



Carátula para entrega de prácticas

Facultad de Ingeniería

Laboratorio de docencia

Laboratorios de computación salas A y B

Profesor: M.I. MARCO ANTONIO MARTINEZ

Asignatura: ESTRUCTURA DE DATOS Y ALGORITMOS

Grupo: 17

No de Práctica(s):

ISLAS ESPINO JESÚS ABRAHAM

Integrante(s):

*No. de Equipo de
cómputo empleado:*

19

No. de Lista o Brigada:

Semestre: 2020-2

Fecha de entrega: 31-mar-2020

Observaciones:

CALIFICACIÓN: _____

INTRODUCCIÓN A PYTHON (1)

Objetivo

Aplicar las bases del lenguaje de programación Python en el ambiente de Jupyter notebook.

Introducción

Python es un lenguaje interpretado, dinámico, multiplataforma y de programación multiparadigma, ya que soporta orientación a objetos, programación imperativa y, programación funcional.

Los nombres de las variables en python son alfanuméricos (a-z, A-Z, 0-9) y empiezan con una letra en minúscula.

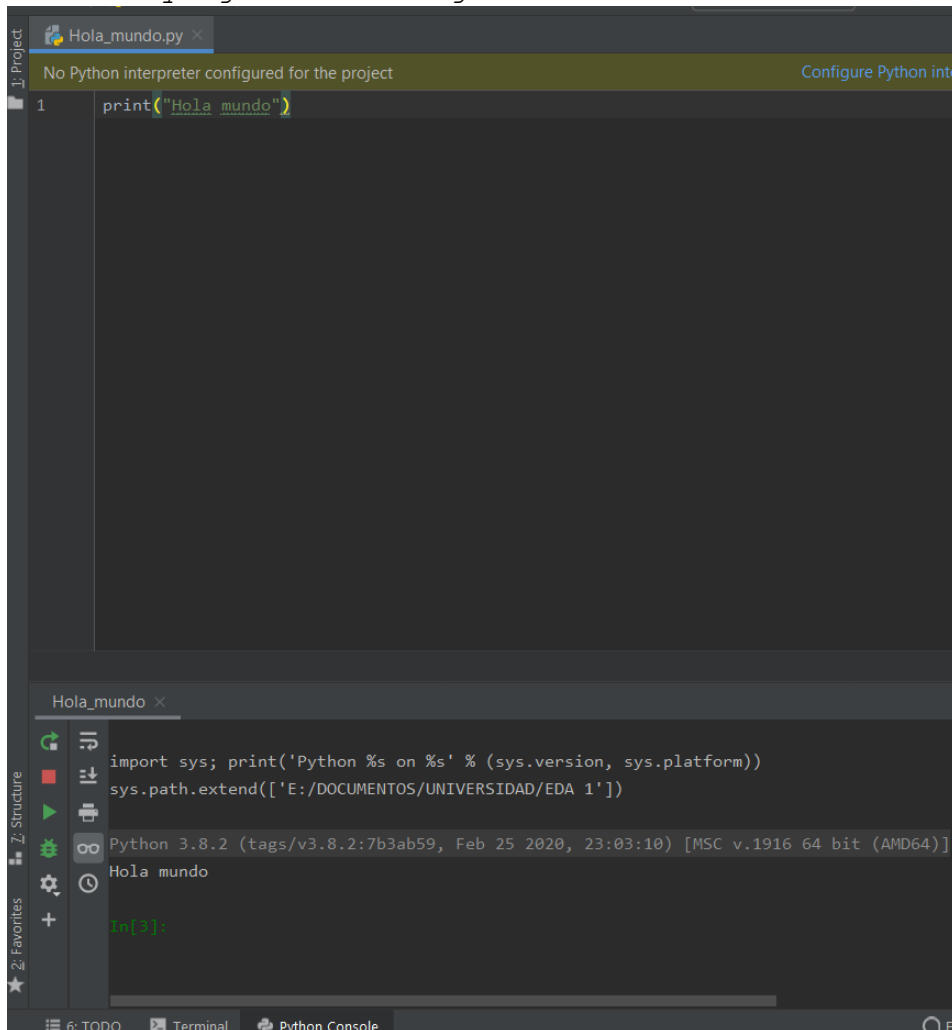
El valor de una variable esta implícito al momento de asignar un valor. No se necesita poner; al final de cada instrucción.

