# AI ASSISTED CODING

## NAME: Kaveti Manohar

## En.No:2403A52079

## Batch:02

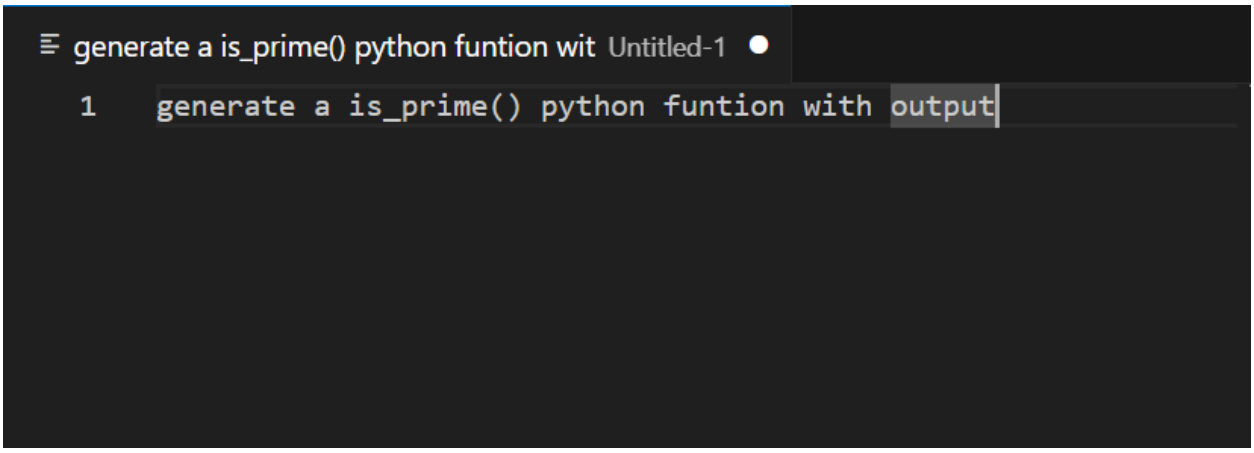
## Task Description#1

- Use Copilot to generate  a is_prime() python function.

## Expected Output#1

- Correct examples verify it True or False

## Question




## Code

```
    if n <= 1:
        return False
    if n == 2:
        return True
    if n % 2 == 0:
        return False
    for i in range(3, int(n ** 0.5) + 1, 2):
        if n % i == 0:
            return False
    return True

# Example usage:
for num in range(1, 21):
```

## Output

```
11 is prime: True
12 is prime: False
13 is prime: True
14 is prime: False
15 is prime: False
16 is prime: False
17 is prime: True
18 is prime: False
19 is prime: True
20 is prime: False
PS C:\Users\NEW\AppData\Local\Programs\Microsoft VS Code>
```

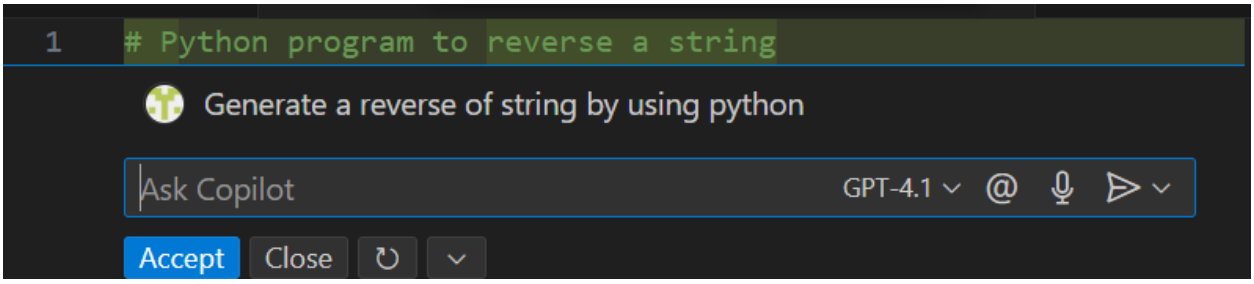
## Task Description#2

- **Generate a python program for Reverse of a string by using copilot.**

## Expected Output#2

➢ **Verify the correct output by giving the reverse string as input**

## Question




## Code

```python
# Python program to reverse a string

input_str = input("Enter a string: ")
reversed_str = input_str[::-1]
print("Reversed string:", reversed_str)
```

## Output

```
e "c:/Users/NEW/Desktop/AI assisted coading/reverse.py"
Enter a string: mom
Reversed string: mom
PS C:\Users\NEW\AppData\Local\Programs\Microsoft VS Code>
```

