


Taller de programación en Python. Parte 1

JULIO CÉSAR NAVARRO
PSICÓLOGO
DESARROLLADOR DE SOFTWARE

COMPUTACIÓN COGNITIVA



Guía del Taller



INTRODUCCIÓN A LA PROGRAMACIÓN CON PYTHON

Fundamentos de programación, entorno y lenguaje.

COMENTARIOS Y VALORES DE ENTRADA Y SALIDA

Comentarios, input y print.

VARIABLES Y OPERACIONES

Variables, tipos de datos y operaciones aritméticas.

CONDICIONALES Y BUCLES

if y while

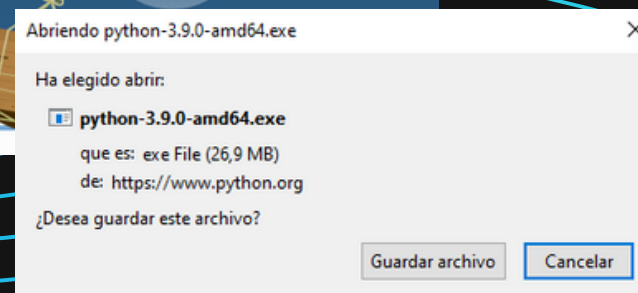
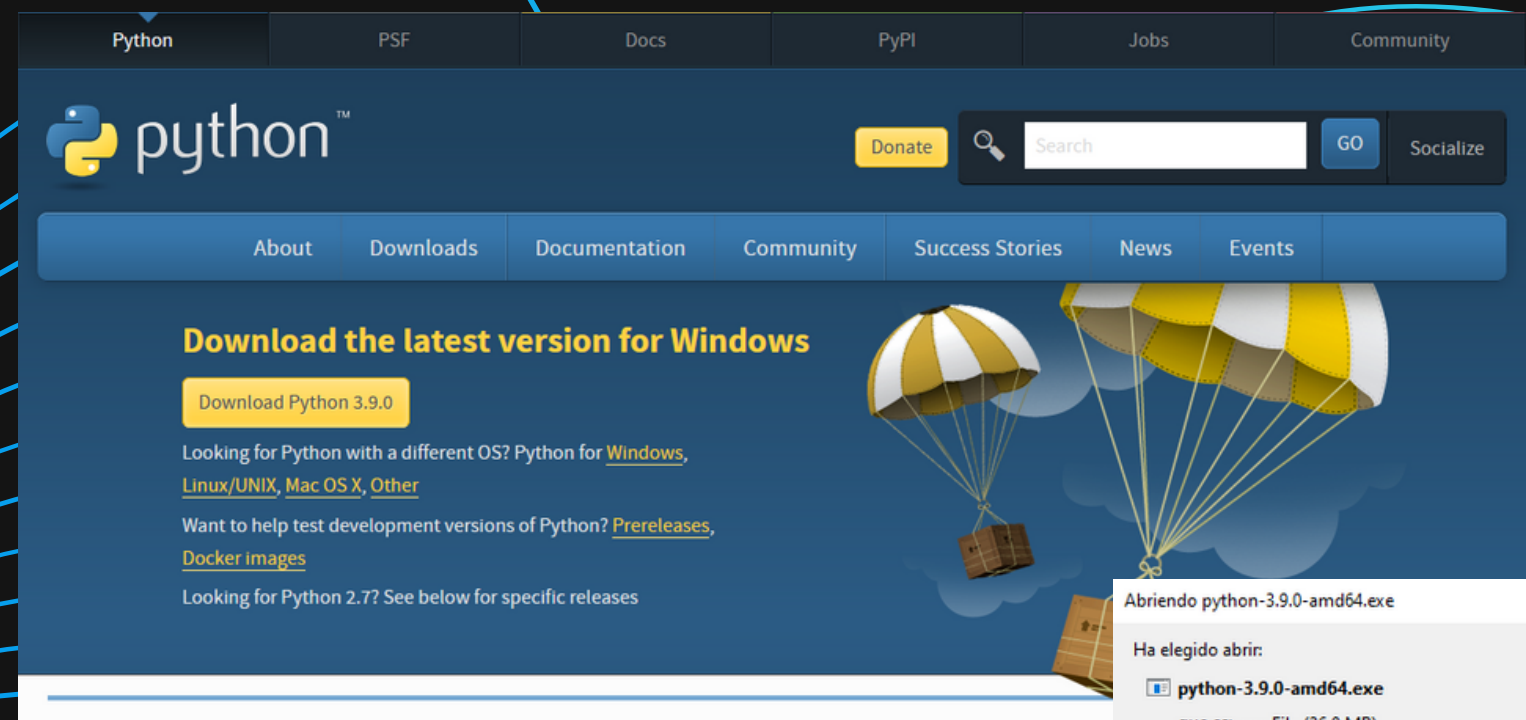
FUNCIONES

Declaración y llamado de funciones

Descarga Python

WWW.PYTHON.ORG/DOWNLOADS

Elegimos la descarga para nuestro sistema operativo.



Guardamos y ejecutamos el instalador



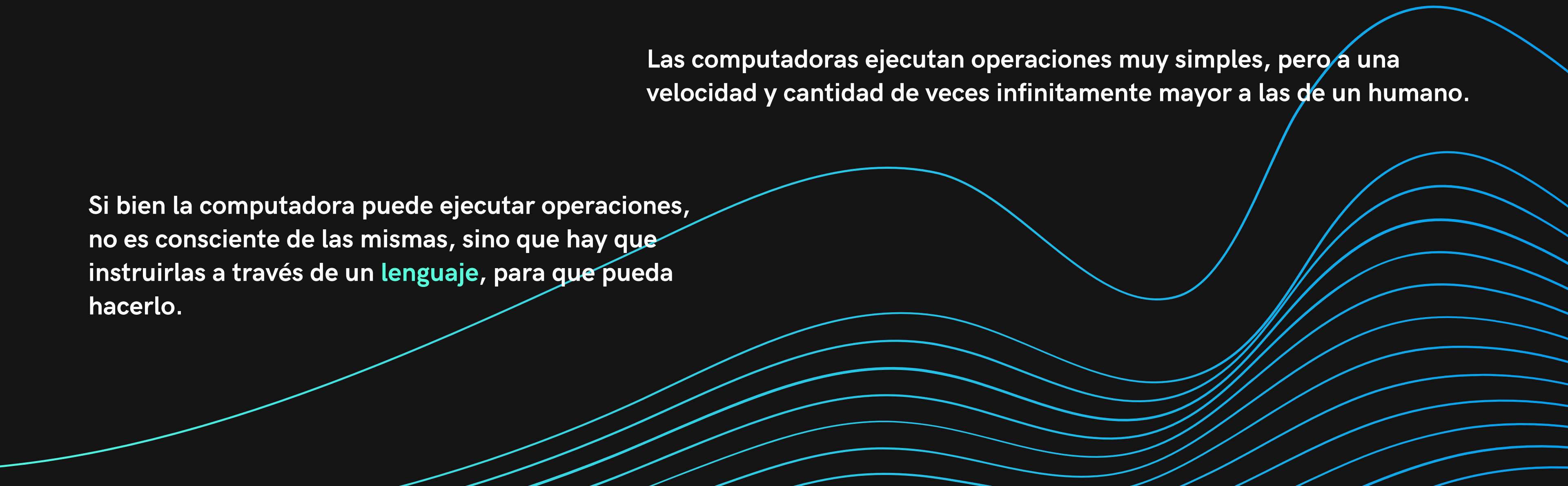
Ejecutamos el asistente

¿Cómo funciona un programa de computadora?

Un programa es un conjunto de instrucciones que nos permite utilizar una computadora.

Las computadoras ejecutan operaciones muy simples, pero a una velocidad y cantidad de veces infinitamente mayor a las de un humano.

Si bien la computadora puede ejecutar operaciones, no es consciente de las mismas, sino que hay que instruirlas a través de un **lenguaje**, para que pueda hacerlo.

The bottom half of the slide features several horizontal, wavy lines in a light blue color, creating a sense of motion or a stylized background element.

Lenguaje natural vs. Lenguajes maquina

Un lenguaje es un medio para expresar y registrar pensamientos, las computadoras también tienen un lenguaje con el cual nos comunicamos con ellas, el **lenguaje máquina**, el lenguaje que utilizamos los huamnos se llama **lenguaje natural**.

Culaquier lenguaje, natural o maquina está compuesto por.

Alfabeto: Conjunto de símbolos para formar palabras en determinado idioma.

Léxico: Conjunto de palabras que el idioma ofrece a sus usuarios.

Sintaxis: Conjunto de reglas utilizadas para determinar si una frase es válida.