Mantener las indentaciones al momento de escribir código.

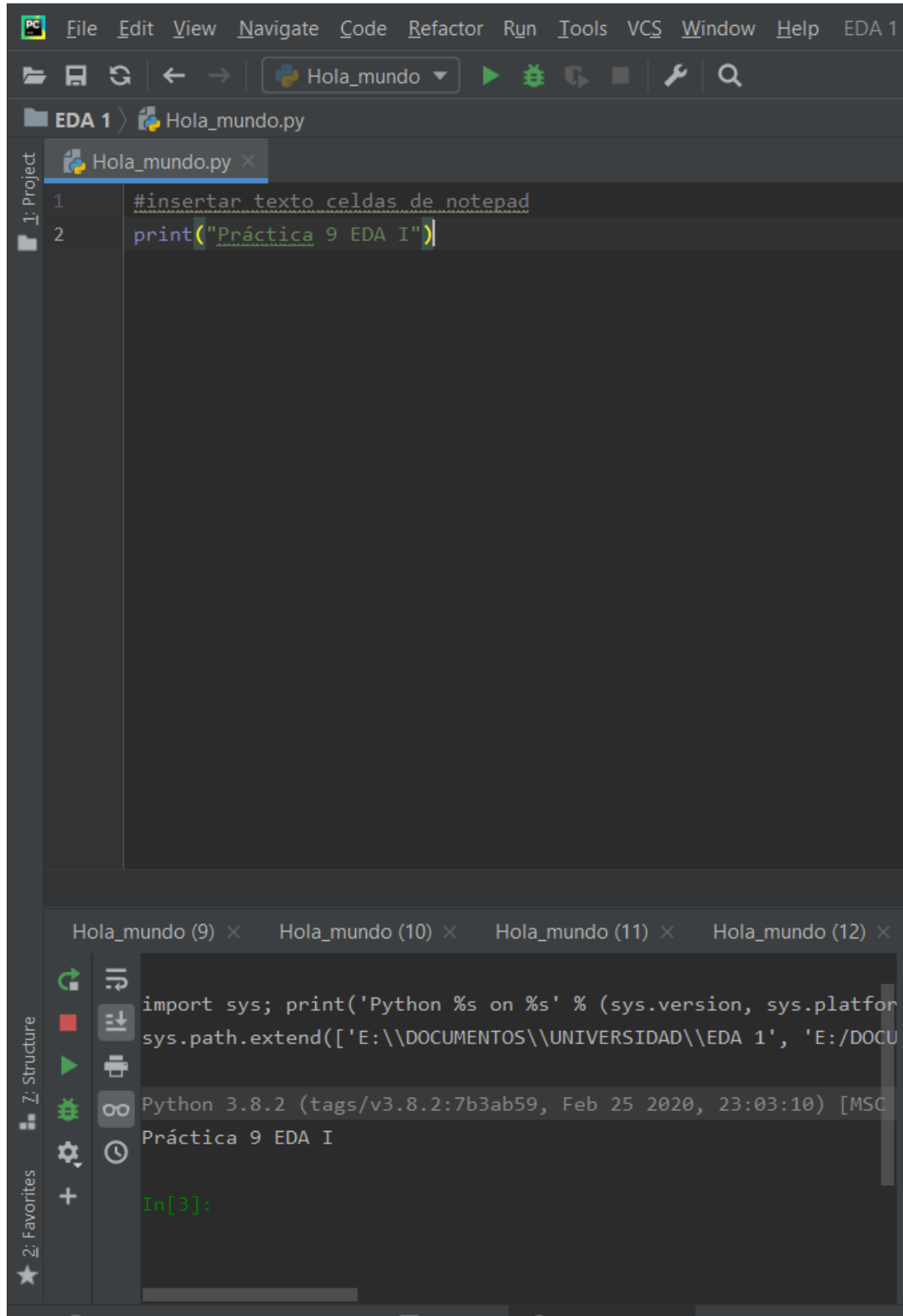
and, as, assert, break, class, continue, def, del, elif, else, except, exec, finally, for, from, global, if, import, in, is, lambda, not, or, pass, print, raise, return, try, while, with, yield.

