



Introducción a la Programación IP

Laboratorio 1 de Nivel 1

N1-L1: Ambientes Básicos de Trabajo

Eduardo Rosales

ee.rosales24@uniandes.edu.co

Departamento de Ingeniería de Sistemas y Computación

Universidad de los Andes



► Entrega

Al final de las diapositivas:

- Instrucciones para enviar el laboratorio mediante Bloque Neón

► Equipo

Puede realizar el laboratorio en:

- un equipo del laboratorio
- ó en su propio equipo

► ¿Necesita ayuda?

Para ayuda, levante la mano

- Nuestros nombres:
 - *Eduardo*
 - *Joshua*
 - *Juan José*



¿Anaconda Instalado?

► Equipo

Si usará un equipo de la sala:

- **Vaya a la diapositiva 6**

► Si usará su propio equipo:

Con Anaconda instalado:

- **Vaya a la diapositiva 6**

Sin Anaconda instalado:

- Continúe a la **siguiente diapositiva**



► Descarga

Descargue el instalador del siguiente link:

- <https://www.anaconda.com/download/success>

Si su sistema es **Windows**

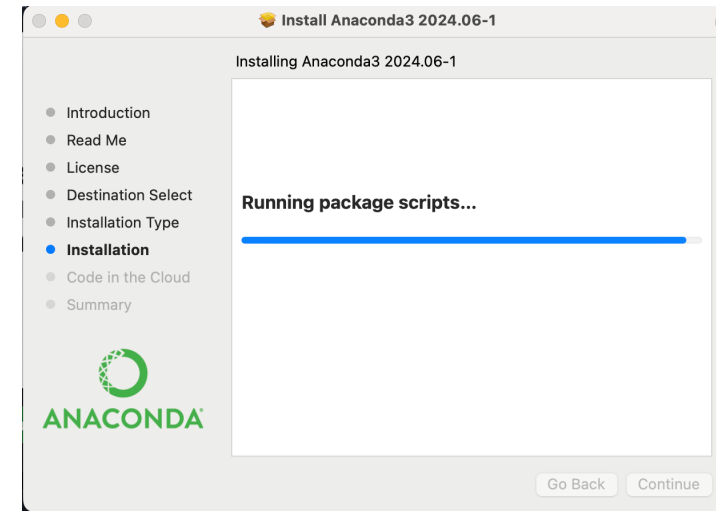
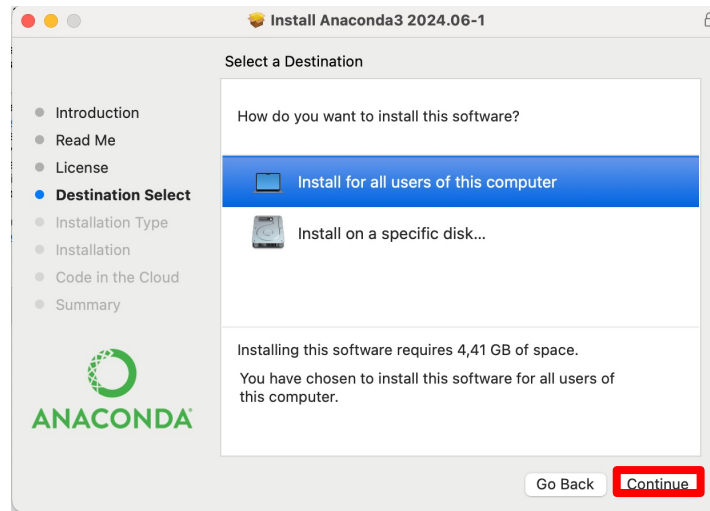
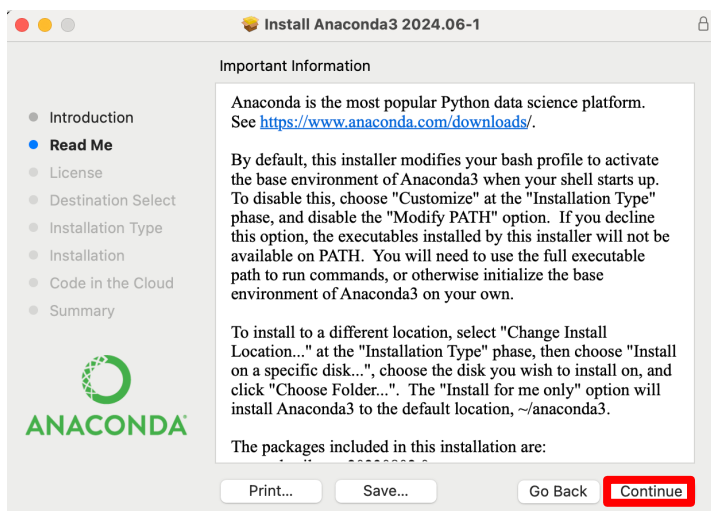
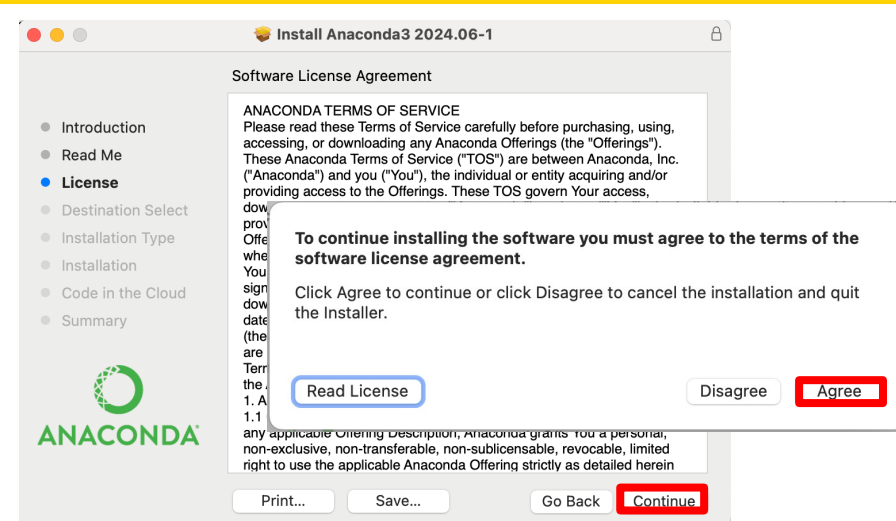
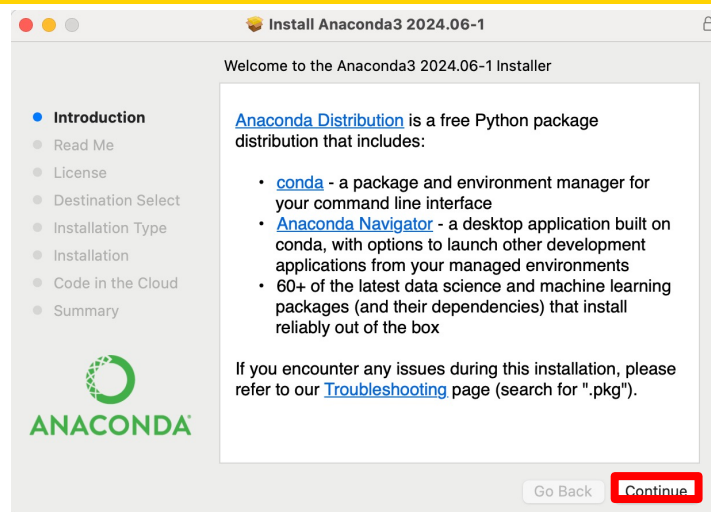
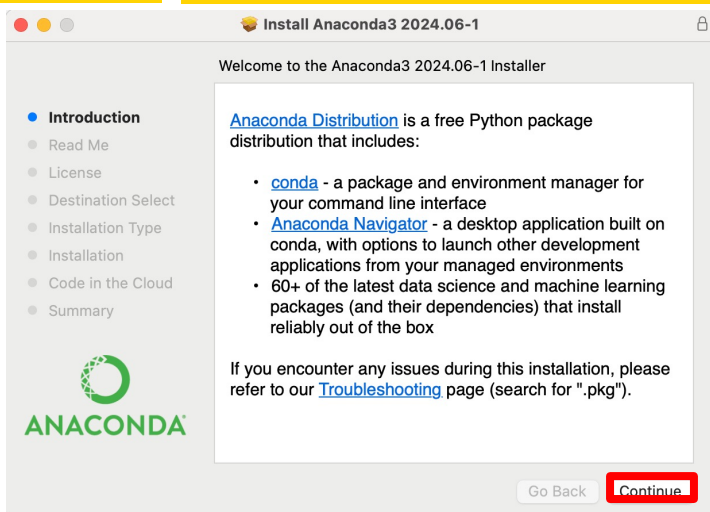
- [64-Bit Graphical Installer \(912.3M\)](#)