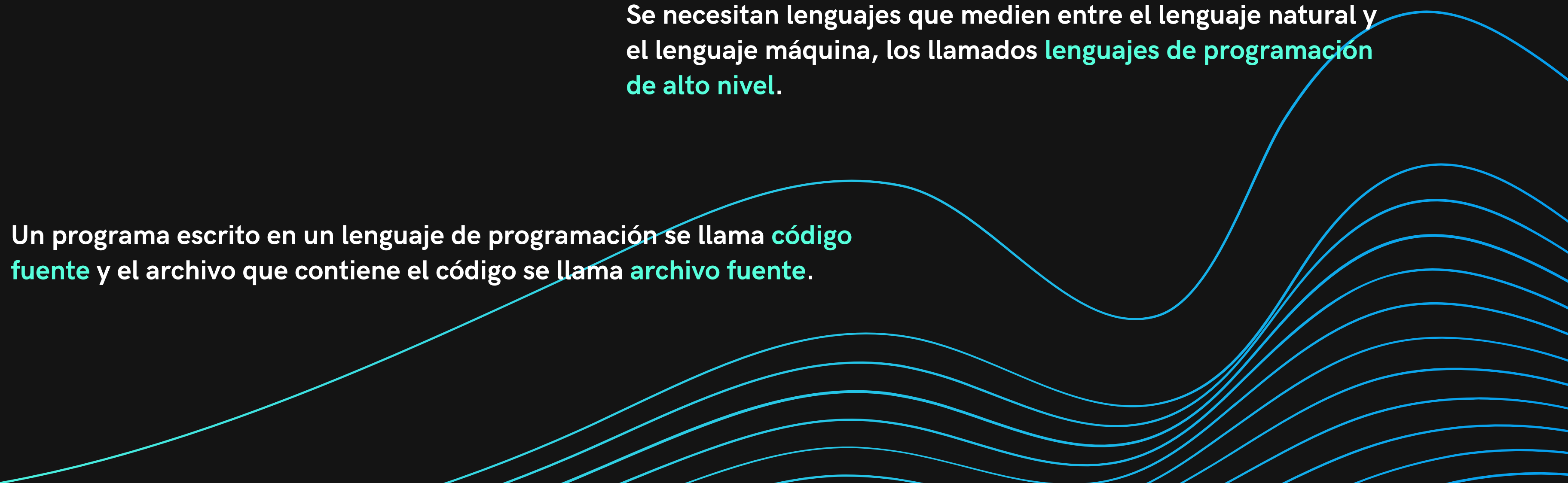
Semántica: Conjunto de reglas utilizadas para determinar el sentido de una frase

Lenguajes de programación

La **IL (Instruction List)** es el alfabeto del lenguaje máquina, pero este está muy lejos de ser entendido por los humanos.

Se necesitan lenguajes que medien entre el lenguaje natural y el lenguaje máquina, los llamados **lenguajes de programación de alto nivel**.

Un programa escrito en un lenguaje de programación se llama **código fuente** y el archivo que contiene el código se llama **archivo fuente**.

A series of light blue wavy lines that sweep across the bottom half of the slide, starting from the left and moving towards the right, creating a sense of motion and design.

Compilación o Interpretación

Si bien el lenguaje de programación nos permite escribir el conjunto de sentencias, la máquina sólo entiende el lenguaje máquina, por lo que se hace necesario **traducir** el lenguaje de programación al lenguaje máquina.

Compilación

El código fuente se traduce una vez, y cada vez que se modifique, para obtener un archivo ejecutable en lenguaje máquina. El programa que realiza la traducción se llama compilador.

Interpretación

El usuario final traduce el código a medida que se ejecuta, es decir que a medida que el código es usado, es traducido. Para esto se necesita un interprete.

Python

Python es un lenguaje de programación de **alto nivel**, **interpretado**, orientado a objetos y de propósito general.

Su nombre proviene de la serie de comedia **Monty Python Flying Circus**. Es el resultado del trabajo de una sola persona, **Guido Van Rossum**, pero su extensión y crecimientos se debe a una gran comunidad de desarrolladores, pues es un proyecto de **código abierto**.

Python3 es uno de los lenguajes más populares en la actualidad y se usa en gran medida en aplicaciones de Ciencia de datos, Aprendizaje automático e inteligencia artificial.

Comenzar en Python3

Para comenzar en Python necesitaremos 3 elementos importantes:

- Un editor de texto, este te ayudara a escribir el código, recuerda que python no reconoce el texto formateado.
- Una consola para iniciar y detener el código.
- Un depurador para ejecutar el código paso a paso

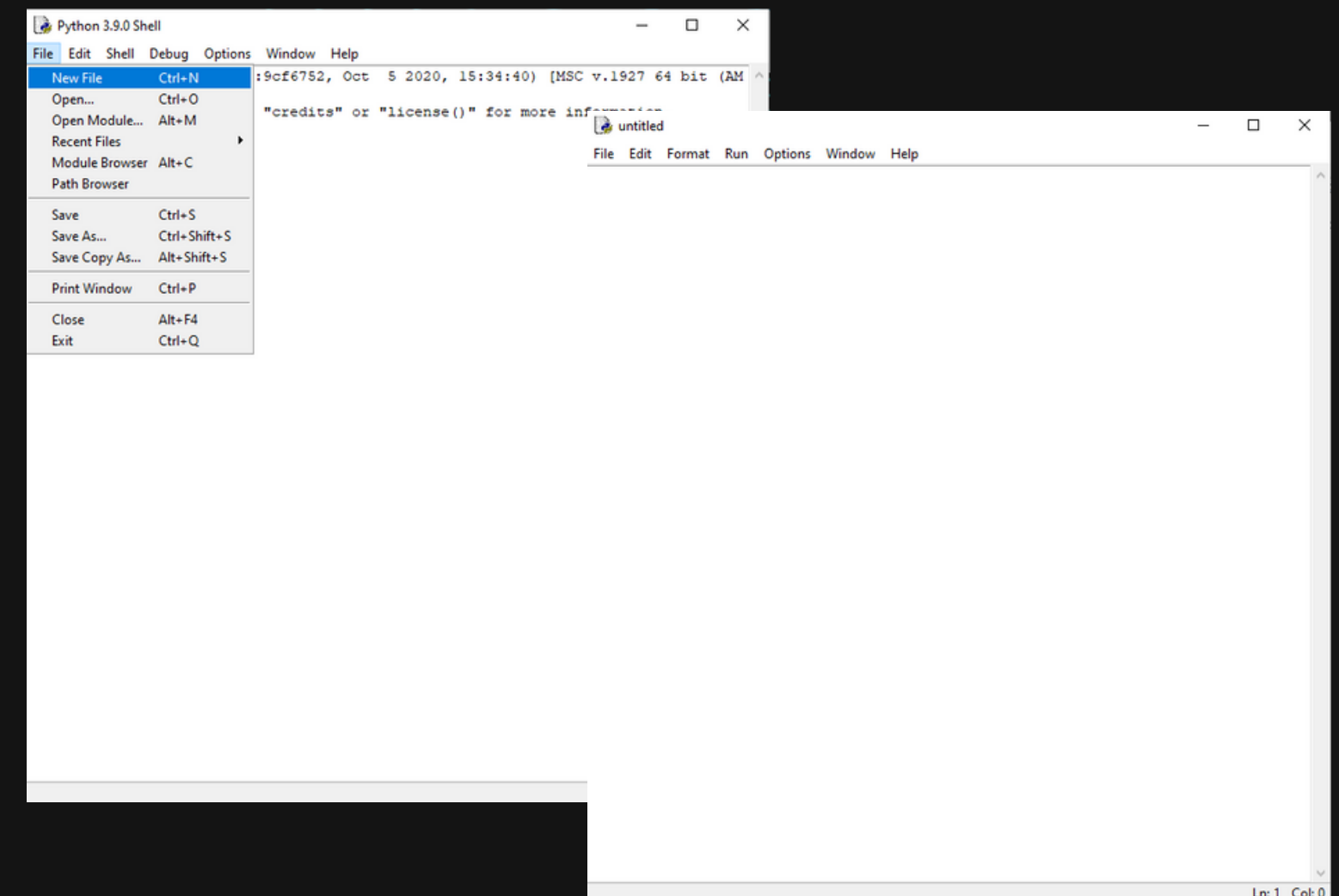
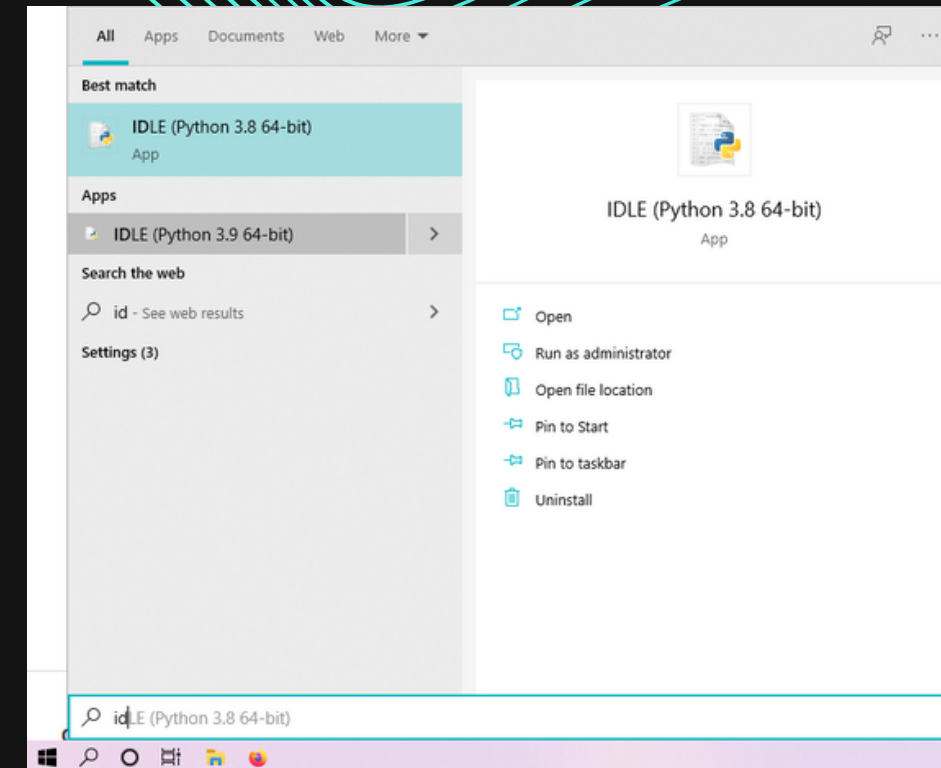
La instalación de python3 contiene un **IDLE** (Integrated Development and Learning Environment)



IDLE

Una vez instalado Python 3 en tu computadora haz lo siguiente:

- Busca IDLE en tus aplicaciones de escritorio y ejecutala.
- Se abrirá una ventana llamada Shell, aquí se ejecuta el código.
- En esa ventana ve a File > New File
- Esta ventana es donde escribirás tu código. Para ejecutar el código haz clic en Run, deberás guardar el archivo primero.