Desarrollo

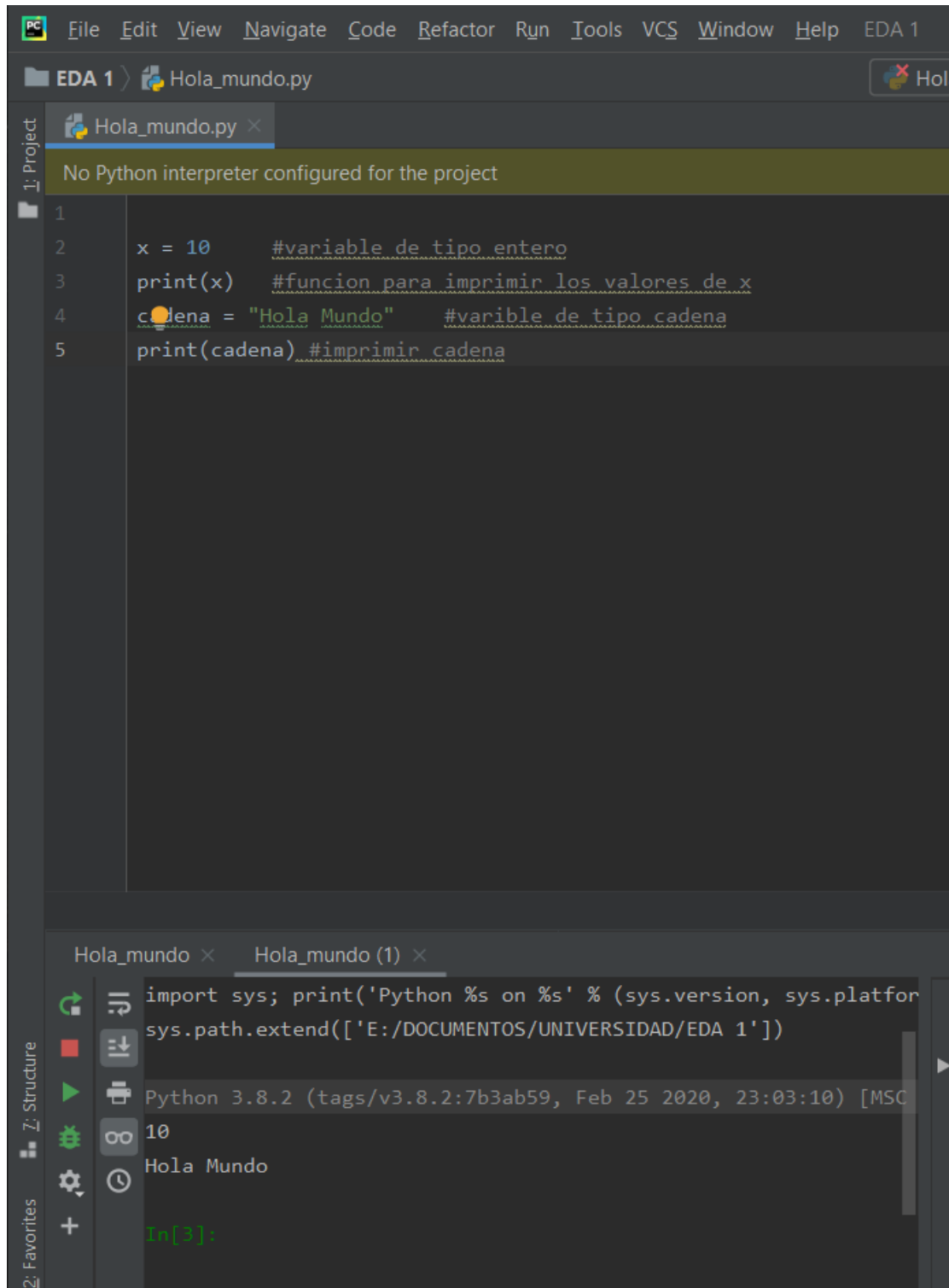
- Insertar y ejecutar código en las celdas de la notebook.



