

```

> open("/path/to/mars.jpg")
[26] 0.1s

...
-----
FileNotFoundError                                Traceback (most recent call last)
<ipython-input-26-bd73c29b34fc> in <module>
----> 1 open("/path/to/mars.jpg")

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

```

```

%run open.py
0.1s

-----
FileNotFoundError                                Traceback (most recent call last)
d:\Users\EduardoMorales\Desktop\CursoInnovaccion\KatasParaEntregar\open.py in <module>
      3
      4 if __name__ == '__main__':
----> 5     main()

d:\Users\EduardoMorales\Desktop\CursoInnovaccion\KatasParaEntregar\open.py in main()
      1 def main():
----> 2     open("/path/to/mars.jpg")
      3
      4 if __name__ == '__main__':
      5     main()

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

```

```

try:
    open('config.txt')
except FileNotFoundError:
    print("Couldn't find the config.txt file!")
0.5s

```

Couldn't find the config.txt file!

```

%run config.py
0.5s

-----
PermissionError                                Traceback (most recent call last)
d:\Users\EduardoMorales\Desktop\CursoInnovaccion\KatasParaEntregar\config.py in <module>
      7
      8 if __name__ == '__main__':
----> 9     main()

d:\Users\EduardoMorales\Desktop\CursoInnovaccion\KatasParaEntregar\config.py in main()
      1 def main():
      2     try:
----> 3         configuration = open('config.txt')
      4     except FileNotFoundError:
      5         print("Couldn't find the config.txt file!")

PermissionError: [Errno 13] Permission denied: 'config.txt'

```

```
%run config.py
✓ 0.1s
Couldn't find the config.txt file!
```

```
%run config.py
✓ 0.6s
Found config.txt but it is a directory, couldn't read it
```

```
%run config.py
✓ 0.8s
Couldn't find the config.txt file!
```

```
try:
    open("mars.jpg")
except FileNotFoundError as err:
    | print("got a problem trying to read the file:", err)
✓ 0.8s
got a problem trying to read the file: [Errno 2] No such file or directory: 'mars.jpg'
```

```
try:
    open("config.txt")
except OSError as err:
    if err.errno == 2:
        print("Couldn't find the config.txt file!")
    elif err.errno == 13:
        | print("Found config.txt but couldn't read it")
✓ 0.5s
Couldn't find the config.txt file!
```

```
water_left(5, 100, 2)
✓ 0.5s
'Total water left after 2 days is: -10 liters'
```

```
water_left(5, 100, 2)
```

⊗ 0.6s

```
-----
RuntimeError                                Traceback (most recent call last)
d:\Users\EduardoMorales\Desktop\CursoInnovaccion\KatasParaEntregar\open.py in <module>
----> 1 water_left(5, 100, 2)

d:\Users\EduardoMorales\Desktop\CursoInnovaccion\KatasParaEntregar\open.py 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!
```

```
water_left("3", "200", None)
```

⊗ 0.5s

```
-----
TypeError                                Traceback (most recent call last)
d:\Users\EduardoMorales\Desktop\CursoInnovaccion\KatasParaEntregar\open.py in <module>
----> 1 water_left("3", "200", None)

d:\Users\EduardoMorales\Desktop\CursoInnovaccion\KatasParaEntregar\open.py in water_left(astronauts, water_left, days_left)
      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     if total_water_left < 0:

TypeError: can't multiply sequence by non-int of type 'NoneType'
```

```
water_left("3", "200", None)
```

⊗ 0.1s

```
-----
TypeError                                Traceback (most recent call last)
d:\Users\EduardoMorales\Desktop\CursoInnovaccion\KatasParaEntregar\open.py in water_left(astronauts, water_left, days_left)
      4         # If argument is an int, the following operation will work
----> 5         argument / 10
      6     except TypeError:

TypeError: unsupported operand type(s) for /: 'str' and 'int'

During handling of the above exception, another exception occurred:

TypeError                                Traceback (most recent call last)
d:\Users\EduardoMorales\Desktop\CursoInnovaccion\KatasParaEntregar\open.py in <module>
----> 1 water_left("3", "200", None)

d:\Users\EduardoMorales\Desktop\CursoInnovaccion\KatasParaEntregar\open.py in water_left(astronauts, water_left, days_left)
      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     daily_usage = astronauts * 11
     11     total_usage = daily_usage * days_left

TypeError: All arguments must be of type int, but received: '3'
```