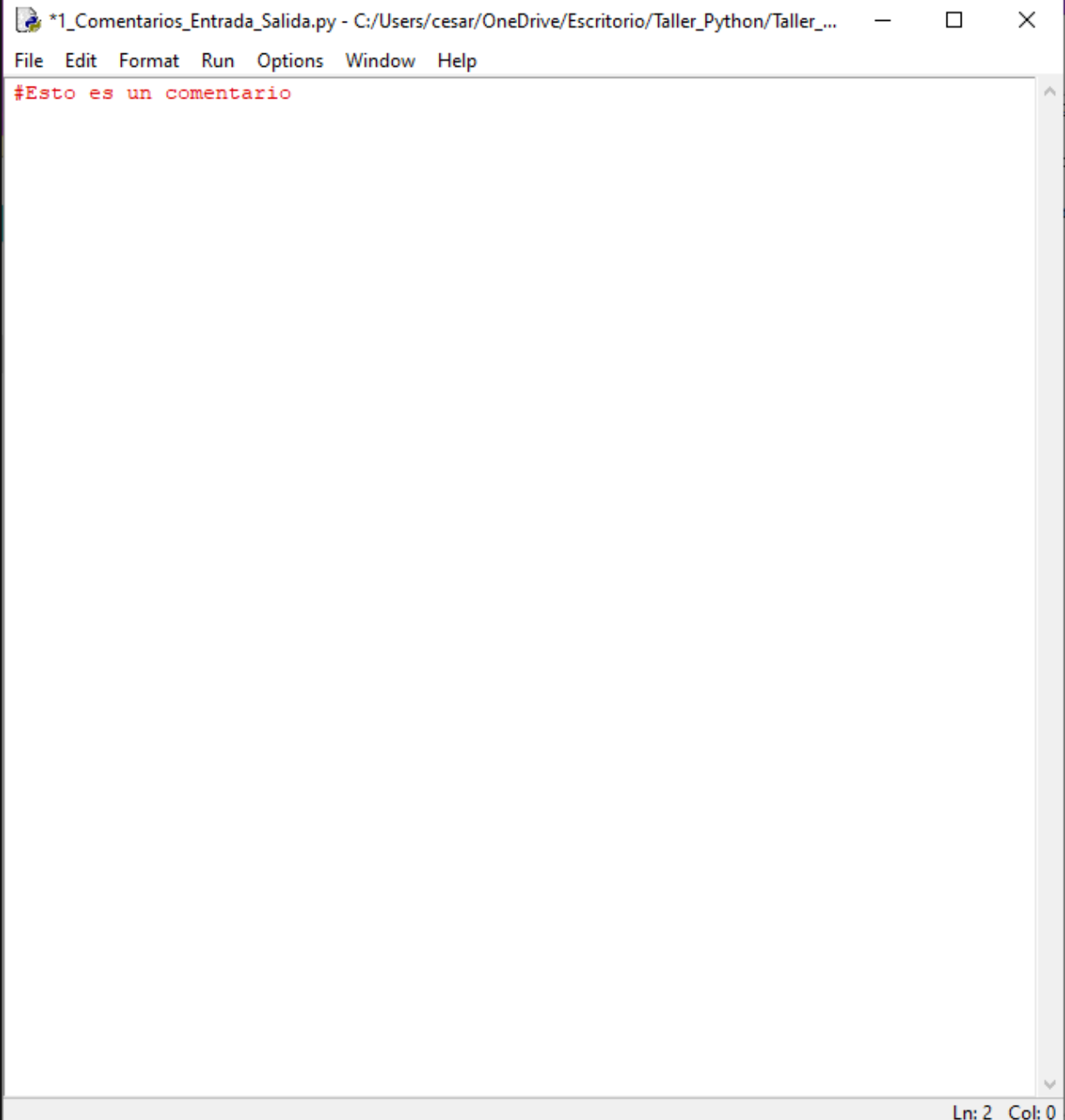


Comentarios

Un comentario en el código es una porción de caracteres que el lenguaje no tiene en cuenta para a la hora de ejecutarlo.

Sirve para informar a las otras personas sobre el código o deshabilitar el código que no se usa.

En python los comentarios se hacen escribiendo un **#** antes de comentar, se debe hacer un comentario por cada línea escrita.

A screenshot of a Python IDE window titled '*1_Comentarios_Entrada_Salida.py'. The window has a menu bar with 'File', 'Edit', 'Format', 'Run', 'Options', 'Window', and 'Help'. The main text area contains a single line of code: '#Esto es un comentario'. The text is in a red font. At the bottom right of the window, the status bar shows 'Ln: 2 Col: 0'.

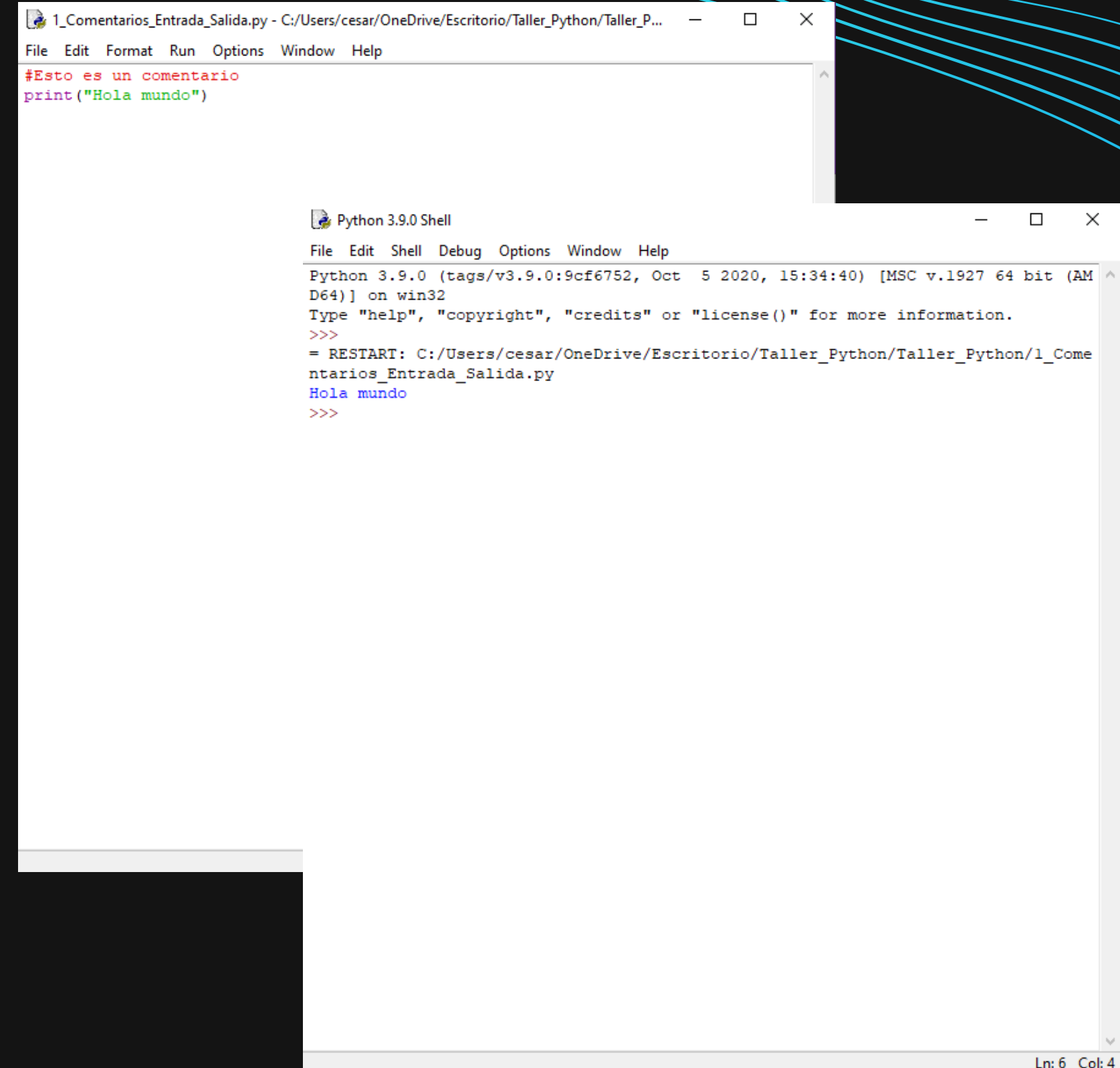
```
*1_Comentarios_Entrada_Salida.py - C:/Users/cesar/OneDrive/Escritorio/Taller_Python/Taller_...
File Edit Format Run Options Window Help
#Esto es un comentario
Ln: 2 Col: 0
```

Salida

`print ()` es una función nativa de Python3, esta función, como su nombre lo indica, nos permite imprimir cosas en la consola, esta función acepta varios parámetros y todo tipo de datos.