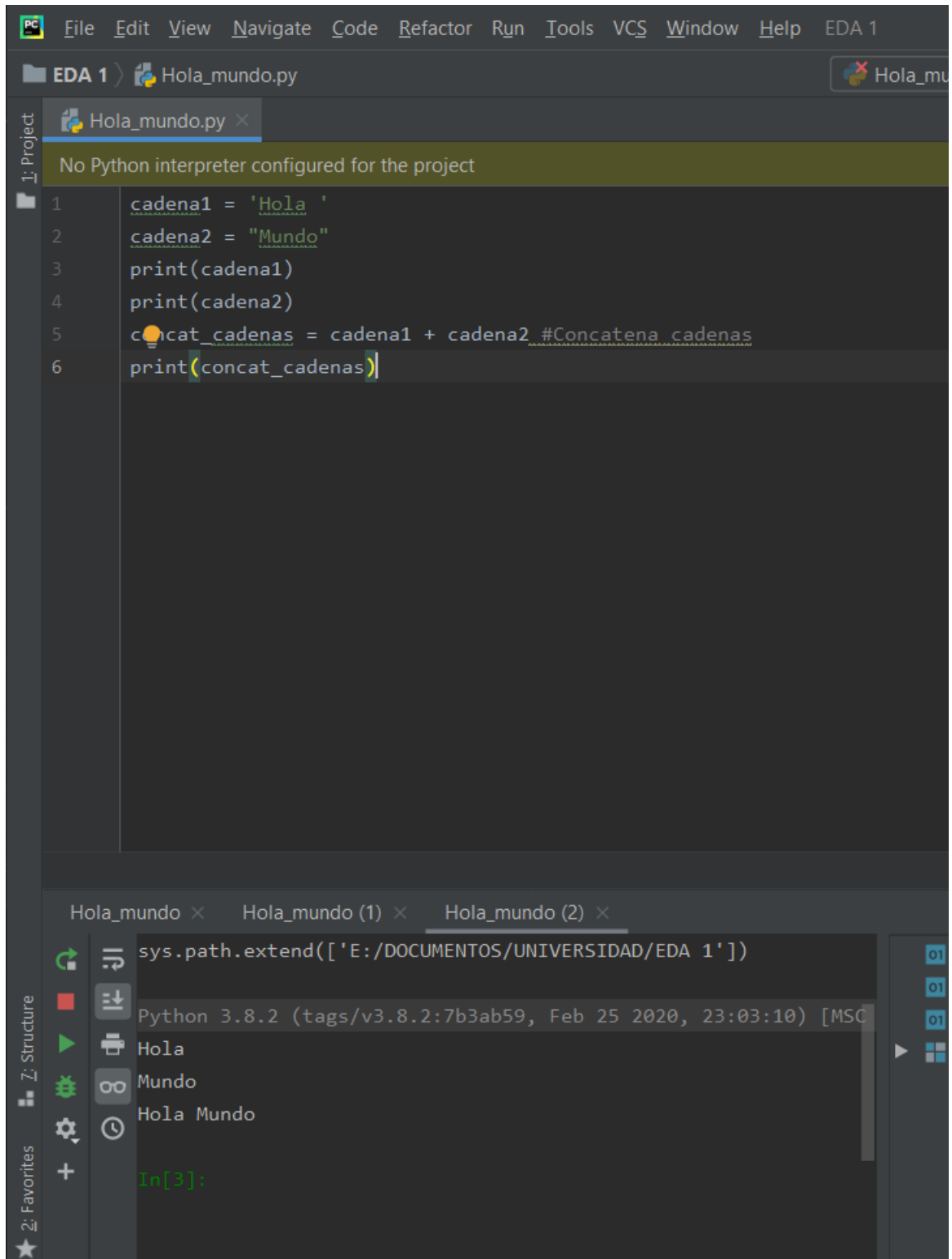
- Insertar texto en las celdas de la notebook



