

---

---

# Manual to run the Python code

Olatz Perez de Viñaspre Garralda  
Ander Soraluze Irureta

---

---

---

# Operating System

We recommend you to install Ubuntu (or other Linux distribution) in your computer:

- In a new partition
- In a virtual machine (VMware, VirtualBox,...)

Plenty tutorials on the Internet

---

---

---

# Environment (IDE)

You can use the IDE of your choice. The lecturers will be using **Visual Studio Code** (installed in the classroom's machines)

---

EXPLORER

TEORIAKO\_KODEA

02-01-Hello\_world.py

02-02-Hello\_name.py

02-03-function\_hello.py

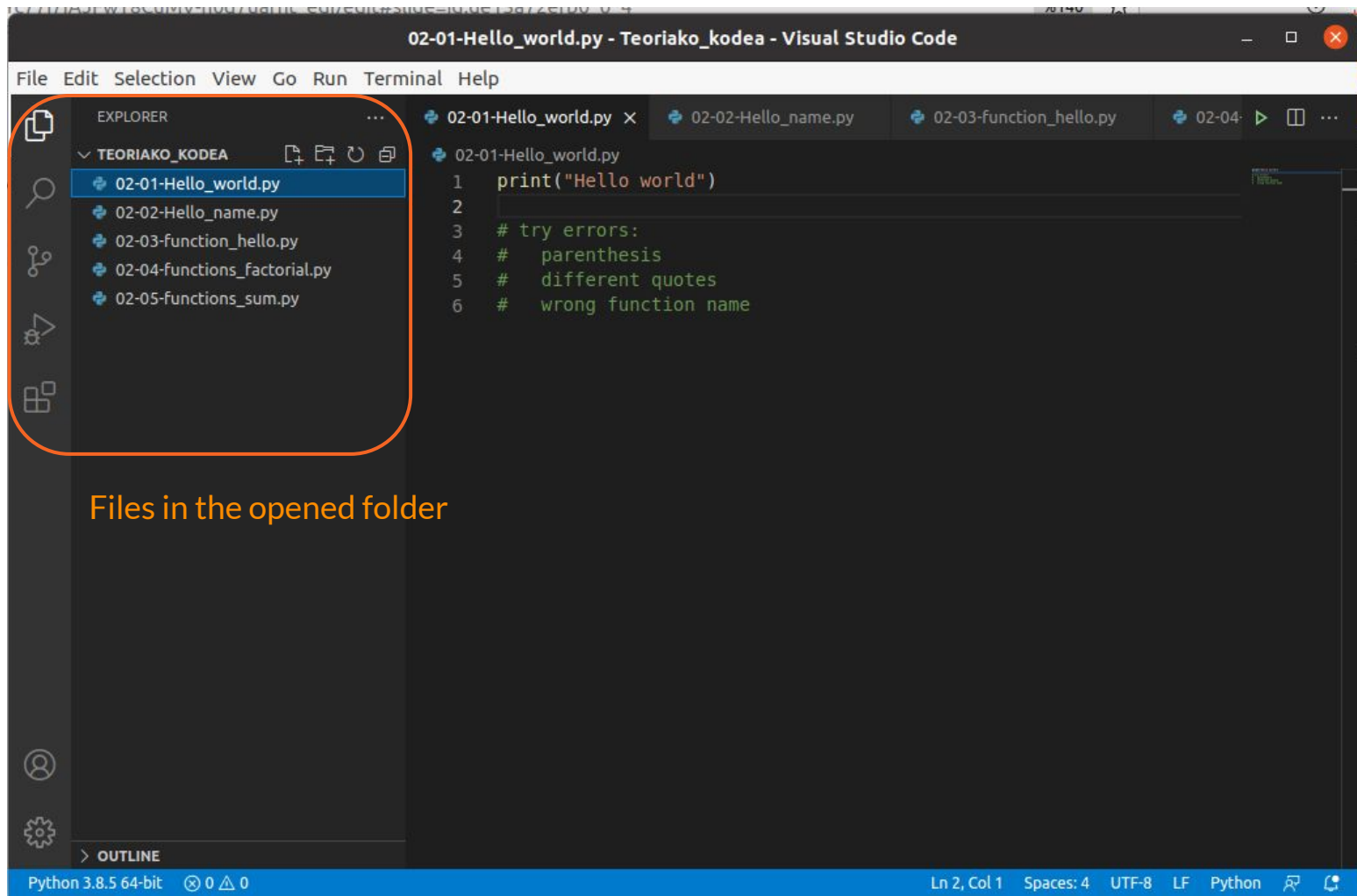
02-04-functions\_factorial.py

02-05-functions\_sum.py

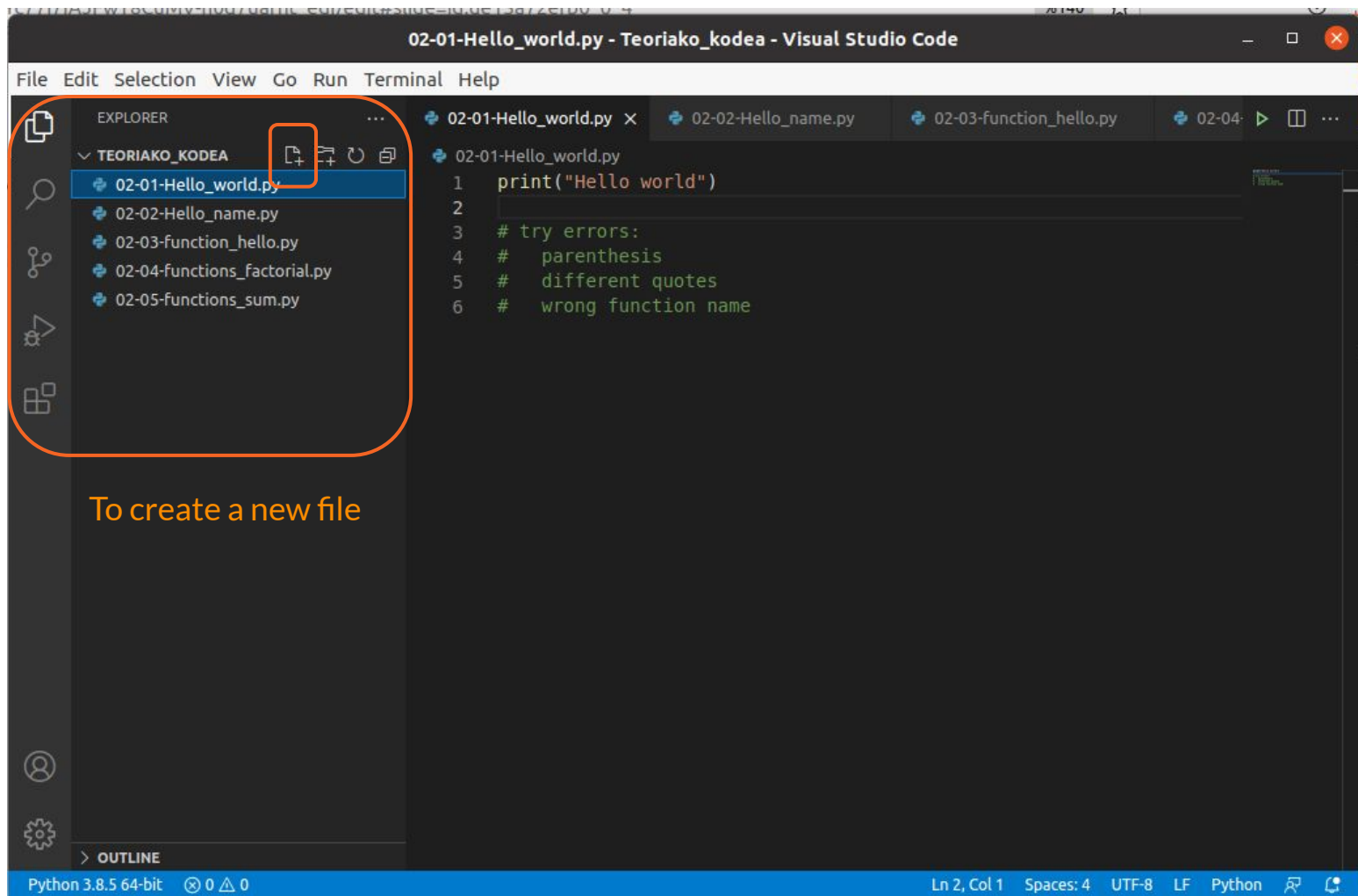
02-01-Hello\_world.py

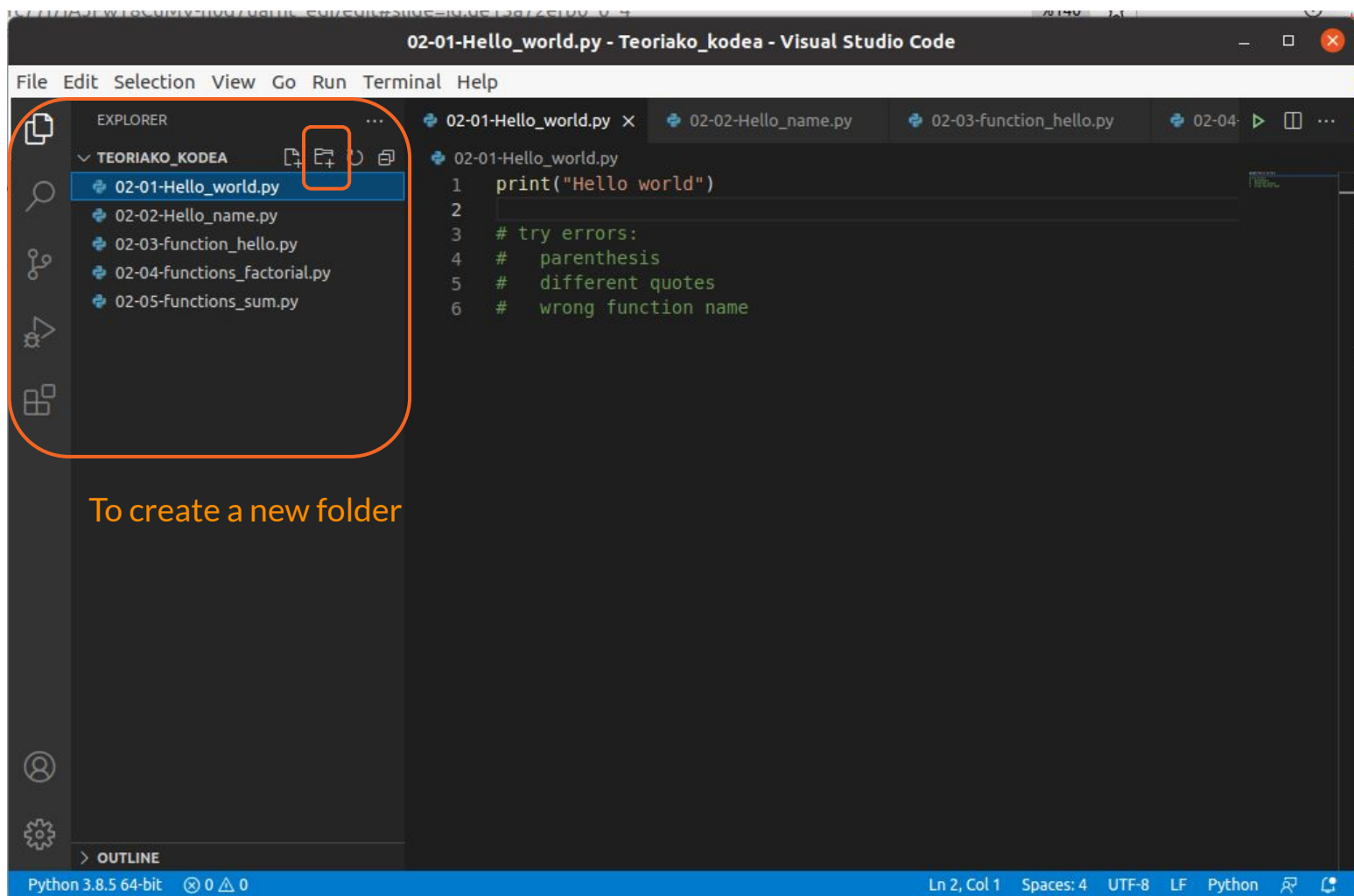
```
1 print("Hello world")
2
3 # try errors:
4 #   parenthesis
5 #   different quotes
6 #   wrong function name
```

