ASSIGNMENT-7.4

NAME:-N.ROHAN

BATCH-16

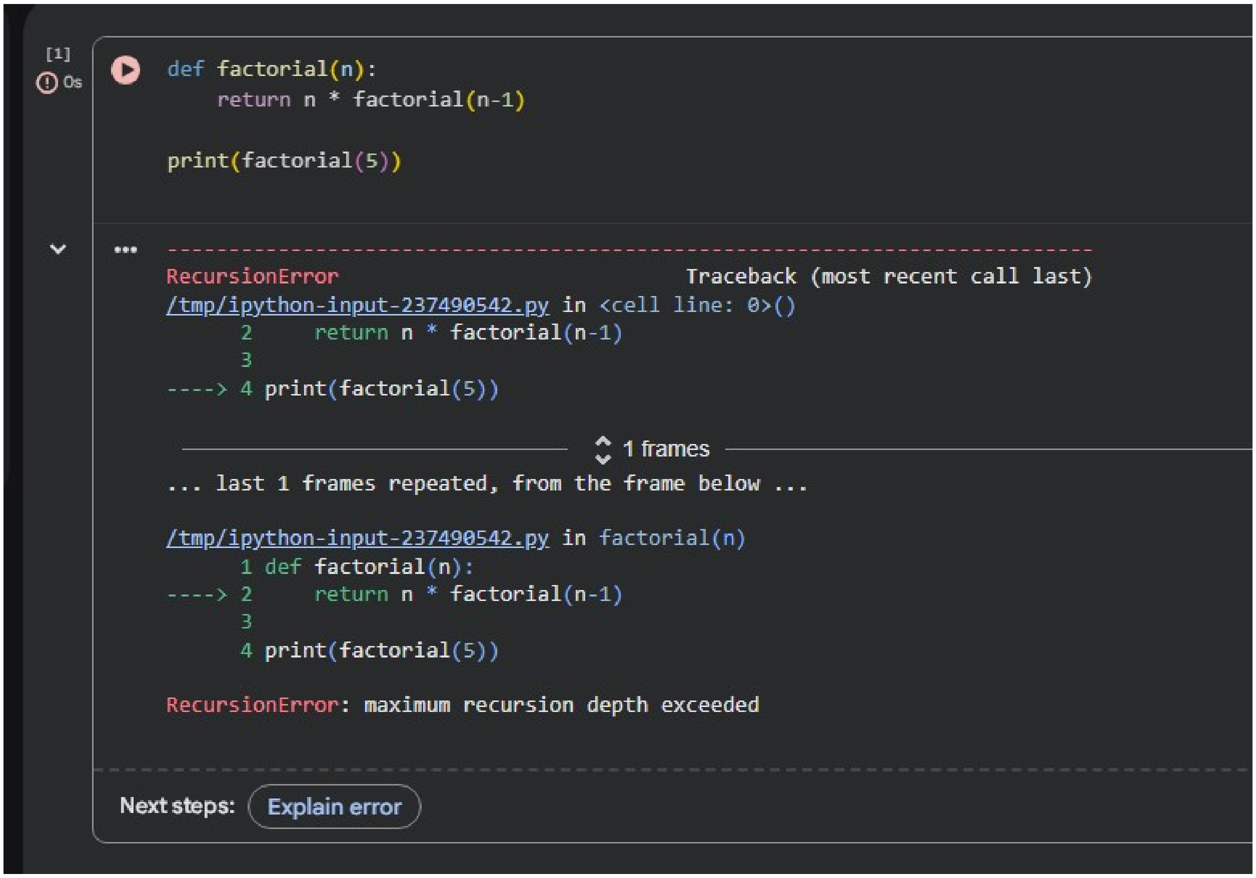
HTNO:-2303A51064

Task 1 – Debugging Recursive Factorial Function

AI Prompt Used

Debug this recursive factorial Python function. It crashes and gives wrong output.