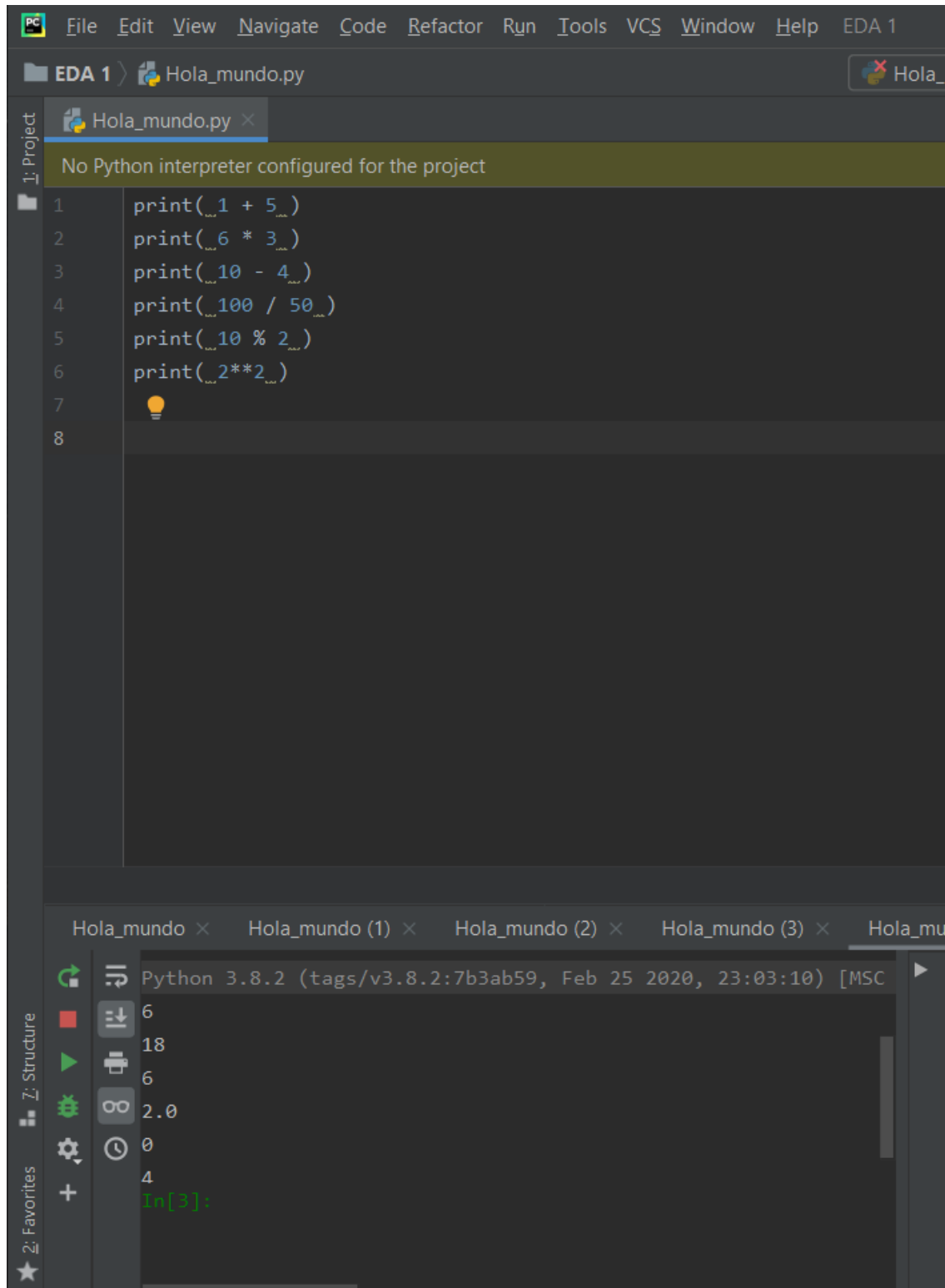
- Declarar variables



- Declarar cadenas



- Aplicar operadores



- Crear y manipular listas, tuplas y diccionarios.

The screenshot displays the EDA 1 IDE interface. The main editor window shows a Python script named `Hola_mundo.py`. The script creates a dictionary `elementos` with keys for chemical elements and their atomic numbers. It then prints the dictionary and updates it with new elements like 'litio' and 'nitrogeno'. A second dictionary `elementos2` is created, containing detailed information for 'Hydrogen' and 'Helium'. The script prints the entire `elementos2` dictionary, the details for 'Hydrogen', and updates the weight of 'Hydrogen' to 4.3. The bottom panel shows the output of the script, including the dictionary structure and the keys of the nested dictionary.

```

1 #creando un diccionario
2 elementos = {'hidrogeno': 1, 'helio': 2, 'carbon': 6}
3
4 #El momento de la impresion, pueden aparecer en diferente orden del intro
5 print__(elementos)
6
7 print__(elementos['hidrogeno'])
8 elementos['litio'] = 3
9 elementos['nitrogeno'] = 8
10
11 print__(elementos)
12 elementos2 = {}
13 elementos2['H'] = {'name': 'Hydrogen', 'number': 1, 'weight': 1.00794}
14 elementos2['He'] = {'name': 'Helium', 'number': 2, 'weight': 4.002602}
15
16 print__(elementos2)
17 print__(elementos2['H'])
18 print__(elementos2['H']['name'])
19 print__(elementos2['H']['number'])
20 elementos2['H']['weight'] = 4.30 #Cambiando el valor de un elemento

