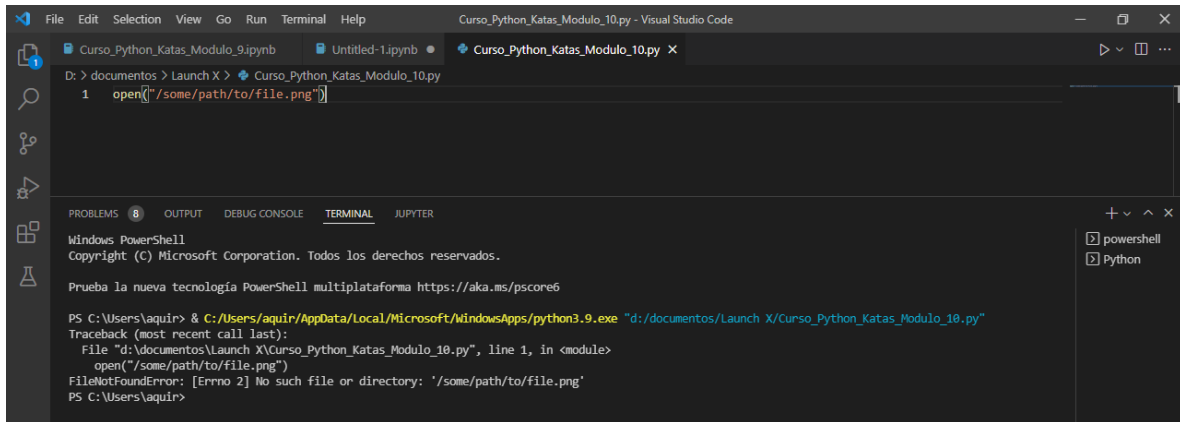


Modulo 10 Curso de Python. Manejo de errores

Katas Modulo 10



```
File Edit Selection View Go Run Terminal Help
Curso_Python_Katas_Modulo_10.py - Visual Studio Code

D:\> documentos > Launch X > Curso_Python_Katas_Modulo_10.py
1 open("/some/path/to/file.png")

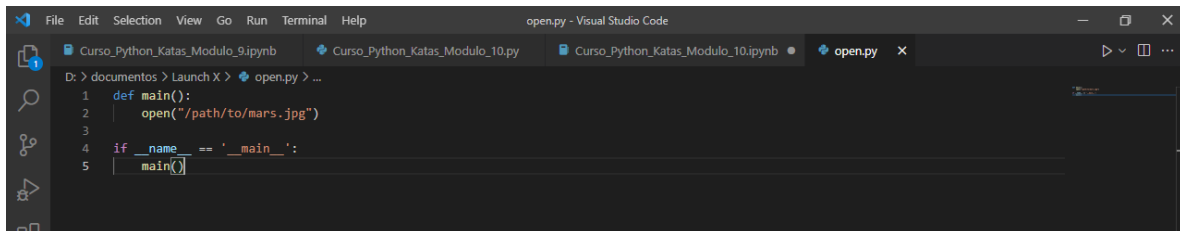
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL JUPYTER

Windows PowerShell
Copyright (C) Microsoft Corporation. Todos los derechos reservados.

Prueba la nueva tecnologia PowerShell multiplataforma https://aka.ms/pscore6

PS C:\Users\aquir> & C:\Users\aquir\AppData\Local\Microsoft\WindowsApps\python3.9.exe "d:\documentos\Launch X\Curso_Python_Katas_Modulo_10.py"
Traceback (most recent call last):
  File "d:\documentos\Launch X\Curso_Python_Katas_Modulo_10.py", line 1, in <module>
    open("/some/path/to/file.png")
FileNotFoundError: [Errno 2] No such file or directory: '/some/path/to/file.png'
PS C:\Users\aquir>
```