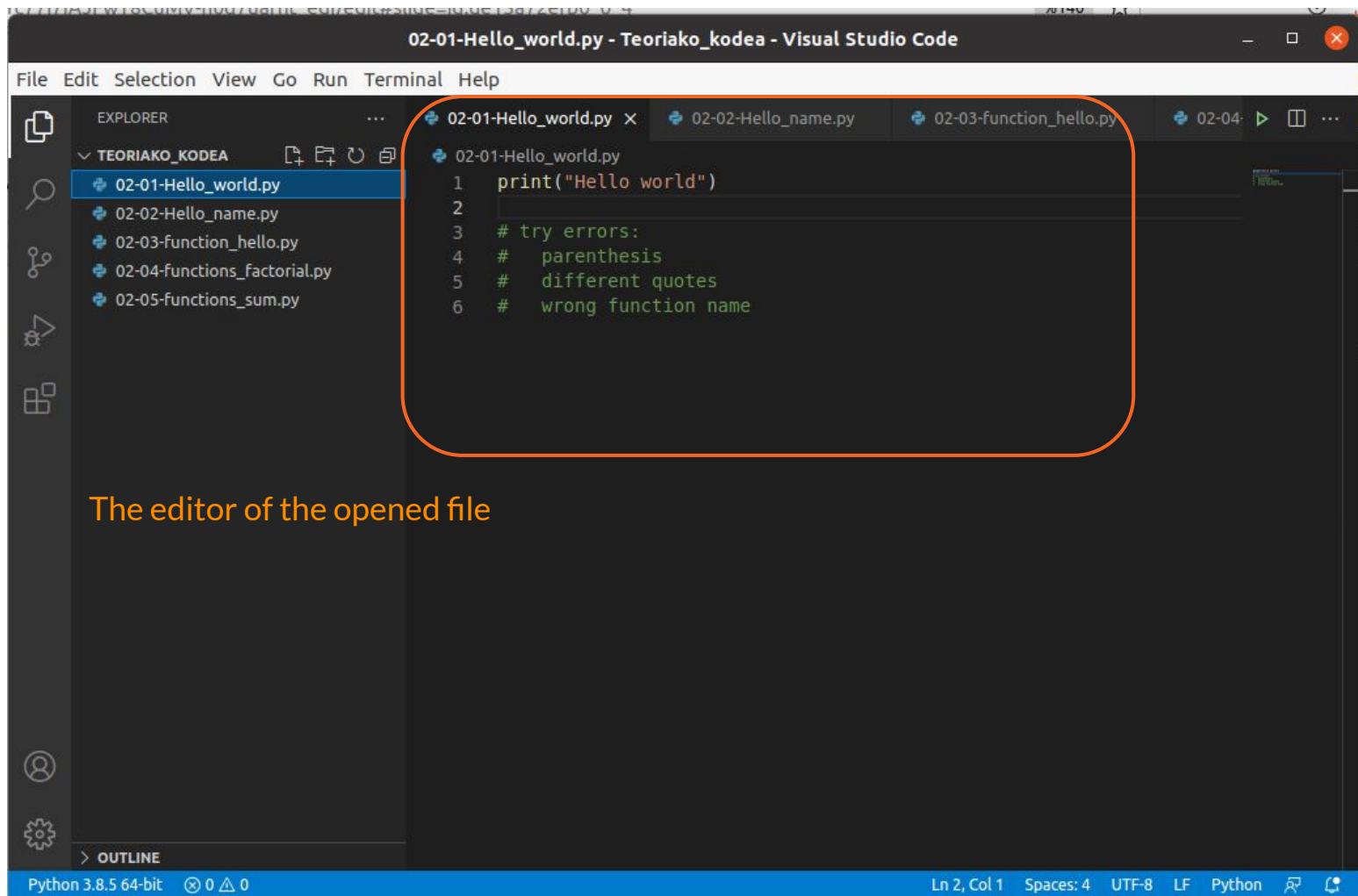
&gt; OUTLINE



Files in the opened folder

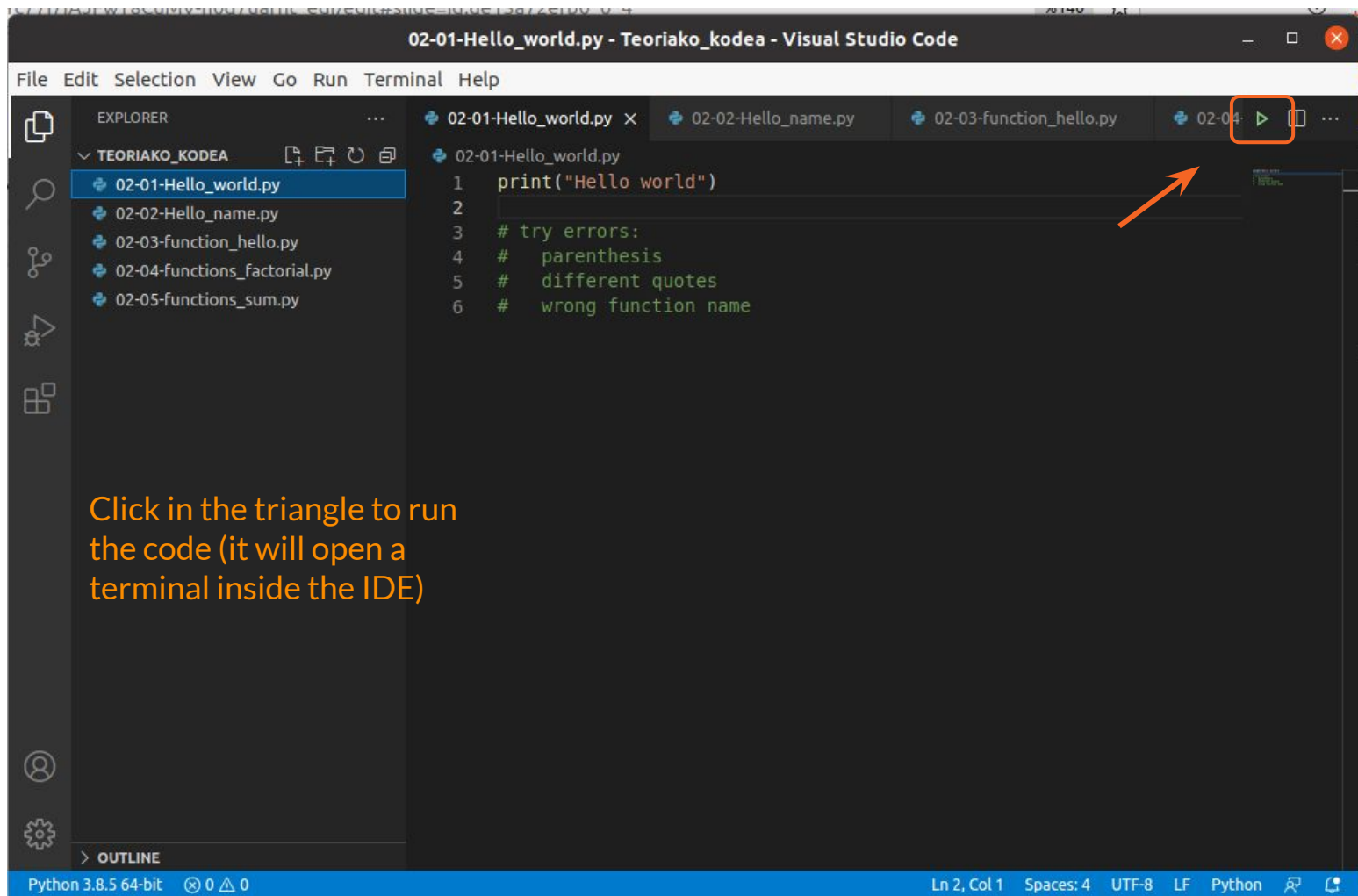






The editor of the opened file







EXPLORER

TEORIAKO\_KODEA

- 02-01-Hello\_world.py
- 02-02-Hello\_name.py
- 02-03-function\_hello.py
- 02-04-functions\_factorial.py
- 02-05-functions\_sum.py



&gt; OUTLINE

02-01-Hello\_world.py x

02-02-Hello\_name.py

02-03-function\_hello.py

02-04-

▶ □ ...

02-01-Hello\_world.py

```
1 print("Hello world")
2
3 # try errors:
4 #   parenthesis
5 #   different quotes
6 #   wrong function name
```

PROBLEMS OUTPUT TERMINAL DEBUG CONSOLE

Python + v ^ x

```
/usr/bin/python3 "/home/olatz/Nextcloud/Irakaskuntza/Introduction to Programming/Teoriako_kodea/02-01-Hello_world.py"
olatz@U111275:~/Nextcloud/Irakaskuntza/Introduction to Programming/Teoriako_kodea$ /usr/bin/python3 "/home/olatz/Nextcloud/Irakaskuntza/Introduction to Programming/Teoriako_kodea/02-01-Hello_world.py"
Hello world
olatz@U111275:~/Nextcloud/Irakaskuntza/Introduction to Programming/Teoriako_kodea$
```

The output of your code after running it in the terminal

---

**Running the code in an  
ordinary terminal (Linux)**

---

Jarduerak



Terminala





olatz@U111275: ~



olatz@U111275:~\$



olatz@U111275: ~



```
olatz@U111275:~$ cd Nextcloud/Irakaskuntza/Introduction\ to\ Programming/Teoriak  
o_kodea/
```



olatz@U111275: ~/Nextcloud/Irakaskuntza/Introduction to P...



```
olatz@U111275:~$ cd Nextcloud/Irakaskuntza/Introduction\ to\ Programming/Teoriak  
o_kodea/
```

```
olatz@U111275:~/Nextcloud/Irakaskuntza/Introduction to Programming/Teoriako_kode  
a$ ls
```

```
02-Basic_data_and_operations    05-Dictionaries_and_files
