TECHNIK NEST

INNOVATIVE MINDS, NESTING SUCCESS

Name: Maaz Sher Muhammad

Intern ID: TN/1N01/003

Email ID : maazshermuhammadofficial@gmail.com

Internship Domain : Python Internee

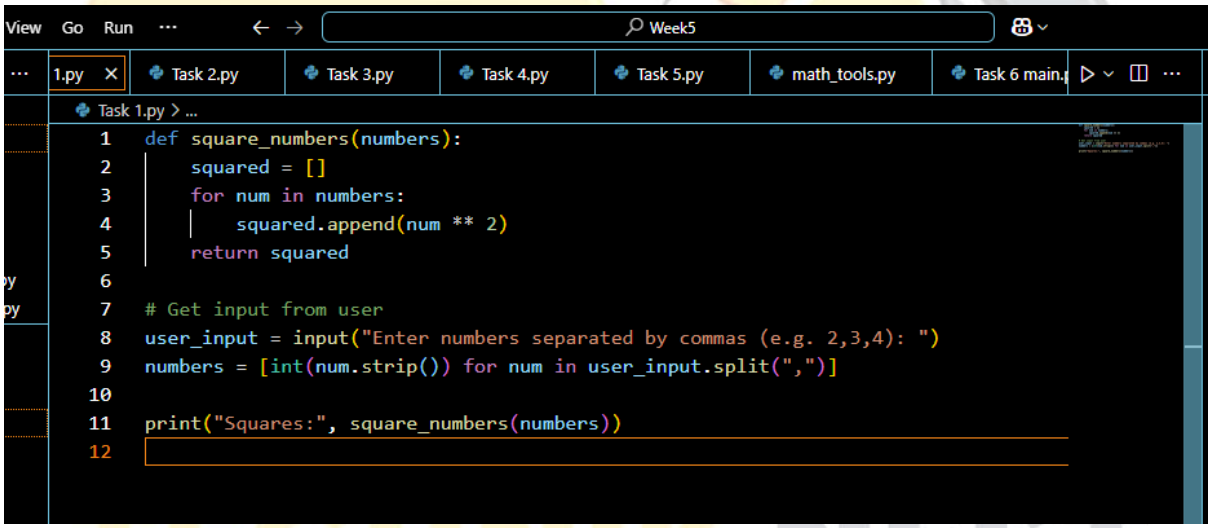
Task Week : 5

Instructor Name : Hassan Ali

Task 1 :

Code Snippet & Screenshot

```
def square_numbers(numbers):
    squared = []
    for num in numbers:
        squared.append(num ** 2)
    return squared
# Get input from user
user_input = input("Enter numbers separated by commas (e.g. 2,3,4): ")
numbers = [int(num.strip()) for num in user_input.split(",")]
print("Squares:", square_numbers(numbers))
```




```
Enter numbers separated by commas (e.g. 2,3,4): 5,7,8
Squares: [25, 49, 64]
```

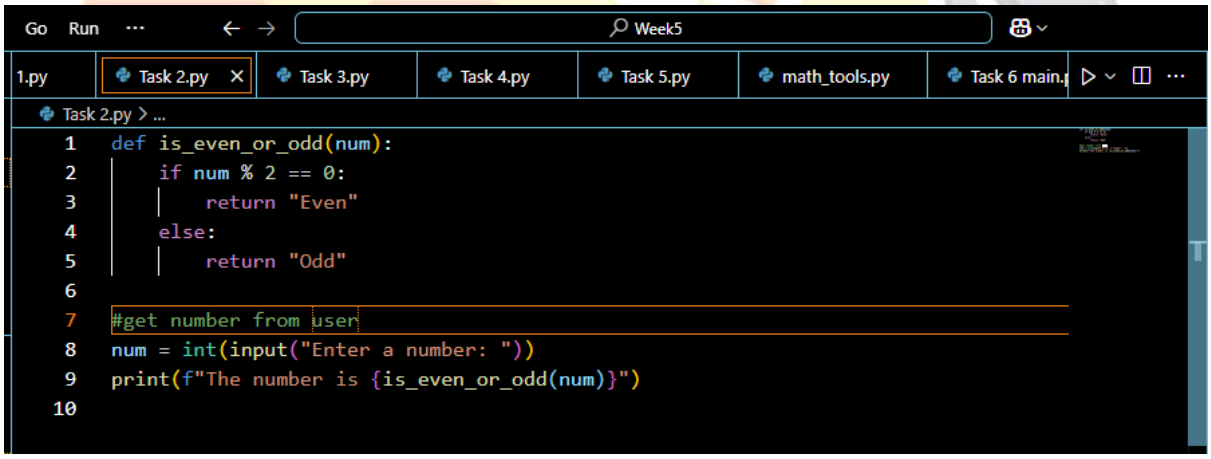
Challenges:

- Had to learn how to **split and clean user input** from a string to integers.
- Faced issues when users entered **spaces or non-numeric characters**, causing crashes.
- Initially forgot to **convert input into a list of integers** using list comprehension.