Haremos que python imprima un mensaje en la consola ejecutando la función `print()`

```
print("Hola Mundo")
```



The screenshot displays two windows from a Python IDE. The top window, titled '1_Comentarios_Entrada_Salida.py', contains the following code:

```
#Esto es un comentario
print("Hola mundo")
```

The bottom window, titled 'Python 3.9.0 Shell', shows the execution output:

```
Python 3.9.0 (tags/v3.9.0:9cf6752, Oct 5 2020, 15:34:40) [MSC v.1927 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
= RESTART: C:/Users/cesar/OneDrive/Escritorio/Taller_Python/Taller_Python/1_Comentarios_Entrada_Salida.py
Hola mundo
>>>
```

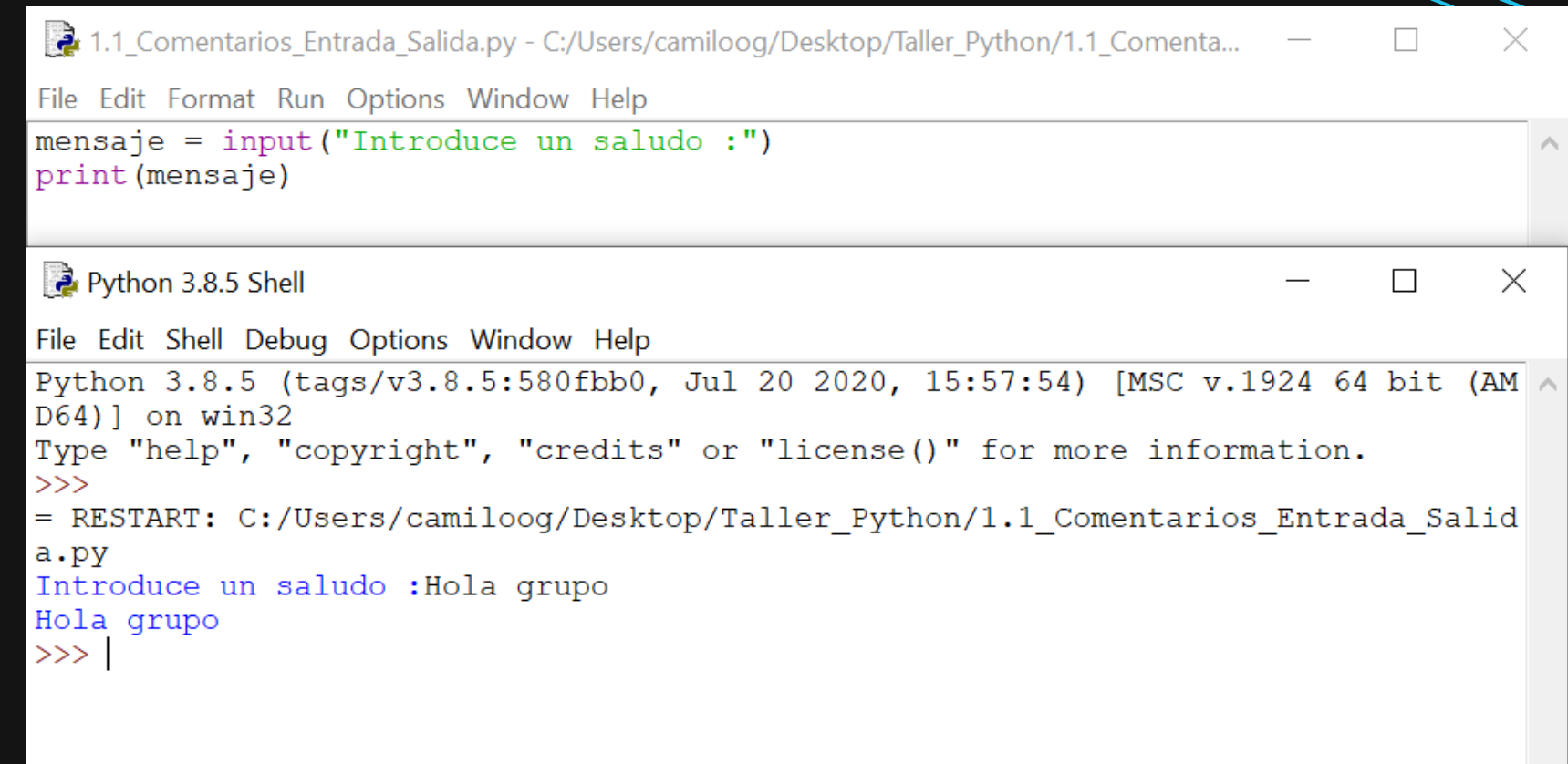
The status bar at the bottom right of the shell window indicates 'Ln: 6 Col: 4'.

Entrada

`input ()` es una función nativa de Python3, esta función, como su nombre lo indica, nos permite introducir cosas en la consola, esta función acepta varios parámetros y todo tipo de datos.

Haremos que Python imprima un mensaje en la consola que previamente le hemos enviado, ejecutando la función `input()`

`input("Hola Mundo")`



The screenshot displays two windows from a Python IDE. The top window, titled '1.1_Comentarios_Entrada_Salida.py', contains the following code: `mensaje = input("Introduce un saludo :")` and `print(mensaje)`. The bottom window, titled 'Python 3.8.5 Shell', shows the execution output. It starts with the Python version and architecture information, followed by a prompt `>>>`. The user enters `= RESTART: C:/Users/camiloog/Desktop/Taller_Python/1.1_Comentarios_Entrada_Salida.py`, then `Introduce un saludo :Hola grupo`, and finally `Hola grupo` is printed. The prompt `>>>` is followed by a vertical bar `|`.

```
1.1_Comentarios_Entrada_Salida.py - C:/Users/camiloog/Desktop/Taller_Python/1.1_Comenta...
File Edit Format Run Options Window Help
mensaje = input("Introduce un saludo :")
print(mensaje)

Python 3.8.5 Shell
File Edit Shell Debug Options Window Help
Python 3.8.5 (tags/v3.8.5:580fbb0, Jul 20 2020, 15:57:54) [MSC v.1924 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
= RESTART: C:/Users/camiloog/Desktop/Taller_Python/1.1_Comentarios_Entrada_Salida.py
Introduce un saludo :Hola grupo
Hola grupo
>>> |
```

Variables

Las variables son contenedores de valores en la memoria, es decir que el programa nos permite definir un espacio de memoria y asignarle un valor determinado.

Las variables se componen de dos elementos

Nombre: El nombre de la variable debe ser significativo, es decir tener sentido por sí mismo.

Valor: Es el valor que se le asigna a ese espacio de memoria.

Variables

El nombre de las variables deben cumplir con ciertas reglas:

1. Puede estar compuesto por mayúsculas, minúsculas, dígitos y guiones bajos (_)
2. Debe comenzar con una letra.
3. Es sensible a las mayusculas y minusculas.
4. No puede ser igual a las palabras reservadas de Python.
5. No debe contener espacios.

False	class	return	is	finally
None	if	for	lambda	continue
True	def	from	while	nonlocal
and	del	global	not	with
as	elif	try	or	yield
assert	else	import	pass	
break	except	in	raise	