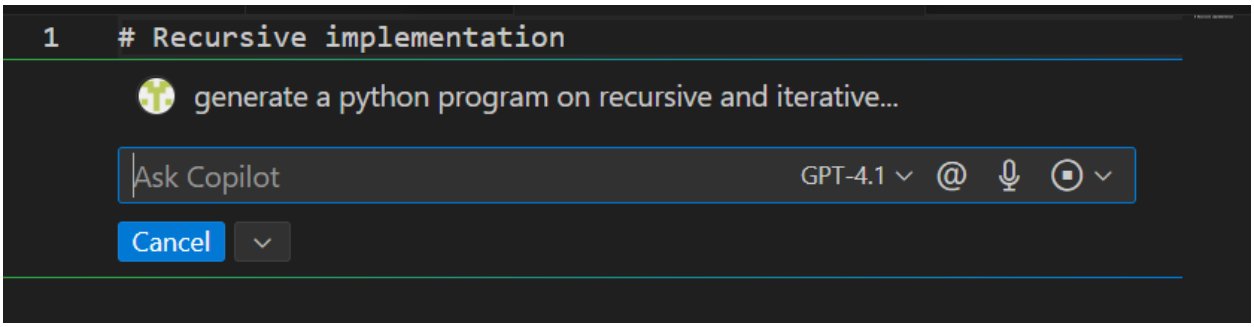
## Task Discription#3

- **Generate a python program for Recursive and Iterate using with python function.**

## Expected Output#3

- **Verify the expected output of the program.**

## Question




## Code

```python
# Recursive implementation of factorial
def factorial_recursive(n):
    if n == 0 or n == 1:
        return 1
    else:
        return n * factorial_recursive(n - 1)

# Iterative implementation of factorial
def factorial_iterative(n):
    result = 1
    for i in range(2, n + 1):
        result *= i
    return result

# Example usage
num = 5
print(f"Recursive factorial of {num}: {factorial_recursive(num
print(f"Iterative factorial of {num}: {factorial_iterative(num
```

## Output

```
:\Users\NEW\AppData\Local\Microsoft\WindowsApps\python3.13.ex
e "c:/Users/NEW/Desktop/AI assisted coading/Recursive.py"
Recursive factorial of 5: 120
Iterative factorial of 5: 120
PS C:\Users\NEW\AppData\Local\Programs\Microsoft VS Code>
```

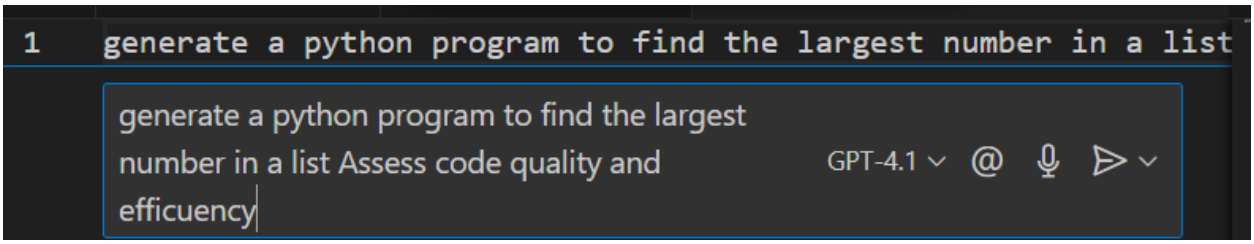
## Task Discription#4

- **Generate the python program for finding the Largest number and assiss by the copilot.**

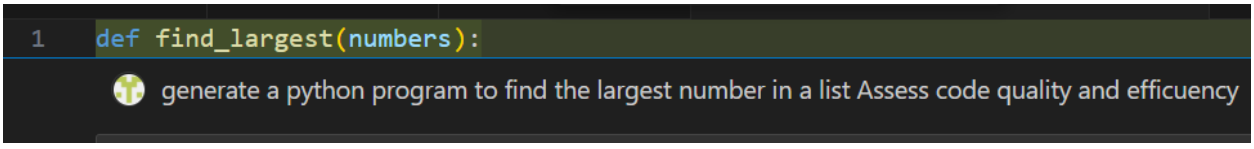
## Expected Output#4

➢ **Verify the expected output and review of the code.**

## Question




## Generated Code




## Code

```python
def find_largest(numbers):
    if not numbers:
        raise ValueError("The list is empty.")
    largest = numbers[0]
    for num in numbers[1:]:
        if num > largest:
            largest = num
    return largest

# Example usage:
nums = [3, 7, 2, 9, 4]
print("Largest number:", find_largest(nums))
```

## Output

```
                                                    > & C:\Users\NEW\AppData\Local\Microsoft
\WindowsApps\python3.13.exe "c:/Users/NEW/Desktop/AI assisted coading/Largest no.py"
Largest number: 9
PS C:\Users\NEW\AppData\Local\Programs\Microsoft VS Code>
```

----------THE END----------