AI Assisted Coding Assignment- 7.2

RUDROJU RUPA SRI

2303A51918

BATCH-30

Task 1 – Runtime Error Due to Invalid Input Type
- A Python program accepts user input and performs arithmetic operations. However, the program throws a runtime error because the input is treated as a string instead of a numeric type.

Example (Buggy Code):

num = input("Enter a number: ")

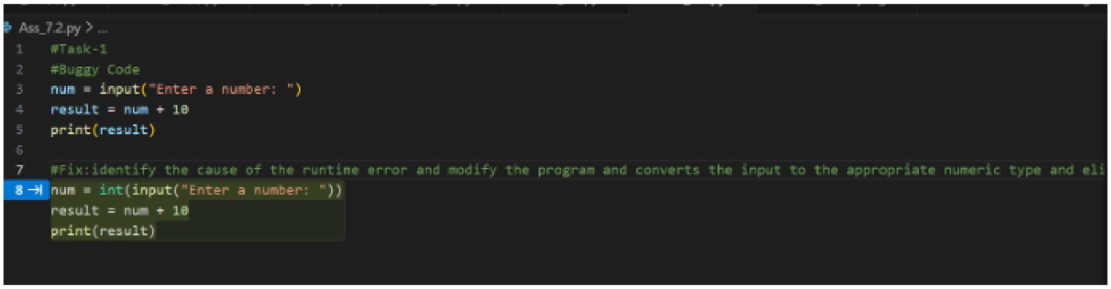
result = num + 10

print(result)

- Task:

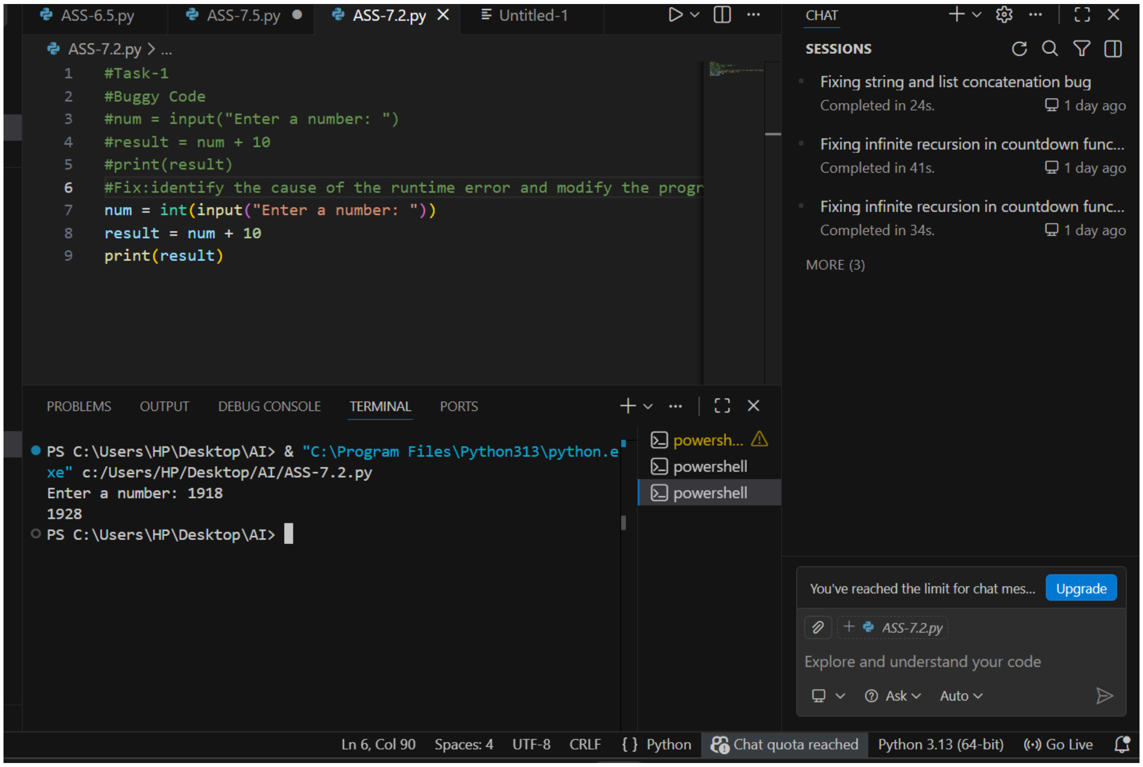
Use AI tools to identify the cause of the runtime error and modify the program so it executes correctly.