```

```
03-Loops                        06-Advanced_files_and_objects
```

```
04-Strings_and_lists           data
```

```
olatz@U111275:~/Nextcloud/Irakaskuntza/Introduction to Programming/Teoriako_kode  
a$
```



olatz@U111275: ~/Nextcloud/Irakaskuntza/Introduction to P...



```
olatz@U111275:~$ cd Nextcloud/Irakaskuntza/Introduction\ to\ Programming/Teoriako_kodea/
```

```
olatz@U111275:~/Nextcloud/Irakaskuntza/Introduction to Programming/Teoriako_kodea$ ls
```

```
02-Basic_data_and_operations    05-Dictionaries_and_files
03-Loops                        06-Advanced_files_and_objects
04-Strings_and_lists           data
```

```
olatz@U111275:~/Nextcloud/Irakaskuntza/Introduction to Programming/Teoriako_kodea$ ls 02-Basic_data_and_operations/
```

```
02-01-Hello_world.py           02-08-conditionals-rain2.py
02-02-Hello_name.py            02-09-booleans.py
02-03-function_hello.py        02-10-booleansNumbers.py
02-04-functions_factorial.py    02-11-nestedConditionalsImproved.py
02-05-functions_sum.py         02-11-nestedConditionals.py
02-06-functions_return.py      02-12-manyConditionals.py
02-07-conditionals-rain.py     02-13-conditionals-weekdays.py
```

```
olatz@U111275:~/Nextcloud/Irakaskuntza/Introduction to Programming/Teoriako_kodea$
```





olatz@U111275: ~/Nextcloud/Irakaskuntza/Introduction to P...



```
olatz@U111275:~$ cd Nextcloud/Irakaskuntza/Introduction\ to\ Programming/Teoriako_kodea/
```

```
olatz@U111275:~/Nextcloud/Irakaskuntza/Introduction to Programming/Teoriako_kodea$ ls
```

```
02-Basic_data_and_operations  05-Dictionaries_and_files
03-Loops                      06-Advanced_files_and_objects
04-Strings_and_lists          data
```