```

The bottom panel shows the output of the script, including the dictionary structure and the keys of the nested dictionary:

```

{'name': 'Hydrogen', 'number': 1, 'weight': 1.00794}
Hydrogen
1
4.3
{'name': 'Hydrogen', 'number': 1, 'weight': 4.3, 'gas noble': T
dict_items([('H', {'name': 'Hydrogen', 'number': 1, 'weight': 4
dict keys(['H', 'He'])
In[3]:

```

- Crear y ejecutar funciones

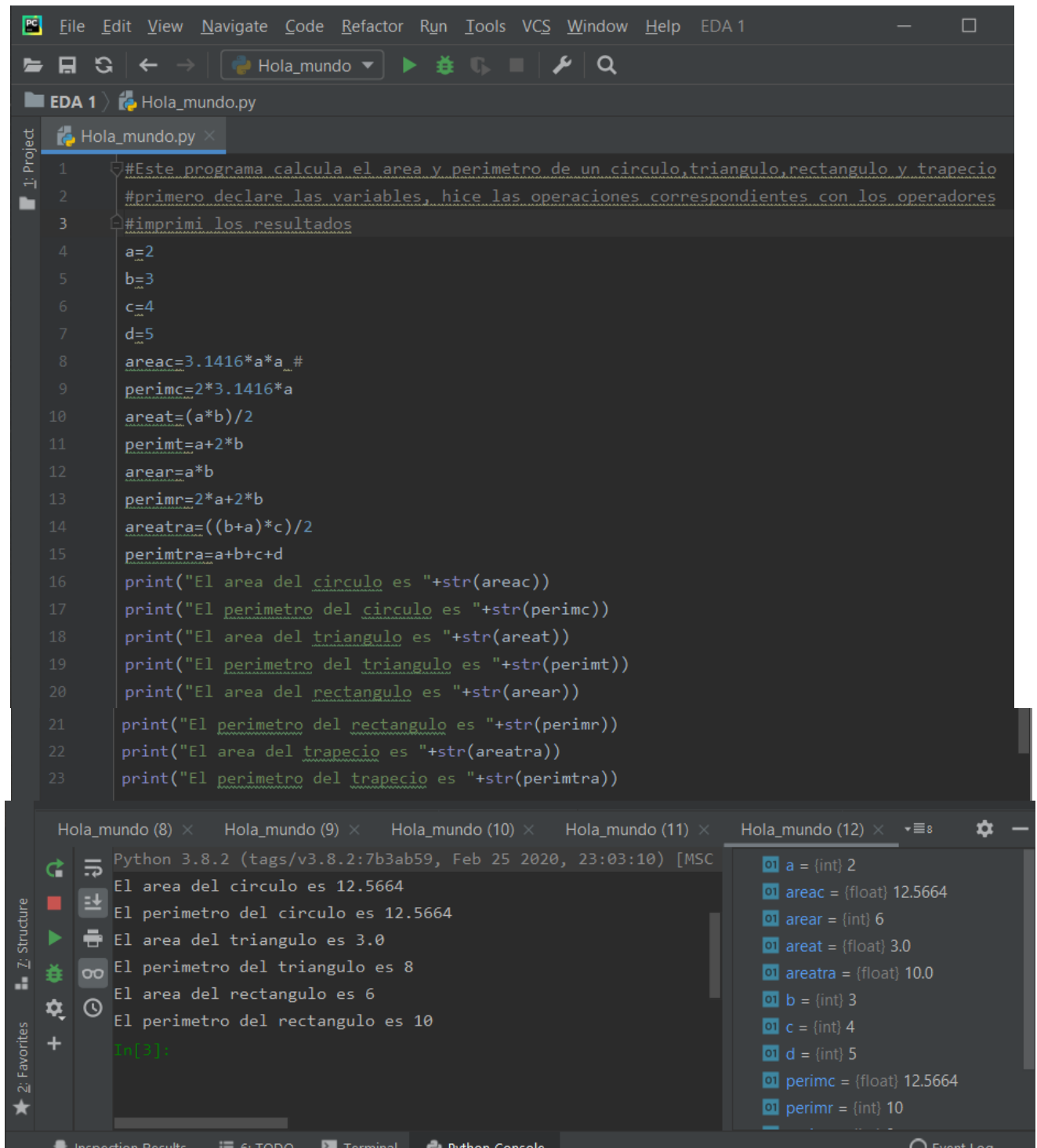
The screenshot shows an IDE window titled 'EDA 1' with a file named 'Hola_mundo.py'. The code defines two functions: `imprime_nombre` and `cuadrado`. The `imprime_nombre` function prints 'hola ' followed by the argument. The `cuadrado` function returns the square of the argument. The script sets `x = 5` and prints the result of `cuadrado(x)` using `format`.

```
1 #Las funciones pueden recibir n número de parámetros, no se necesita indicar el tipo
2 def imprime_nombre(nombre):
3     print("hola "+nombre)
4     imprime_nombre("Abraham")
5 def cuadrado(x):
6     return x**2
7 x = 5
8 #La función format() sirve para convertir los parámetros que recibe, en cadenas; éstos valores
9 #por las llaves de la cadena.
10 print("El cuadrado de {} es {}".format(x, cuadrado(x)))
11
```