Expected Output -1:
- AI converts the input to the appropriate numeric type and eliminates the runtime error.



```
#Task-1
#Buggy Code
num = input("Enter a number: ")
result = num + 10
print(result)

#Fix:identify the cause of the runtime error and modify the program and converts the input to the appropriate numeric type and eli
num = int(input("Enter a number: "))
result = num + 10
print(result)
```



```
#Task-1
#Buggy Code
#num = input("Enter a number: ")
#result = num + 10
#print(result)
#Fix:identify the cause of the runtime error and modify the progr
num = int(input("Enter a number: "))
result = num + 10
print(result)
```

```
PS C:\Users\HP\Desktop\AI> & "C:\Program Files\Python313\python.e
xe" c:/Users/HP/Desktop/AI/ASS-7.2.py
Enter a number: 1918
1928
PS C:\Users\HP\Desktop\AI>
```

Fixing string and list concatenation bug
Completed in 24s. 1 day ago

Fixing infinite recursion in countdown func...
Completed in 41s. 1 day ago

Fixing infinite recursion in countdown func...
Completed in 34s. 1 day ago

MORE (3)

You've reached the limit for chat mes...

Explore and understand your code

Task 2 – Incorrect Function Return Value

A function is designed to calculate the square of a number, but it does not return the computed result properly.

Example (Buggy Code):

def square(n):

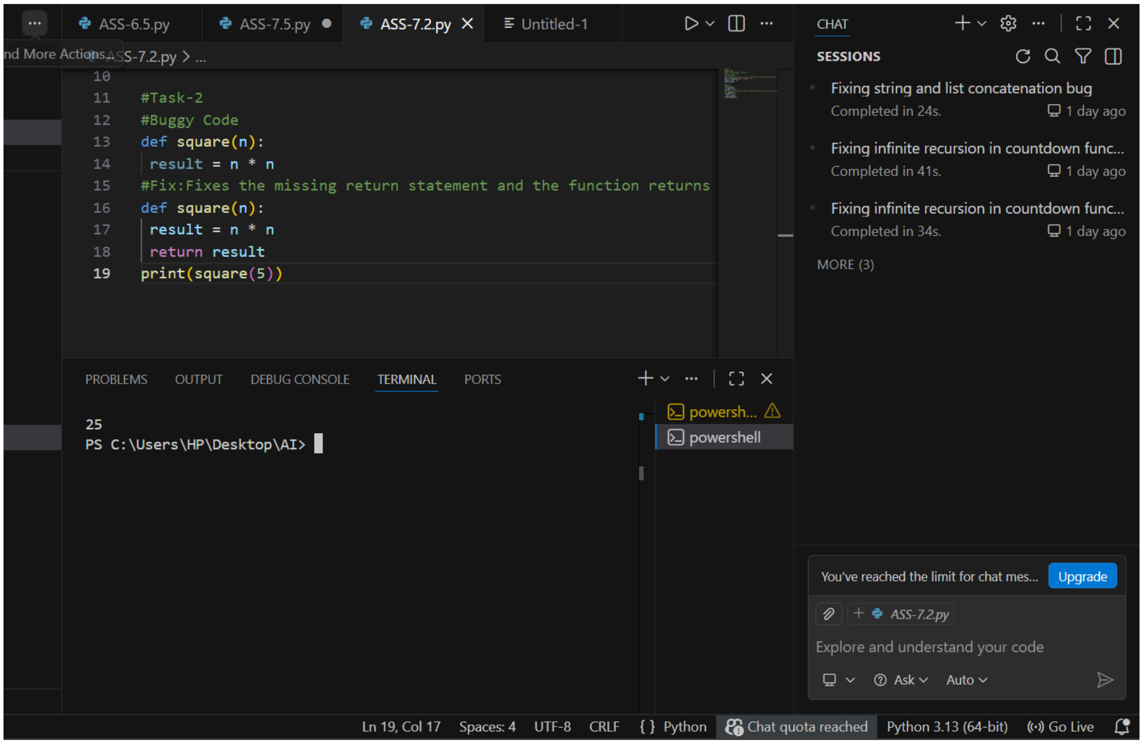
result = n * n

Task:

Use AI assistance to analyze the function and ensure the correct value is returned.