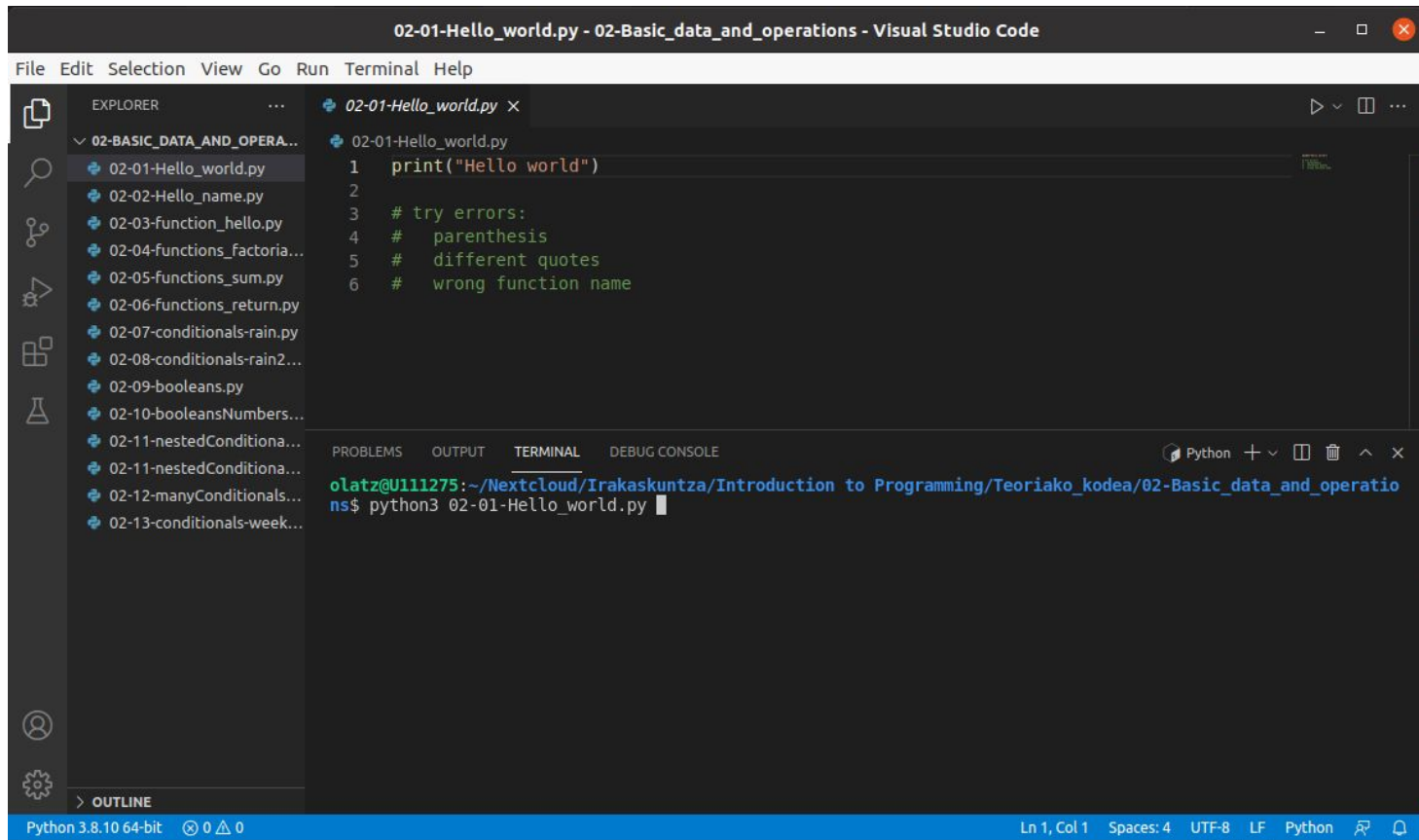
```
olatz@U111275:~/Nextcloud/Irakaskuntza/Introduction to Programming/Teoriako_kodea$ ls 02-Basic_data_and_operations/
```

```
02-01-Hello_world.py          02-08-conditionals-rain2.py
02-02-Hello_name.py           02-09-booleans.py
02-03-function_hello.py       02-10-booleansNumbers.py
02-04-functions_factorial.py   02-11-nestedConditionalsImproved.py
02-05-functions_sum.py        02-11-nestedConditionals.py
02-06-functions_return.py     02-12-manyConditionals.py
02-07-conditionals-rain.py     02-13-conditionals-weekdays.py
```

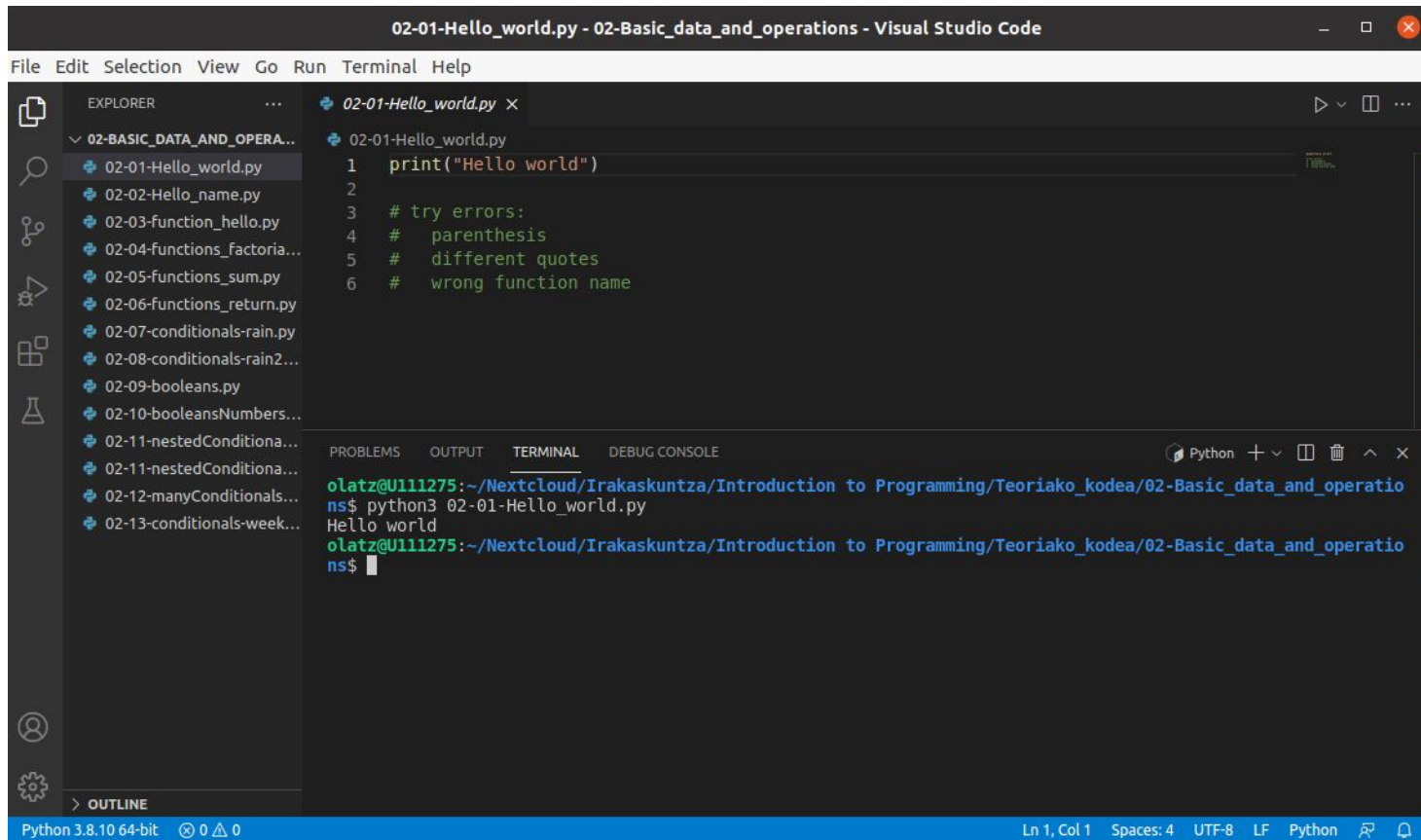
```
olatz@U111275:~/Nextcloud/Irakaskuntza/Introduction to Programming/Teoriako_kodea$ python3 02-Basic_data_and_operations/02-01-Hello_world.py
Hello world
```

```
olatz@U111275:~/Nextcloud/Irakaskuntza/Introduction to Programming/Teoriako_kodea$
```