Task 2 :

Code Snippet & Screenshot

```
def is_even_or_odd(num):
    if num % 2 == 0:
        return "Even"
    else:
        return "Odd"
#get number from user
num = int(input("Enter a number: "))
print(f"The number is {is_even_or_odd(num)}")
```




```
Enter a number: 78
The number is Even
```

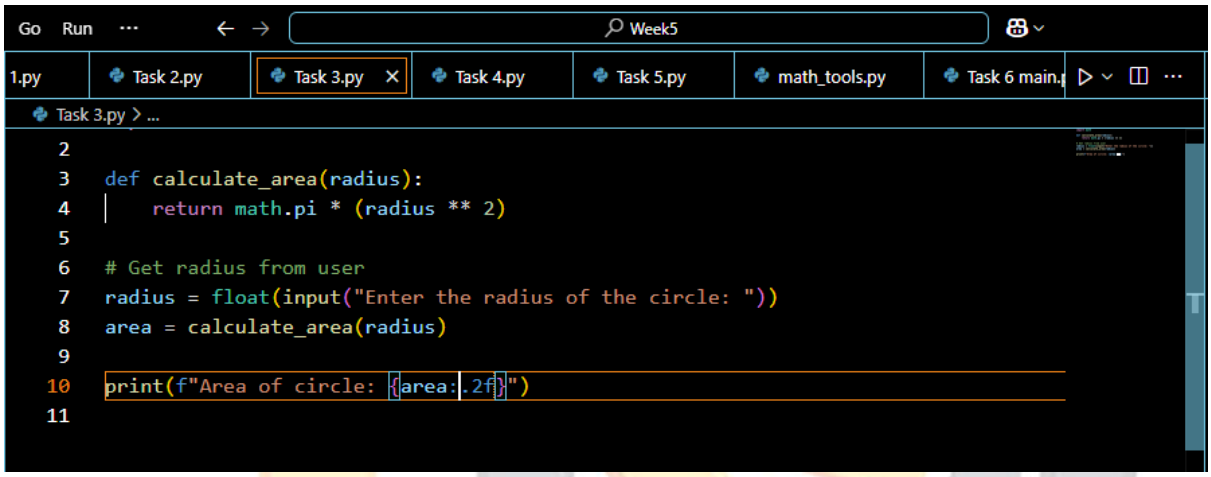
Challenges:

- Forgot to wrap input() inside int() — caused type errors.
- Had to understand how to handle **odd/even logic using modulo %**.
- Entered float values like 4.5 initially, which caused unexpected results or errors.

Task 3 :

Code Snippet & Screenshot

```
import math

def calculate_area(radius):

    return math.pi * (radius ** 2)

# Get radius from user

radius = float(input("Enter the radius of the circle: "))

area = calculate_area(radius)

print(f"Area of circle: {area:.2f}")
```




```
Enter the radius of the circle: 6767676
Area of circle: 143889462530015.34
```

Challenges:

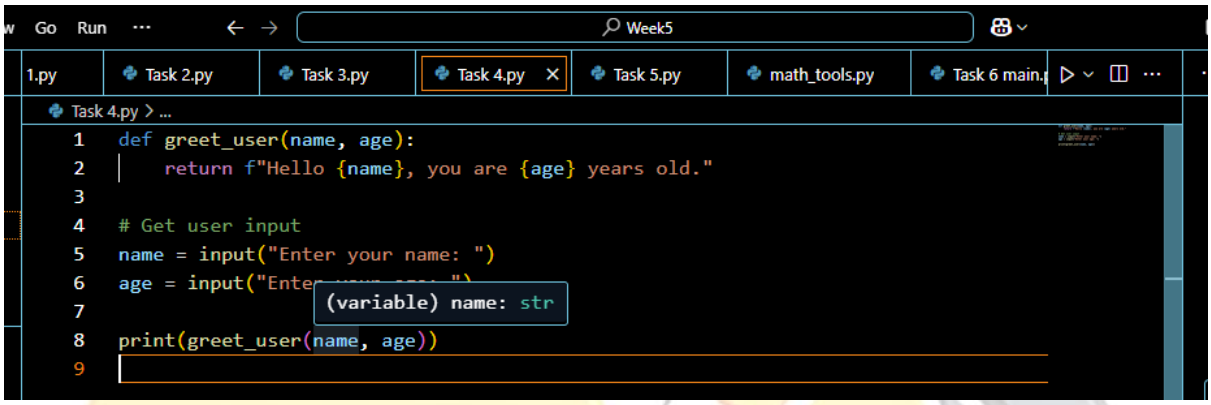
- Was confused whether to use math.pi or just 3.14.
- Inputting negative numbers gave wrong results — had to decide whether to **validate radius**.
- Got very long decimals in output — learned to format with :.2f.