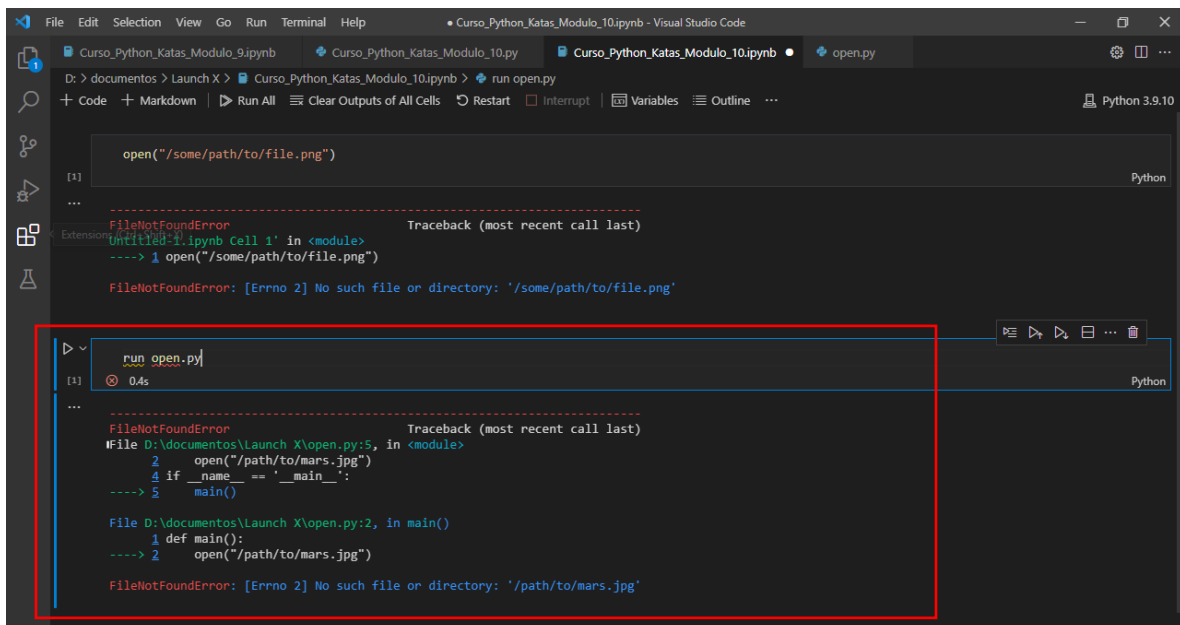
Creamos el archivo open.py



```
File Edit Selection View Go Run Terminal Help
open.py - Visual Studio Code

D:\> documentos > Launch X > open.py > ...
1 def main():
2     open("/path/to/mars.jpg")
3
4 if __name__ == '__main__':
5     main()
```

Y al ejecutarlo nos muestra el siguiente error



```
File Edit Selection View Go Run Terminal Help
Curso_Python_Katas_Modulo_10.ipynb - Visual Studio Code

D:\> documentos > Launch X > Curso_Python_Katas_Modulo_10.ipynb > run open.py
+ Code + Markdown | Run All | Clear Outputs of All Cells | Restart | Interrupt | Variables | Outline ... Python 3.9.10

open("/some/path/to/file.png")
[1]

-----
FileNotFoundError
Traceback (most recent call last)
Untitled-1.ipynb Cell 1: in <module>
----> 1 open("/some/path/to/file.png")

FileNotFoundError: [Errno 2] No such file or directory: '/some/path/to/file.png'

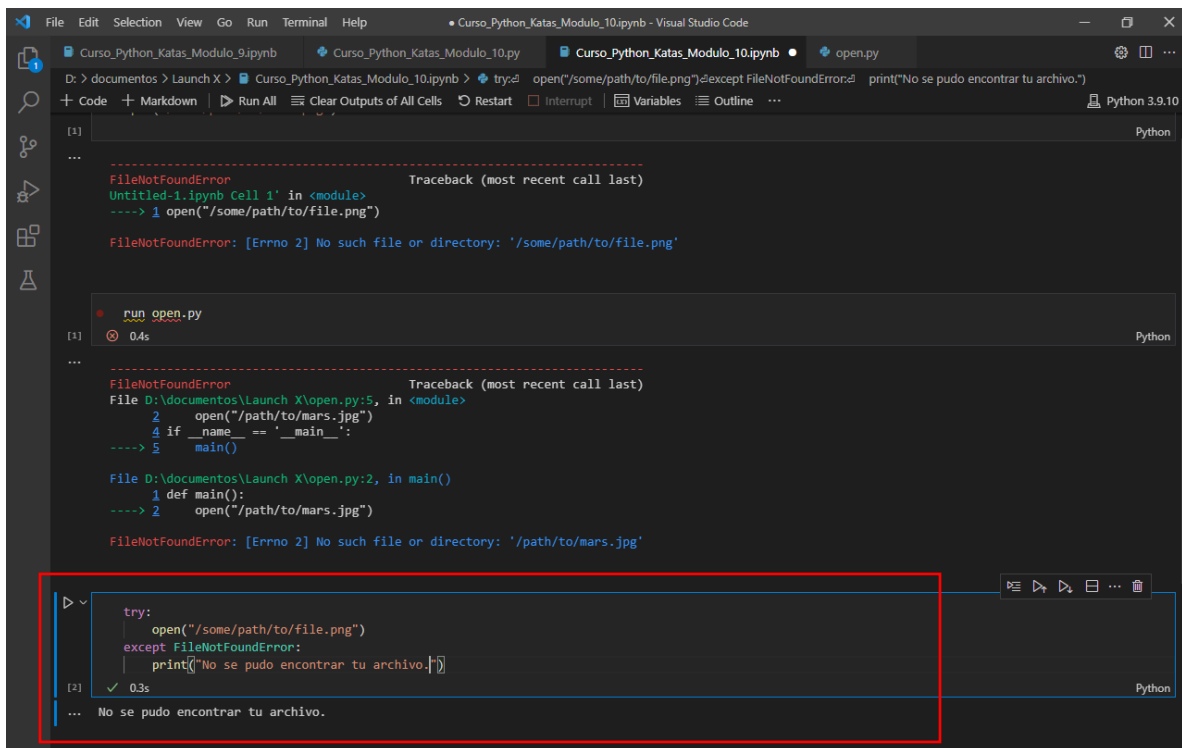
run open.py
[1] 0.4s

-----
FileNotFoundError
Traceback (most recent call last)
File D:\documentos\Launch X\open.py:5, in <module>
      2 open("/path/to/mars.jpg")
      4 if __name__ == '__main__':
----> 5     main()

File D:\documentos\Launch X\open.py:2, in main()
      1 def main():
----> 2     open("/path/to/mars.jpg")

FileNotFoundError: [Errno 2] No such file or directory: '/path/to/mars.jpg'
```