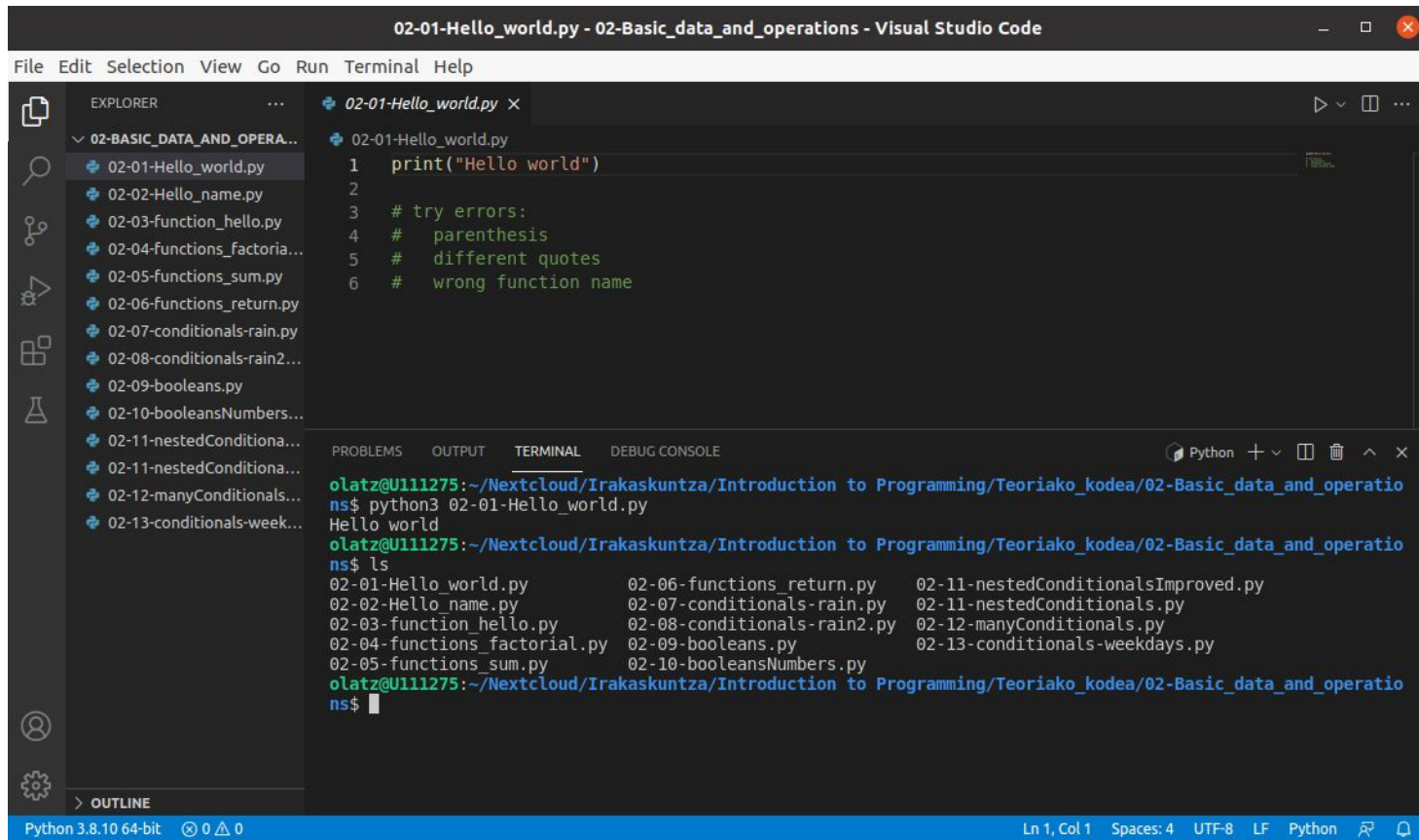
# Ordinary terminal == VSC terminal



# Ordinary terminal == VSC terminal



# Ordinary terminal == VSC terminal



The screenshot shows the Visual Studio Code interface with a file named `02-01-Hello_world.py` open in the editor. The file contains the following Python code:

```
1 print("Hello world")
2
3 # try errors:
4 #   parenthesis
5 #   different quotes
6 #   wrong function name
```

The Explorer sidebar on the left shows a project structure with files named `02-01-Hello_world.py` through `02-13-conditionals-weekdays.py`. The Terminal panel at the bottom shows the execution of the script using `python3` and the output of the `ls` command, which lists all files in the directory.

```
olatz@U111275:~/Nextcloud/Irakaskuntza/Introduction to Programming/Teoriako kodea/02-Basic_data_and_operatio
ns$ python3 02-01-Hello_world.py
Hello world
olatz@U111275:~/Nextcloud/Irakaskuntza/Introduction to Programming/Teoriako kodea/02-Basic_data_and_operatio
ns$ ls
02-01-Hello_world.py      02-06-functions_return.py    02-11-nestedConditionalsImproved.py
02-02-Hello_name.py      02-07-conditionals-rain.py    02-11-nestedConditionals.py
02-03-function_hello.py  02-08-conditionals-rain2.py   02-12-manyConditionals.py
02-04-functions_factorial.py 02-09-booleans.py            02-13-conditionals-weekdays.py
02-05-functions_sum.py     02-10-booleansNumbers.py
olatz@U111275:~/Nextcloud/Irakaskuntza/Introduction to Programming/Teoriako kodea/02-Basic_data_and_operatio
ns$
```

The status bar at the bottom indicates the Python version is 3.8.10 64-bit, and the current line is Ln 1, Col 1.