En el caso de **Mac**

- Si su procesador es un Silicon:
 - [64-Bit \(Apple silicon\) Graphical Installer \(704.7M\)](#)
- Si su procesador es Intel:
 - [64-Bit \(Intel chip\) Graphical Installer \(734.7M\)](#)

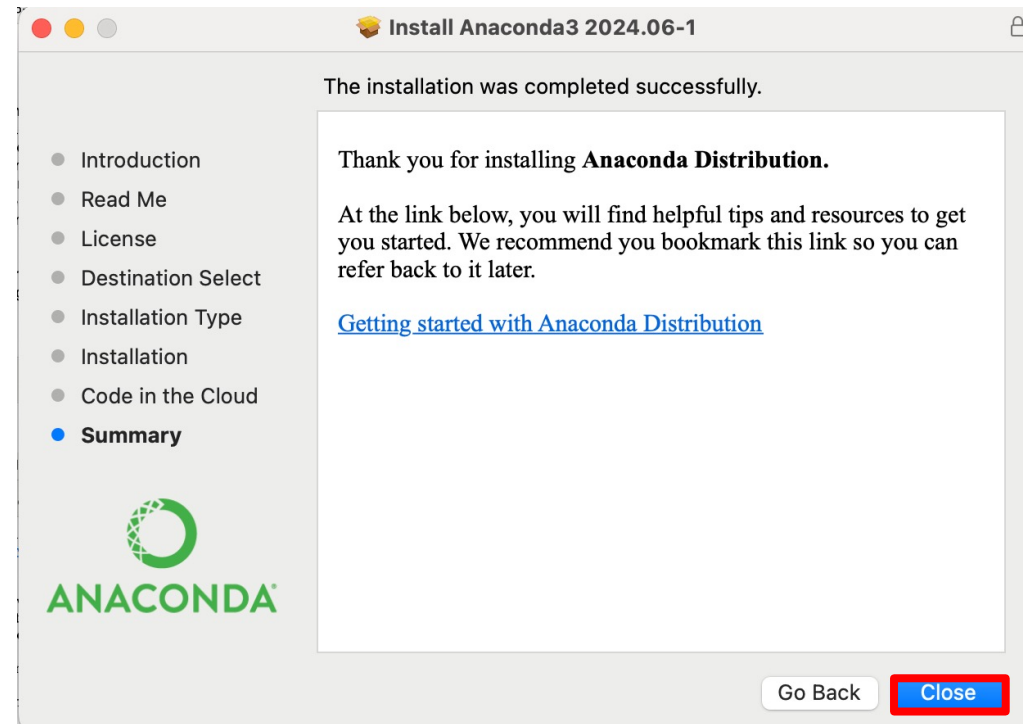
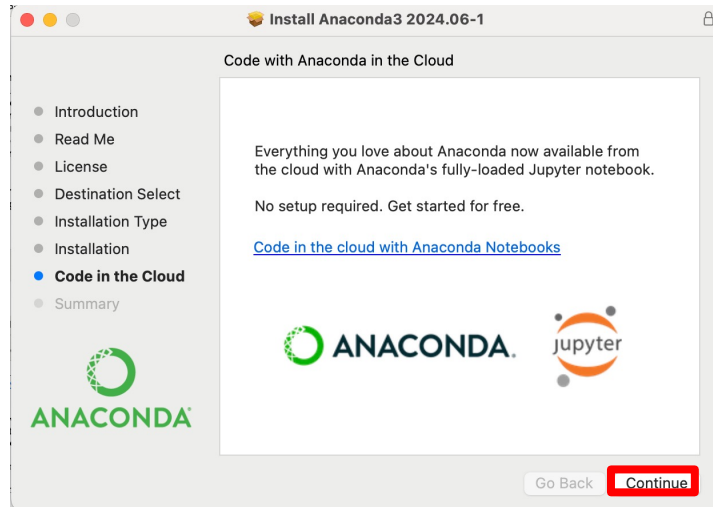


Instalación de Anaconda (2 de 3)





Instalación de Anaconda (3 de 3)

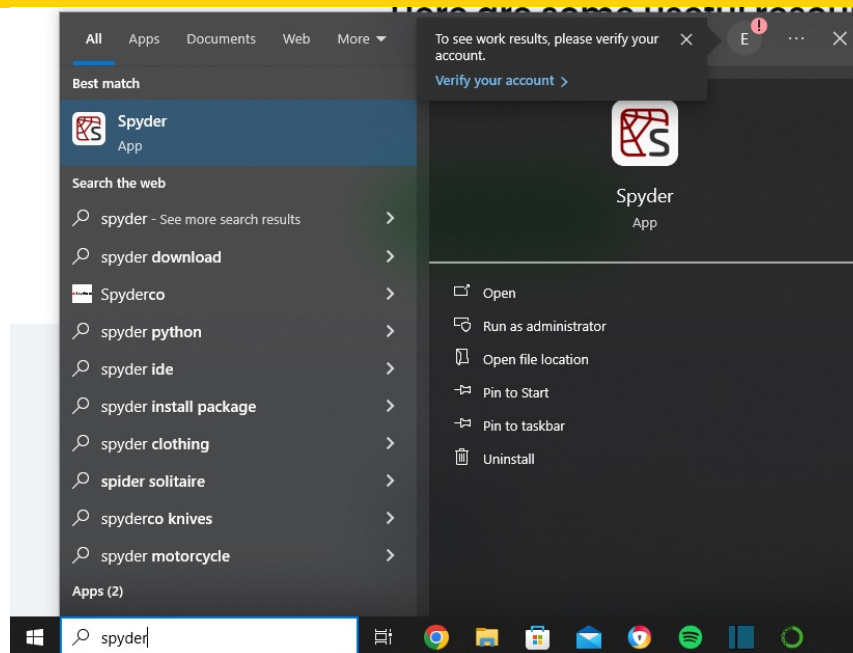




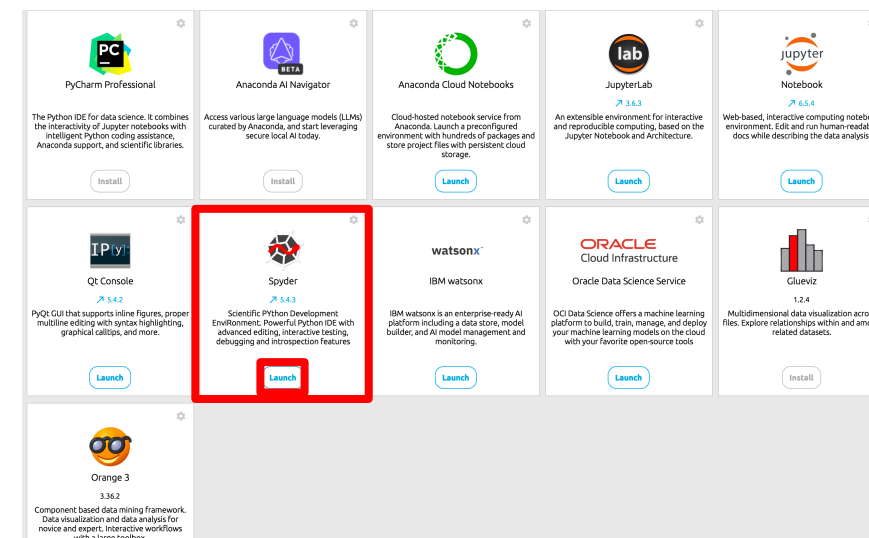
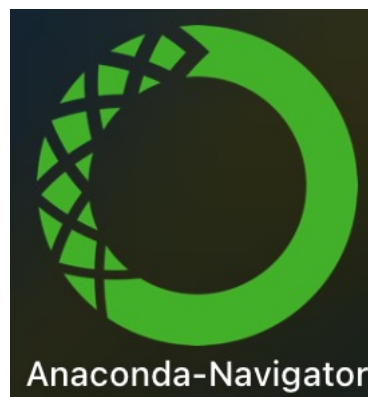
► Spyder

