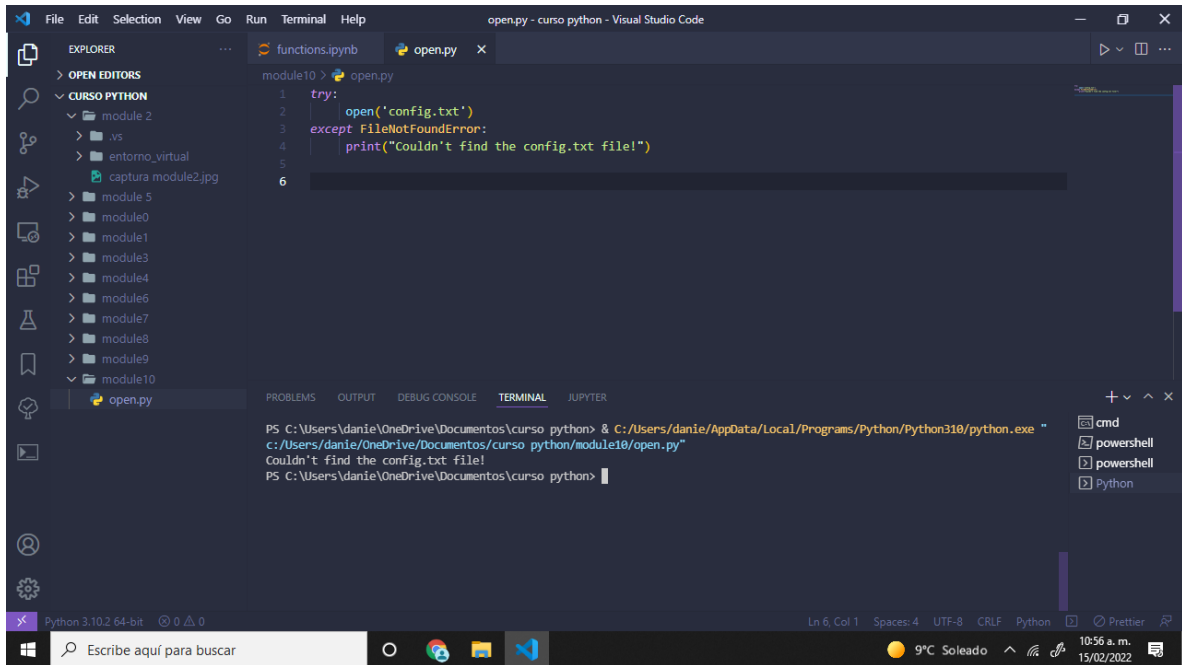


Excepciones (Control try-except)

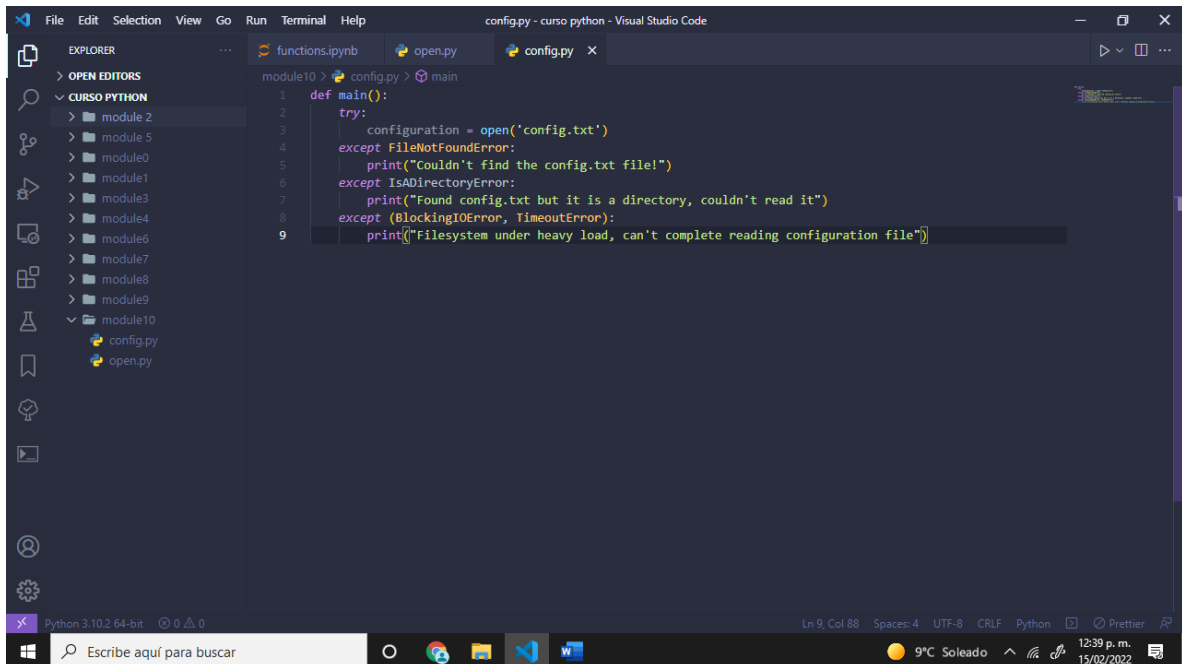


The screenshot shows the Visual Studio Code interface with the file explorer on the left displaying a project structure under 'CURSO PYTHON'. The main editor window shows the file 'open.py' with the following code:

```
1 try:
2     open('config.txt')
3 except FileNotFoundError:
4     print("Couldn't find the config.txt file!")
5
6
```

