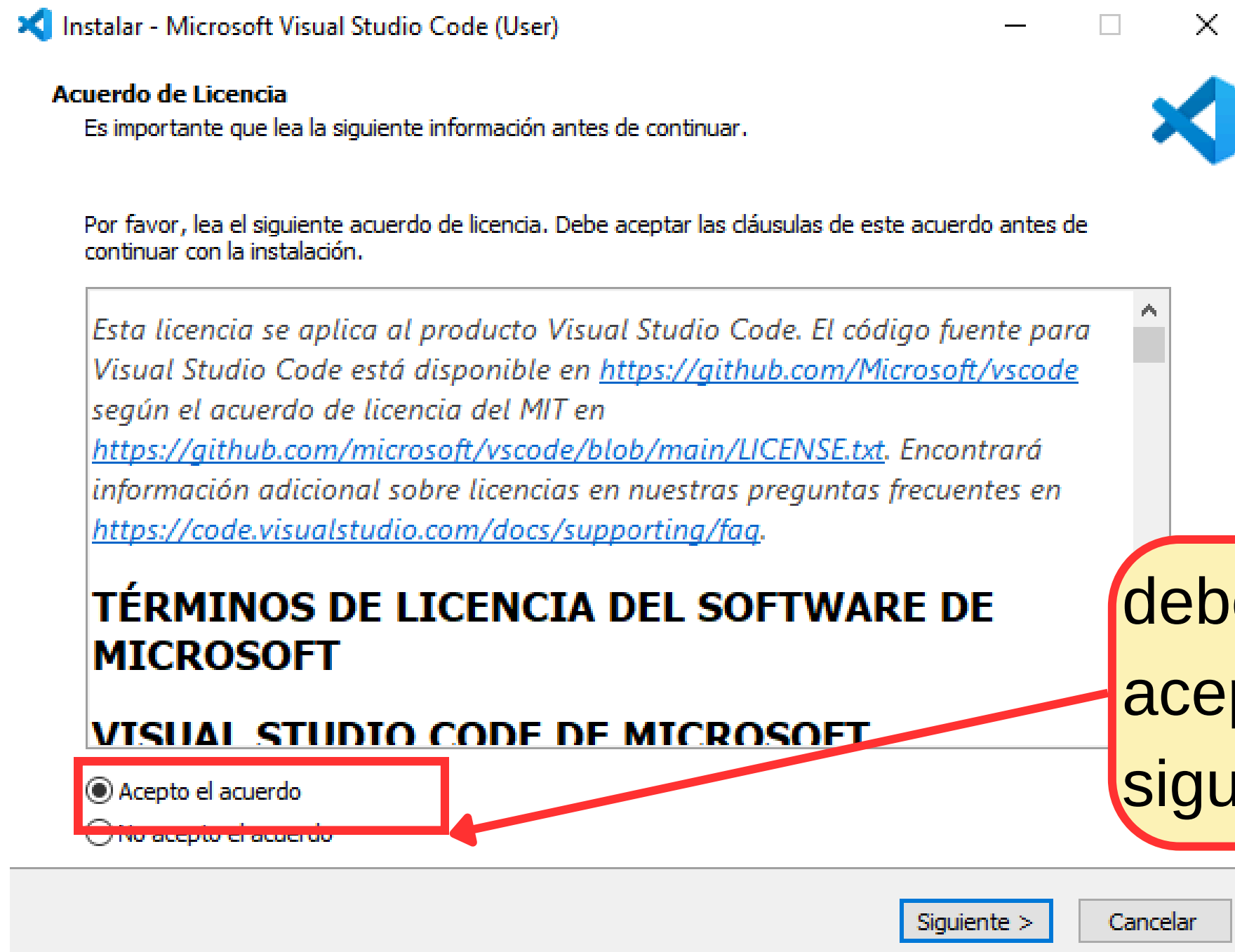
By using VS Code, you agree to its [license](#) and [privacy statement](#).

Le damos clic en descargar

Una vez descargado ejecutamos el instalador

# Preparar Nuestro Entorno de Desarrollo

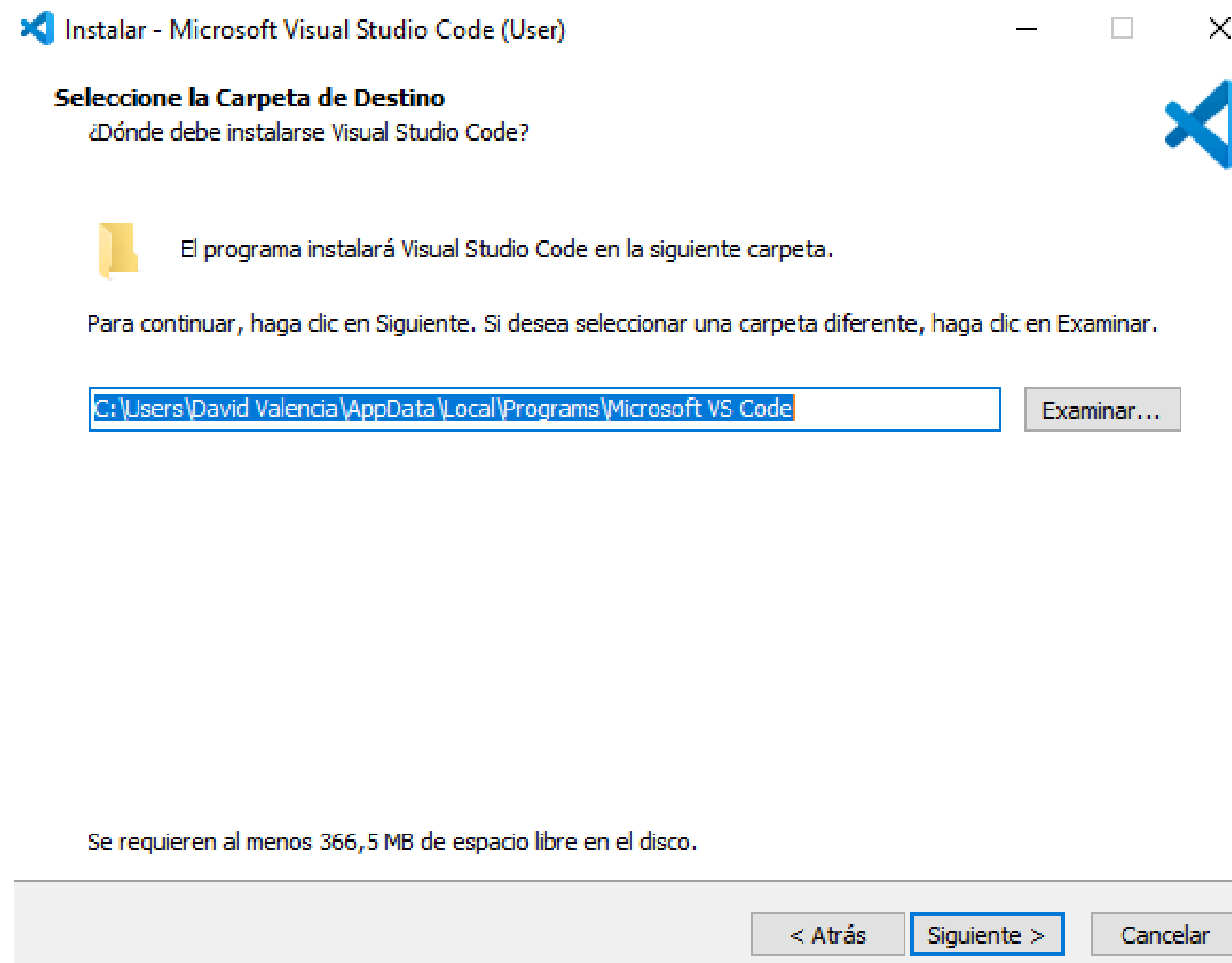
2.1 Luego de ejecutar la instalador nos saldrá la siguiente pantalla:



debemos seleccionar la opción acepto el acuerdo y luego en siguiente

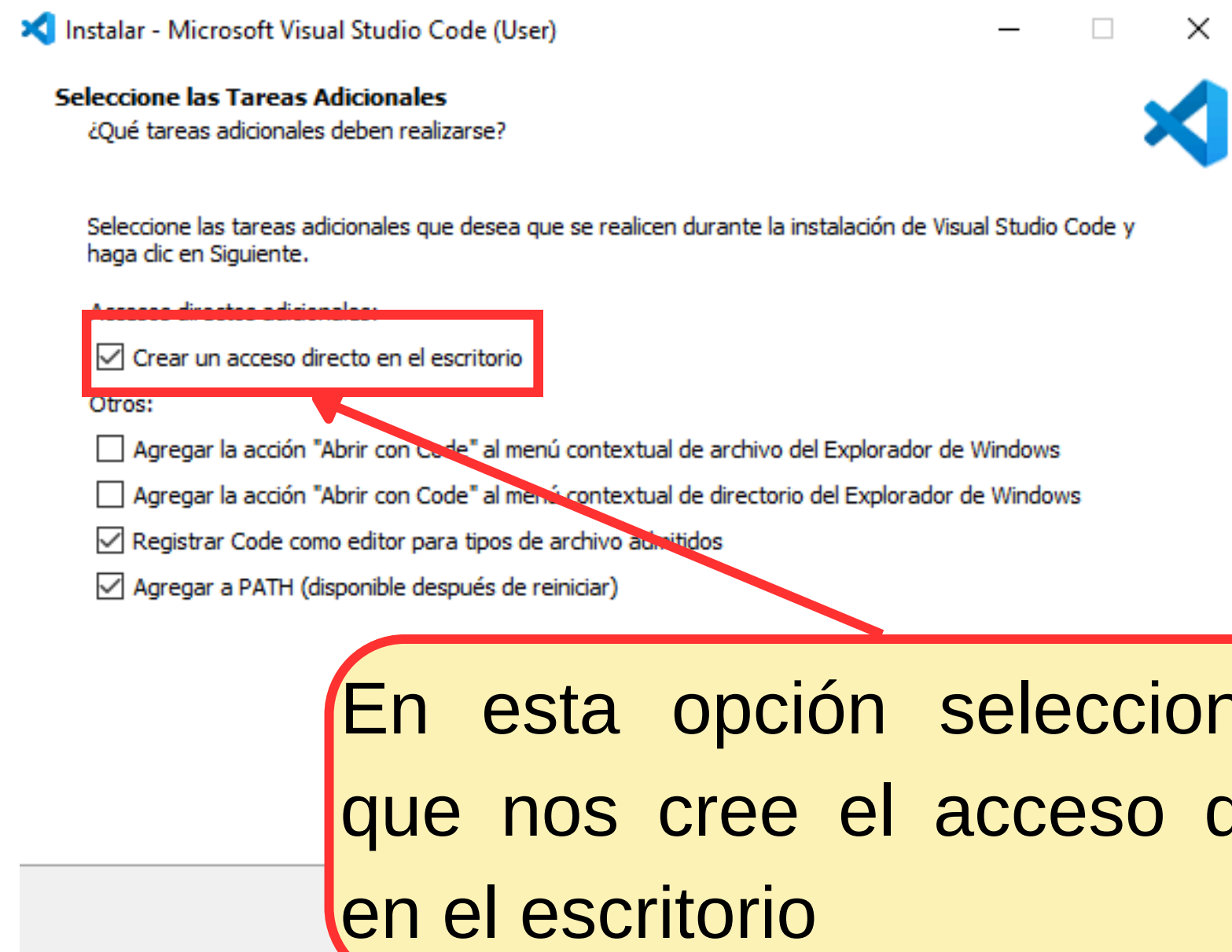
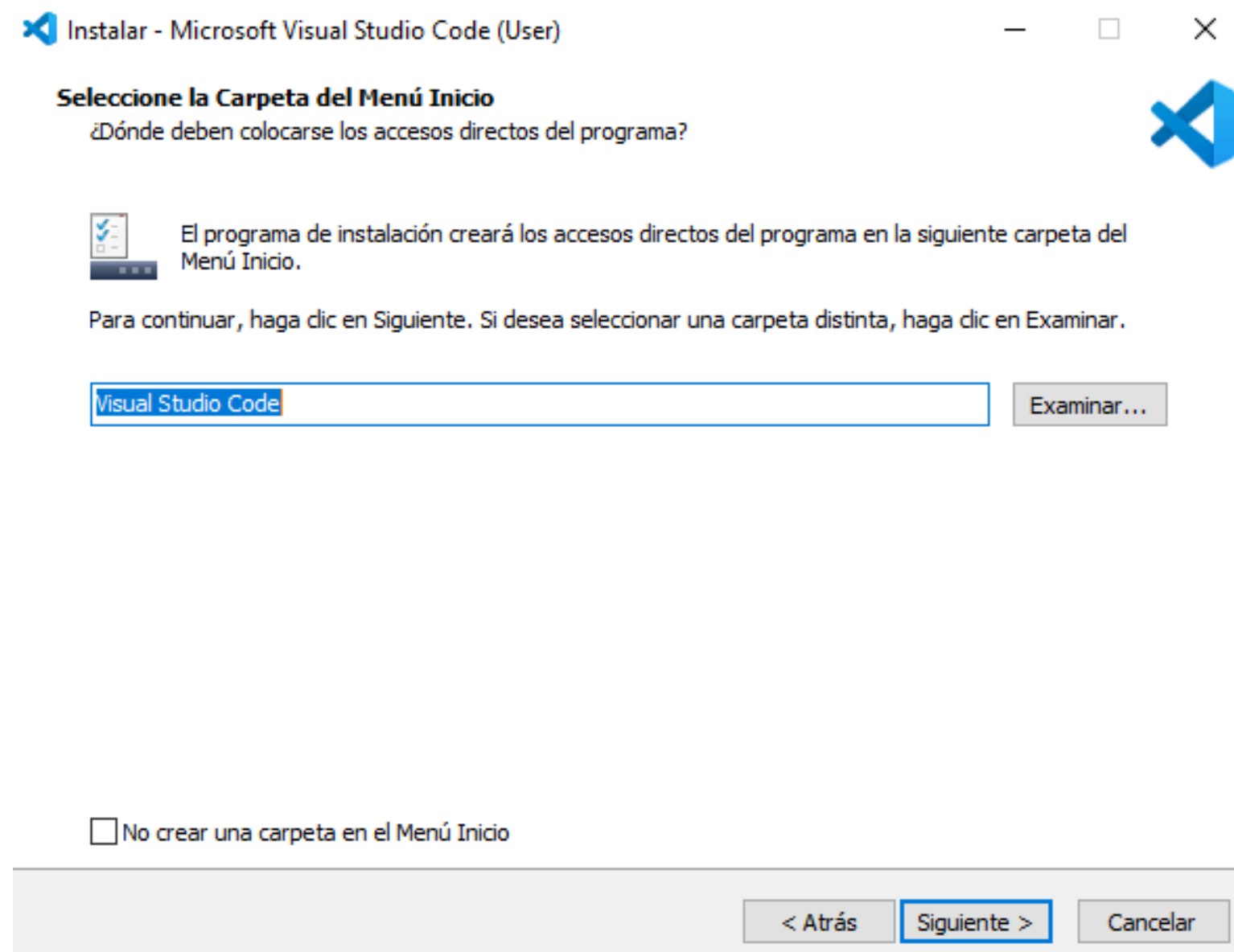
# Preparar Nuestro Entorno de Desarrollo

2.2 en la siguiente opción nos pedirá en que carpeta lo vamos a instalar, lo dejamos con la que viene por defecto y luego clic en el botón siguiente:



# Preparar Nuestro Entorno de Desarrollo

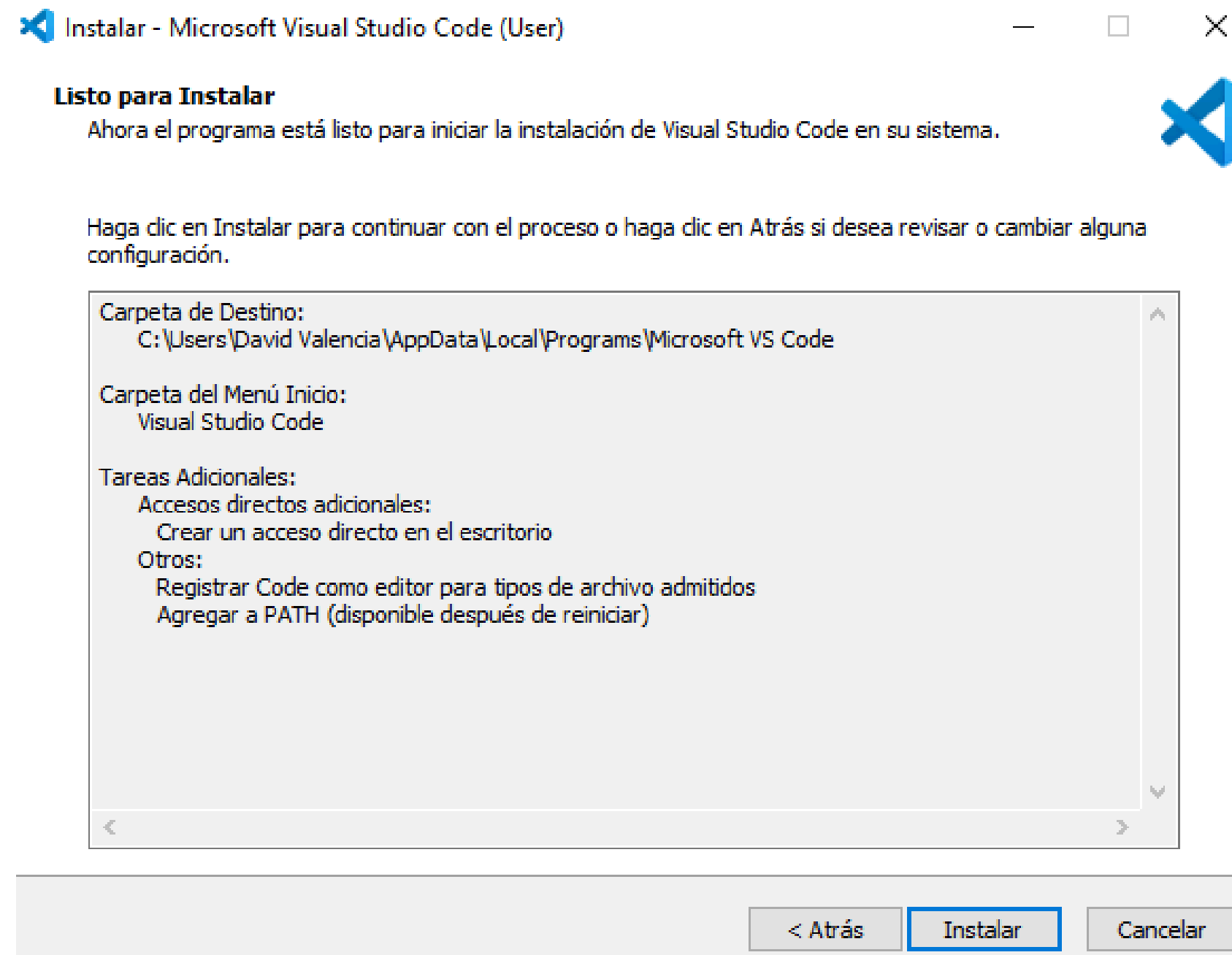
2.3 en las siguientes opciones le damos en siguiente dejamos todo lo que viene por defecto:



En esta opción seleccionamos que nos cree el acceso directo en el escritorio

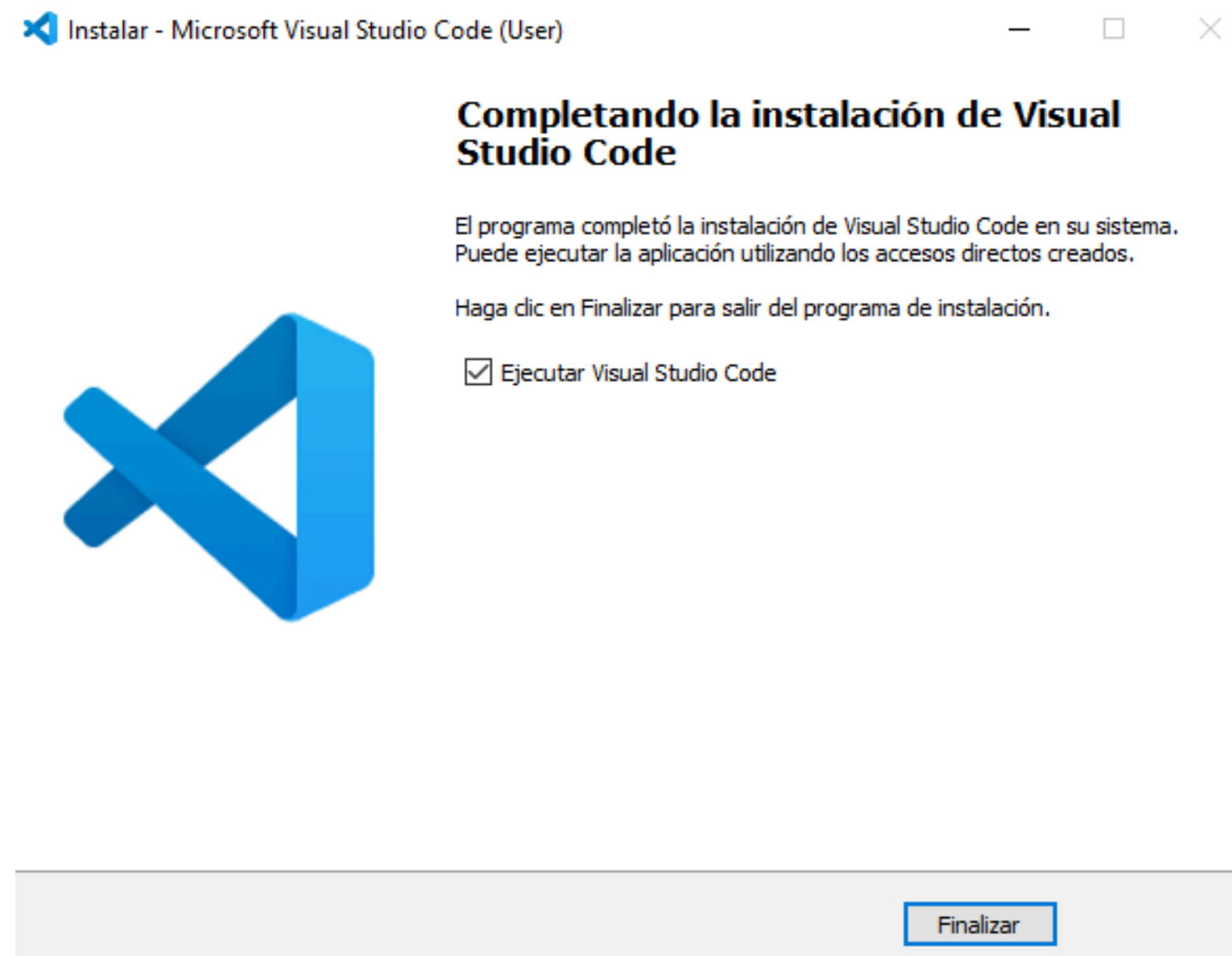
# Preparar Nuestro Entorno de Desarrollo

2.4 Luego nos saldrá la siguiente pantalla y solo le damos clic donde dice instalar:



# Preparar Nuestro Entorno de Desarrollo

2.5 Luego de darle en la opción nos saldrá la pantalla de carga de la instalación y por último nos saldrá el mensaje de que Visual se ha instalado correctamente:

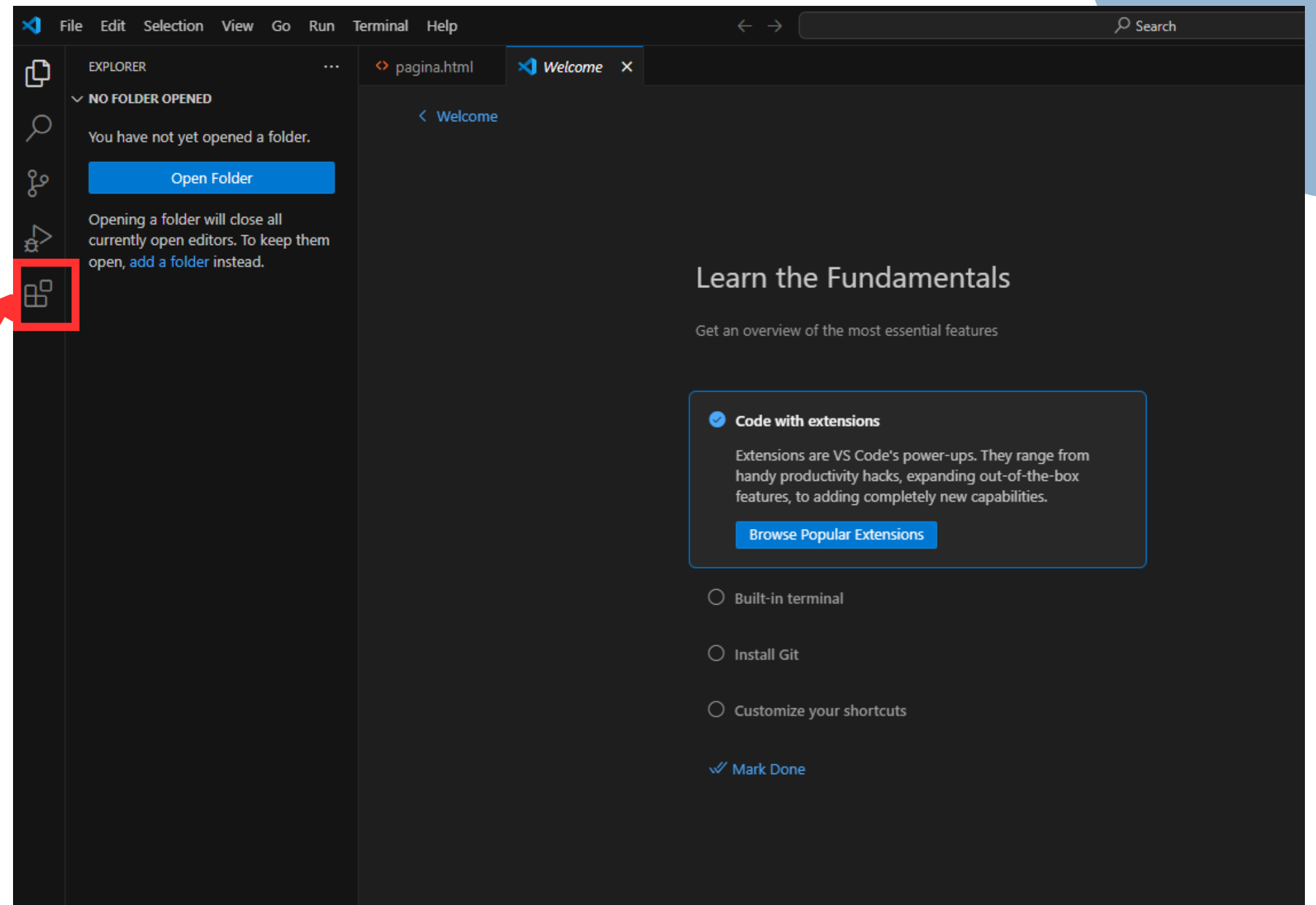


# Preparar Nuestro Entorno de Desarrollo

3. cuando instalamos visual studio code, el por defecto viene en idioma ingles para cambiarlo a español debemos de realizar los siguientes pasos:

3.1 Abrimos Visual Studio Code:

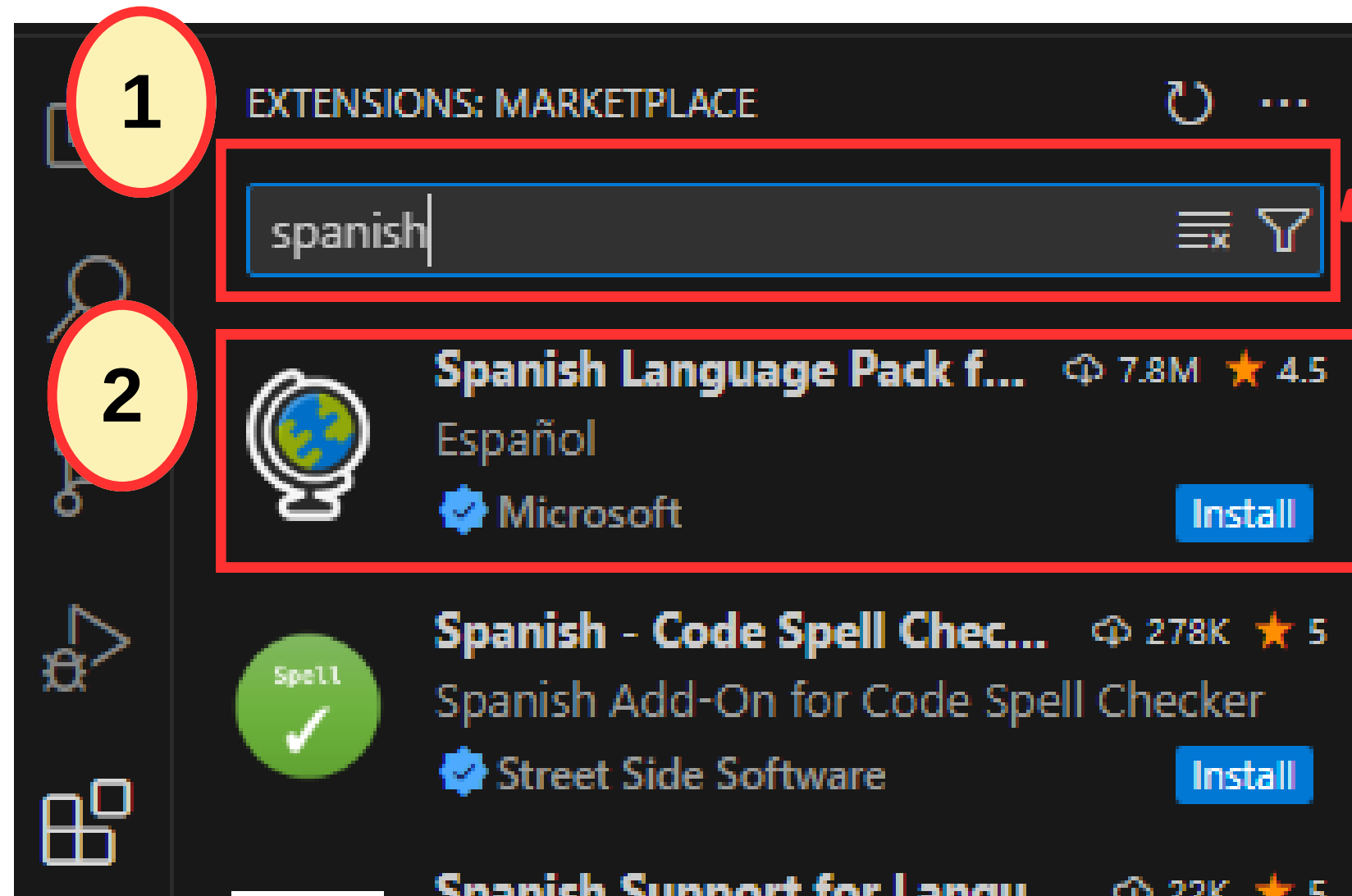
damos clic en extensiones





# Preparar Nuestro Entorno de Desarrollo

3.2 nos saldra la siguiente pantalla:



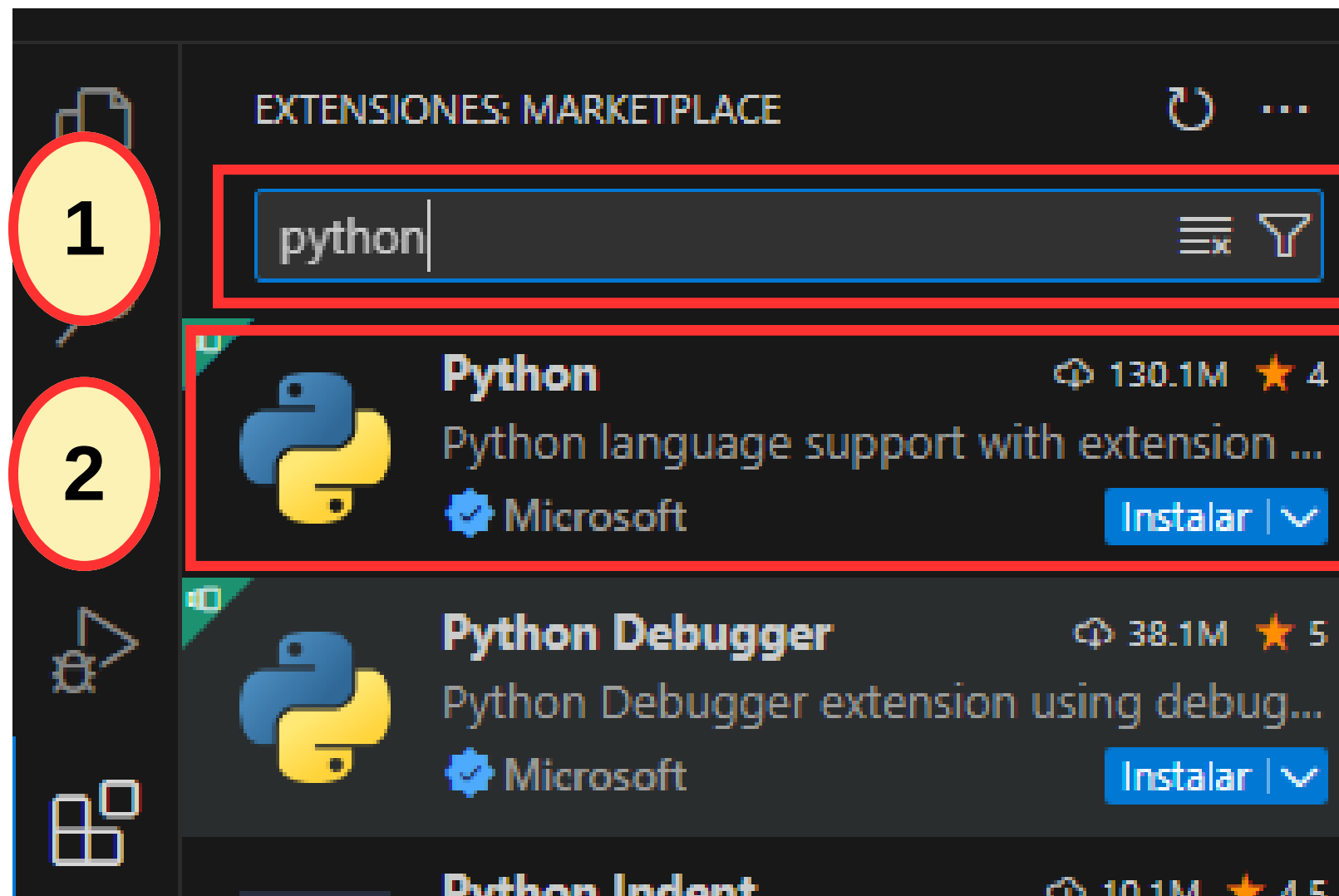
en el buscador escribimos spanish

le damos clic al primero y luego donde dice install

Luego nos saldrá esta ventana le damos clic al botón azul

# Preparar Nuestro Entorno de Desarrollo

4. para trabajar con python desde visual studio code debemos instalar la siguiente extensión:



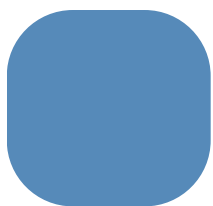
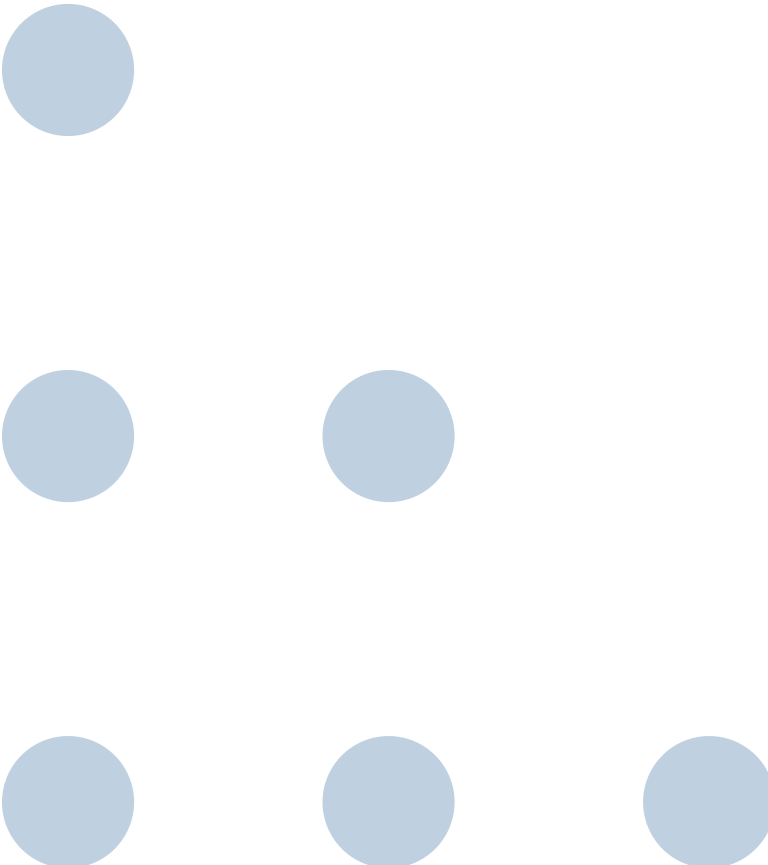
en el buscador escribimos python