INCORRECT CODE:-



```
def factorial(n):
    return n * factorial(n-1)

print(factorial(5))
```

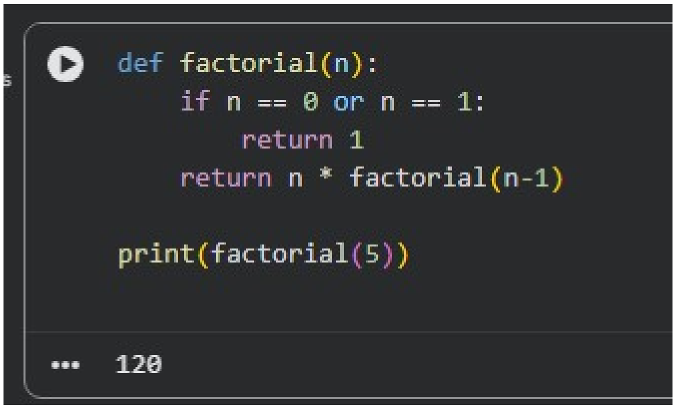
```
---------------------------------------------------------------------------
RecursionError                            Traceback (most recent call last)
/tmp/ipython-input-237490542.py in <cell line: 0>()
      2     return n * factorial(n-1)
      3
----> 4 print(factorial(5))

    ⇕ 1 frames
... last 1 frames repeated, from the frame below ...

/tmp/ipython-input-237490542.py in factorial(n)
      1 def factorial(n):
----> 2     return n * factorial(n-1)
      3
      4 print(factorial(5))

RecursionError: maximum recursion depth exceeded
```

CORRECT CODE:-



```
def factorial(n):
    if n == 0 or n == 1:
        return 1
    return n * factorial(n-1)

print(factorial(5))
```

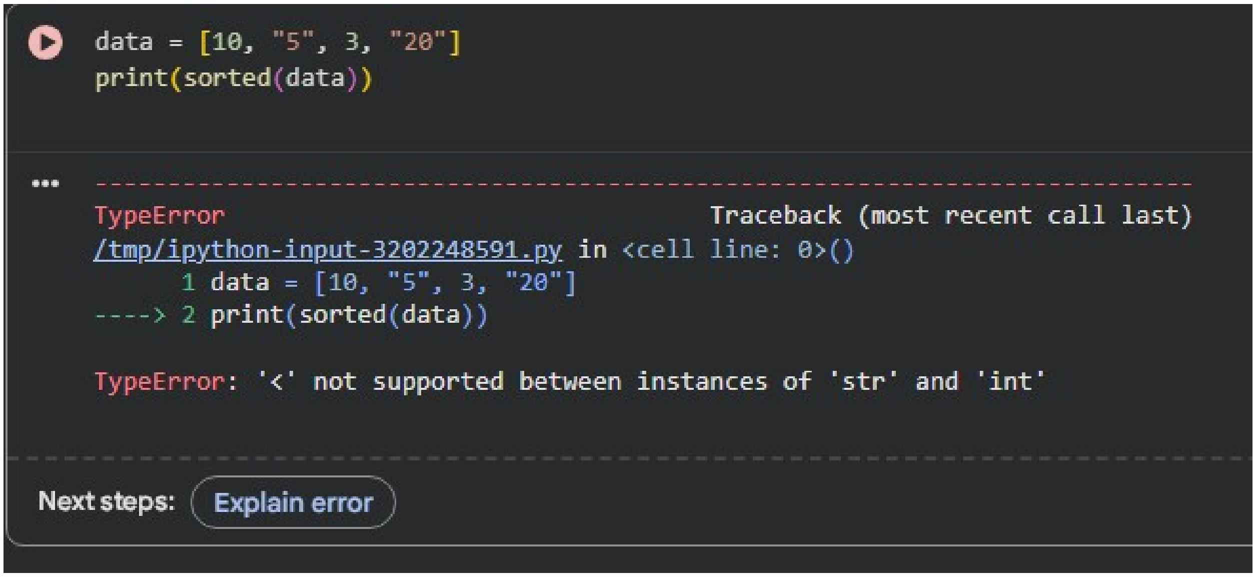
```
120
```

Task 2 – Fixing Data Type Errors in Sorting

AI Prompt Used

Why does this Python sorting code fail with mixed data types?