Busque en sus Programas a Spyder

- Si lo encuentra
 - Ábralo



- Si no lo encuentra
 - Abra Anaconda y luego
 - **Abra Spyder**





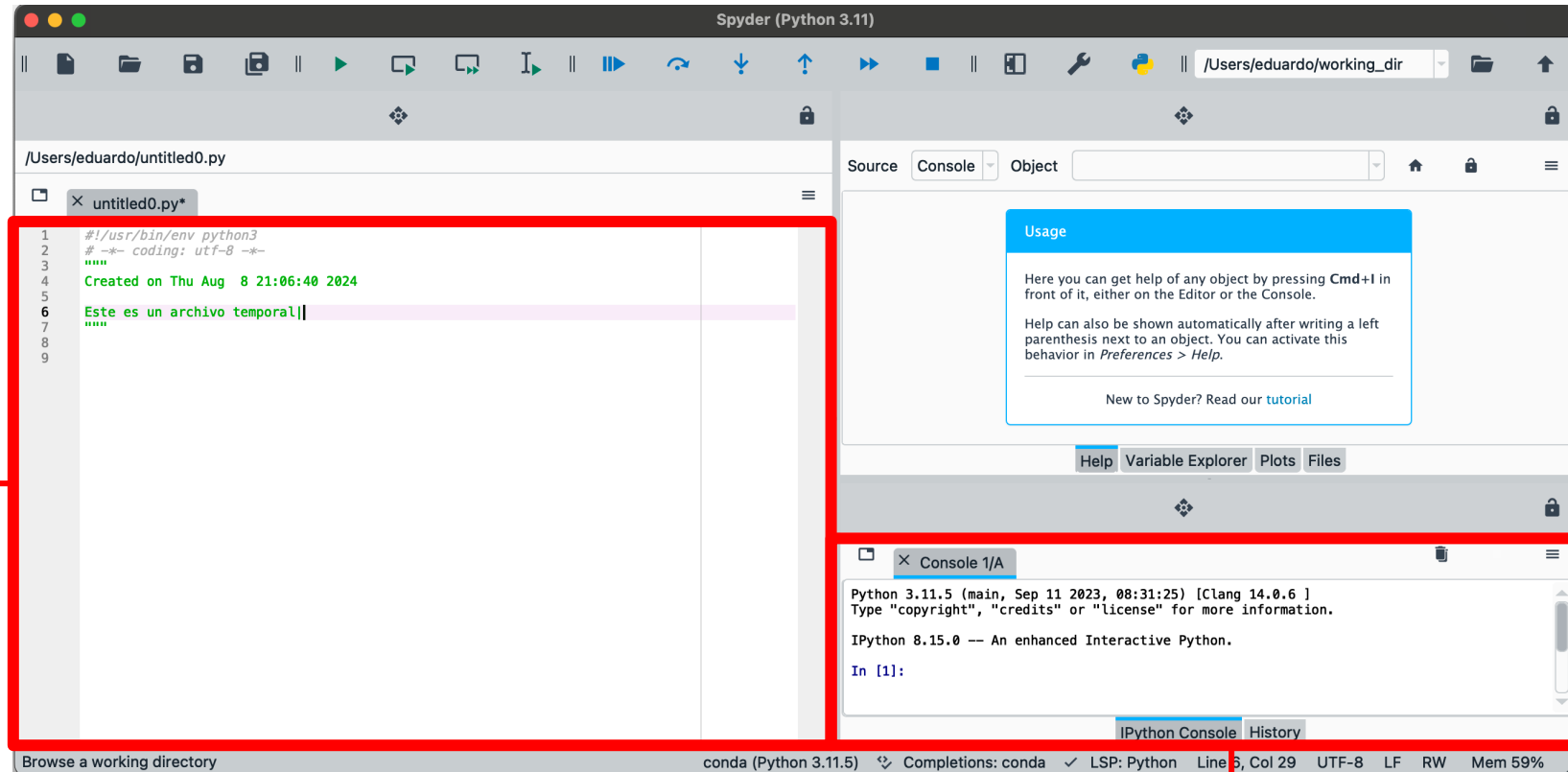
Editor:

- Aquí puede escribir programas

- Múltiples instrucciones

- Los programas se pueden ejecutar

- Los programas se pueden guardar



Interprete (terminal):

- Aquí puede usar el intérprete de Python
- No se puede guardar

-Aquí también se muestran resultados de ejecución



ACTIVIDAD 1:

Usando el Intérprete de Python



Actividad 1: Usando el Intérprete (1 de 1)

► Actividad 1:

In [35]: 2 + 2
Out[35]: 4

In [36]:

Aprenda a usar el intérprete

- Escriba los siguientes ejemplos (puede copiar y pegar lo que está en negrilla)
 - Uno a la vez
 - Presione **Enter**
 - Compruebe el resultado

| Operación | Descripción | Ejemplo | Resultado esperado |
|----------------|--|---------------|--------------------|
| Suma | Suma de dos valores | 2 + 2 | 4 |
| Resta | Resta de dos valores | 4 - 2 | 2 |
| Multiplicación | Multiplica dos valores | 3 * 3 | 9 |
| División | Cociente de la división de dos valores | 10 / 2 | 5 |



FIN DE LA ACTIVIDAD 1



ACTIVIDAD 2:

Comprendiendo Errores en Python



Actividad 2: Comprendiendo Errores en Python (1 de 4)

► Actividad 2:

Escriba en el intérprete el siguiente valor y presione **Enter**

4 000

- Note que hay un **espacio** entre el 4 y los ceros

Debería obtener un error similar al siguiente:

```
In [3]: 4 000
        Cell In[3], line 1
          4 000
            ^
SyntaxError: invalid syntax
```

```
In [4]:
```



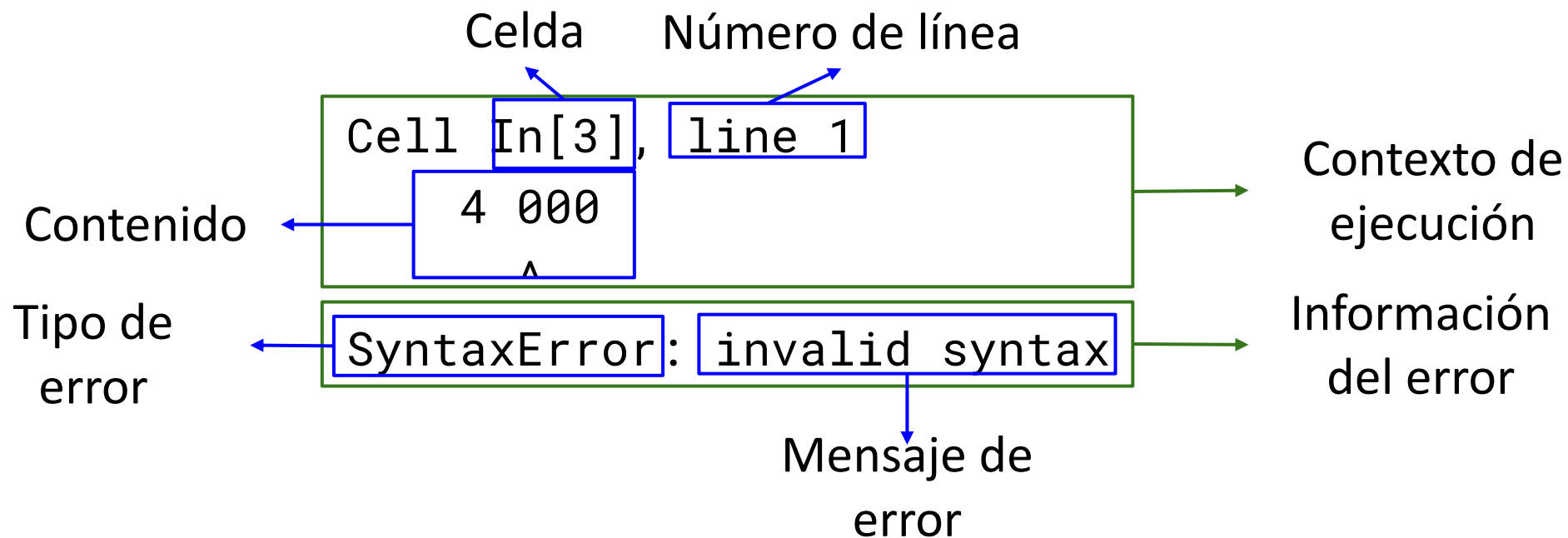
Actividad 2: Comprendiendo Errores en Python (2 de 4)

