

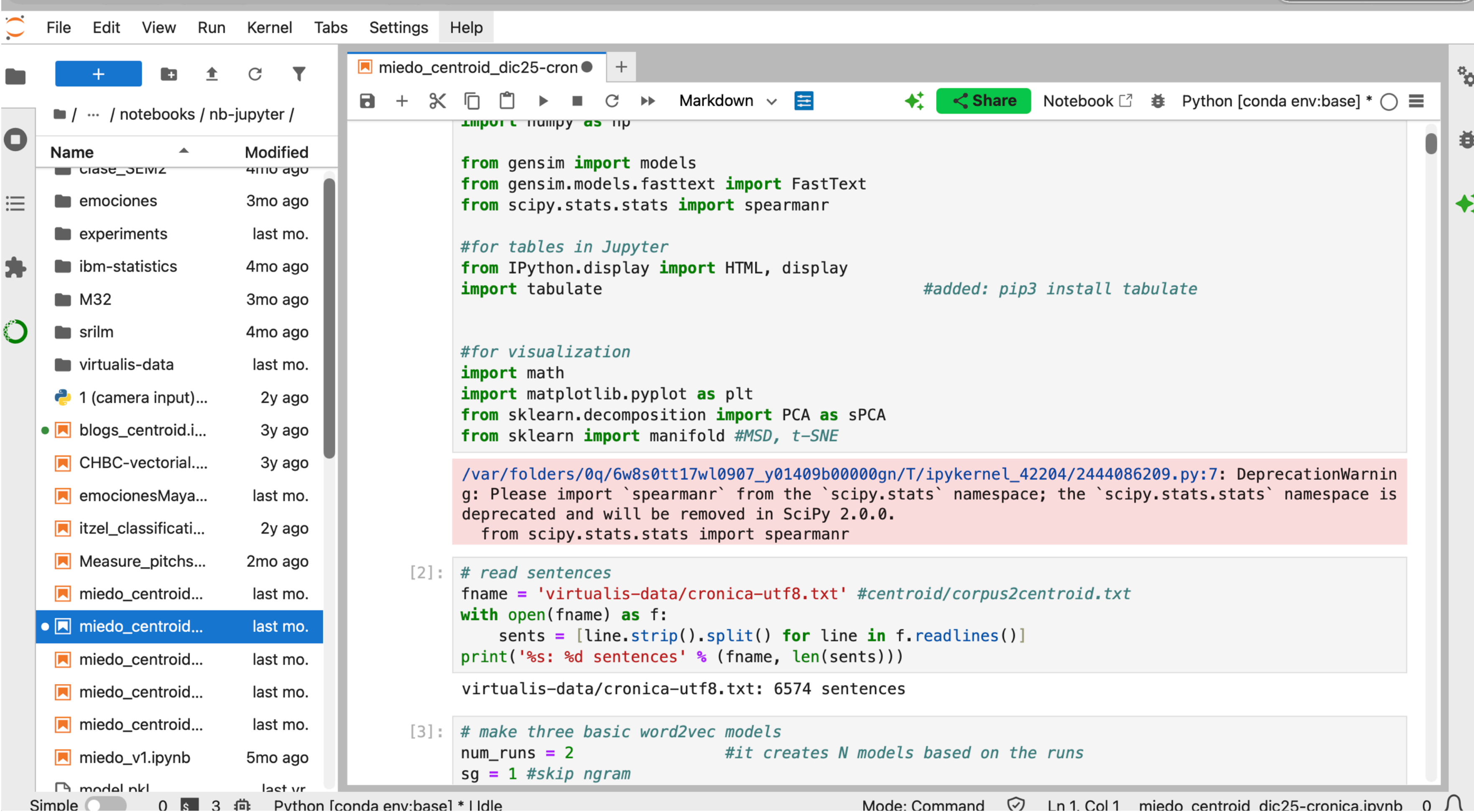
Introducción a Python: Hola mundo 🐍

Reflexiones iniciales

- ¿Qué significa programar?
- ¿Cuáles son las características de un lenguaje?
- ¿Cuáles son las diferencias entre lenguaje natural y artificial?

Instalación de Python

- Requisitos
- Download
- Versiones: un gran problema para los desarroladores



¿Qué versión elegir?