le damos clic al primero y luego donde dice install

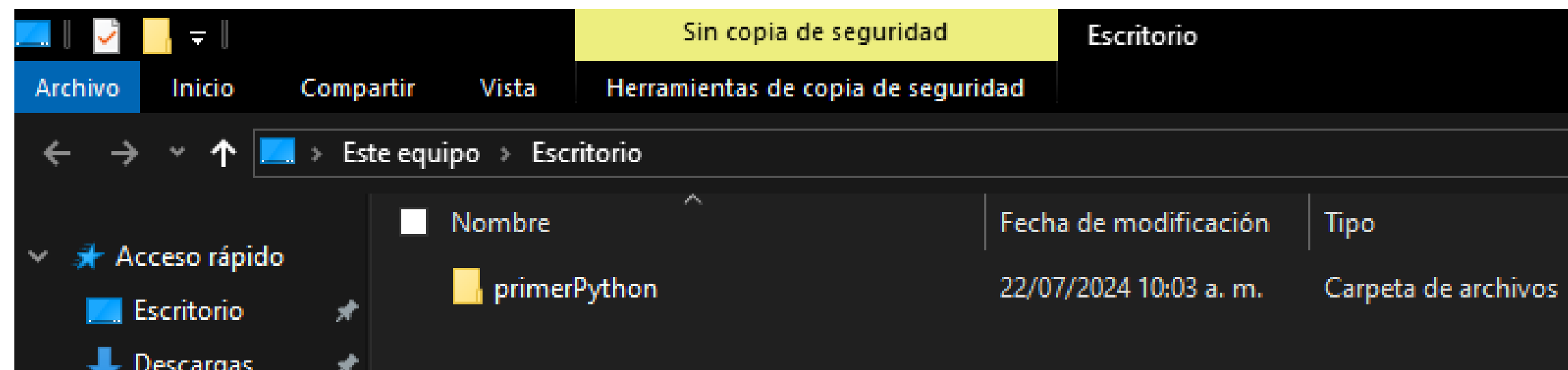


Primer

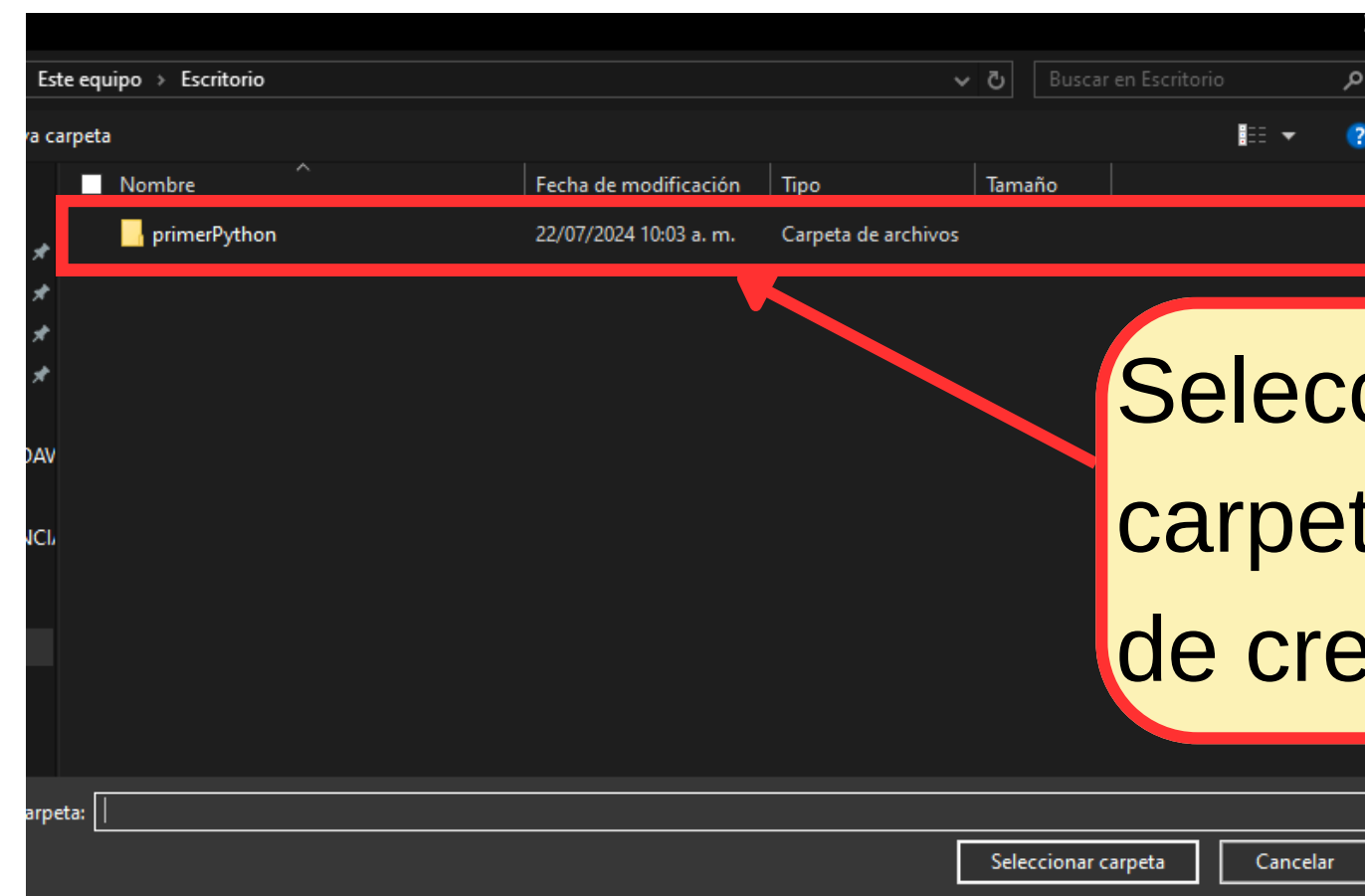
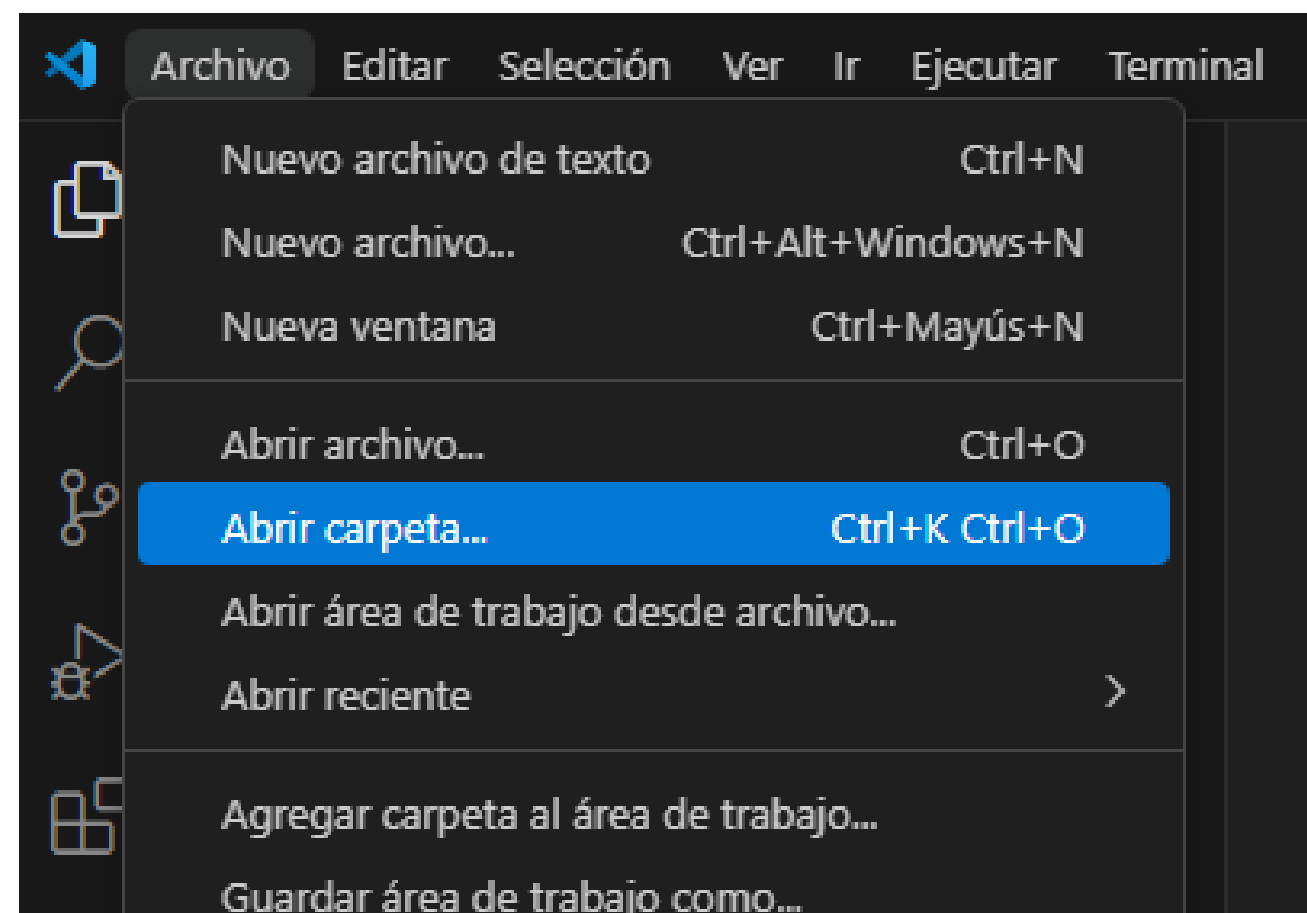
Programa en  
Python