Expected Output -2:

AI fixes the missing return statement and the function returns the correct output.



```python
#Task-2
#Buggy Code
def square(n):
    result = n * n
#Fix:Fixes the missing return statement and the function returns
def square(n):
    result = n * n
    return result
print(square(5))
```

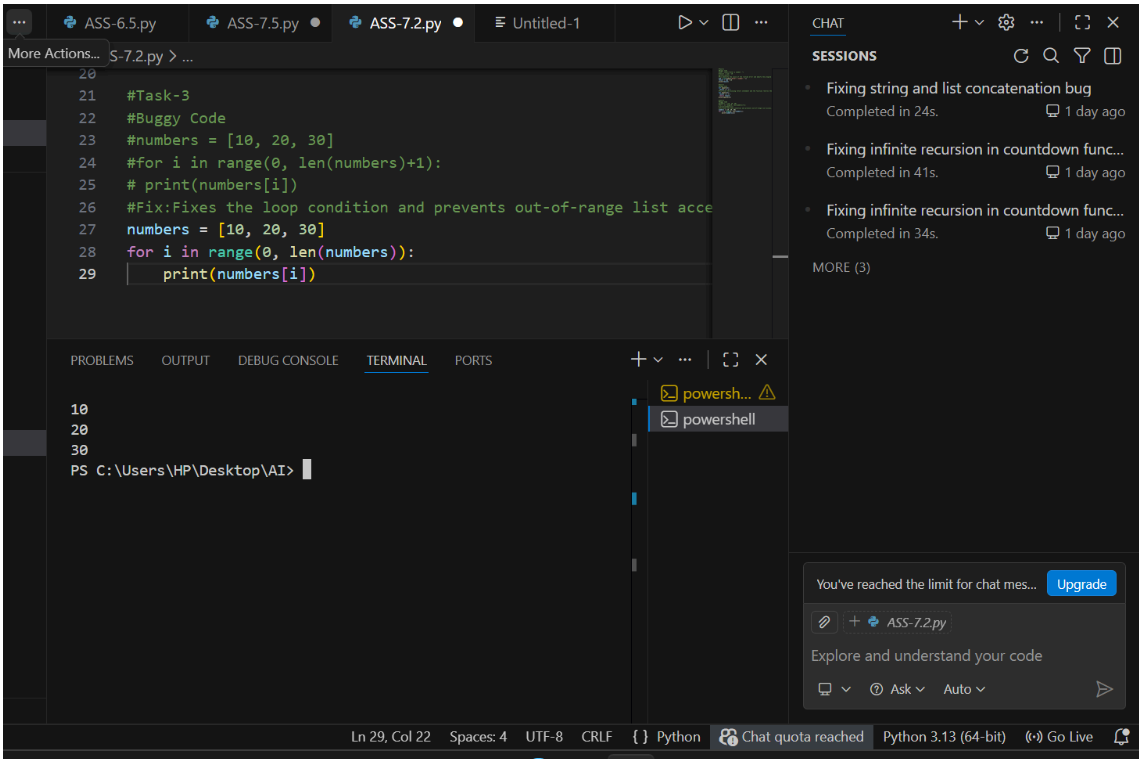
```
25
PS C:\Users\HP\Desktop\AI>
```

Fixing string and list concatenation bug
Completed in 24s. 1 day ago

Fixing infinite recursion in countdown func...
Completed in 41s. 1 day ago

Fixing infinite recursion in countdown func...
Completed in 34s. 1 day ago

Task 3 – IndexError in List Traversal
A Python program iterates over a list using incorrect index limits, causing an IndexError.
Example (Buggy Code):
numbers = [10, 20, 30]
for i in range(0, len(numbers)+1):
print(numbers[i])
Task:
Use AI to identify the incorrect loop boundary and correct the iteration logic.
Expected Output -3:
AI fixes the loop condition and prevents out-of-range list access.



```
#Task-3
#Buggy Code
#numbers = [10, 20, 30]