► Actividad 2:

Lea cuidadosamente la siguiente explicación

```
In [3]: 4 000
        Cell In[3], line 1
          4 000
          ^
        SyntaxError: invalid syntax
```

```
In [4]:
```





► Error

Problema relacionado con el código

Impide la ejecución exitosa del programa

► Error de sintaxis

Error resultante de usar una **sintaxis inválida en Python**

Ocurre en el momento del análisis (parse time)

- Antes de la ejecución

Hace que el intérprete de Python falle

No puede procesar la línea de código

Solución:

Usar la sintaxis correcta de Python



Actividad 2: Comprendiendo Errores en Python (3 de 4)

► Actividad 2:

- ¿Qué pasa al intentar ejecutar la siguiente instrucción?

4 000

```
Cell In[3], line 1
```

```
4 000
```

```
^
```

```
SyntaxError: invalid syntax
```

► Respuesta:

Un error de sintaxis

- Ocurrió un **SyntaxError** dentro de la celda 3, en la línea 1
- `^` señala los ceros finales problemáticos después del número 4
- Usar espacios dentro de un número es una sintaxis inválida en Python



Actividad 2: Comprendiendo Errores en Python (4 de 4)

► Actividad 2:

- ¿Cómo se soluciona el error de sintaxis producido por la siguiente instrucción?

4 000

```
Cell In[3], line 1
```

```
4 000
```

```
^
```

```
SyntaxError: invalid syntax
```

► Respuesta:

Remover los espacios entre el 4 y los ceros

4000



FIN DE LA ACTIVIDAD 2



ACTIVIDAD 3:

Usando Funciones de Python



Actividad 3: Usando Funciones en Python (1 de 1)

► Actividad 3:

En el intérprete

- Escriba los siguientes ejemplos (puede copiar y pegar lo que está en negrilla)
 - Uno a la vez
 - Presione **Enter**
 - Compruebe el resultado

| Ejemplo | Resultado esperado |
|------------------------------|---------------------|
| <code>type('0.0')</code> | str |
| <code>type(0.0)</code> | float |
| <code>type("")</code> | str |
| <code>type("""abc""")</code> | str |
| <code>type(.1)</code> | float |
| <code>type(3.14e-10)</code> | float |
| <code>type(4_000)</code> | int |
| <code>type(4 000)</code> | ... SyntaxError ... |



FIN DE LA ACTIVIDAD 3



ACTIVIDAD 4:

Usando la Función `print(...)`



► print(...)

Muestra información en la consola

Sintaxis: `print(*objects, sep=' ', end='\n', ...)`

- **objects:** objetos a imprimir (puede ser varios, separados por coma)
- **sep:** separador entre objetos (por defecto, espacio)
- **end:** carácter al final de la línea (por defecto, nueva línea)

Usa `help(print)` para más información

Útil para mostrar resultados, mensajes de error y depurar

Ejemplo:

```
print('Hola, mundo!') # → Hola, mundo!
```

```
print('Nombre:', 'Isaac Newton') # → Nombre: Isaac Newton
```



Actividad 4: Usando la Función print(...) (1 de 2)

► Actividad 4:

En el intérprete

- Escriba los siguientes ejemplos (puede copiar y pegar lo que está en negrilla)
 - Uno a la vez
 - Presione **Enter**
 - Compruebe el resultado

Analice el por qué de los resultados obtenidos

| Ejemplo | Resultado esperado |
|---|---|
| <code>print("")</code> | |
| <code>print('Hola Mundo')</code> | Hola Mundo |
| <code>print("Ella dijo: 'hola'.")</code> | Ella dijo: 'hola'. |
| <code>print('Ella dijo: "bienvenidos".')</code> | Ella dijo: "bienvenidos". |
| <code>print('El alumno dijo: \'¡He terminado mi tarea!\'')</code> | El alumno dijo: '¡He terminado mi tarea!' |
| <code>print("Ella dijo: \"Hola, mundo\" en su código.")</code> | Ella dijo: "Hola, mundo" en su código. |



Actividad 4: Usando la Función print(...) (2 de 2)

► Actividad 4:

Cree otros ejemplos usando la función `print(. . .)` para imprimir:

- Su primer nombre
- Su primer y segundo nombre (si alguno)
- Sus nombres y su primer apellido
- Su nombre completo (todos los nombres y apellidos)

- Su nombre completo y luego la frase: *!Pasaré IP con más de 4!*
- Su nombre completo y luego la frase: dijo: *“¡Python es genial!”*
- Su nombre completo y luego la frase: dijo en voz alta: *‘¡soy un programador!’*

Nota1: Esta doble comilla: “ no es la misma que esta: ”

Nota2: Esta comilla simple: ‘ no es la misma que esta: ’

Cree otros ejemplos usando la función `print(. . .)`

- Que impriman mensajes personalizados



FIN DE LA ACTIVIDAD 4



ACTIVIDAD 5:

Escribiendo un Programa Usando Spyder

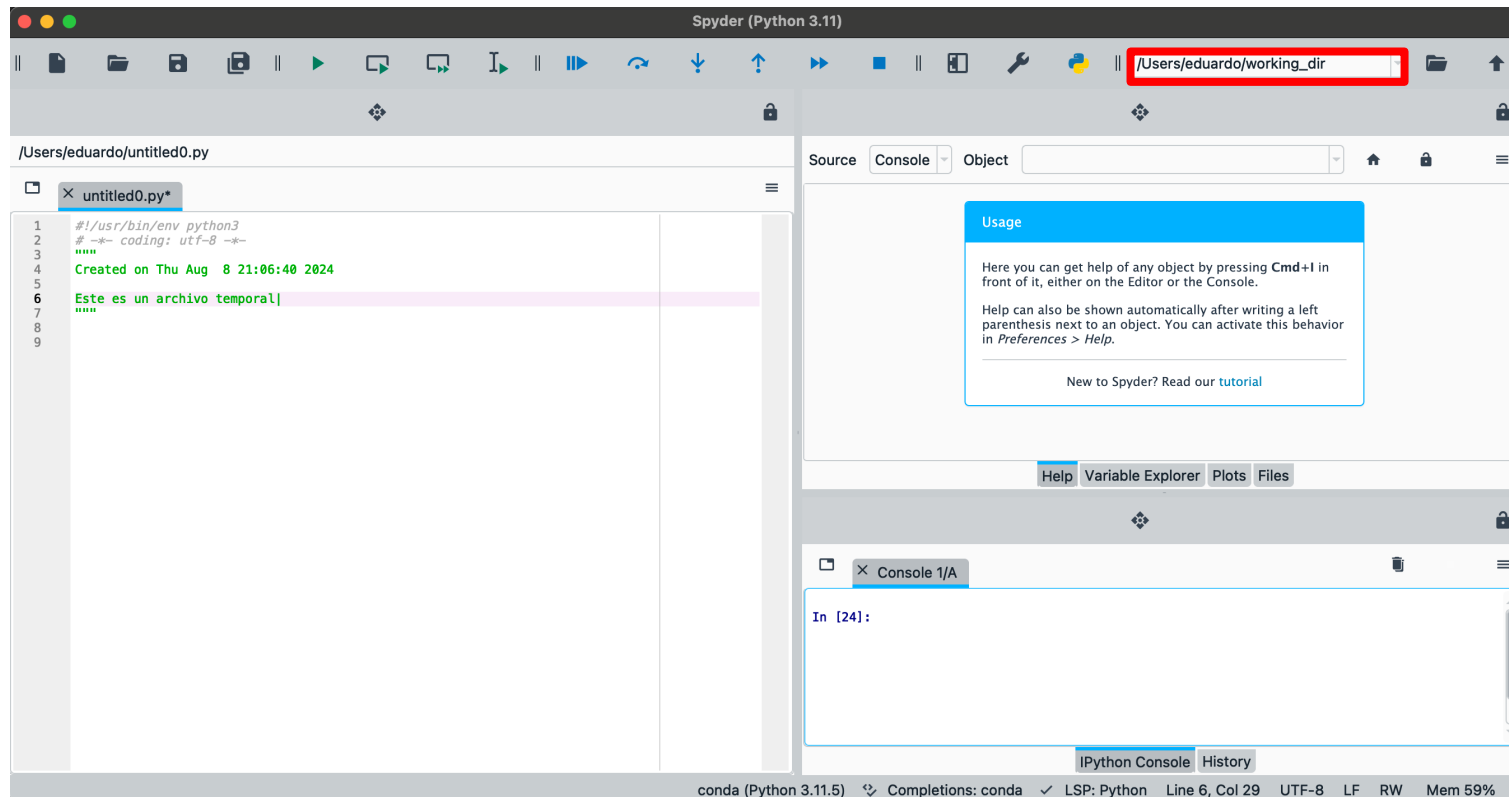


Actividad 5: Escribiendo un Programa Usando Spyder (1 de 11)