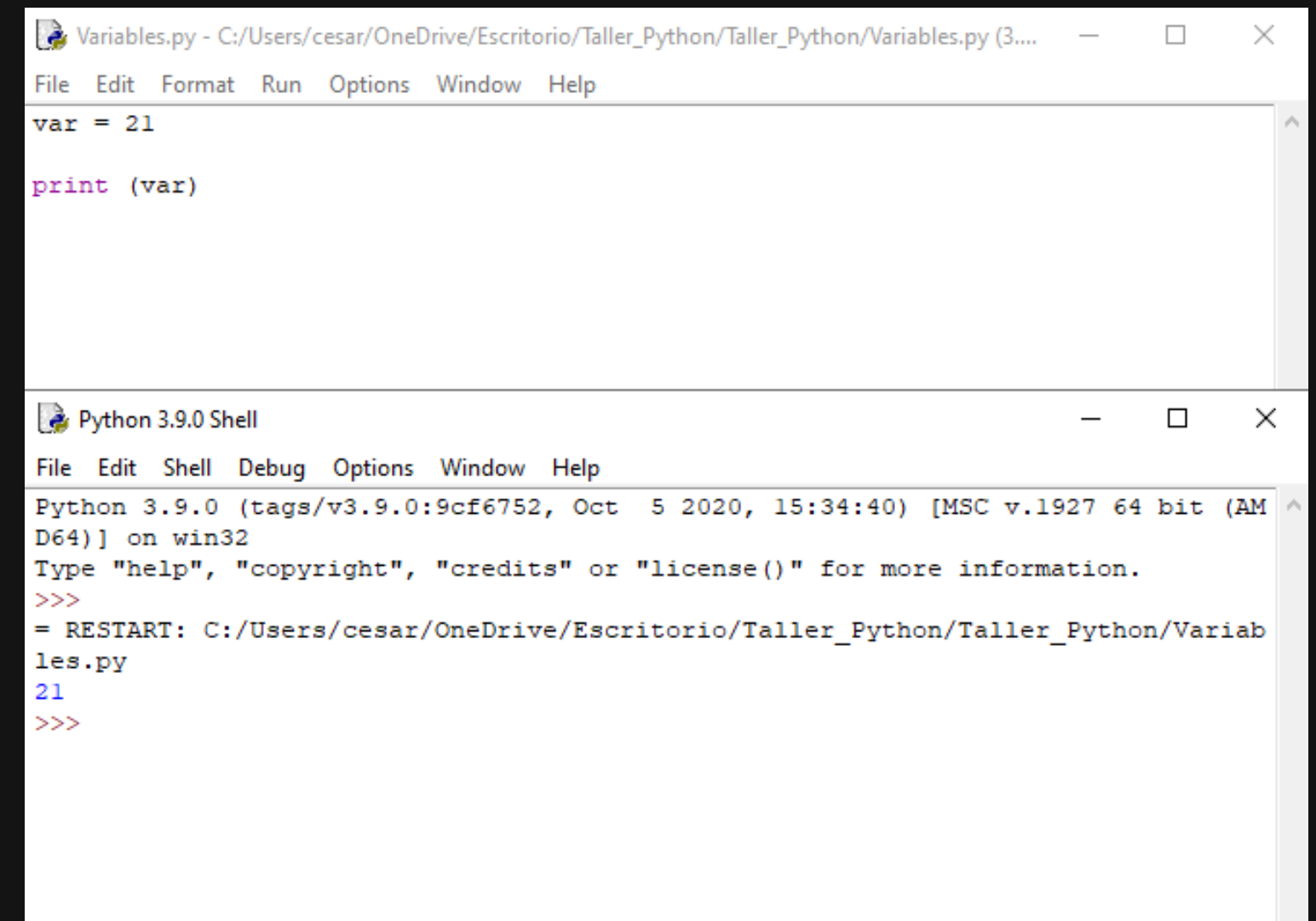
Variables

Para declararla sólo es necesario el nombre de la variable, el operador de asignación y el valor.

```
var = 21
```

Esto se lee: asigna el valor de 21 a una variable llamada var

Una variable se crea en el momento en que se le asigna un valor, no es necesario declararla.



The screenshot displays two windows from a Python IDE. The top window, titled 'Variables.py', contains the following code:

```
var = 21  
  
print (var)
```

The bottom window, titled 'Python 3.9.0 Shell', shows the output of running the script:

```
Python 3.9.0 (tags/v3.9.0:9cf6752, Oct 5 2020, 15:34:40) [MSC v.1927 64 bit (AMD64)] on win32  
Type "help", "copyright", "credits" or "license()" for more information.  
>>>  
= RESTART: C:/Users/cesar/OneDrive/Escritorio/Taller_Python/Taller_Python/Variables.py  
21  
>>>
```

Tipos de datos

Entero: `int` Son aquellos números que no tienen fracciones

Flotante: `float` El tipo flotante está designado para representar los números que tienen, o pueden tener una parte fraccionada después del punto decimal, por ejemplo 2.5 0 -0.4

Cadena: `string` Las cadenas se emplean cuando se quiere procesar texto y requieren comillas simples ' ' o dobles " " para que el programa entienda que se deben tomar de forma literal.

Booleano: `bool` Se utilizan para evaluar si algo es verdadero o falso, como por ejemplo $1 > 2$, esto solo tiene dos posibles respuestas, que sea verdad o que sea falso. Su notación en Python es `True` y `False`, respetando las mayúsculas y minúsculas.

Operadores aritméticos

Suma

Para la suma usamos el signo de más +

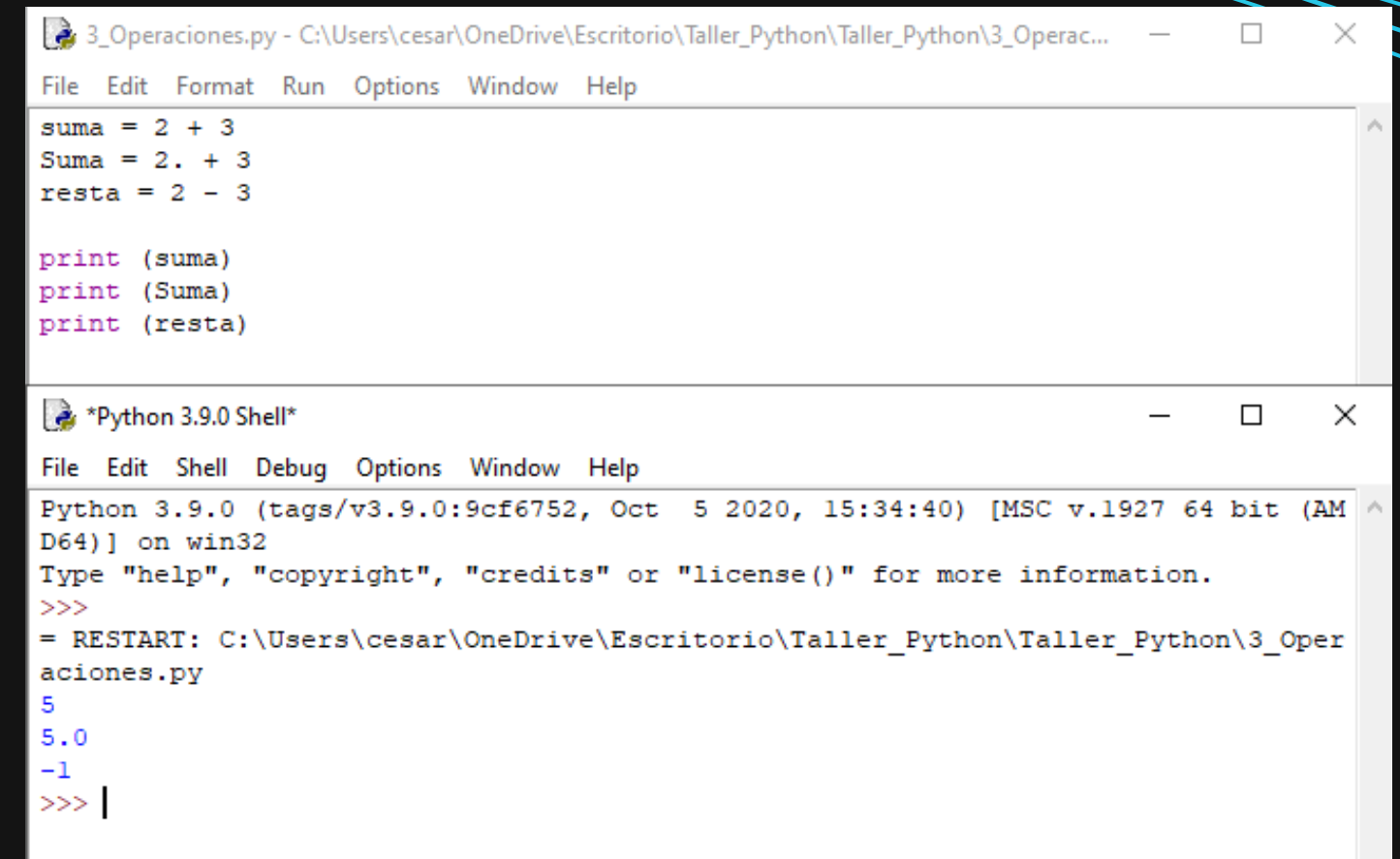
Se debe tener en cuenta que si ambos números a sumar son enteros, el resultado será un entero.

Si al menos uno de los número es flotante, el resultado será un flotante.

Resta

Para la resta usamos el signo de menos -

Así mismo cumple con la regla entero vs flotante



The image shows two windows from a Windows operating system. The top window is a text editor titled '3_Operaciones.py' containing the following Python code:

```
suma = 2 + 3
Suma = 2. + 3
resta = 2 - 3

print (suma)
print (Suma)
print (resta)
```

The bottom window is a 'Python 3.9.0 Shell' showing the execution of the script. It displays the output of the print statements:

```
= RESTART: C:\Users\cesar\OneDrive\Escritorio\Taller_Python\Taller_Python\3_Operaciones.py
5
5.0
-1
>>> |
```