- Versiones estables
- Versión sugerida por el desarrollador
- Entorno ENV

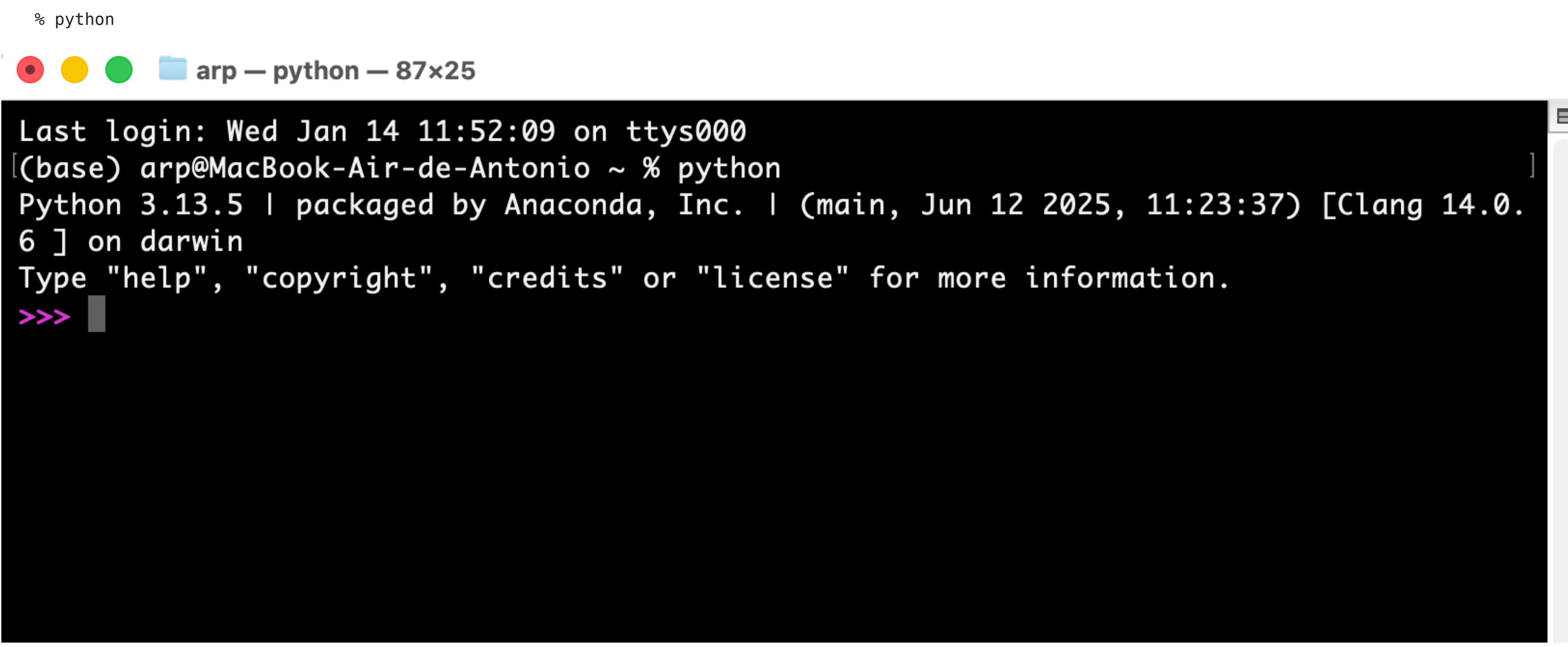
Primeros pasos

Distinción entre source y editor

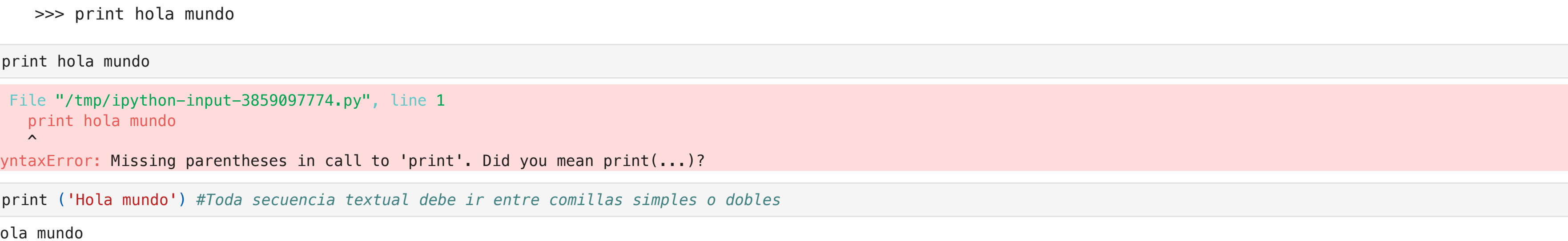
- Source: código de cualquier lenguaje – estructura (gramática)
- Editor (IDLE: Integrated DeveLopment Environment): interfaz de comunicación (teoría/escuela)
- Python tiene su propio IDLE: Shell
- Se puede crear código en cualquier editor de texto, pero es mejor usar uno especializado para sacar provecho a recursos, herramientas e, incluso, motores de IA
- Hay varios editores para Python: Terminal, PyCharm, Spyder, PyDev.