INCORRECT CODE:-

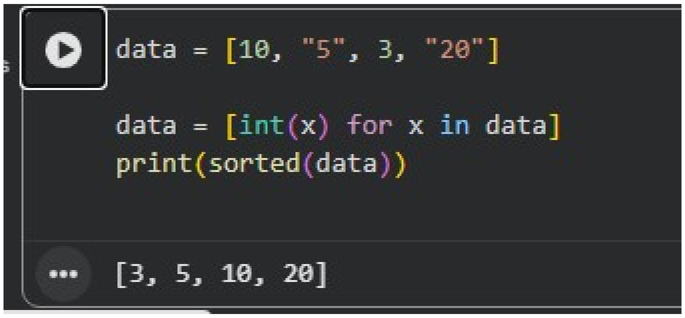


```
data = [10, "5", 3, "20"]
print(sorted(data))
```

```
---------------------------------------------------------------------------
TypeError                                 Traceback (most recent call last)
/tmp/ipython-input-3202248591.py in <cell line: 0>()
      1 data = [10, "5", 3, "20"]
----> 2 print(sorted(data))

TypeError: '<' not supported between instances of 'str' and 'int'
```

CORRECTED CODE:-



```
data = [10, "5", 3, "20"]

data = [int(x) for x in data]
print(sorted(data))
```

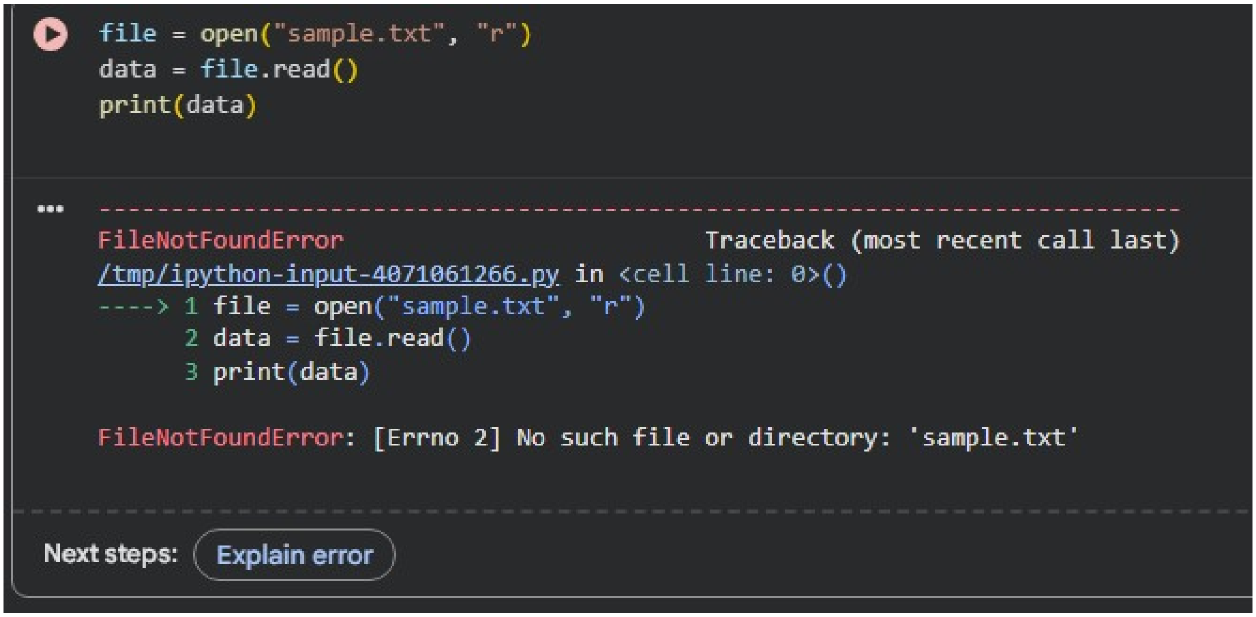
```
[3, 5, 10, 20]
```

Task 3 – Improving File Handling Reliability

AI Prompt Used

Identify issue in Python file handling code that doesn’t close files.

