

```

Microsoft Windows [Version 10.0.19043.1526]
(c) Microsoft Corporation. All rights reserved.

C:\Users\Woodstock>python
Python 3.9.4 (tags/v3.9.4:1f2e308, Apr  4 2021, 13:27:16) [MSC v.1928 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license" for more information.
>>> open('/path/to/mars.jpg')
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
FileNotFoundError: [Errno 2] No such file or directory: '/path/to/mars.jpg'
>>> exit()

C:\Users\Woodstock>cd xLaunch

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

C:\Users\Woodstock\xLaunch>python
Python 3.9.4 (tags/v3.9.4:1f2e308, Apr  4 2021, 13:27:16) [MSC v.1928 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license" for more information.
>>> try:
...     open('config.txt')
... except:
...
KeyboardInterrupt
>>> try:
...     open('config.txt')
... except FileNotFoundError:
...     print("Couldn't find the config.txt file!")
...
Couldn't find the config.txt file!
>>> exit()

C:\Users\Woodstock\xLaunch>python config.py
Couldn't find the config.txt file!

```

```

C:\Users\Woodstock\xLaunch>python config.py
Couldn't find the config.txt file!

C:\Users\Woodstock\xLaunch>python config.py
Traceback (most recent call last):
  File "C:\Users\Woodstock\xLaunch\config.py", line 9, in <module>
    main()
  File "C:\Users\Woodstock\xLaunch\config.py", line 3, in main
    configuration = open('config.txt')
PermissionError: [Errno 13] Permission denied: 'config.txt'

C:\Users\Woodstock\xLaunch>python config.py
You have no permissions to either open or execute this file!

```

```

C:\Users\Woodstock\xLaunch>python
Python 3.9.4 (tags/v3.9.4:1f2e308, Apr  4 2021, 13:27:16) [MSC v.1928 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license" for more information.
>>> try:
...     open('Mars.jpg')
... except:
...
KeyboardInterrupt
>>> try:
...     open('Mars.jpg')
... except FileNotFoundError as err:
...     print("got a problem trying to read the file:", err)
...
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")
...
Couldn't find the config.txt file!

```

```
>>> 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)
'Total water left after 2 days is: -10 liters'
>>> 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, 100, 2)
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
  File "<stdin>", line 6, in water_left
RuntimeError: There is not enough water for 5 astronauts after 2 days!
>>> try:
...     water_left(5, 100, 2)
... except RuntimeError as err:
...     alert_navigation_system(err)
...
Traceback (most recent call last):
  File "<stdin>", line 2, in <module>
  File "<stdin>", line 6, in water_left
RuntimeError: There is not enough water for 5 astronauts after 2 days!

During handling of the above exception, another exception occurred:

Traceback (most recent call last):
  File "<stdin>", line 4, in <module>
NameError: name 'alert_navigation_system' is not defined
>>> 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'
```

```
>>> 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"
...
>>> water_left("3", "200", None)
Traceback (most recent call last):
  File "<stdin>", line 5, in water_left
TypeError: unsupported operand type(s) for /: 'str' and 'int'

During handling of the above exception, another exception occurred:

Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
  File "<stdin>", line 9, in water_left
TypeError: All arguments must be of type int, but received: '3'
```