1. Vamos a crear una carpeta en el escritorio que se llama **primerPython**

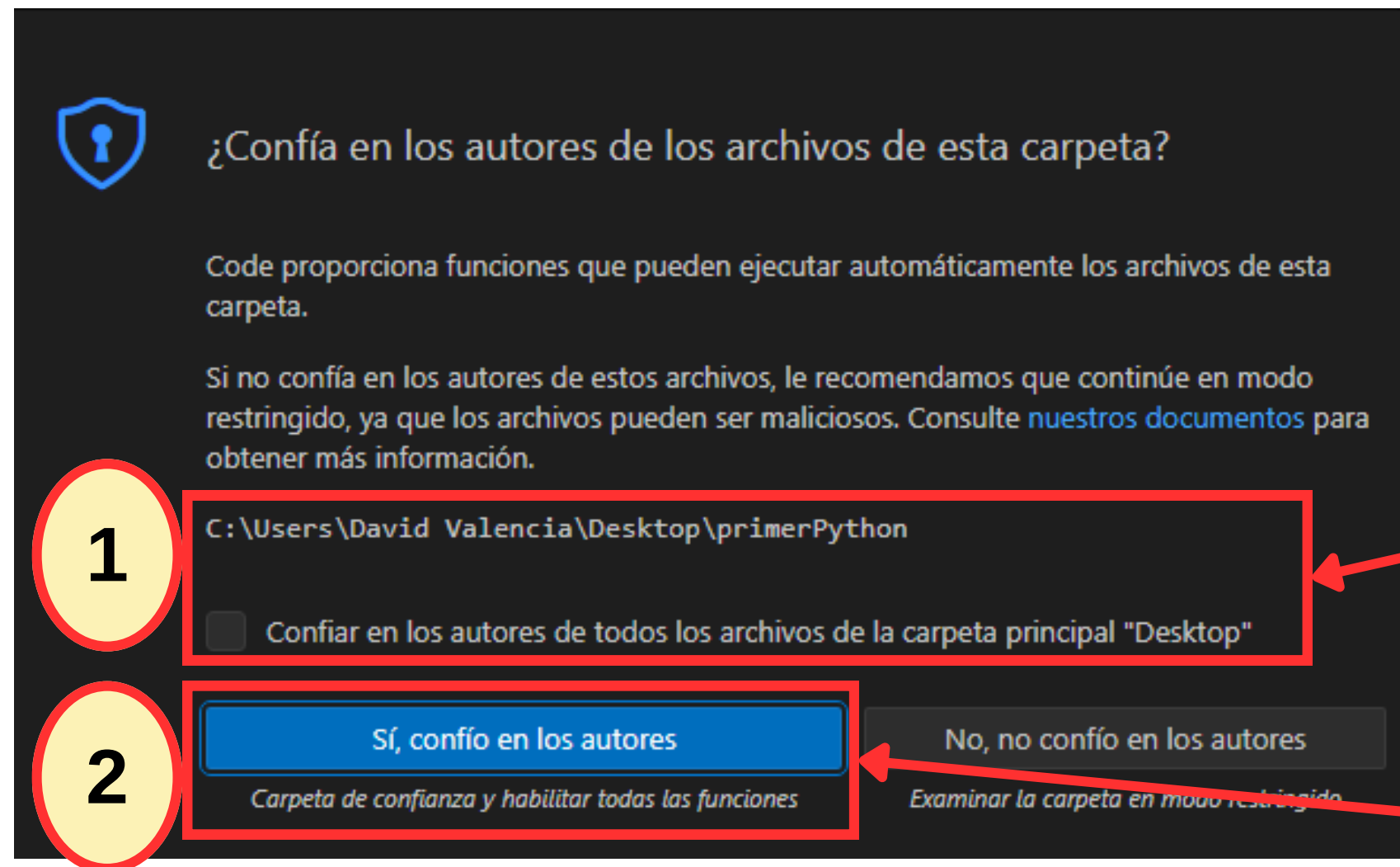


2. Luego abrimos visual estudio code, damos en archivo y abrir carpeta



Seleccionamos la carpeta que acabamos de crear

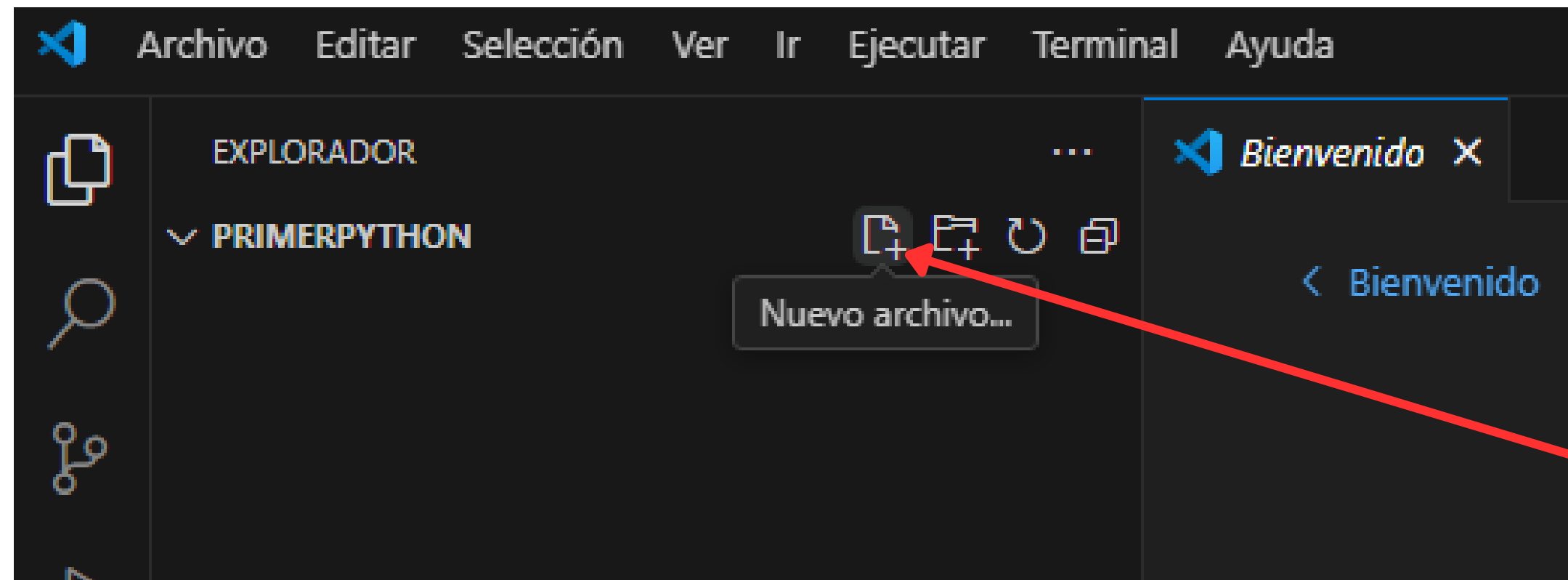
Si nos sale este mensajes hacemos los siguiente:



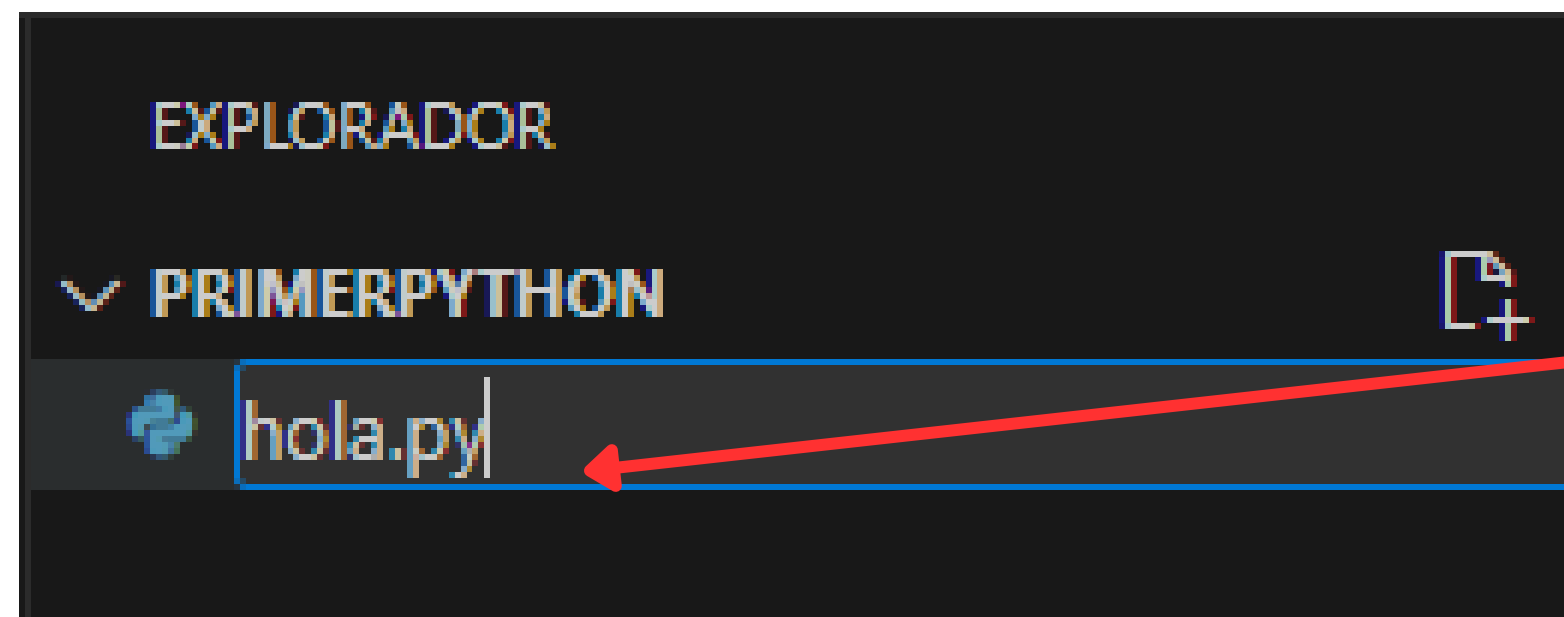
marcamos en la casilla

damos clic en si, confié en los autores

### 3. Vamos a crear un nuevo archivo python:



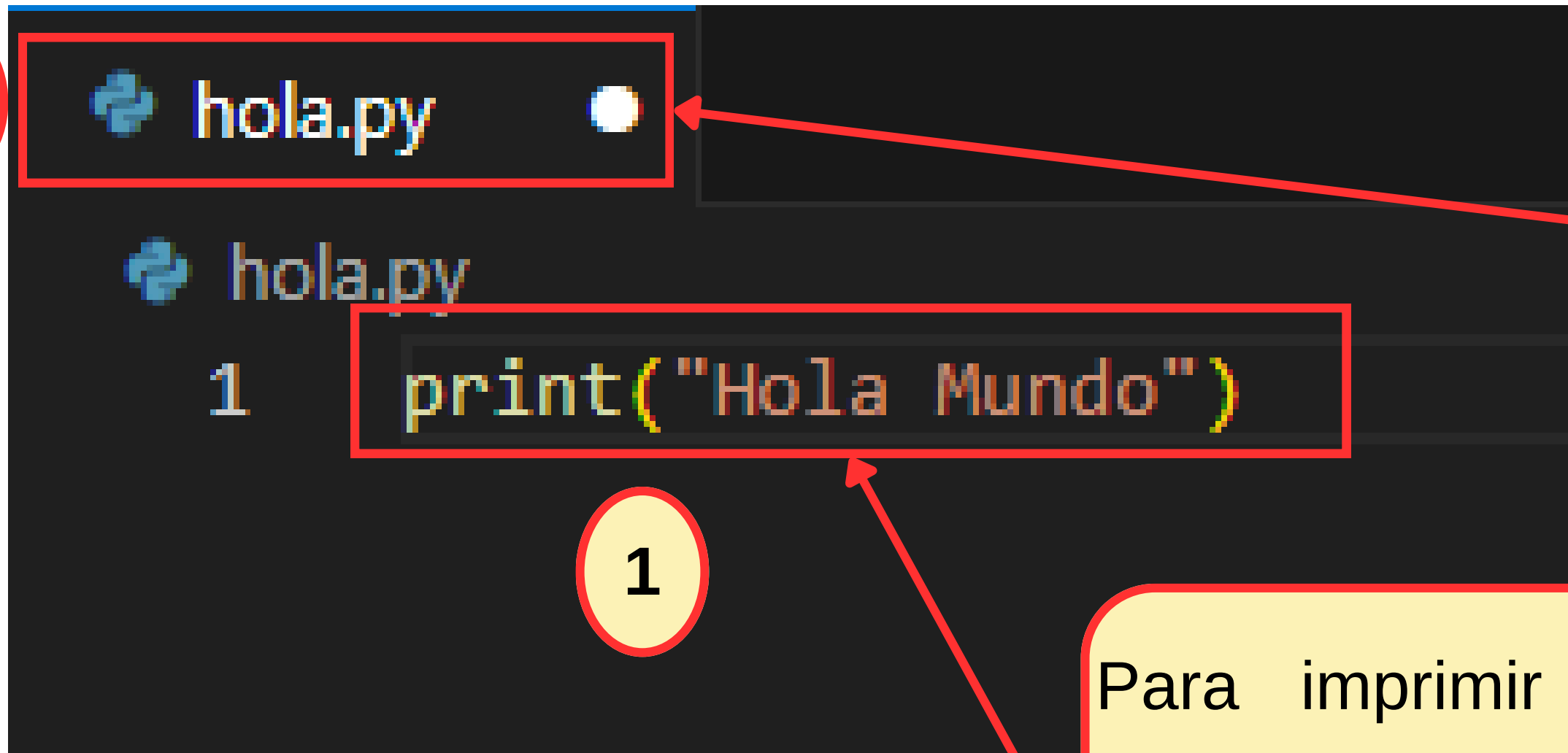
damos clic en nuevo archivo



nos saldra un cuadro de texto donde le colocaremos como nombre al archivo hola.py

**TODOS LOS ARCHIVO PYHTON TERMINAN EN .py como extensión**

4. cada lenguaje de programación tiene su sintaxis de código en este primer programa vamos a imprimir algo en pantalla:



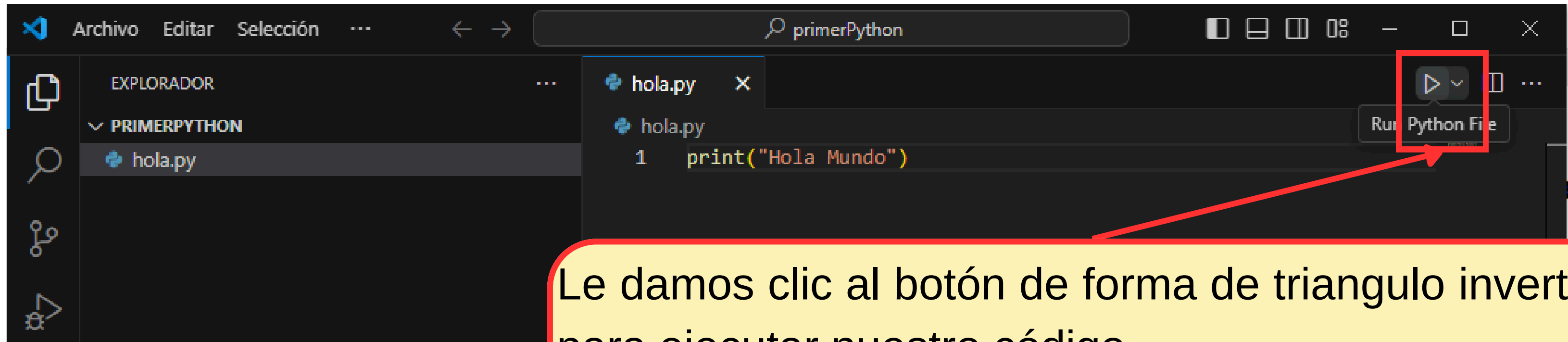
The image shows a code editor interface with a dark background. At the top, a file named 'hola.py' is open, indicated by a blue Python logo and a white dot. Below this, the code editor shows a single line of Python code: `print("Hola Mundo")`. The line is numbered '1' on the left. Two red boxes highlight specific elements: one box around the file name 'hola.py' with a red arrow pointing to it from a yellow callout box labeled '2', and another box around the code line with a red arrow pointing to it from a yellow callout box labeled '1'.