► Actividad 5:

Defina su **directorio de trabajo** (*working directory*)

- Para ello cree una carpeta cuyo nombre **no tenga espacios**
- Copie la ruta a esa carpeta en la parte superior de Spyder (ver cuadro rojo)
 - Presione **Enter**



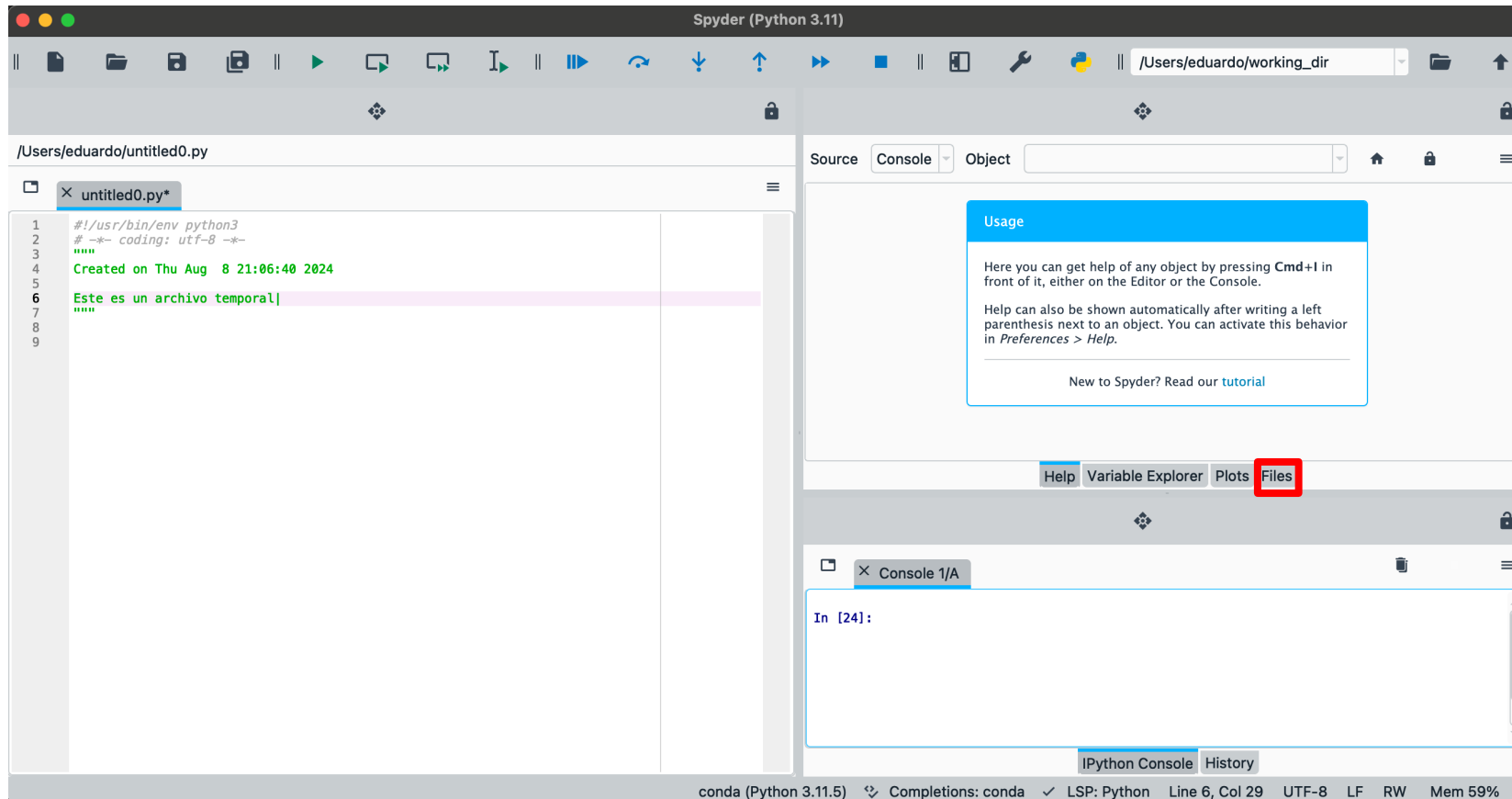


Actividad 5: Escribiendo un Programa Usando Spyder (2 de 11)

► Actividad 5:

Cambie la vista al Explorador de archivos (*Files*)

- Para ello, haga clic en **Explorador de archivos (*Files*)** (ver cuadro rojo)



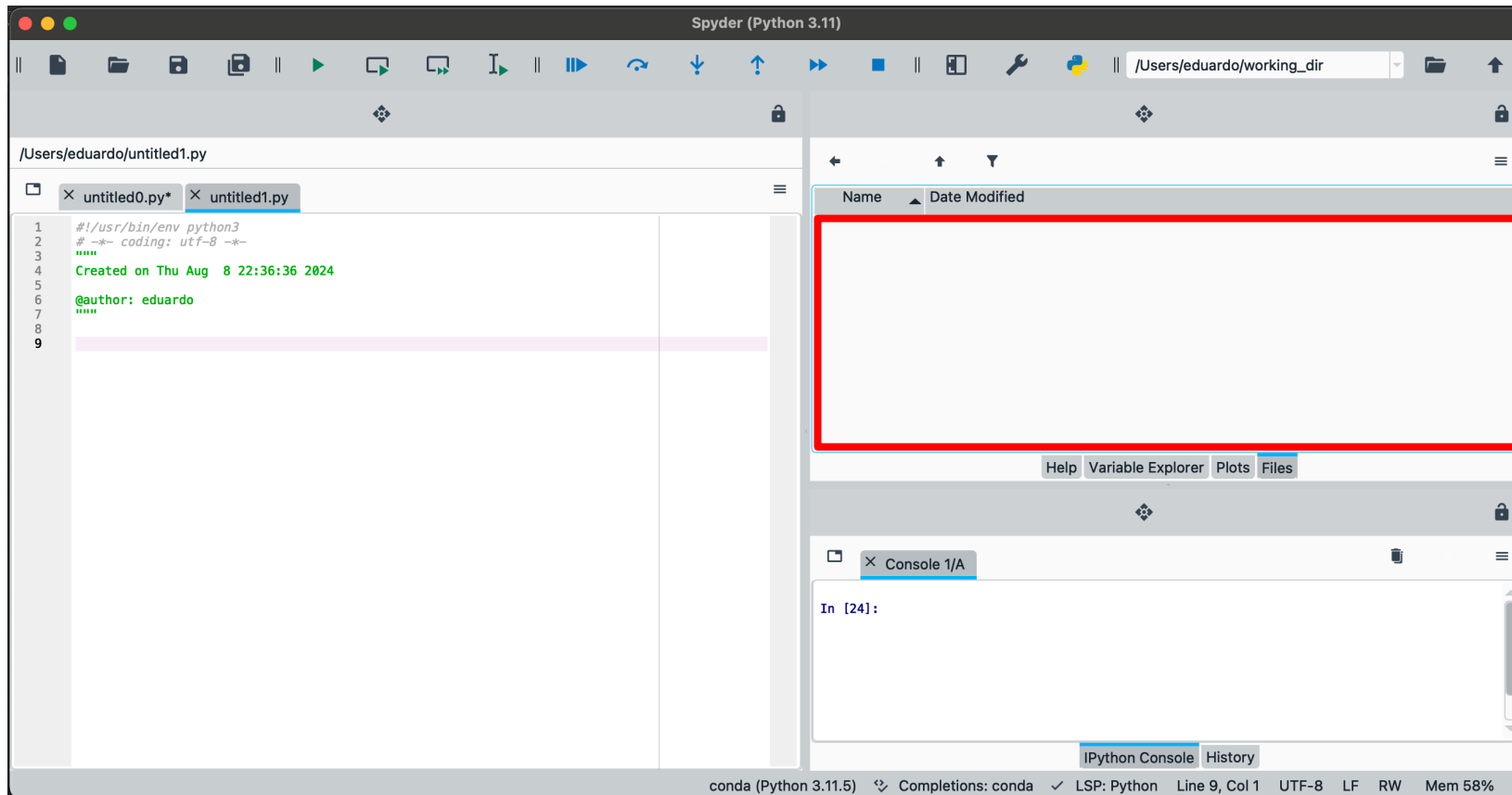


Actividad 5: Escribiendo un Programa Usando Spyder (3 de 11)