INCORRECT CODE:-

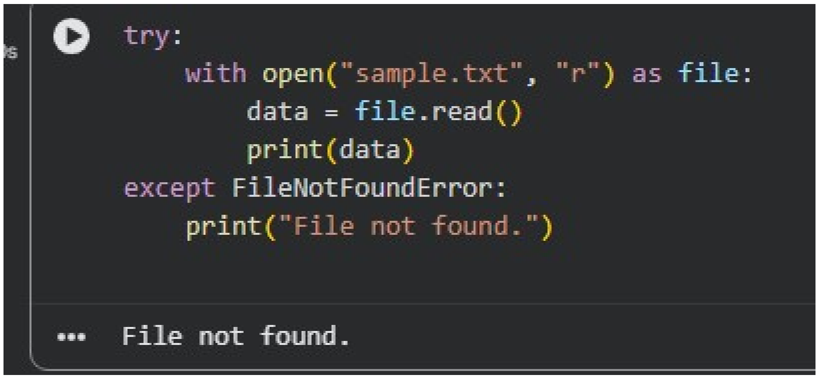


```
file = open("sample.txt", "r")
data = file.read()
print(data)
```

```
---------------------------------------------------------------------------
FileNotFoundError                         Traceback (most recent call last)
/tmp/ipython-input-4071061266.py in <cell line: 0>()
----> 1 file = open("sample.txt", "r")
      2 data = file.read()
      3 print(data)

FileNotFoundError: [Errno 2] No such file or directory: 'sample.txt'
```

CORRECT CODE:-



```
try:
    with open("sample.txt", "r") as file:
        data = file.read()
        print(data)
except FileNotFoundError:
    print("File not found.")
```