Task 4 :

Code Snippet & Screenshot

```
def greet_user(name, age):

  return f"Hello {name}, you are {age} years old."
```

```
# Get user input
name = input("Enter your name: ")
age = input("Enter your age: ")
print(greet_user(name, age))
```




```
Enter your name: maaz sher muhammad
Enter your age: 21
Hello maaz sher muhammad, you are 21 years old.
```

Challenges:

- Initially used + for string joining — later learned f-strings are cleaner and better.
- Forgot to cast age as a string when joining — caused TypeError.
- Tried to add validation to stop empty names or non-numeric ages but wasn't sure how.

.

Task 5 :

Code Snippet & Screenshot

```
counter = 0  # Global variable


def change_counter():
    global counter
    counter += 1
```

```
    return counter

# Call multiple times

print("Counter now:", change_counter())

print("Counter now:", change_counter())
```


```
Counter now: 1
Counter now: 2
```

Challenges:

- Didn't understand why the function needed global — thought the variable would update automatically.
- Tried using counter += 1 without declaring it global — resulted in UnboundLocalError.
- Wasn't sure where the global variable should be declared in the script.

Task 6 :

Code Snippet & Screenshot

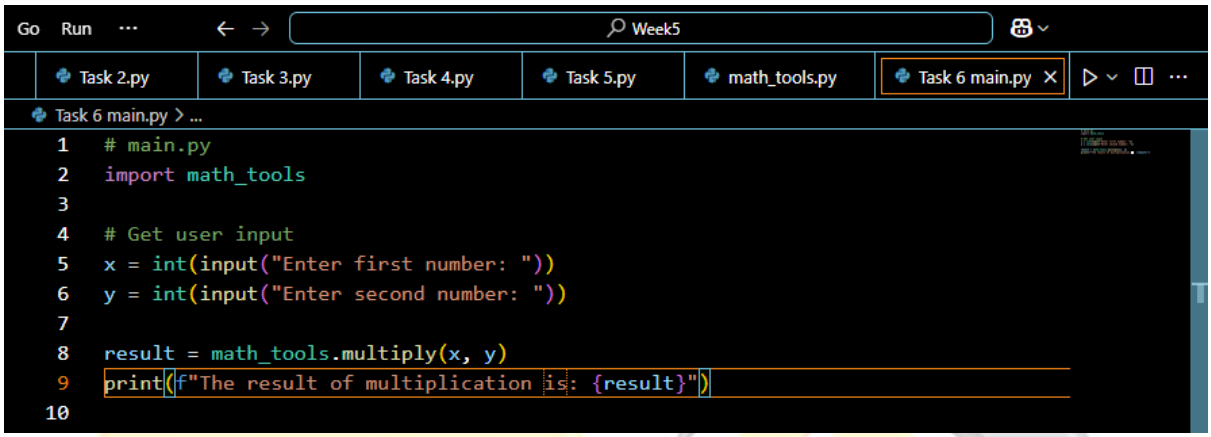
```
# main.py

import math_tools


# Get user input
```

x = int(input("Enter first number: "))

y = int(input("Enter second number: "))

result = math_tools.multiply(x, y)

print(f"The result of multiplication is: {result}")



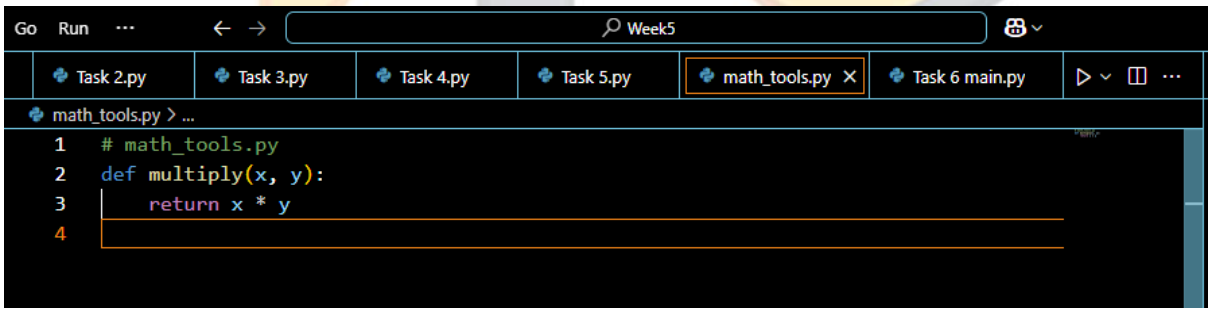

```
Enter first number: 01
Enter second number: 21
The result of multiplication is: 21
```

math_tools.py:

math_tools.py

def multiply(x, y):

    return x * y



Challenges:

- Faced error ModuleNotFoundError because filename had a **space instead of underscore**.
- Didn't know both .py files must be in the **same folder** for import to work.
- Wasn't sure how to name files properly or avoid using special characters in filenames.