► Actividad 5:

Clic derecho en el área blanca (ver cuadro rojo)

- Si hacer clic derecho le funcionó, vaya a la **siguiente diapositiva**
- Si hacer clic derecho no funciona, vaya a la **diapositiva 31**

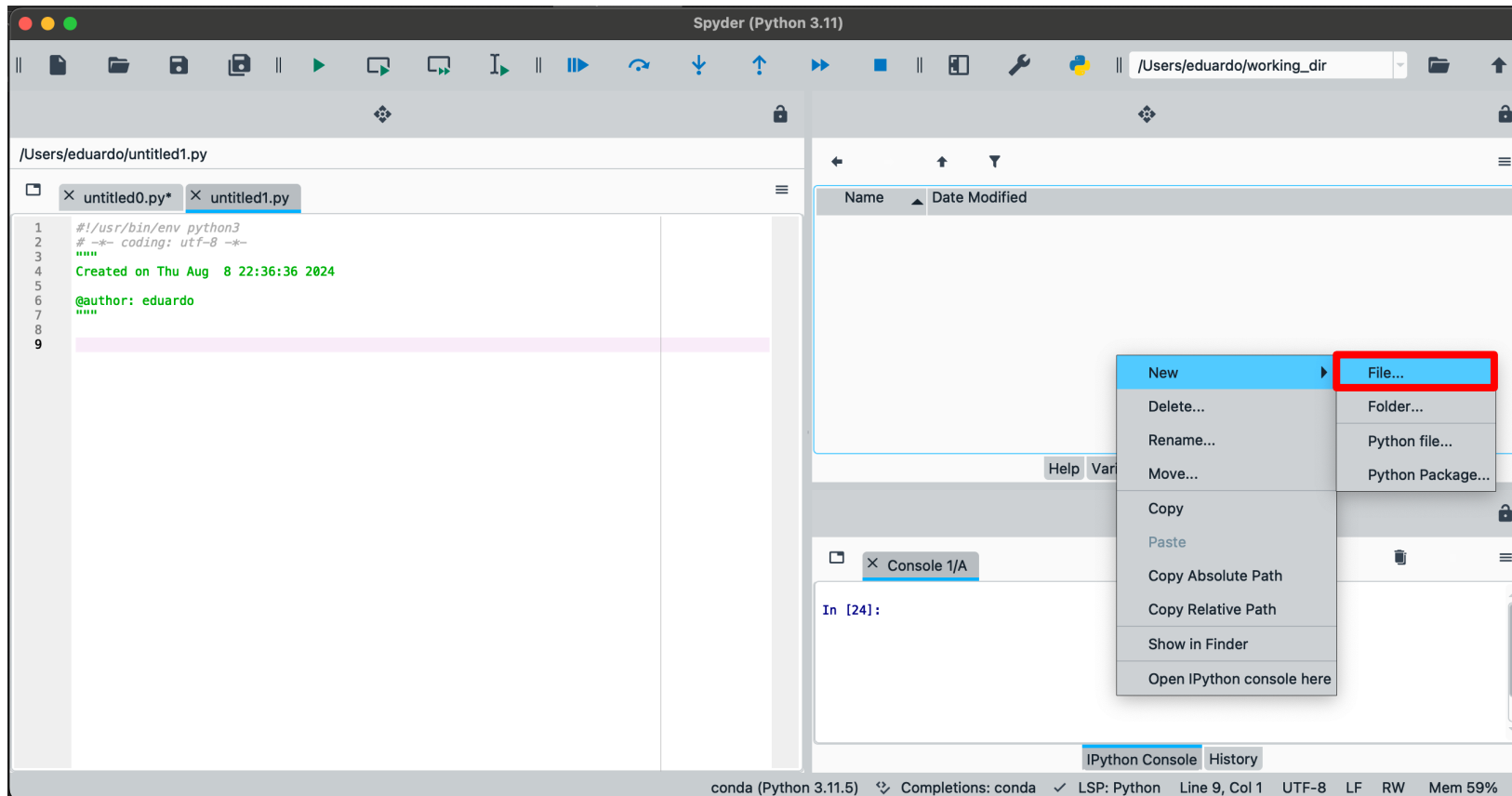




Actividad 5: Escribiendo un Programa Usando Spyder (4a de 11)

► Actividad 5:

Seleccione: Nuevo – Archivo (*New – File*) (ver cuadro rojo)





Actividad 5: Escribiendo un Programa Usando Spyder (4b de 11)

► Actividad 5:

Si hacer clic derecho no funcionó entonces

- Vaya a:

► Archivo (File)

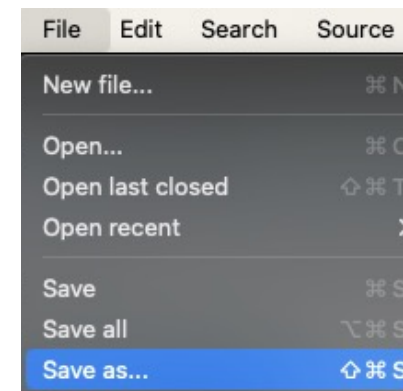
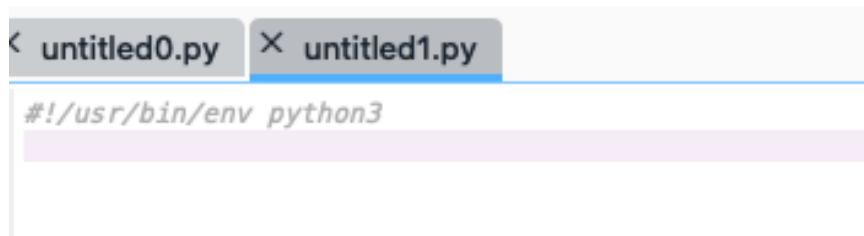
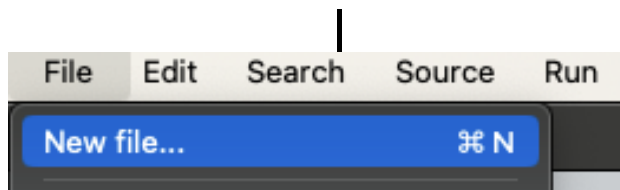
Nuevo Archivo (New File)

- *Esto creará un nuevo archivo llamado untitled.py o similares*

► Archivo (File)

Guardar como (Save as)

- *Esto abrirá una ventana en donde debe especificar en dónde guardar el archivo*
- *Debe ser el directorio de trabajo que definió antes*





Actividad 5: Escribiendo un Programa Usando Spyder (5 de 11)