Abrir la terminal

- Windows: menú inicio > CMD o PowerShell
- Mac: Spotlight > Terminal
- Ejecutar Python

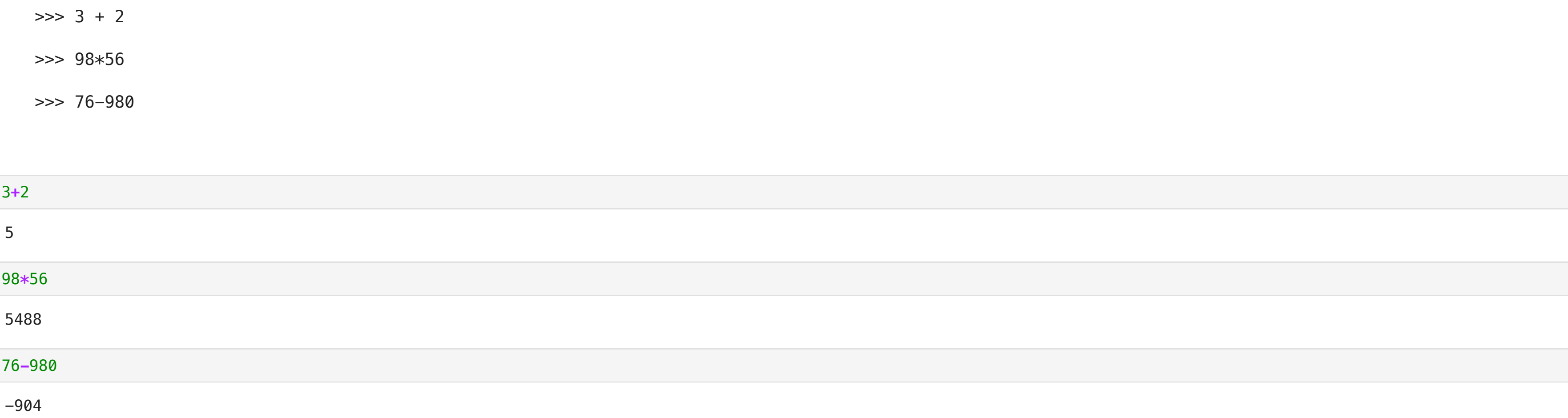


Empezando a leer los errores



Procesos básicos

- Operaciones matemáticas



- Orden y secuencias
- ¿Por qué se da ese resultado?
- Estandar de orden: **PEMDAS** (Parentheses, Exponentiation, Multiplication, Division, Addition, Subtraction)

Ejercicio:

- Escribe el código para sumar $3 + 2 - 1 * 3$ y obtener **12** como resultado

```
>>>
```

Ejercicio:

- Explica por qué el resultado del siguiente código es **2**

```
>>> (3+2) - 1 * 3
```

Variables

Una analogía lingüística

Las variables se pueden entender como signos que representan algo de la realidad: un dato, un número, una operación, un resultado, un corpus, etc.

- /h/ = sonido fricativo, labiodental, sordo
- it = sufijo apreciativo
- silla = objeto sólido con cuatro patas y una base que sirve para sentarse

Toda variable debe estar referenciada.

- Prefiero tomarla en el verano, luego la coloco ahí y de inmediato le sucede lo que ya te había comentado.
- la = cerveza

No debe existir ambigüedad, de lo contrario, el resultado puede ser sobre un dato que no es el que necesitamos.

- la = botella

Hay que tener mucho cuidado con el tratamiento de las variables ya que éstas se pueden sobrescribir y, por lo tanto, representar(guardar) datos que han cambiado.

- la no siempre es anáfora de cerveza.

Definición de variables en Python

Toda variable debe definirse respetando la siguiente sintaxis:

- VARIABLE = algo

```
>>> x = 3+2
```

Una vez que se asigna un valor a una variable, éste se puede usar para realizar operaciones sin tener que volver a poner el *algo*.

```
>>> x * 2 - 1
```

El proceso se puede volver recursivo y la variable guardará nuevos datos, sobrescribiendo los previos:

```
>>> x = x * 2 - 1
```

Ojo con este tipo de procesos por lo que se mencionó anteriormente sobre la reescritura de datos.

- El código anterior modificó la variable **x**, ¿cómo puedo ver el valor actualizado?

```
>>> print VARIABLE
```

Ejercicio:

- Define dos variables y asigna a cada una el resultado de una operación matemática en la que incluyas secuencias parentéticas. Posteriormente, obten un siguiente resultado que represente la división de la variable 1 entre la variable 2. El resultado lo guardas en una variable 3. Finalmente, imprime el dato que tiene la variable 3.

Las variables también pueden almacenar datos textuales. Para ello, Python exige que este tipo de datos cumpla con la sintaxis que se estableció anteriormente: uso de **comillas simples o dobles**.

Ejercicio:

- Toma el código anterior y genera una variable 4 que almacene el siguiente texto: El resultado de la operación es:
- Imprime las variables necesarias para que en pantalla aparezca lo siguiente: El resultado de la operación es: VALOR

Nota: Python permite hacer operaciones sobre texto. Prueba sumar o multiplicar una cadena de texto.