Operadores aritméticos

Multiplicación

Para la multiplicación se usa un solo asterisco `*`

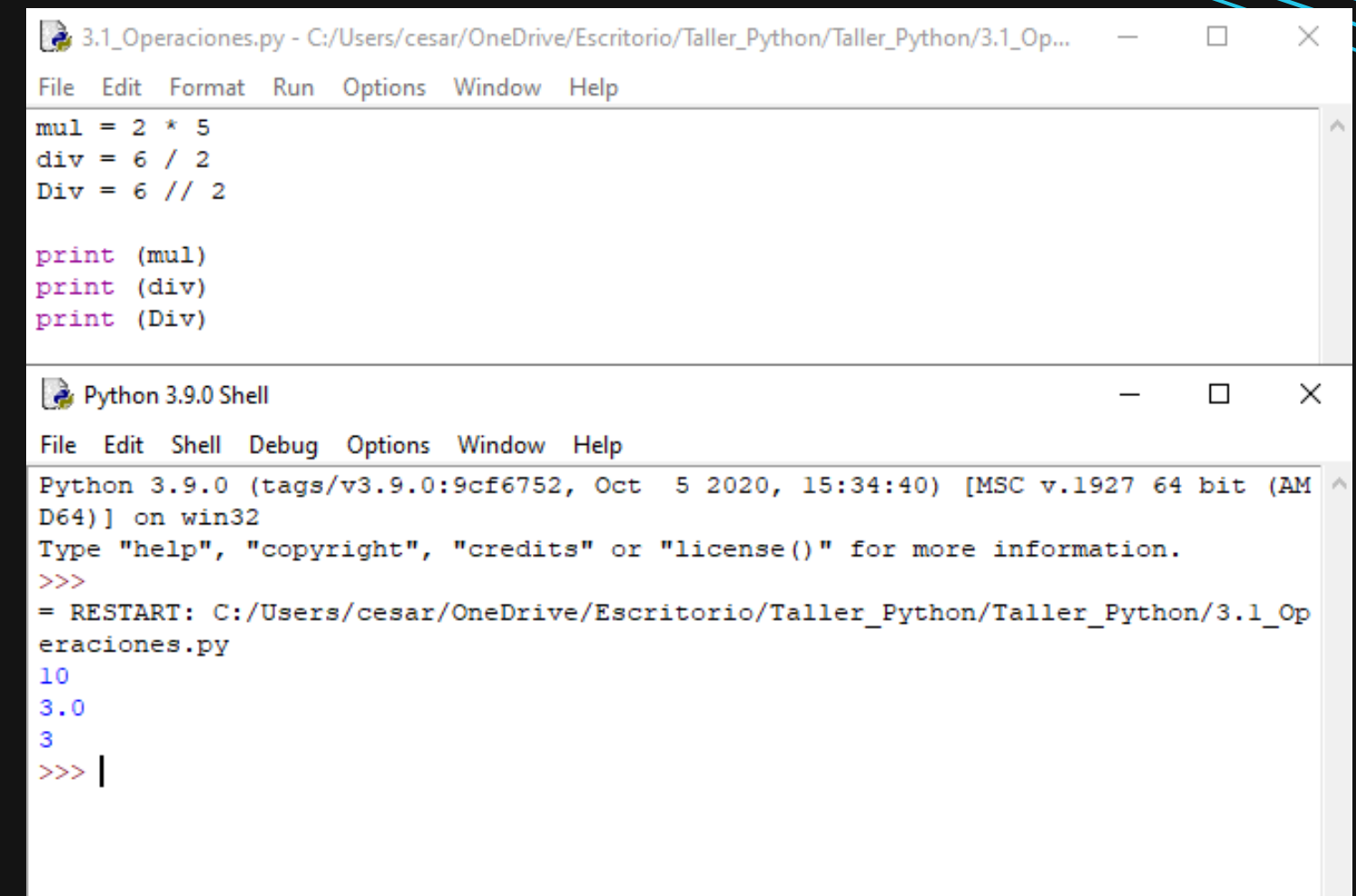
Sigue aplicando la regla de entero vs flotante.

División

Para la división usamos el la barra diagonal `/` el valor antes de la barra es dividendo, el valor después de la barra es el divisor.

El resultado de una división siempre será flotante.

Para la división entera usamos la doble diagonal `//` esto hará que todo resultado sea redondeado a su entero más bajo, pero obedece a la regla entero vs flotante.



The screenshot displays two windows from a Python IDE. The top window, titled '3.1_Operaciones.py', contains the following Python code:

```
mul = 2 * 5
div = 6 / 2
Div = 6 // 2

print (mul)
print (div)
print (Div)
```

The bottom window, titled 'Python 3.9.0 Shell', shows the output of running the script. It includes the Python version and architecture information, followed by the results of the print statements:

```
Python 3.9.0 (tags/v3.9.0:9cf6752, Oct 5 2020, 15:34:40) [MSC v.1927 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
= RESTART: C:/Users/cesar/OneDrive/Escritorio/Taller_Python/Taller_Python/3.1_Operaciones.py
10
3.0
3
>>> |
```


Operadores aritméticos

Exponenciación

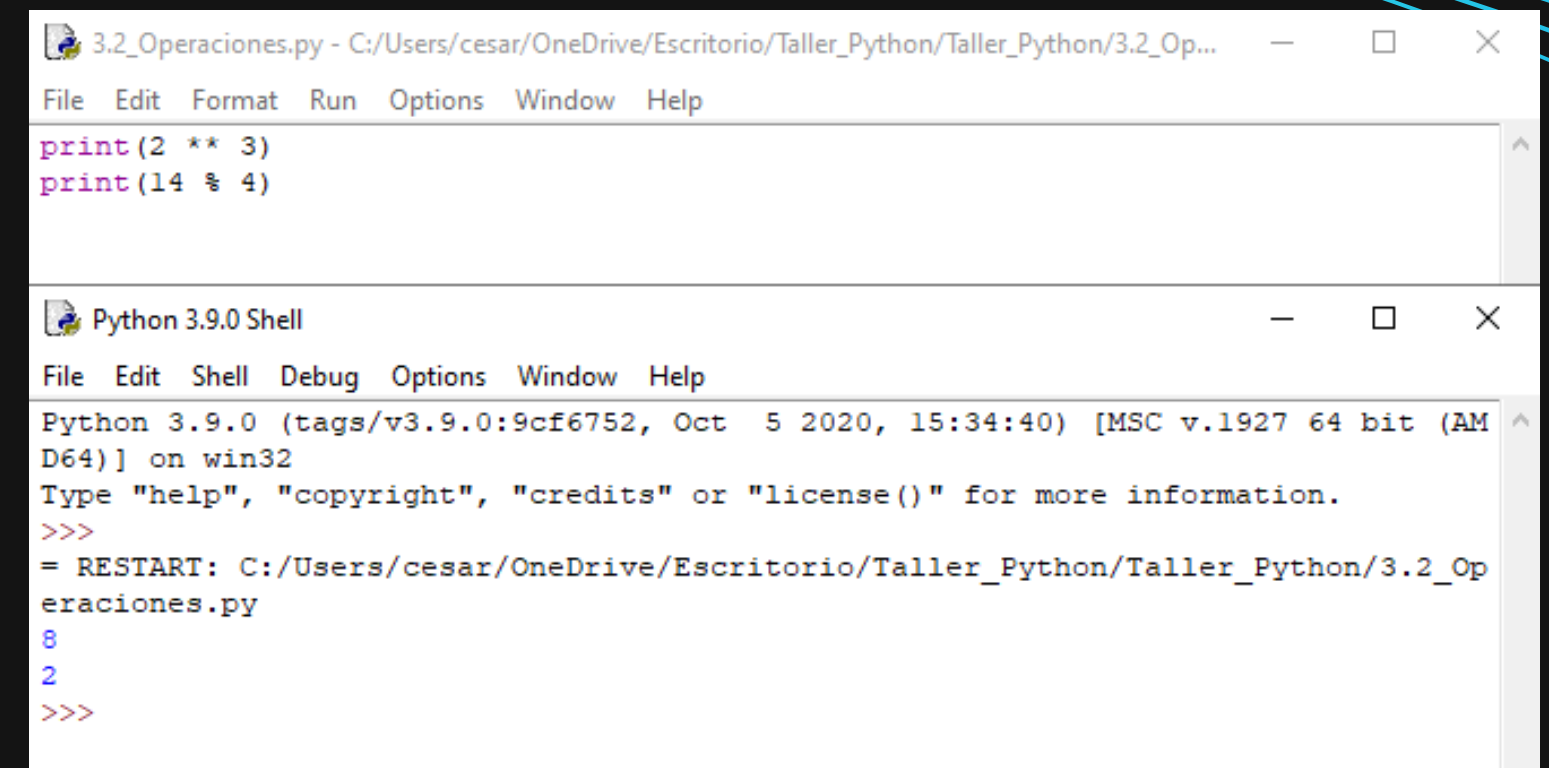
Para la exponenciación usamos el doble asterisco ****** el argumento de la izquierda es la base, el de la derecha la potencia.

Es una convención dejar espacio alrededor de los asteriscos.

Módulo

Este operador es especial pues no tiene un equivalente en los operadores aritméticos tradicionales.

En Python se representa como **%** y lo que hace es devolver el residuo de una división



The screenshot shows two windows from a Python IDE. The top window, titled '3.2_Operaciones.py', contains the following code:

```
print(2 ** 3)
print(14 % 4)
```

The bottom window, titled 'Python 3.9.0 Shell', shows the output of running the script:

```
Python 3.9.0 (tags/v3.9.0:9cf6752, Oct 5 2020, 15:34:40) [MSC v.1927 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
= RESTART: C:/Users/cesar/OneDrive/Escritorio/Taller_Python/Taller_Python/3.2_Operaciones.py
8
2
>>>
```


Operadores prioridades y enlaces

Python reconoce la prioridad de los operadores en las expresiones que usan varias operaciones y ejecuta la operación según su jerarquía.

Esto es como comúnmente se hace en aritmética.

1. Operaciones en paréntesis
2. Exponenciación, Radicación y Módulo
3. Multiplicación y División
4. Suma y Resta

Operadores relacionales

Los operadores relacionales son aquellos que establecen situaciones de igualdad o diferencia entre dos valores, es decir, comparaciones.

