LAB ASSIGNMENT-7

TASK-1:

Prompt:

def add(a, b)

    return a + b

Fix the syntax error

Code and Output:

```python
def add(a, b):
    return a + b
```

Code Explanation:

- The colon : is required at the end of the def line
- The return line must be indented (typically 4 spaces)

If you had copied this from somewhere and the indentation was off, Python would throw an IndentationError.

TASK-2:

Prompt:

def count_down(n):

  while n >= 0:

    print(n)

    n += 1

Identify and fix a logic error in a loop that causes infinite iteration.

Code and Output:

```python
def count_down(n):
    while n >= 0:
        print(n)
        n -= 1  # Decrement n to avoid infinite loop
```

Code Explanation:

1)n += 1 increases the value of n, which means n will **never be less than 0**, so the loop never ends.

2)Changing it to n -= 1 ensures that n eventually drops below 0, exiting the loop.

TASK-3:

Prompt:

```
def divide(a,b):
    return a/b
print(divide(10,0))
```

Debug a runtime error caused by division by zero and insert try-except

Code and Output:

```python
def divide(a, b):
    try:
        return a / b
    except ZeroDivisionError:
        return "Error: Cannot divide by zero"

print(divide(10, 0))
```

Code Explanation:

Try attempts to perform the division.

Except ZeroDivisionError catches the error if b is zero. Instead of crashing, it returns a friendly error message.

TASK - 4:

Prompt:

class Rectangle:

def __init__(length, width):

self.length = length

self.width = width

Fix the given code.

Code and Output:

```python
class Rectangle:
    def __init__(self, length, width):
        self.length = length
        self.width = width
```

Code Explanation:

self is required as the first argument in all instance methods (including __init__) to refer to the current object.

Without self, Python will raise a TypeError.

TASK-5:

Prompt:

numbers = [1, 2, 3]

print(numbers[5])
Fix the given code.

Code and Output:

```python
numbers = [1, 2, 3]
print(numbers[2])
```

Code Explanation:

The error in the code is an **IndexError** because you're trying to access index 5, but the list numbers = [1, 2, 3] only has indices 0, 1, and 2.
numbers[0] → 1

numbers[1] → 2

numbers[2] → 3 ← This is valid