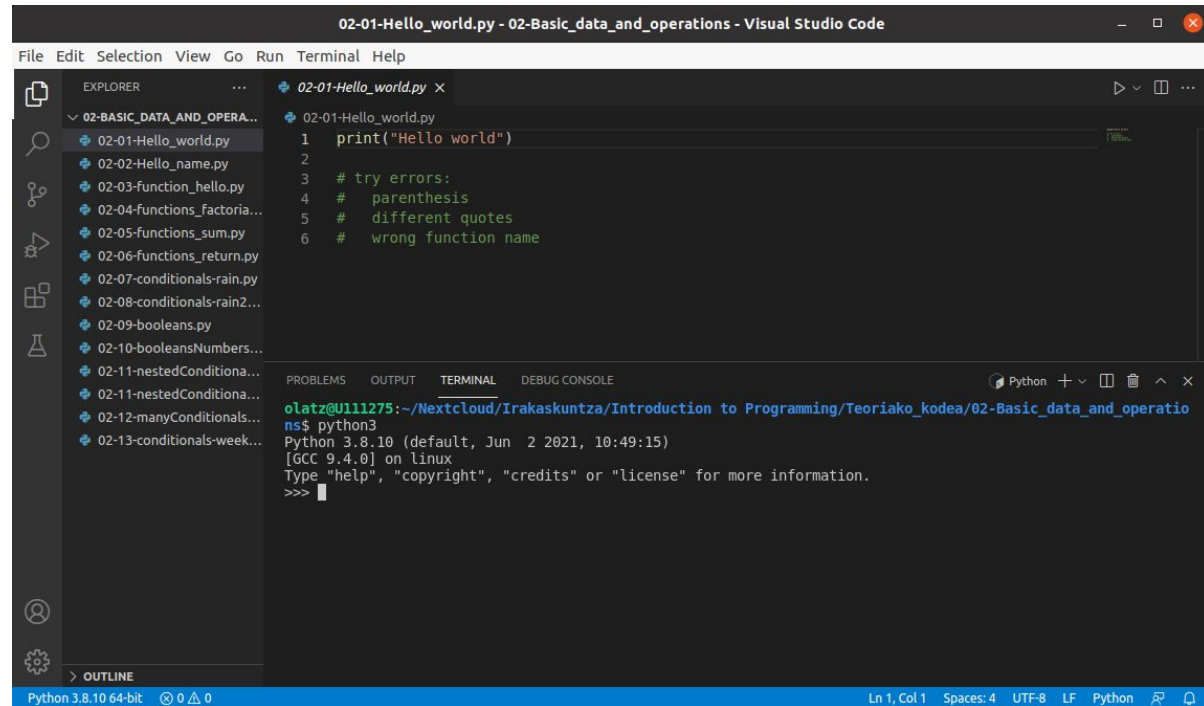
---

**Using the terminal to check  
python pieces of code**

---

# Check code

You can run the python interpreter by  
executing "python3" command



The screenshot shows the Visual Studio Code interface. The title bar reads "02-01-Hello\_world.py - 02-Basic\_data\_and\_operations - Visual Studio Code". The Explorer sidebar on the left shows a project structure with a folder "02-BASIC\_DATA\_AND\_OPERA..." containing several Python files, with "02-01-Hello\_world.py" selected. The editor window displays the content of "02-01-Hello\_world.py":

```
1 print("Hello world")
2
3 # try errors:
4 # parenthesis
5 # different quotes
6 # wrong function name
```

The TERMINAL panel at the bottom shows a shell prompt with the following output:

```
olatz@U111275:~/Nextcloud/Irakaskuntza/Introduction to Programming/Teoriako kodea/02-Basic_data_and_operatio
ns$ python3
Python 3.8.10 (default, Jun  2 2021, 10:49:15)
[GCC 9.4.0] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>> 
```

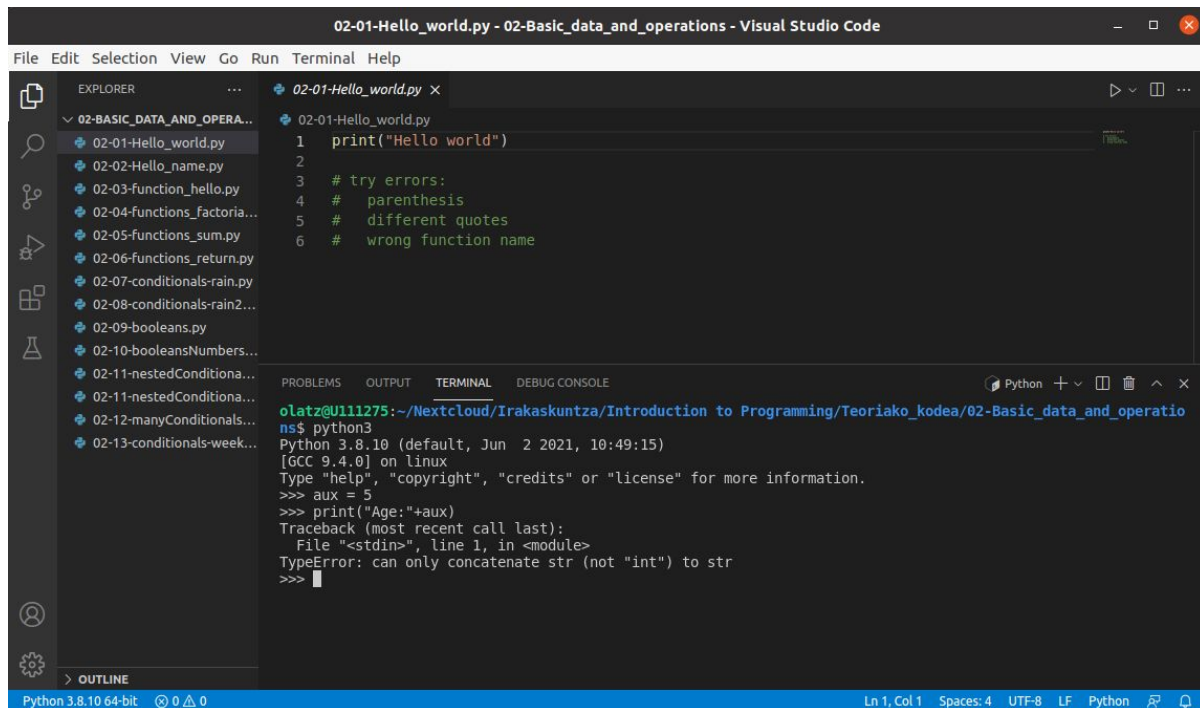
The status bar at the bottom indicates "Python 3.8.10 64-bit" and "Ln 1, Col 1".