Siempre que hagamos un cambio debemos de ir a guardar, hay dos formas:

1. archivo, guardar.
2. dando al tiempo control S.

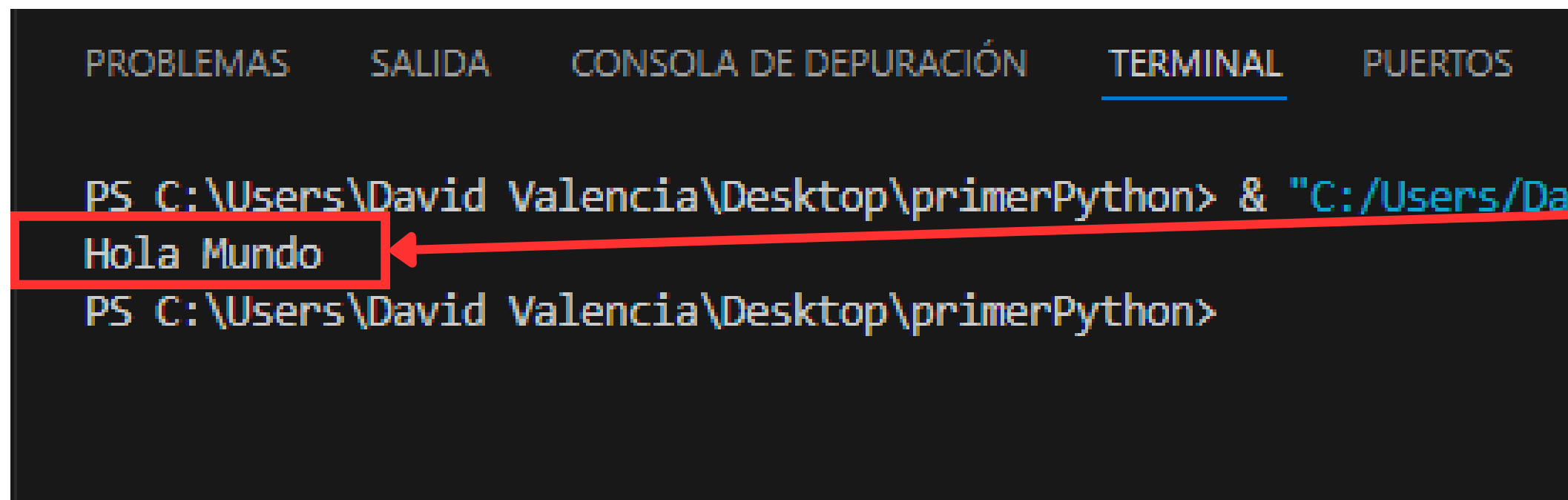
Para imprimir en la consola de python siempre usamos la instrucción **print** seguido de paréntesis y el valor que vamos a imprimir, recuerden si es una cadena va dentro de comillas.

4. para ejecutar un archivo python lo hacemos de la siguiente manera:



Le damos clic al botón de forma de triángulo invertido para ejecutar nuestro código

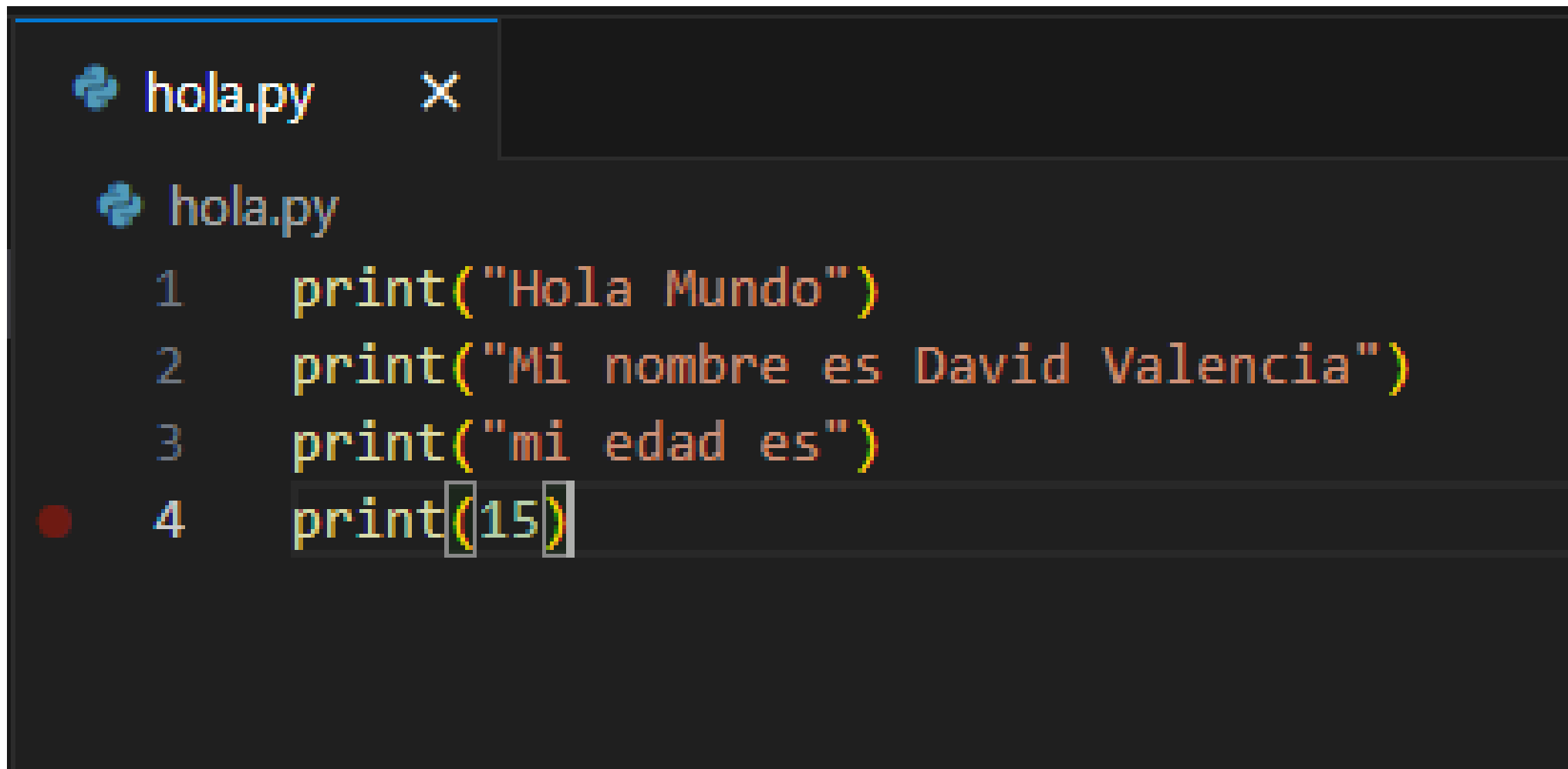
En la parte de abajo abrirá una ventana llamada terminal mostrando el resultado del código



Este es el resultado de la impresión realizada en nuestro código

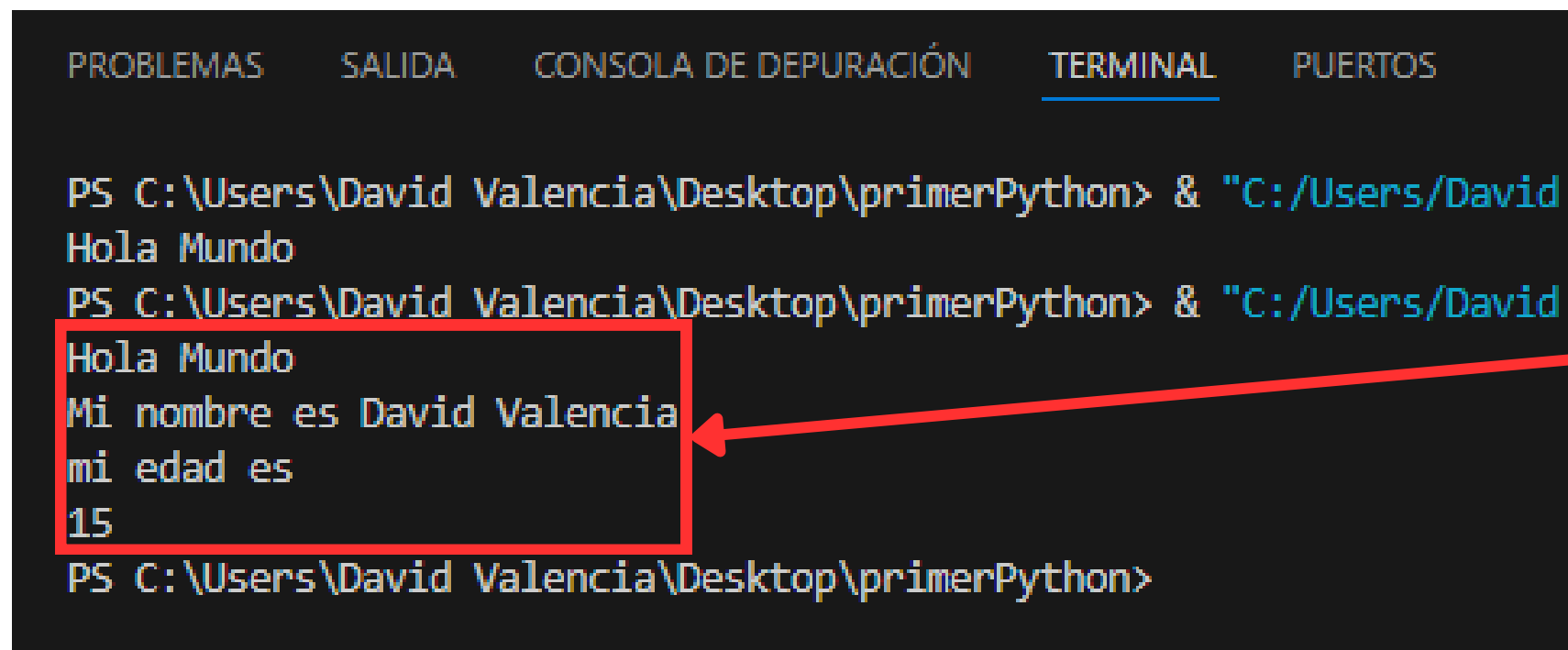


5. si deseamos imprimir mas cosas en terminal lo podemos hacer el simultaneo con otro codigo:



```
hola.py X
hola.py
1 print("Hola Mundo")
2 print("Mi nombre es David Valencia")
3 print("mi edad es")
4 print(15)
```