== Igualdad

!= Diferencia

< Menor que

<= Menor o igual que

> Mayor que

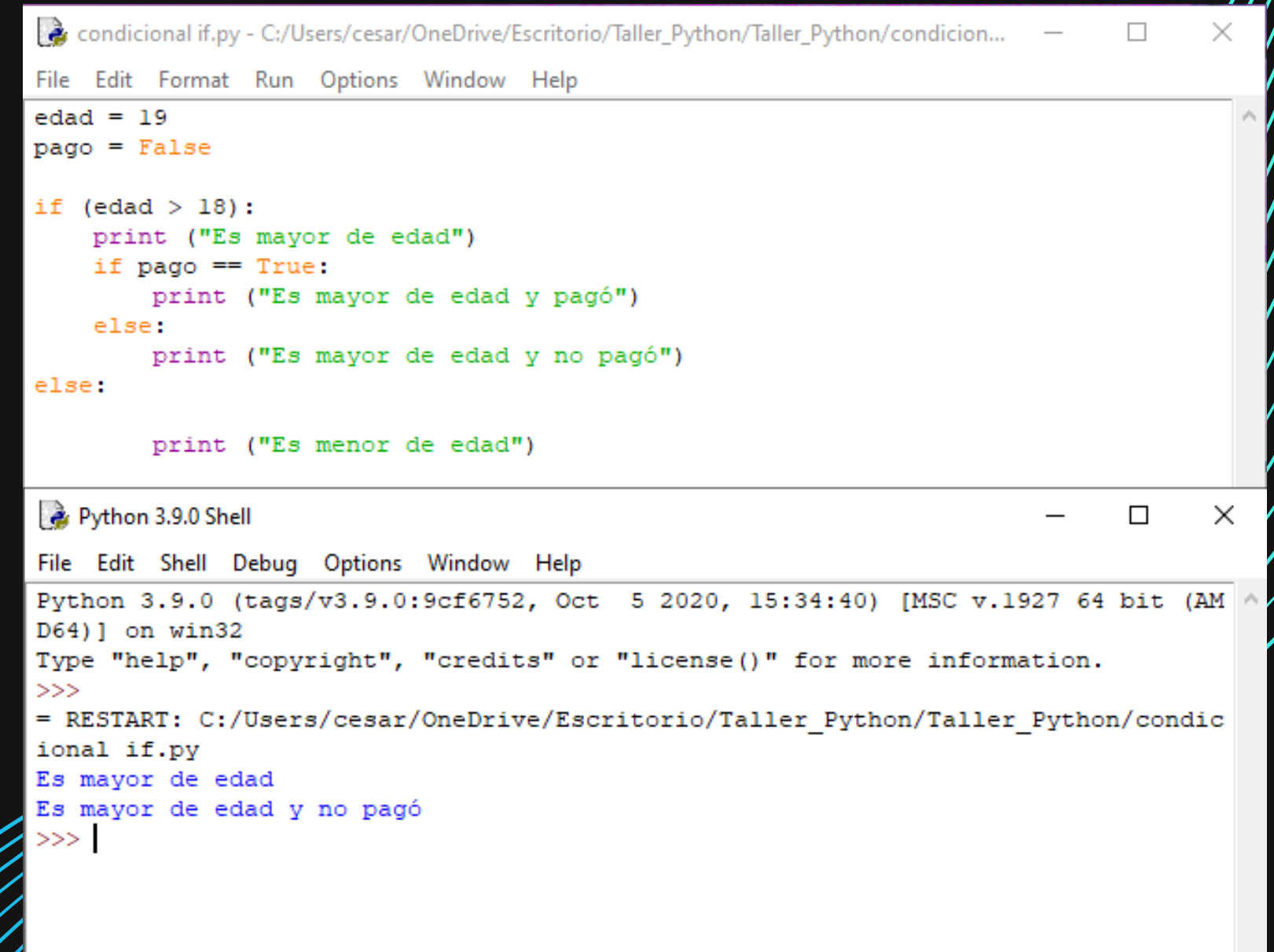
>= Mayor o igual que

Condicional if

Un condicional es una sentencia que nos permite tomar decisiones si se cumple determinada condición que ha sido establecida.

If anidados

Los if anidados nos sirven para establecer condiciones dentro de otras condiciones



The image shows a screenshot of a Python IDE with two windows. The top window, titled 'condicional if.py', contains the following Python code:

```
edad = 19
pago = False

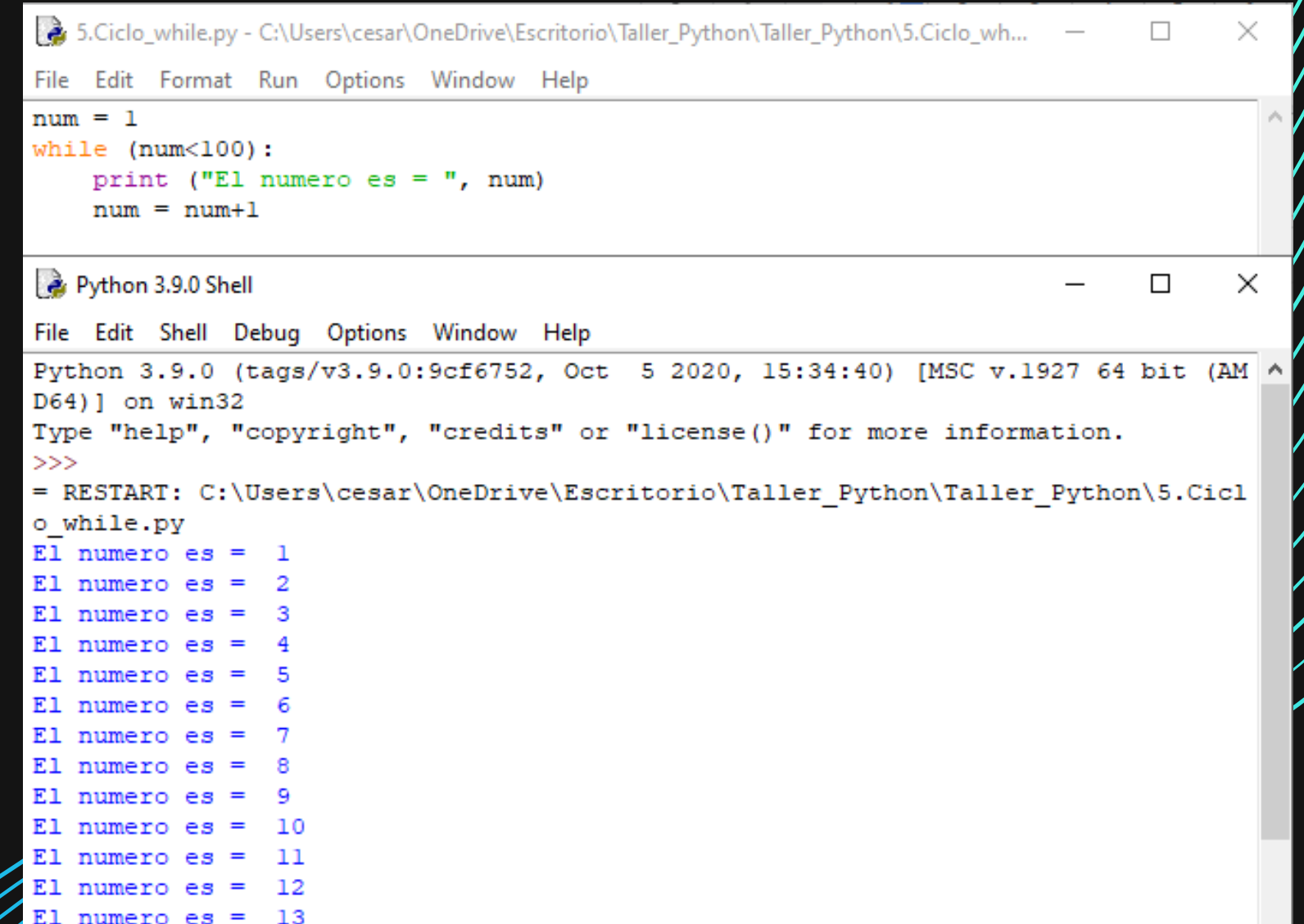
if (edad > 18):
    print ("Es mayor de edad")
    if pago == True:
        print ("Es mayor de edad y pagó")
    else:
        print ("Es mayor de edad y no pagó")
else:
    print ("Es menor de edad")
```

The bottom window, titled 'Python 3.9.0 Shell', shows the output of running the script. It displays the version information and the execution results of the code:

```
Python 3.9.0 (tags/v3.9.0:9cf6752, Oct 5 2020, 15:34:40) [MSC v.1927 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
= RESTART: C:/Users/cesar/OneDrive/Escritorio/Taller_Python/Taller_Python/condicional if.py
Es mayor de edad
Es mayor de edad y no pagó
>>> |
```

Bucle while

El ciclo **while** ejecuta el código mientras la condición devuelva True, es decir, se cumpla.