Usamos TRY y EXCEPT para controlar el error al abrir el archivo



The screenshot shows the Visual Studio Code interface with a Python file named `open.py`. The code in the file is as follows:

```
1 open("/some/path/to/file.png")
```

The output console shows a `FileNotFoundError` traceback:

```
FileNotFoundError: [Errno 2] No such file or directory: '/some/path/to/file.png'
```

Below the error, the code is shown again with a `try-except` block:

```
1 def main():
2     open("/path/to/mars.jpg")
3
4 if __name__ == '__main__':
5     main()
```

The output console shows the execution of the script:

```
FileNotFoundError: [Errno 2] No such file or directory: '/path/to/mars.jpg'
```

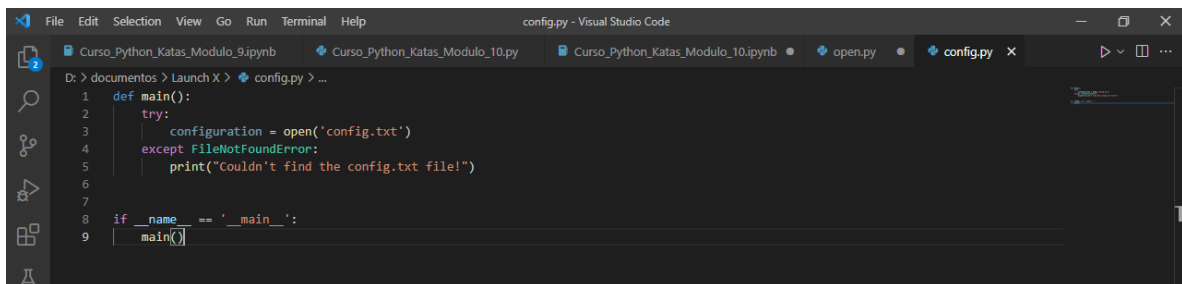
At the bottom, a red box highlights the `try-except` block in the code editor:

```
try:
    open("/some/path/to/file.png")
except FileNotFoundError:
    print("No se pudo encontrar tu archivo.")
```

The output console shows the result of the execution:

```
No se pudo encontrar tu archivo.
```