#for i in range(0, len(numbers)+1):
# print(numbers[i])
#Fix:Fixes the loop condition and prevents out-of-range list acce
numbers = [10, 20, 30]
for i in range(0, len(numbers)):
    print(numbers[i])
```

```
10
20
30
PS C:\Users\HP\Desktop\AI>
```

Task 4 – Uninitialized Variable Usage

A program uses a variable in a calculation before assigning it any value.

Example (Buggy Code):

if True:

pass

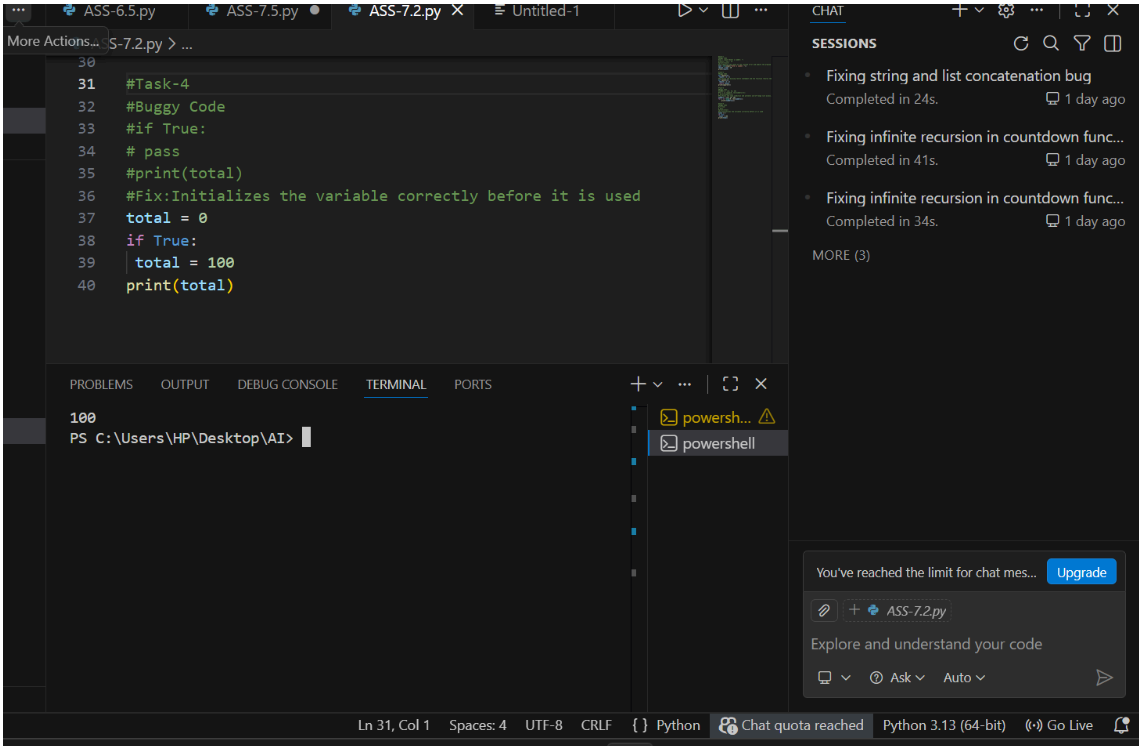
print(total)

Task:

Use AI tools to detect the uninitialized variable and correct the program.

Expected Output -4:

AI initializes the variable correctly before it is used



```python
#Task-4
#Buggy Code
#if True:
# pass
#print(total)
#Fix:Initializes the variable correctly before it is used
total = 0
if True:
    total = 100
print(total)
```

```
100
PS C:\Users\HP\Desktop\AI>
```

Fixing string and list concatenation bug
Completed in 24s. 1 day ago

Fixing infinite recursion in countdown func...
Completed in 41s. 1 day ago

Fixing infinite recursion in countdown func...
Completed in 34s. 1 day ago

You've reached the limit for chat mes...

