

Creación de un archivo en Python llamado "open.py" y ejecutándolo en la terminal de VSCode para obtener una excepción de tipo traceback:

```

CicFor.ipynb  Cadenas.ipynb  FCadenas.ipynb  Diccionarios.ipynb  programa.py  open.py U X
Modulo10 > open.py > ...
1  def main():
2      open("/path/to/mars.jpg")
3
4  if __name__ == "__main__":
5      main()

PS C:\Users\... \Documents\KatasRicardoLaunchX> cd Modulo10
PS C:\Users\... \Documents\KatasRicardoLaunchX\Modulo10> python3 open.py
Traceback (most recent call last):
  File "C:\Users\... \Documents\KatasRicardoLaunchX\Modulo10\open.py", line 5, in <module>
    main()
  File "C:\Users\... \Documents\KatasRicardoLaunchX\Modulo10\open.py", line 2, in main
    open("/path/to/mars.jpg")
FileNotFoundError: [Errno 2] No such file or directory: '/path/to/mars.jpg'
```

Luego se incluyó y trabajo con las excepciones try y except con el fin de controlar el flujo del programa (función):

```

Modulo10 > open.py > main
1  def main():
2      try:
3          archivo_config = open("/path/to/config.txt")
4      except FileNotFoundError:
5          print("Archivo no encontrado")
6
7  if __name__ == "__main__":
8      main()
9
10 # Como primer paso se incluye la excepcion try y except a la funcion creada.
11

PS C:\Users\... \Documents\KatasRicardoLaunchX\Modulo10> python3 open.py
Archivo no encontrado
```

Ahora se creó un programa para calcular el suministro de agua para una nave espacial llamada "Apolo" con el fin de que calcule el programa cuanta agua queda conforme transcurren los días esto con excepciones:

```
Modulo10 > ProgramApolo.py > ...
1  # Definimos la funcion para determinar cuanta agua queda con el transcurso de los dias
2  def agua(astronautas, agua_por_dia, dias): # Astronautas, agua por dia, dias
3      agua_usada = astronautas * 5 # 5 litros de agua por astronauta
4      total_agua = agua_usada * dias # Total de litros de agua
5      agua_restante = agua_por_dia - agua_usada # Litros de agua restante
6      return f"El agua que queda es {agua_restante * dias} litros" # Retorna el valor de la variable
7
8  # Usando excepciones
9  try: # Se define la excepcion
10     agua(5, 10, 5) # 5 astronautas, 10 litros por dia, 5 dias
11 except RuntimeError as err: # RuntimeError, excepcion para generar la alerta.
12     print(f"Error: {err}") # Imprime el error
13
```