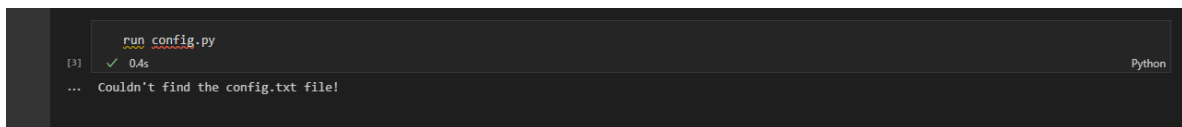
Creamos el archivo config



The screenshot shows the Visual Studio Code interface with a Python file named `config.py`. The code in the file is as follows:

```
1 def main():
2     try:
3         configuration = open('config.txt')
4     except FileNotFoundError:
5         print("Couldn't find the config.txt file!")
6
7
8 if __name__ == '__main__':
9     main()
```

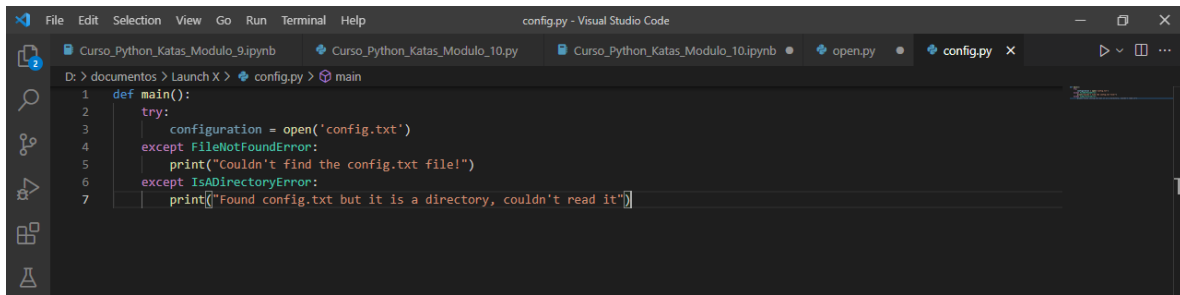
Corremos el archivo config.py



The screenshot shows the output console of the `config.py` script. The output is:

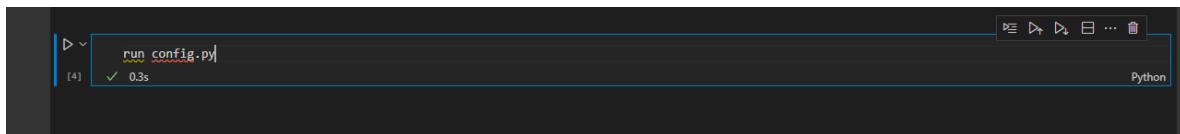
```
... Couldn't find the config.txt file!
```

Editamos el archivo config.py con el siguiente código



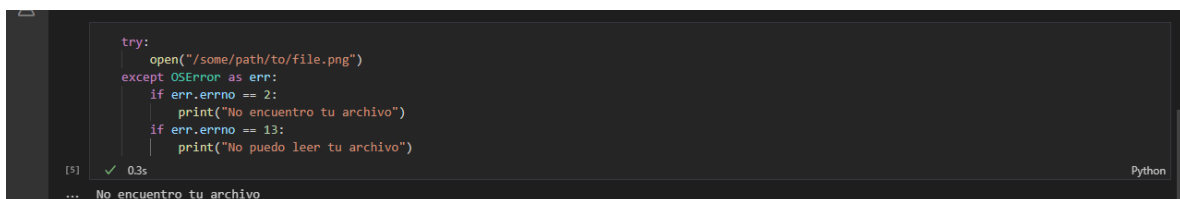
```
1 def main():
2     try:
3         configuration = open('config.txt')
4     except FileNotFoundError:
5         print("Couldn't find the config.txt file!")
6     except IsADirectoryError:
7         print("Found config.txt but it is a directory, couldn't read it")
```

Al ejecutarse nos muestra que todo funciona correctamente por lo que ya no esta el error.



```
run config.py
[4] ✓ 0.3s Python
```

Ejecutamos el siguiente script y nos muestra este resultado

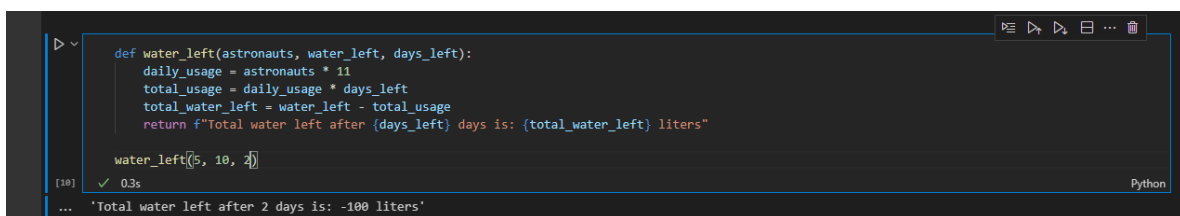


```
try:
    open("/some/path/to/file.png")
except OSError as err:
    if err.errno == 2:
        print("No encuentro tu archivo")
    if err.errno == 13:
        print("No puedo leer tu archivo")
[5] ✓ 0.3s Python
... No encuentro tu archivo
```

