

## School of Computer Science and Artificial Intelligence

---

**Lab Assignment # 7.2**

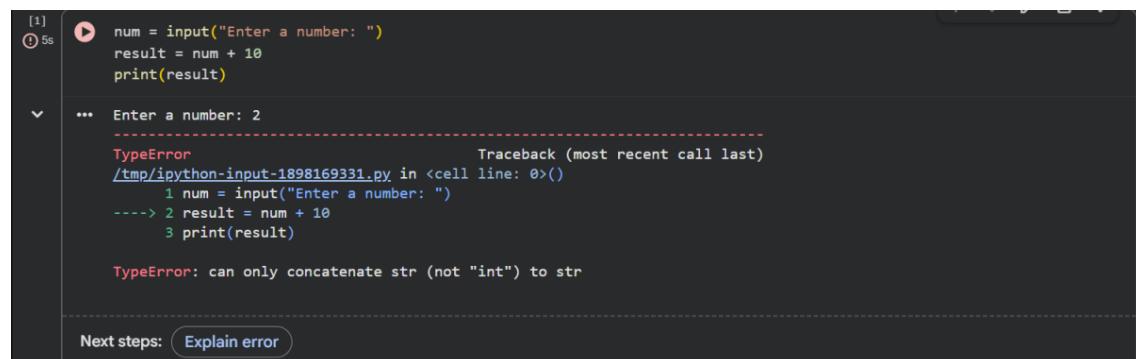
---

Program : B. Tech (CSE)  
Specialization : -  
Course Title : AI Assisted Coding  
Course Code : 23CS002PC304  
Semester : II  
Academic Session : 2025-2026  
Name of Student : paindla vamshi  
Enrollment No. : 2403A51L24  
Batch No. : 51  
Date : 30/01/26

---

**Submission Starts here****Screenshots:****Task 1 – Runtime Error Due to Invalid Input Type****(Buggy Code):**

```
num = input("Enter a number: ")
result = num + 10
print(result)
```



The screenshot shows a Jupyter Notebook cell with the following code:

```
[1]: num = input("Enter a number: ")
      result = num + 10
      print(result)

... Enter a number: 2
```

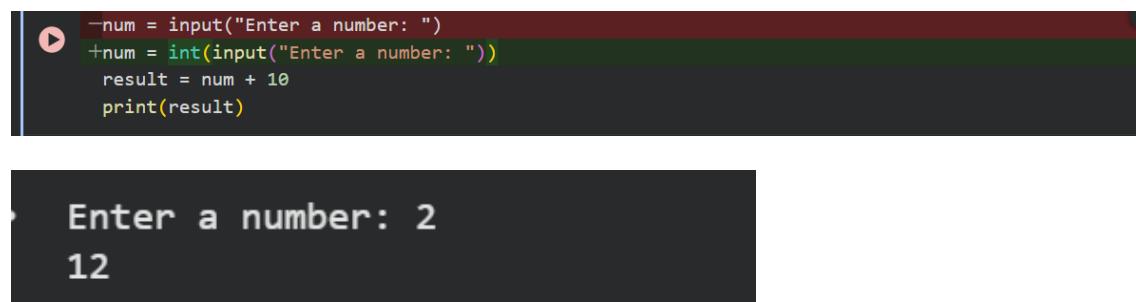
A red error message is displayed:

```
TypeError: can only concatenate str (not "int") to str
```

Traceback (most recent call last):

```
/tmp/ipython-input-1898169331.py in <cell line: 0>()
      1 num = input("Enter a number: ")
----> 2 result = num + 10
      3 print(result)
```

Next steps: Explain error

**Output:**

The screenshot shows a Jupyter Notebook cell with the following corrected code:

```
-num = input("Enter a number: ")
+num = int(input("Enter a number: "))
result = num + 10
print(result)
```

The output window shows the user entering the number 2 and the resulting output:

```
Enter a number: 2
12
```

## Task 2 – Incorrect Function Return Value

(Buggy Code):

```
def square(n):  
    result = n * n
```

The screenshot shows a code editor with the following code:

```
def square(n):  
    result = n * n  
...     File "/tmp/ipython-input-3910404483.py", line 2  
          result = n * n  
          ^  
        IndentationError: expected an indented block after function definition on line 1
```

Below the code, there is a button labeled "Next steps: Explain error".

Output:

The screenshot shows a code editor with the following code:

```
[10]  def square(n):  
    -result = n * n  
    +    result = n * n
```

On the left, there is a green checkmark icon and the text "[10]". On the right, there is a play button icon.

## Task 3 – IndexError in List Traversal

(Buggy Code):

```
numbers = [10, 20, 30]  
for i in range(0, len(numbers)+1):  
    print(numbers[i])
```

The screenshot shows a code editor with the following code:

```
[11]  numbers = [10, 20, 30]  
    for i in range(0, len(numbers)+1):  
        print(numbers[i])  
...     File "/tmp/ipython-input-726334973.py", line 3  
          print(numbers[i])  
          ^  
        IndentationError: expected an indented block after 'for' statement on line 2
```

Below the code, there is a button labeled "Next steps: Explain error".

Output:

The screenshot shows a code editor with the following code:

```
[1]  numbers = [10, 20, 30]  
    -for i in range(0, len(numbers)+1):  
    -print(numbers[i])  
    +for i in range(len(numbers)):  
    +    print(numbers[i])
```

On the left, there is a green checkmark icon and the text "[1]". On the right, there is a play button icon.

The screenshot shows a terminal window with the following output:

```
... 10  
20  
30
```

## Task 4 – Uninitialized Variable Usage

(Buggy Code):

if True:

pass

print(total)

```
[13] if True:  
    pass  
    print(total)  
...   File "/tmp/ipython-input-1170978020.py", line 2  
        pass  
        ^  
        IndentationError: expected an indented block after 'if' statement on line 1  
Next steps: Explain error
```

Output:

```
◆ Gemini  
[13] if True:  
    pass  
+ total = 0 # Or any other initial value  
    print(total)  
-  
... 0
```

## Task 5 – Logical Error in Student Grading System

(Buggy Code):

marks = 85

if marks &gt;= 90:

grade = "A"

elif marks &gt;= 80:

grade = "C"

else:

grade = "B"

print(grade)

```
[16] ① 0s
▶ marks = 85
if marks >= 90:
    grade = "A"
elif marks >= 80:
    grade = "C"
else:
    grade = "B"
print(grade)

...
File "/tmp/ipython-input-2691675298.py", line 3
    grade = "A"
^
IndentationError: expected an indented block after 'if' statement on line 2

Next steps: Explain error
```

Output:

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

...
*** C
```