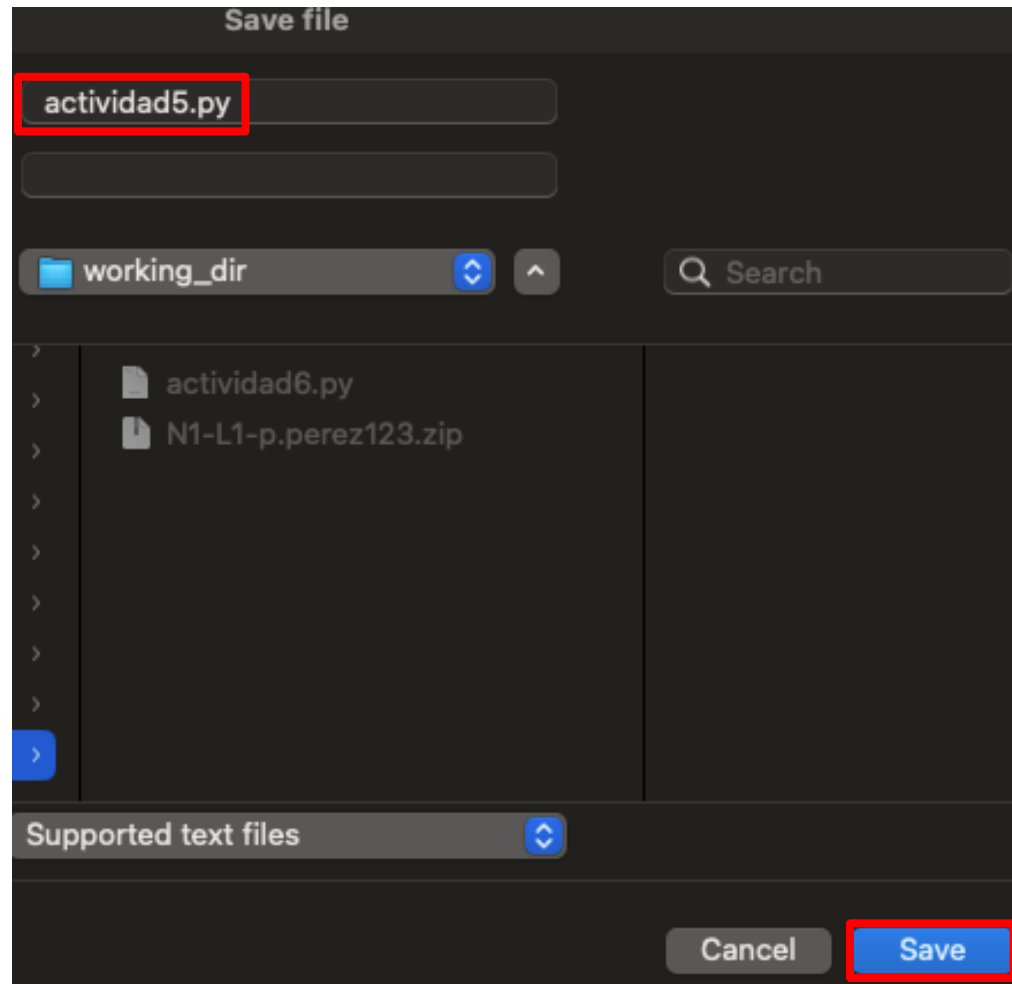
► Actividad 5:

Digite el nombre:

- **actividad5.py**

Elija la opción

- Guardar (*Save*)

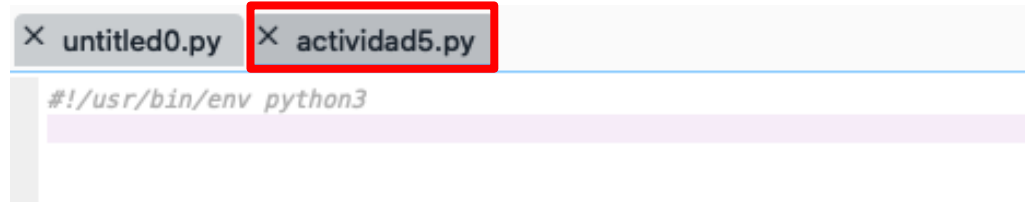




Actividad 5: Escribiendo un Programa Usando Spyder (6 de 11)

► Actividad 5:

Verifique que el nuevo archivo esté abierto





Actividad 5: Escribiendo un Programa Usando Spyder (7 de 11)

► Actividad 5:

Copie y pegue en el archivo, el siguiente programa:

```
def suma(num1: int, num2: int) -> int:
```

```
    """
```

```
    Suma dos números enteros.
```

```
    Parámetros:
```

```
        num1 (int): El primer número entero.
```

```
        num2 (int): El segundo número entero.
```

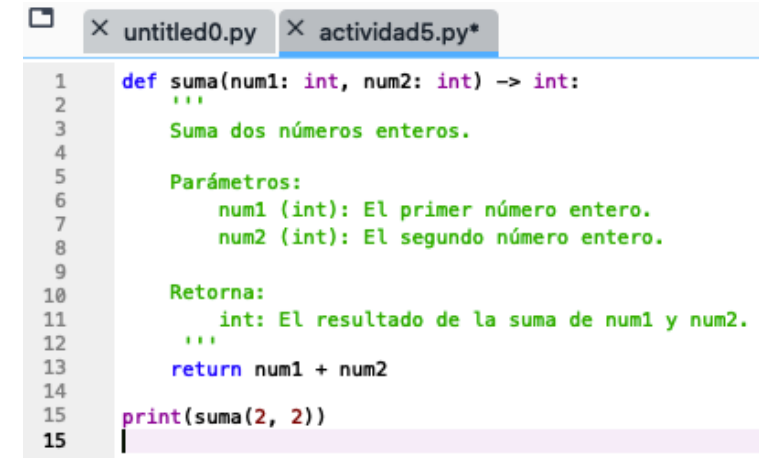
```
    Retorna:
```

```
        int: El resultado de la suma de num1 y num2.
```

```
    """
```

```
    return num1 + num2
```

```
print(suma(2, 2))
```



```
def suma(num1: int, num2: int) -> int:
    """
    Suma dos números enteros.

    Parámetros:
        num1 (int): El primer número entero.
        num2 (int): El segundo número entero.

    Retorna:
        int: El resultado de la suma de num1 y num2.
    """
    return num1 + num2

print(suma(2, 2))
```

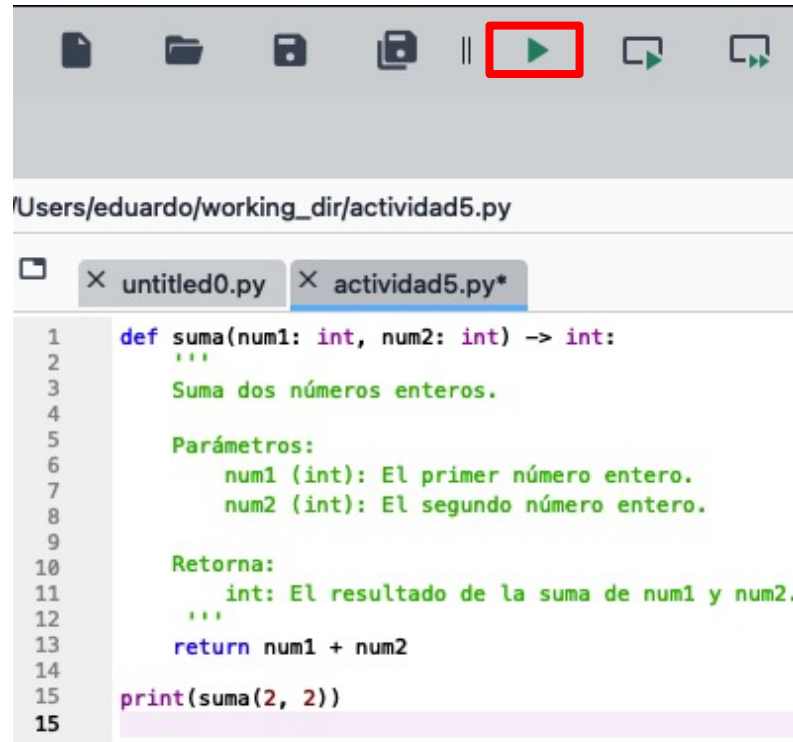


Actividad 5: Escribiendo un Programa Usando Spyder (8 de 11)

► Actividad 5:

Ejecute el programa

- Para ello, clic en el ícono triangular verde (ver cuadro rojo)



```
1 def suma(num1: int, num2: int) -> int:
2     """
3     Suma dos números enteros.
4
5     Parámetros:
6         num1 (int): El primer número entero.
7         num2 (int): El segundo número entero.
8
9     Retorna:
10        int: El resultado de la suma de num1 y num2.
11    """
12
13    return num1 + num2
14
15 print(suma(2, 2))
15
```



Actividad 5: Escribiendo un Programa Usando Spyder (9 de 11)

► Actividad 5:

Verifique el resultado

- Para ello, observe la terminal (ver cuadro rojo)
 - Es 4 en este caso

The screenshot shows the Spyder IDE interface. On the left, the code editor displays a Python function `suma` and its execution. The function `suma` takes two integers as input and returns their sum. The code is as follows:

```
def suma(num1: int, num2: int) -> int:
    """
    Suma dos números enteros.

    Parámetros:
        num1 (int): El primer número entero.
        num2 (int): El segundo número entero.

    Retorna:
        int: El resultado de la suma de num1 y num2.
    """
    return num1 + num2

print(suma(2, 2))
```

On the right, the IPython console shows the execution of the script. The output is `4`, which is highlighted with a red box. The console also shows the IPython version (8.15.0) and the command used to run the script.

| Nam | Type | Size | Value |
|-----|------|------|-------|
|-----|------|------|-------|

Help Variable Explorer Plots Files

Console 1/A

IPython 8.15.0 -- An enhanced Interactive Python.