Explore and understand your code

Task 5 – Logical Error in Student Grading System

A grading program assigns incorrect grades due to improper conditional logic.

Example (Buggy Code):

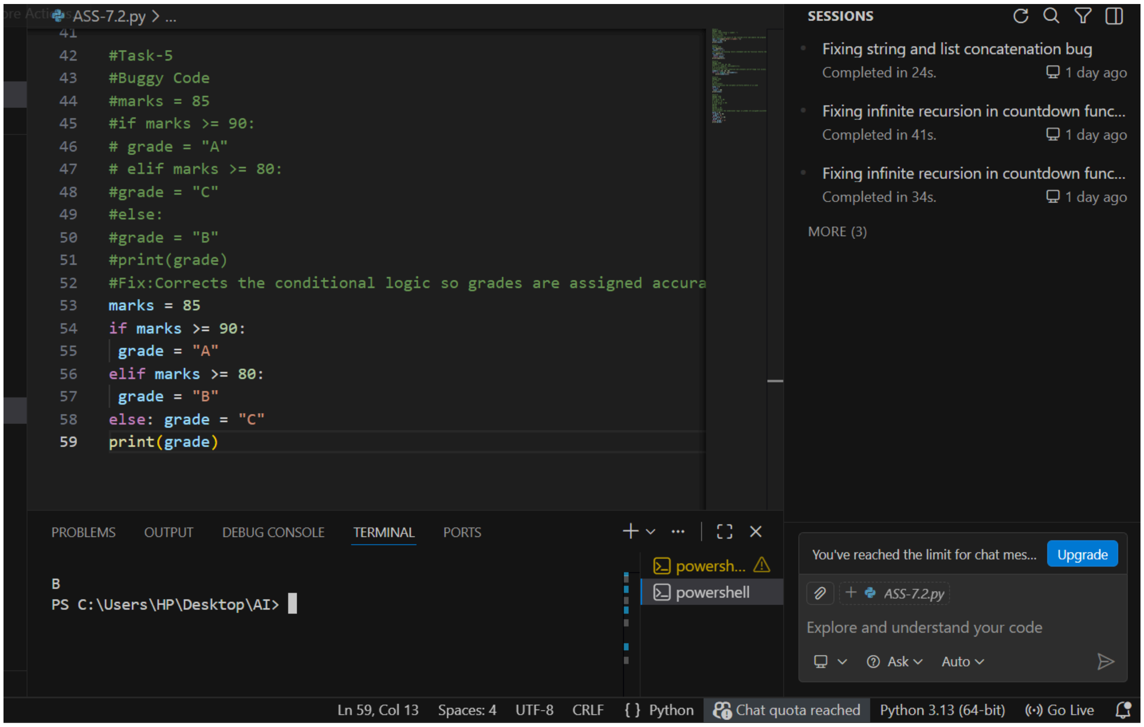
```
marks = 85
if marks >= 90:
grade = "A"
elif marks >= 80:
grade = "C"
else:
grade = "B"
print(grade)
```

Task:

Use AI to analyze the grading conditions and correct the logical flow.

Expected Output -5:

AI corrects the conditional logic so grades are assigned accurately.



```python
#Task-5
#Buggy Code
#marks = 85
#if marks >= 90:
# grade = "A"
# elif marks >= 80:
#grade = "C"
#else:
#grade = "B"
#print(grade)
#Fix:Corrects the conditional logic so grades are assigned accura
marks = 85
if marks >= 90:
 grade = "A"
elif marks >= 80:
 grade = "B"
else: grade = "C"
print(grade)
```

```
B
PS C:\Users\HP\Desktop\AI>
```

Fixing string and list concatenation bug
Completed in 24s. 1 day ago

Fixing infinite recursion in countdown func...
Completed in 41s. 1 day ago

Fixing infinite recursion in countdown func...
Completed in 34s. 1 day ago

You've reached the limit for chat mes...

Explore and understand your code