Recuerden cada vez que hagan un cambio guardar los mismos sino no van a ver reflejado lo que escriben

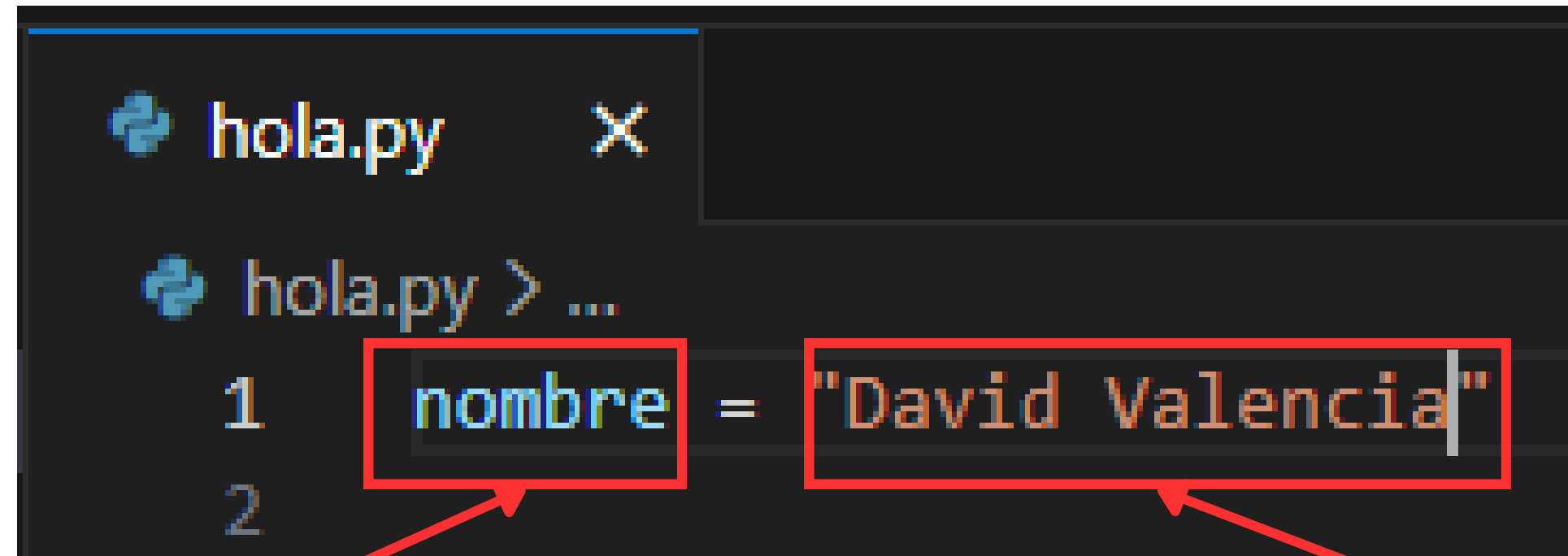


```
PROBLEMAS  SALIDA  CONSOLA DE DEPURACIÓN  TERMINAL  PUERTOS

PS C:\Users\David Valencia\Desktop\primerPython> & "C:/Users/David
Hola Mundo
PS C:\Users\David Valencia\Desktop\primerPython> & "C:/Users/David
Hola Mundo
Mi nombre es David Valencia
mi edad es
15
PS C:\Users\David Valencia\Desktop\primerPython>
```

Este seria el resultado en la terminal

6. para declarar o crear variables en python se hace de la siguiente manera:

A screenshot of a Python IDE window titled 'hola.py'. The code editor shows two lines of code: line 1 contains 'nombre = "David Valencia"' and line 2 is empty. The word 'nombre' is highlighted with a red box, and the string value '"David Valencia"' is also highlighted with a red box. Two red arrows point from these boxes to callout boxes below the code.

```
hola.py x
hola.py > ...
1 nombre = "David Valencia"
2
```

como nombramos la variable  
en este caso se llama **nombre**

el valor que le estamos dando a  
la variable **nombre** en este  
caso David Valencia

6. para imprimir esta variable en la terminal lo hacemos de la siguiente manera:

```
hola.py x
hola.py > ...
1 nombre = "David Valencia"
2 print(nombre)
```

Lo hacemos por medio de un **print** y le enviamos la variable creada anteriormente.

```
PROBLEMAS SALIDA CONSOLA DE DEPURACIÓN TERMINAL PUERTOS
PS C:\Users\David Valencia\Desktop\primerPython> & "C:/Users/Da
David Valencia
PS C:\Users\David Valencia\Desktop\primerPython>
```

Nos mostrará el valor de la variable que se imprimió

7. si deseo imprimir dos variables en un mismo **print** python nos permite concatenar por medio de una “,”

```
hola.py x
hola.py > ...
1 nombre = "David"
2 apellido = "Valencia"
3 print(nombre, apellido)
```

Declaramos las dos variables nombre y apellido con sus valores.

Luego lo mandamos a imprimir por medio del **print** y le pasamos las dos variables separadas por coma.

```
PROBLEMAS SALIDA CONSOLA DE DEPURACIÓN TERMINAL PUERTOS

PS C:\Users\David Valencia\Desktop\primerPython> & "C:/Users/Da
David Valencia
PS C:\Users\David Valencia\Desktop\primerPython>
```

Tendremos este resultado

8. También podemos agregarle texto adicional dentro de mi **print**:

```
hola.py x
hola.py > ...
1 nombre = "David"
2 apellido = "Valencia"
3 print("Mi nombre es: ", nombre, apellido)
```

Texto adicional inicial entre comillas porque es un cadena y con un coma después para concatenarlo

```
PROBLEMAS  SALIDA  CONSOLA DE DEPURACIÓN  TERMINAL  PUERTOS

PS C:\Users\David Valencia\Desktop\primerPython> & "C:/Users/D
Mi nombre es: David Valencia
PS C:\Users\David Valencia\Desktop\primerPython>
```

Tendremos este resultado

9. las variables pueden cambiar su valor en el transcurso de la ejecución del código:

```
hola.py X
hola.py > ...
1 nombre = "David"
2 apellido = "Valencia"
3 print("Mi nombre es: ", nombre, apellido)
4 nombre = "Diego"
5 print("Mi nombre es: ", nombre, apellido)
```

La variable nombre cambio su valor a Diego y si se vuelve a imprimir aparecerá **Mi nombre es: Diego Valencia**

```
PROBLEMAS  SALIDA  CONSOLA DE DEPURACIÓN  TERMINAL  PUERTOS

PS C:\Users\David Valencia\Desktop\primerPython> & "C:/Users/Dav
Mi nombre es: David Valencia
Mi nombre es: Diego Valencia
PS C:\Users\David Valencia\Desktop\primerPython>
```

Tendremos este resultado

10. todos los lenguajes de programación permiten poder realizar comentarios dentro del código para alguna necesidad puntual:

```
hola.py  X
hola.py > ...
1  #este programa sirve para saber el nombre de las personas
2  nombre = "David" # así se declara una variable
3  apellido = "Valencia" # esta es la variable apellido de la persona
4  print("Mi nombre es: ", nombre, apellido)
5  nombre = "Diego"
6  print("Mi nombre es: ", nombre, apellido) # así puedo imprimir
7  |
```

El texto que aparece en verde u otro color diferente son comentarios dentro del sistema, este texto el programa no lo tiene en cuenta solo es texto que el programador escribe para recordar algo. Para crear un comentario se debe iniciar con el signo # y luego el texto

11. para trabajar con diferentes tipos de datos, **python** lo hace de la siguiente manera:

```
hola.py x
hola.py > ...
1 nombre = "David"
2 apellido = "Valencia"
3 print("Mi nombre es: ", nombre, apellido)
4 edad = 15
5 print("Mi edad es", edad)
6 estatura = 1.86
7 print("Mi estatura es", estatura)
```

Cuando están dentro de comillas son cadena de texto

Cuando no tienen comillas y son números enteros lo toma como variable numerica

Cuando no tienen comillas y son numeros separados por puntos lo toma como decimal o numero flotante

```
PROBLEMAS  SALIDA  CONSOLA DE DEPURACIÓN  TERMINAL  PUERTOS

PS C:\Users\David Valencia\Desktop\primerPython> & "C:/Users/Da
Mi nombre es: David Valencia
Mi edad es 15
Mi estatura es 1.86
PS C:\Users\David Valencia\Desktop\primerPython>
```

Tendremos este resultado



11. si se desea operar o iterar con una variable numérica o decimal se hace de la siguiente manera:

```
nombre = "David"
apellido = "Valencia"
print("Mi nombre es: ", nombre, apellido)
edad = 15+15
print("Mi edad es", edad)
estatura = 1.86+0.050
print("Mi estatura es", estatura)
```

a las variables numéricas le podemos sumar mas números o también otras variables numéricas o decimales

A las variables decimales también le podemos sumar mas números o también otras variables sean numéricas o decimales

```
PROBLEMAS  SALIDA  CONSOLA DE DEPURACIÓN  TERMINAL  PUERTOS

PS C:\Users\David Valencia\Desktop\primerPython> & "C:/Users/D
Mi nombre es:  David Valencia
Mi edad es 30
Mi estatura es 1.9100000000000001
PS C:\Users\David Valencia\Desktop\primerPython>
```

Tendremos este resultado