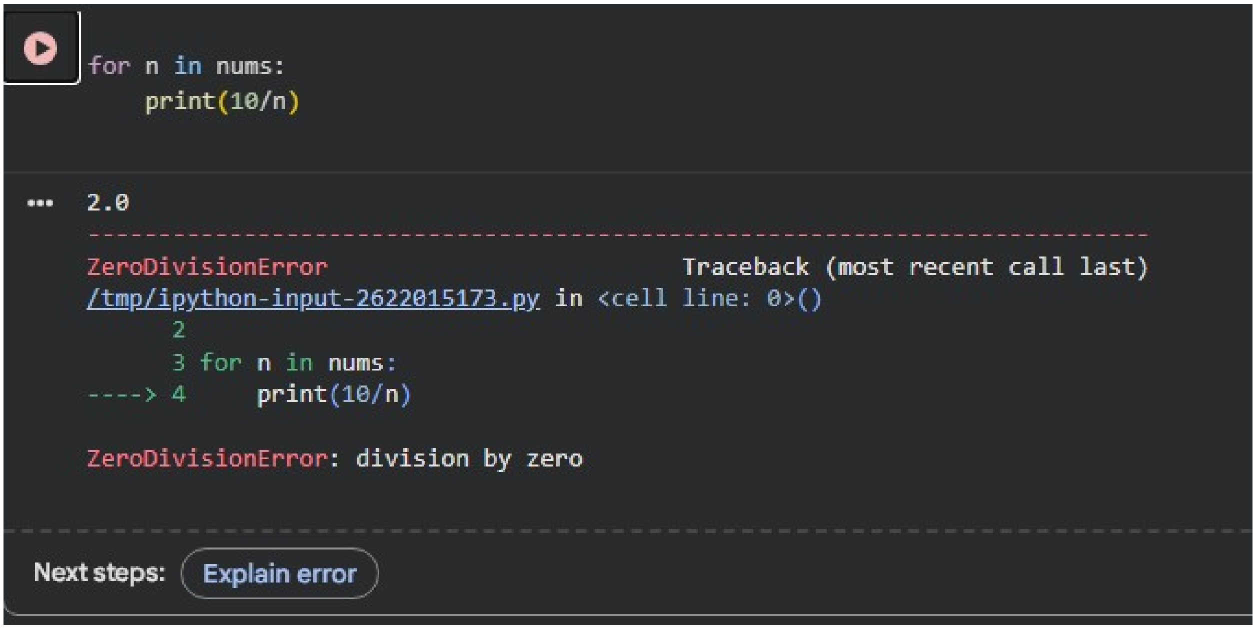
```
File not found.
```

Task 4 – Handling Runtime Errors in Loop

AI Prompt Used

Fix ZeroDivisionError in loop so program continues running.

INCORRECT CODE:-

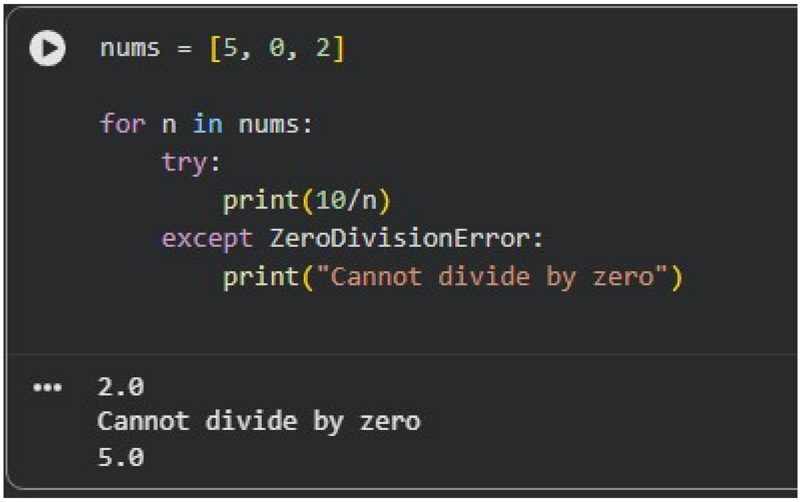


```
for n in nums:
    print(10/n)
```

```
2.0
---------------------------------------------------------------------------
ZeroDivisionError                         Traceback (most recent call last)
/tmp/ipython-input-2622015173.py in <cell line: 0>()
      2
      3 for n in nums:
----> 4     print(10/n)

ZeroDivisionError: division by zero
```

CORRECT CODE:-



```python
nums = [5, 0, 2]

for n in nums:
    try:
        print(10/n)
    except ZeroDivisionError:
        print("Cannot divide by zero")
```