In [1]: runfile('/Users/eduardo/working_dir/actividad5.py', wdir='/Users/eduardo/working_dir')

4

In [2]:

IPython Console History



Actividad 5: Escribiendo un Programa Usando Spyder (10 de 11)

► Actividad 5:

Lea el programa e intente entender **qué** hace y **cómo** lo hace

- **Tiempo límite: 10:30am**

Modifique el programa para que ahora el resultado sea cada uno de estos valores (ejecuciones separadas):

- 10
- 0
- -10

- **Tiempo límite: 10:40am**



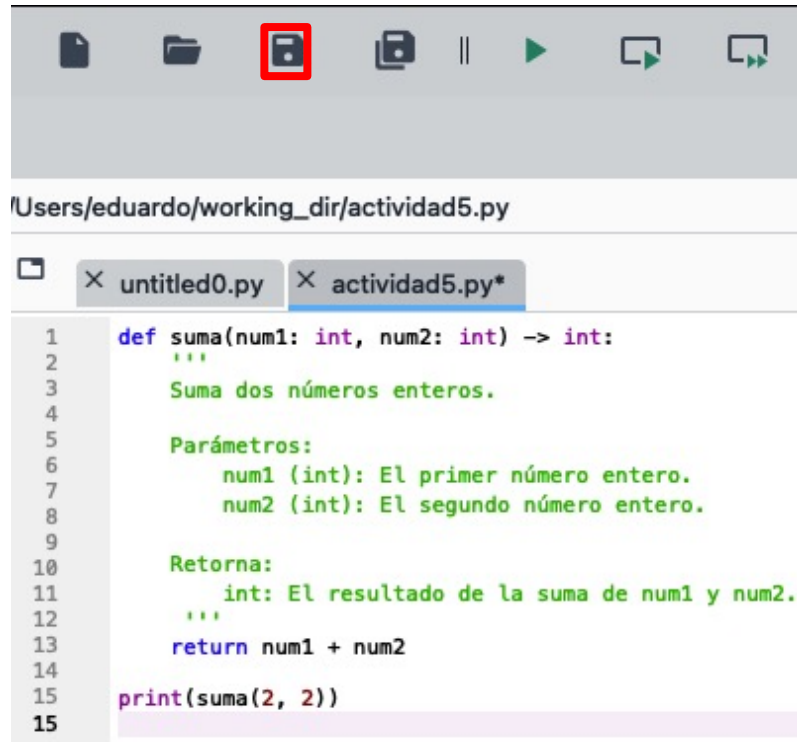
Actividad 5: Escribiendo un Programa Usando Spyder (11 de 11)

► Actividad 5:

Guarde su programa

- Clic en el ícono de guardar (ver cuadro rojo)

Nota: No importa cuál versión guarde (Ej. La original, la que reporta 10 ó 0 ó -10, cualquiera está bien)



```
1 def suma(num1: int, num2: int) -> int:
2     """
3     Suma dos números enteros.
4
5     Parámetros:
6         num1 (int): El primer número entero.
7         num2 (int): El segundo número entero.
8
9     Retorna:
10        int: El resultado de la suma de num1 y num2.
11    """
12
13    return num1 + num2
14
15 print(suma(2, 2))
```



FIN DE LA ACTIVIDAD 5



ACTIVIDAD 6:
Enviar su Laboratorio a BN
Comprobar envío exitoso

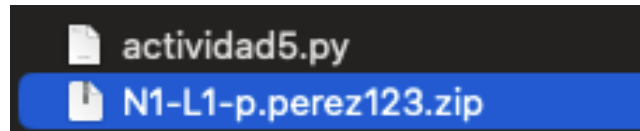
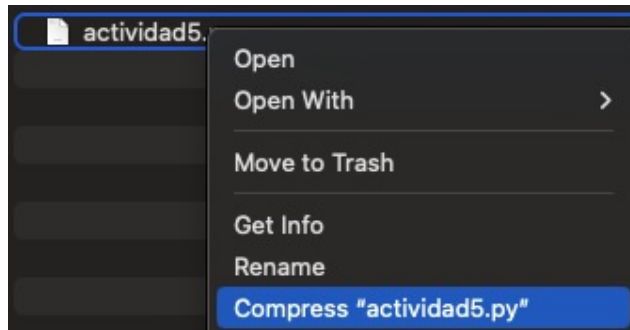


Actividad 6: Enviar su Laboratorio a BN (1 de 4)

► Actividad 6:

Comprima en un **.zip** el archivo:

- actividad5.py
- El archivo debería llamarse **N1-L1-*login*.zip**,
 - donde *login* es su nombre de usuario de Uniandes (Por ejemplo: N1-L1-p.perez123.zip).





Actividad 6: Enviar su Laboratorio a BN (2 de 4)

► Actividad 6:

Envío

- Inicie sesión en [Bloque Neón](#) (BN)
 - *Actividades*
 - *L1: Ambientes básicos de trabajo*
 - *Agregar un archivo*

Enviar actividad

Archivos para enviar *

(0) archivo(s) para enviar

Después de la carga, debe hacer clic en Enviar para completar el envío.

Agregar un archivo

Grabar audio

Grabar video

Comentarios

Párrafo **B** *I* U ~~A~~

Enviar

Cancelar

Enviar un archivo - INTRODUCCIÓN A LA PROGRAMACIÓN -
Universidad de los Andes

Mi PC

Mi casillero

Casillero grupal

Enviar un archivo - INTRODUCCIÓN A LA PROGRAMACIÓN -
Universidad de los Andes

¡Soltar los archivos aquí o hacer clic en la opción de abajo!

Cargar

Puede cargar archivos hasta un máximo de 2 GB.

Enviar un archivo - INTRODUCCIÓN A LA PROGRAMACIÓN -
Universidad de los Andes

¡Soltar los archivos aquí o hacer clic en la opción de abajo!

Cargar

Puede cargar archivos hasta un máximo de 2 GB.

N1-L1-p.perez123.zip (387 bytes) X

Agregar

Anterior

Cancelar



Actividad 6: Comprobar Envío Exitoso (3 de 4)

► **Actividad 6:**

En BN, vaya a *Actividades*

- Debe aparecer el envío

Actividades

 Ayuda

Ver historial

| Actividad | Estado de avance | Puntuación | Estado de evaluación |
|--|-------------------------------|------------|----------------------|
| Ninguna categoría | | | |
| L1: Ambientes básicos de trabajo | 1 material enviado, 1 archivo | - / - | |



Actividad 6: Comprobar Envío Exitoso (4 de 4)

► Actividad 6:

En BN, vaya a *Actividades*

- Descargue su envío
 - **Verifique que el contenido en el .zip enviado es aquel que usted deseaba enviar**

[Actividades](#) > [Ver historial](#)

Historial de envíos



Actividad

L1: Ambientes básicos de trabajo ▾

Aplicar

Tipo de actividad

Actividad individual

| ID de envío | Envíos | Fecha de envío ▾ |
|-------------|--|---------------------------|
| 2803951 |   N1-L1-p.perez123.zip (3.06 MB) | 9 de agosto de 2024 06:47 |



FIN DEL LABORATORIO 1