The image shows a screenshot of a Python IDE with two windows. The top window, titled '5.Ciclo_while.py', contains the following code:

```
num = 1
while (num<100):
    print ("El numero es = ", num)
    num = num+1
```

The bottom window, titled 'Python 3.9.0 Shell', shows the output of running the script. It displays the file path and the execution of the while loop, printing the numbers 1 through 13.

```
Python 3.9.0 (tags/v3.9.0:9cf6752, Oct 5 2020, 15:34:40) [MSC v.1927 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
= RESTART: C:\Users\cesar\OneDrive\Escritorio\Taller_Python\Taller_Python\5.Ciclo_while.py
El numero es = 1
El numero es = 2
El numero es = 3
El numero es = 4
El numero es = 5
El numero es = 6
El numero es = 7
El numero es = 8
El numero es = 9
El numero es = 10
El numero es = 11
El numero es = 12
El numero es = 13
```

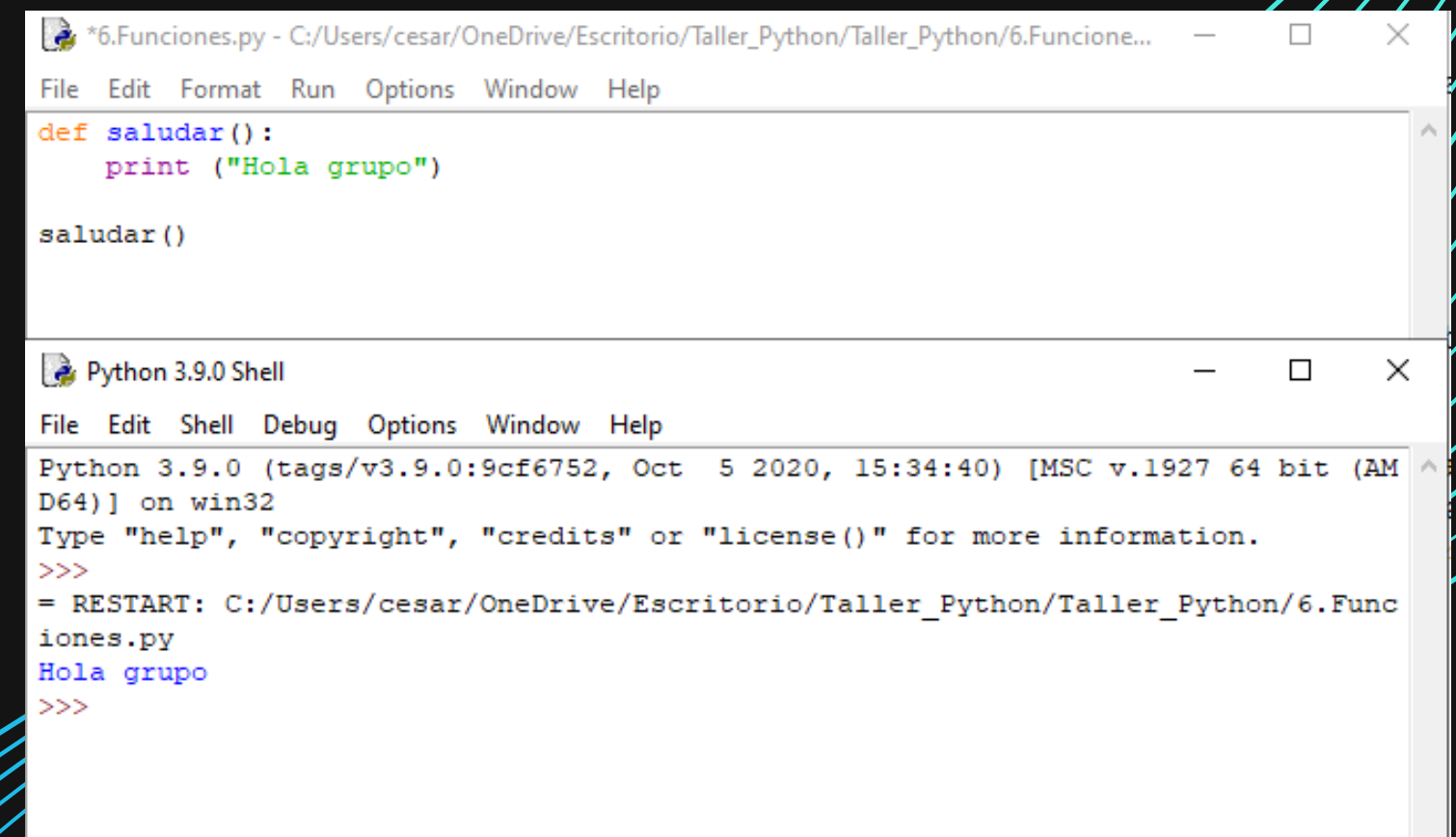
Funciones

Una función consiste encapsular una porción de código para hacerlo reutilizable, esta porción de código debe devolver siempre un valor, si no lo hace no sería una función, sino un método

Una función se declara poniendo la palabra reservada `def`, el nombre de la función, paréntesis y dos puntos.

El nombre de la función debe ser un verbo, que indique que acción realiza.

Al momento de declarar las funciones estas debe llevar el código que va a ejecutar cuando sea invocada, si no se coloca Python lanzará un error.



The screenshot displays two windows from a Python IDE. The top window, titled '*6.Funciones.py', contains the following code:

```
def saludar():  
    print("Hola grupo")  
  
saludar()
```

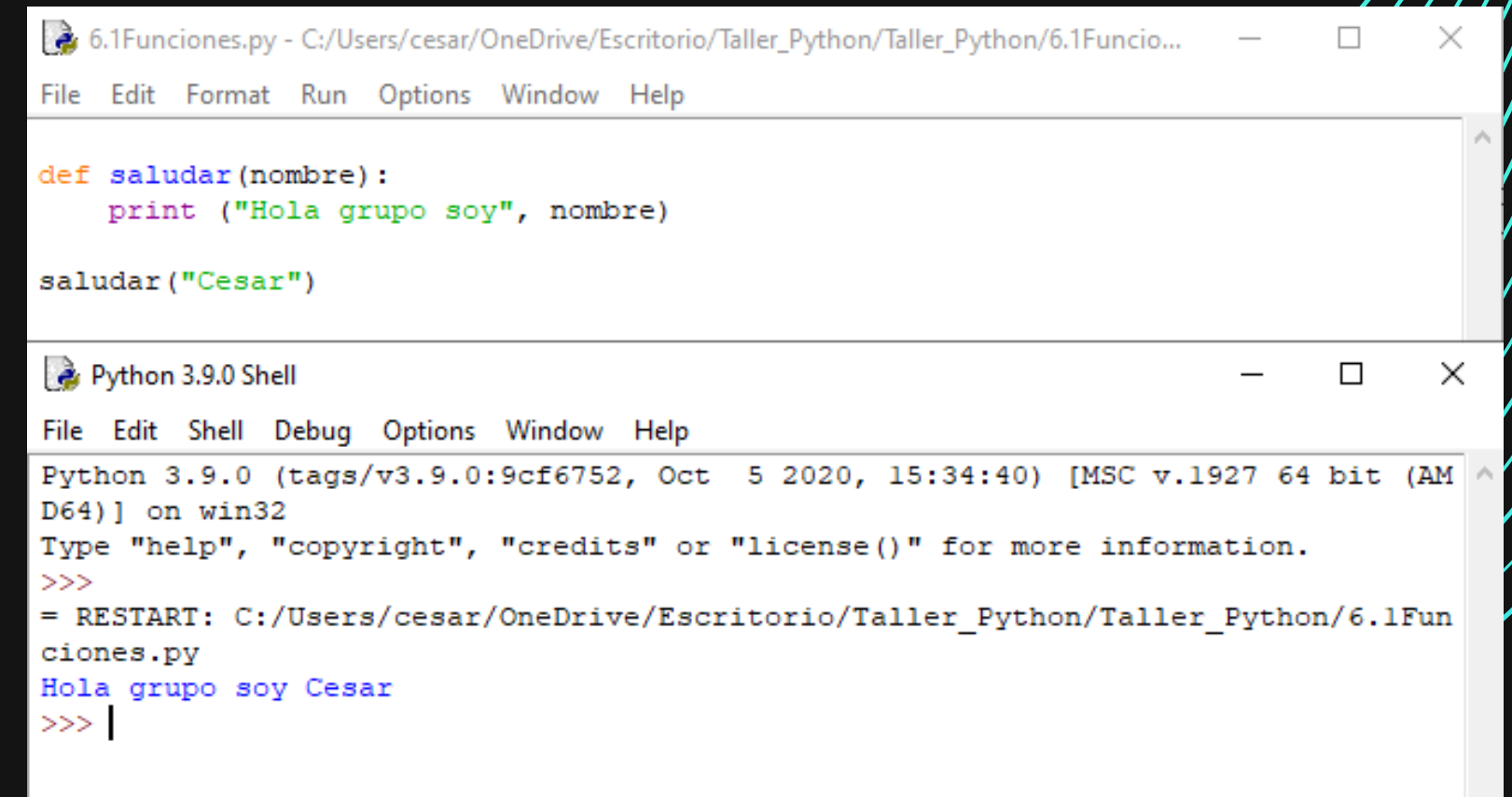
The bottom window, titled 'Python 3.9.0 Shell', shows the execution of the code. It displays the Python version and system information, followed by the output of the function call:

```
Python 3.9.0 (tags/v3.9.0:9cf6752, Oct 5 2020, 15:34:40) [MSC v.1927 64 bit (AMD64)] on win32  
Type "help", "copyright", "credits" or "license()" for more information.  
>>>  
= RESTART: C:/Users/cesar/OneDrive/Escritorio/Taller_Python/Taller_Python/6.Funciones.py  
Hola grupo  
>>>
```


Funciones y parametros.

Las funciones pueden recibir parámetros o variables, esto se hace poniendo entre los paréntesis la variable que va a acompañar a la función y definiendo en qué lugar se va a usar la variable dentro del código.

El paso de valores o de parámetros a una función sirve para utilizar datos que vienen de otra función o del exterior y manipularlos dentro de la propia función.



The screenshot shows two windows from a Python IDE. The top window, titled '6.1Funciones.py', contains the following Python code:

```
def saludar(nombre):  
    print ("Hola grupo soy", nombre)  
  
saludar("Cesar")
```

The bottom window, titled 'Python 3.9.0 Shell', shows the output of running the code:

```
Python 3.9.0 (tags/v3.9.0:9cf6752, Oct 5 2020, 15:34:40) [MSC v.1927 64 bit (AMD64)] on win32  
Type "help", "copyright", "credits" or "license()" for more information.  
>>>  
= RESTART: C:/Users/cesar/OneDrive/Escritorio/Taller_Python/Taller_Python/6.1Funciones.py  
Hola grupo soy Cesar  
>>> |
```