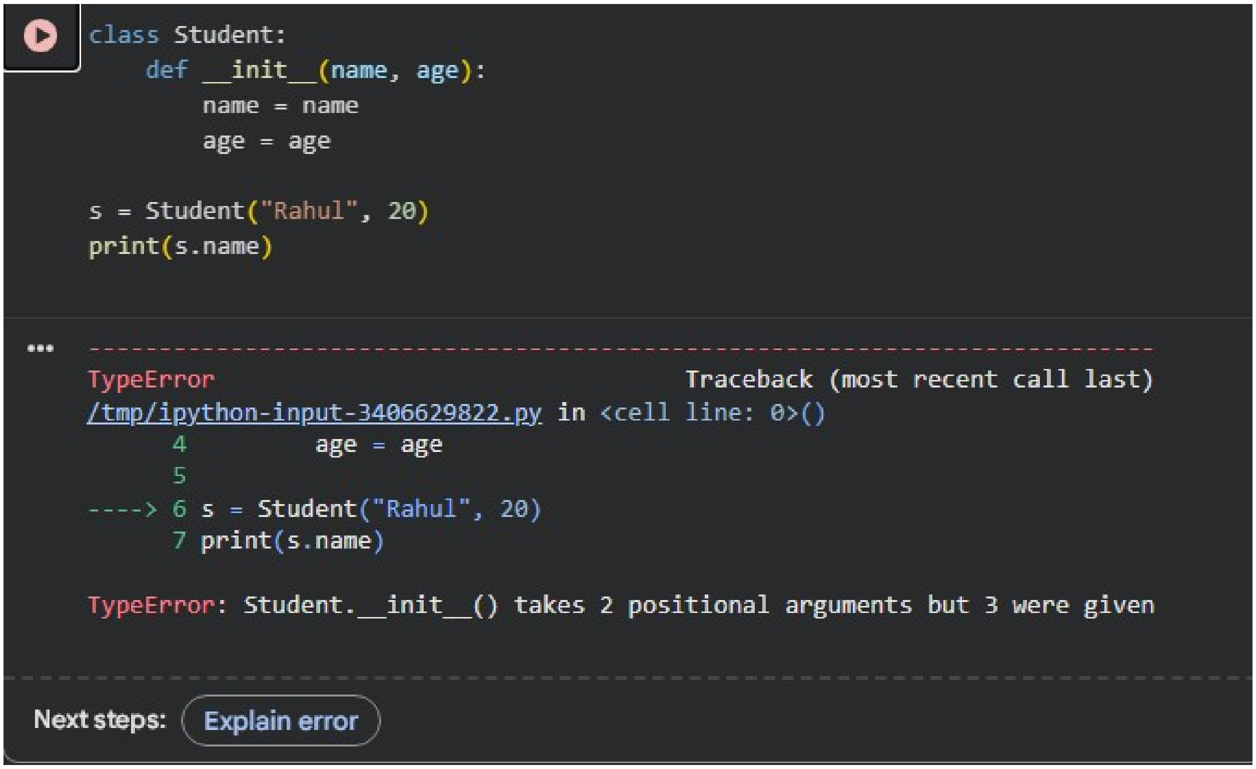
```
2.0
Cannot divide by zero
5.0
```

Task 5 – Debugging Class Initialization Errors

AI Prompt Used

Debug this Python class constructor error.

INCORRECT CODE:-



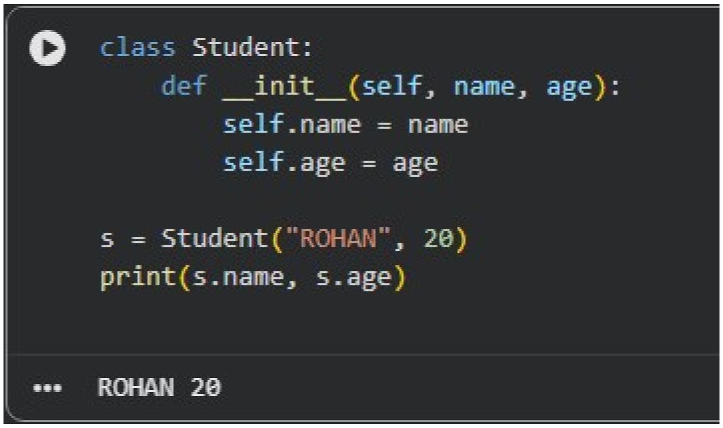
```python
class Student:
    def __init__(name, age):
        name = name
        age = age

s = Student("Rahul", 20)
print(s.name)
```

```
---------------------------------------------------------------------------
TypeError                                 Traceback (most recent call last)
/tmp/ipython-input-3406629822.py in <cell line: 0>()
      4         age = age
      5
----> 6 s = Student("Rahul", 20)
      7 print(s.name)

TypeError: Student.__init__() takes 2 positional arguments but 3 were given
```

CORRECT CODE:-



```
class Student:
    def __init__(self, name, age):
        self.name = name
        self.age = age

s = Student("ROHAN", 20)
print(s.name, s.age)
```

```
ROHAN 20
```