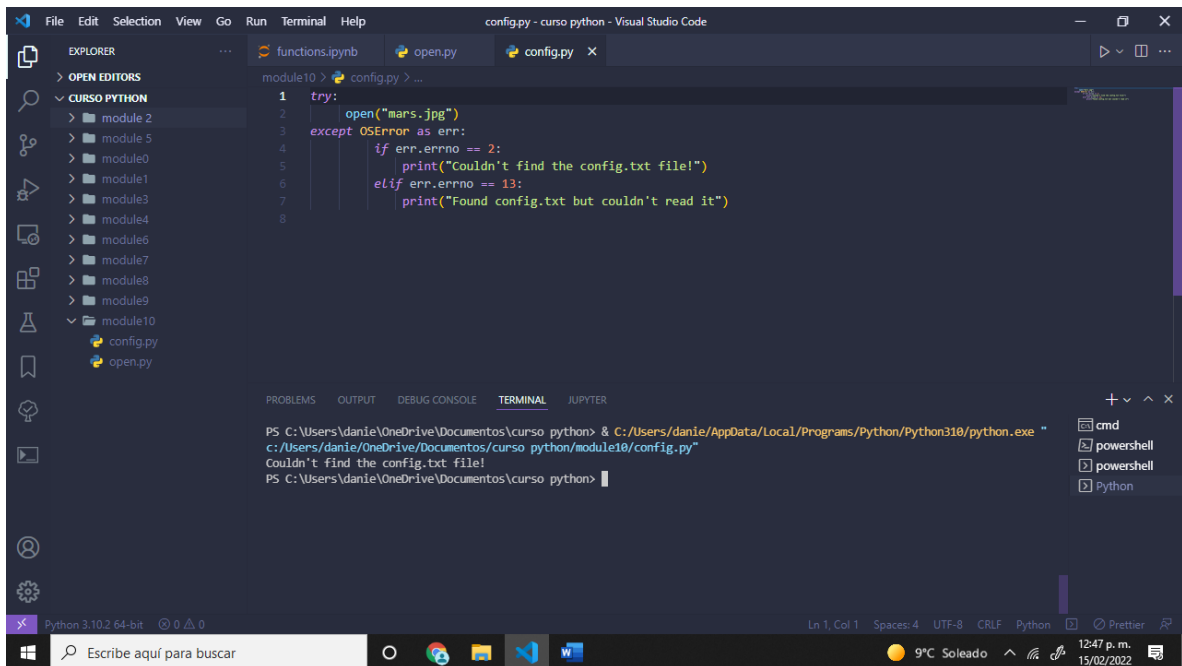
The terminal at the bottom shows the command prompt output:

```
PS C:\Users\danie\OneDrive\Documentos\curso python> & C:\Users\danie\AppData\Local\Programs\Python\Python310\python.exe "c:\Users\danie\OneDrive\Documentos\curso python\module10\open.py"
Couldn't find the config.txt file!
PS C:\Users\danie\OneDrive\Documentos\curso python>
```



The screenshot shows the Visual Studio Code interface with the file explorer on the left displaying a project structure under 'CURSO PYTHON'. The main editor window shows the file 'config.py' with the following code:

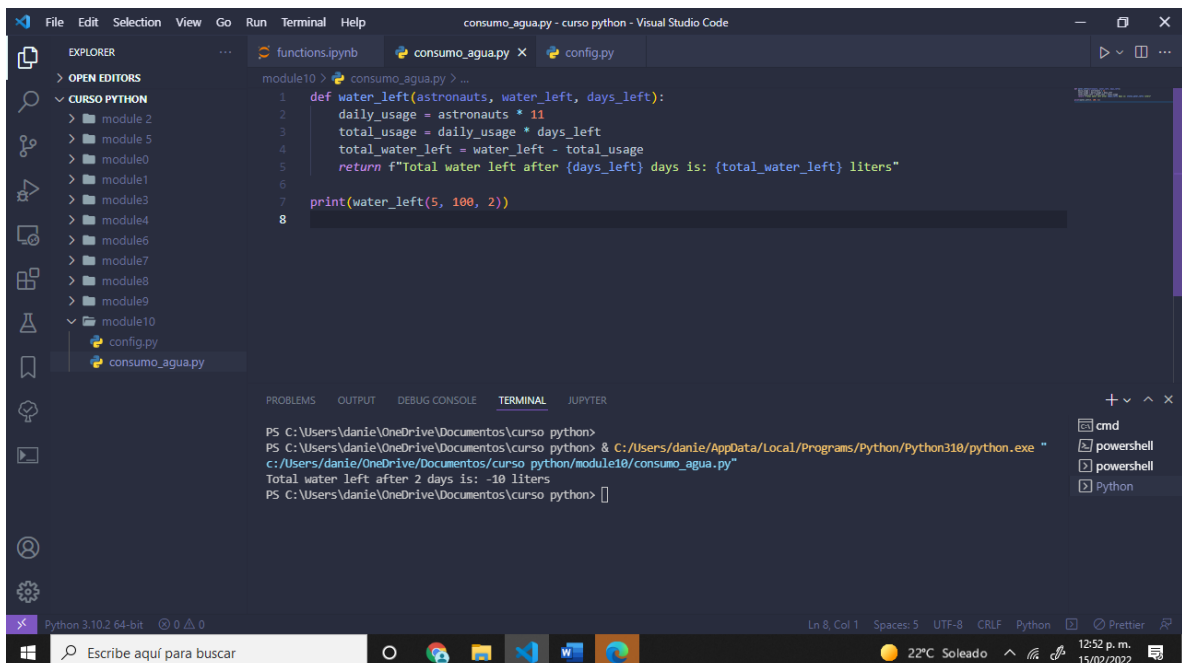
```
1 def main():
2     try:
3         configuration = open('config.txt')
4     except FileNotFoundError:
5         print("Couldn't find the config.txt file!")
6     except IsADirectoryError:
7         print("Found config.txt but it is a directory, couldn't read it")
8     except (BlockingIOError, TimeoutError):
9         print("Filesystem under heavy load, can't complete reading configuration file")
```



```
1 try:
2     open("mars.jpg")
3 except OSError as err:
4     if err.errno == 2:
5         print("Couldn't find the config.txt file!")
6     elif err.errno == 13:
7         print("Found config.txt but couldn't read it")
8
```

```
PS C:\Users\danie\OneDrive\Documentos\curso python> & C:\Users\danie\AppData\Local\Programs\Python\Python310\python.exe "
c:\Users\danie\OneDrive\Documentos\curso python\module10\config.py"
Couldn't find the config.txt file!
PS C:\Users\danie\OneDrive\Documentos\curso python>
```

Sobranche de agua de astronautas después de consumo por día



```
1 def water_left(astronauts, water_left, days_left):
2     daily_usage = astronauts * 11
3     total_usage = daily_usage * days_left
4     total_water_left = water_left - total_usage
5     return f"Total water left after {days_left} days is: {total_water_left} liters"
6
7 print(water_left(5, 100, 2))
8
```

```
PS C:\Users\danie\OneDrive\Documentos\curso python>
PS C:\Users\danie\OneDrive\Documentos\curso python> & C:\Users\danie\AppData\Local\Programs\Python\Python310\python.exe "
c:\Users\danie\OneDrive\Documentos\curso python\module10\consumo_agua.py"
Total water left after 2 days is: -10 liters
PS C:\Users\danie\OneDrive\Documentos\curso python>
```

The image shows a Visual Studio Code window with the file explorer on the left, a code editor in the center, and a terminal at the bottom. The code editor displays a Python script named `consumo_agua.py` with the following content:

```
1 def water_left(astronauts, water_left, days_left):
2     for argument in [astronauts, water_left, days_left]:
3         try:
4             # If argument is an int, the following operation will work
5             argument / 10
6         except TypeError:
7             # TypeError will be raised only if it isn't the right type
8             # Raise the same exception but with a better error message
9             raise TypeError(f"All arguments must be of type int, but received: '{argument}'")
10
11     daily_usage = astronauts * 11
12     total_usage = daily_usage * days_left
13     total_water_left = water_left - total_usage
14     if total_water_left < 0:
15         raise RuntimeError(f"There is not enough water for {astronauts} astronauts after {days_left} days!")
16     return f"Total water left after {days_left} days is: {total_water_left} liters"
17
18 print(water_left(5, 100, 2))
```

The terminal shows the execution of the script, which results in a `RuntimeError` because the `print` statement calls `water_left(5, 100, 2)`, but the function expects three arguments: `astronauts`, `water_left`, and `days_left`. The traceback indicates the error occurred in `water_left` at line 14.

```
PS C:\Users\danie\OneDrive\Documents\curso python> & C:\Users\danie\AppData\Local\Programs\Python\Python310\python.exe "
c:\Users\danie\OneDrive\Documents\curso python\module10\consumo_agua.py"
Traceback (most recent call last):
  File "c:\Users\danie\OneDrive\Documents\curso python\module10\consumo_agua.py", line 17, in <module>
    print(water_left(5, 100, 2))
  File "c:\Users\danie\OneDrive\Documents\curso python\module10\consumo_agua.py", line 14, in water_left
    raise RuntimeError(f"There is not enough water for {astronauts} astronauts after {days_left} days!")
RuntimeError: There is not enough water for 5 astronauts after 2 days!
PS C:\Users\danie\OneDrive\Documents\curso python>
```

The status bar at the bottom indicates the Python version is 3.10.2 64-bit, the file encoding is UTF-8, and the line/col is Ln 9, Col 84.