The bottom panel shows the execution output in the 'Python Console' tab. It displays the output of the script: 'Python 3.8.2 (tags/v3.8.2:7b3ab59, Feb 25 2020, 23:03:10) [MSC...]' and 'hola Abraham'. The output of the `print` statement is 'El cuadrado de 5 es 25'. The 'In[3]:' prompt is visible.

PEP 8: expected 2 blank lines after class or function definition, found 0 4:24 CRLF UTF-8 4 spaces <No interpreter>

EJERCICIO



```
File Edit View Navigate Code Refactor Run Tools VCS Window Help EDA 1
Hola_mundo
EDA 1 > Hola_mundo.py
1: Project
1 #Este programa calcula el area y perimetro de un circulo, triangulo, rectangulo y trapecio
2 #primero declare las variables, hice las operaciones correspondientes con los operadores
3 #imprimi los resultados
4 a=2
5 b=3
6 c=4
7 d=5
8 areac=3.1416*a*a #
9 perimc=2*3.1416*a
10 areat=(a*b)/2
11 perimt=a+2*b
12 arear=a*b
13 perimr=2*a+2*b
14 areatra=((b+a)*c)/2
15 perimtra=a+b+c+d
16 print("El area del circulo es "+str(areac))
17 print("El perimetro del circulo es "+str(perimc))
18 print("El area del triangulo es "+str(areat))
19 print("El perimetro del triangulo es "+str(perimt))
20 print("El area del rectangulo es "+str(arear))
21 print("El perimetro del rectangulo es "+str(perimr))
22 print("El area del trapecio es "+str(areatra))
23 print("El perimetro del trapecio es "+str(perimtra))

Python 3.8.2 (tags/v3.8.2:7b3ab59, Feb 25 2020, 23:03:10) [MSC
El area del circulo es 12.5664
El perimetro del circulo es 12.5664
El area del triangulo es 3.0
El perimetro del triangulo es 8
El area del rectangulo es 6
El perimetro del rectangulo es 10
In[3]:
a = (int) 2
areac = (float) 12.5664
arear = (int) 6
areat = (float) 3.0
areatra = (float) 10.0
b = (int) 3
c = (int) 4
d = (int) 5
perimc = (float) 12.5664
perimr = (int) 10
```

Conclusión

Esta fue una buena práctica, ya que en pocas palabras aplicamos lo aprendido en C a Python, de primera instancia Python suena difícil; pero ya una vez entendiendo como funciona, se te hace muy fácil, me atrevo a decir que es más fácil que C, No lo hice en Jupyter y decidí hacerlo en pycharm.

Bibliografía

<http://lcp02.fi-b.unam.mx/>

<https://www.python.org/>

<http://docs.python.org.ar/tutorial/3/real-index.html>