# Check code

You can run the python interpreter by  
executing "python3" command

Now, you can write any code to check  
it's behaviour without having to run the  
whole program.



The screenshot shows the Visual Studio Code interface. The Explorer panel on the left lists files under the folder "02-BASIC\_DATA\_AND\_OPERA...". The file "02-01-Hello\_world.py" is selected. The editor window shows the code for "02-01-Hello\_world.py":

```
1 print("Hello world")
2
3 # try errors:
4 #   parenthesis
5 #   different quotes
6 #   wrong function name
```

The TERMINAL panel at the bottom shows the output of running the code:

```
olatz@U111275:~/Nextcloud/Irakaskuntza/Introduction to Programming/Teoriako_kodea/02-Basic_data_and_operatio
ns$ python3
Python 3.8.10 (default, Jun  2 2021, 10:49:15)
[GCC 9.4.0] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>> aux = 5
>>> print("Age:"+aux)
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
TypeError: can only concatenate str (not "int") to str
>>>
```

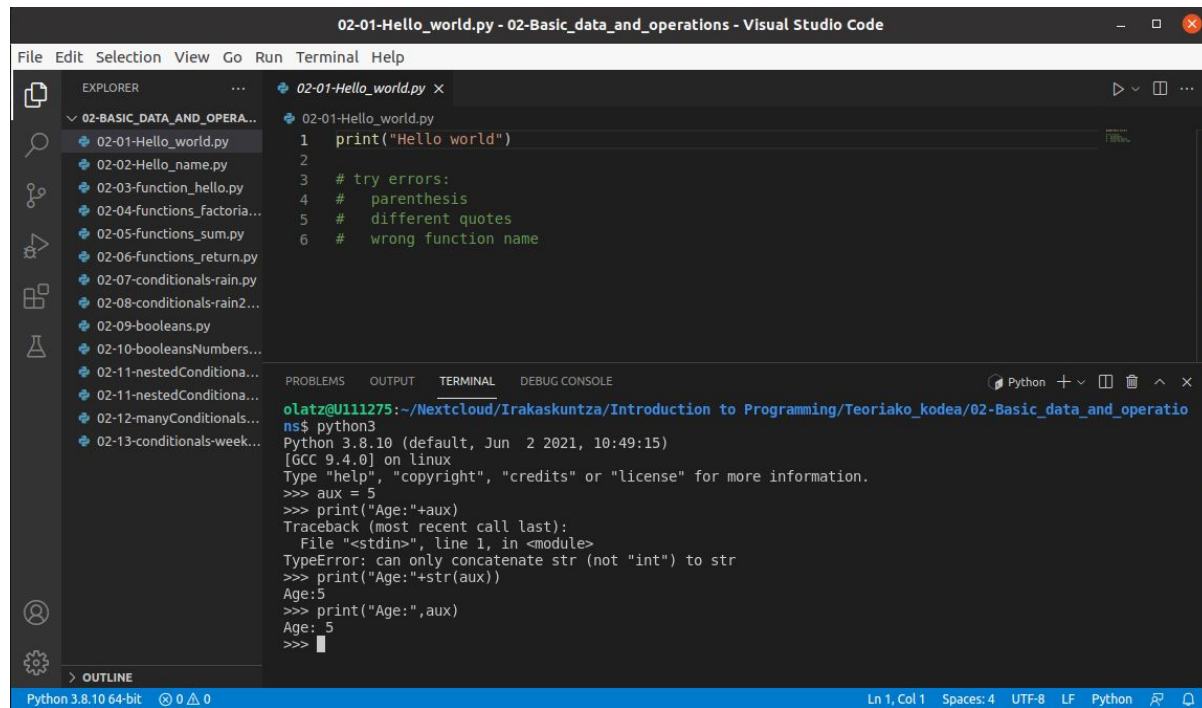
The status bar at the bottom indicates "Python 3.8.10 64-bit" and "Ln 1, Col 1".

# Check code

You can run the python interpreter by executin “python3” command

Now, you can write any code to check it's behaviour without having to run the whole program.

It is especially interesting to check built-in functions



The screenshot shows the Visual Studio Code interface. The Explorer panel on the left lists files under the folder '02-BASIC\_DATA\_AND\_OPERA...'. The file '02-01-Hello\_world.py' is selected. The editor window shows the following code:

```
02-01-Hello_world.py X
02-01-Hello_world.py
1 print("Hello world")
2
3 # try errors:
4 # parenthesis
5 # different quotes
6 # wrong function name
```

The TERMINAL panel at the bottom shows the output of running the Python interpreter:

```
Python 3.8.10 (default, Jun 2 2021, 10:49:15)
[GCC 9.4.0] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>> aux = 5
>>> print("Age:"+aux)
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
TypeError: can only concatenate str (not "int") to str
>>> print("Age:"+str(aux))
Age:5
>>> print("Age:",aux)
Age: 5
>>>
```

The status bar at the bottom indicates 'Python 3.8.10 64-bit' and 'Ln 1, Col 1 Spaces: 4 UTF-8 LF Python'.



---

# Ctrl+d

To run out the python interpreter

Or you can call the `exit()` function

---

—

# Important shortcuts to work with any terminal

---

# Main shortcuts to learn

## Autocomplete file names

You can use the tabular key to autocomplete the filename

To do so, you should start writing the filename and if it uniq, it will be autocompleted.

If it is not uniq, you can press the tab key again and will show you the options.

## Repeat previous commands

You can use the upwards arrow key to recover previous commands.

You can go through the history commands with the upwards and downwards keys.

You can even edit the command once you get to it.

---



**Ready** to start working!