Problema 2 Generación de excepciones

Creamos la siguiente función con los siguientes datos:

water_left(5, 100, 2)



```
def water_left(astronauts, water_left, days_left):
    daily_usage = astronauts * 11
    total_usage = daily_usage * days_left
    total_water_left = water_left - total_usage
    return f"Total water left after {days_left} days is: {total_water_left} liters"

water_left(5, 100, 2)
[10] ✓ 0.3s Python
... 'Total water left after 2 days is: -100 liters'
```

Observamos que nos da error en los litros de agua ya que es un numero negativo

Arreglamos el programa para que nos arroje un error en esos casos

```
def water_left(astronauts, water_left, days_left):
    daily_usage = astronauts * 11
    total_usage = daily_usage * days_left
    total_water_left = water_left - total_usage
    if total_water_left < 0:
        raise RuntimeError(f"There is not enough water for {astronauts} astronauts after {days_left} days!")
    return f"Total water left after {days_left} days is: {total_water_left} liters"

water_left(5, 10, 2)

[12] 0.7s Python

RuntimeError                                Traceback (most recent call last)
d:\documentos\Launch X\Curso_Python_Katas_Modulo_10.ipynb Cell 8' in <module>
      6         raise RuntimeError(f"There is not enough water for {astronauts} astronauts after {days_left} days!")
      7     return f"Total water left after {days_left} days is: {total_water_left} liters"
----> 9 water_left(5, 10, 2)

d:\documentos\Launch X\Curso_Python_Katas_Modulo_10.ipynb Cell 8' in water_left(astronauts, water_left, days_left)
      4 total_water_left = water_left - total_usage
      5 if total_water_left < 0:
----> 6         raise RuntimeError(f"There is not enough water for {astronauts} astronauts after {days_left} days!")
      7 return f"Total water left after {days_left} days is: {total_water_left} liters"

RuntimeError: There is not enough water for 5 astronauts after 2 days!
```

Cambiamos el código

```
def water_left(astronauts, water_left, days_left):
    daily_usage = astronauts * 11
    total_usage = daily_usage * days_left
    total_water_left = water_left - total_usage
    if total_water_left < 0:
        raise RuntimeError(f"There is not enough water for {astronauts} astronauts after {days_left} days!")
    return f"Total water left after {days_left} days is: {total_water_left} liters"
    try:
        water_left(5, 100, 2)
    except RuntimeError as err:
        alert_navigation_system(err)

[24] 0.6s Python

RuntimeError                                Traceback (most recent call last)
d:\documentos\Launch X\Curso_Python_Katas_Modulo_10.ipynb Cell 9' in <module>
      8 try:
----> 9     water_left(5, 100, 2)
     10 except RuntimeError as err:

d:\documentos\Launch X\Curso_Python_Katas_Modulo_10.ipynb Cell 9' in water_left(astronauts, water_left, days_left)
      5 if total_water_left < 0:
----> 6         raise RuntimeError(f"There is not enough water for {astronauts} astronauts after {days_left} days!")
      7 return f"Total water left after {days_left} days is: {total_water_left} liters"

RuntimeError: There is not enough water for 5 astronauts after 2 days!

During handling of the above exception, another exception occurred:

NameError                                Traceback (most recent call last)
d:\documentos\Launch X\Curso_Python_Katas_Modulo_10.ipynb Cell 9' in <module>
      9     water_left(5, 100, 2)
     10 except RuntimeError as err:
----> 11     alert_navigation_system(err)

NameError: name 'alert_navigation_system' is not defined
```

Actualizamos el código para que no permita datos no permitidos

```
water_left("3", "200", None)
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
  File "<stdin>", line 3, in water_left
TypeError: can't multiply sequence by non-int of type 'NoneType'

[35] 0.4s Python

Input In [35]
Traceback (most recent call last):
SyntaxError: invalid syntax
```

Nos arroja error. Actualizamos el código y debe correr correctamente



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

[31] ✓ 0.5s

Python