```
al\Programs\Python\Python312\python.exe' 'c:\Users\RAFTB\.vscode\extensions\ms-python.debugpy-2025.0.0-win32-x64\bundled\libs\debugpy\launcher' '58481
' '--' 'c:\Users\RAFTB\OneDrive\Documents\GroupProjectCOMSCI\fibr.py'
Enter the n: 7
F(7) = 8
PS C:\Users\RAFTB\OneDrive\Documents\GroupProjectCOMSCI>
```

```python
def fibonacci_recursive(n):
    """
    - parameter n: nth Fibonacci term
    - return: nth Fibonacci number
    """
    # 1. Base Cases
    if n <= 0: # F(0) = X
        return "n must be a positive integer >= 1"
    elif n == 1:
        return 0  # F(1) = 0
    elif n == 2:
        return 1  # F(2) = 1

    """
    F(0) = X
    F(1) = 0
    F(2) = 1
    F(3) = F(2) + F(1) = 1 + 0 = 1
    F(4) = F(3) + F(2) = 1 + 1 = 2
     > F(n) = F(n - 2) + F(n - 1)
    """
    # 2. Recursive Formula
    return fibonacci_recursive(n - 1) + fibonacci_recursive(n - 2)


# 3. User Input
try:  # Exception
    n = int(input("Enter the n: "))
    print(f"F({n}) = {fibonacci_recursive(n)}")
except ValueError:
    print("Please enter a valid integer!")


"""
 - Pros: Simple and intuitive code
 - Cons: Very slow (O(2^n)), high memory usage (stack overflow risk)
```