

ASSIGNMENT-7.2

NAME: TEENESWARI

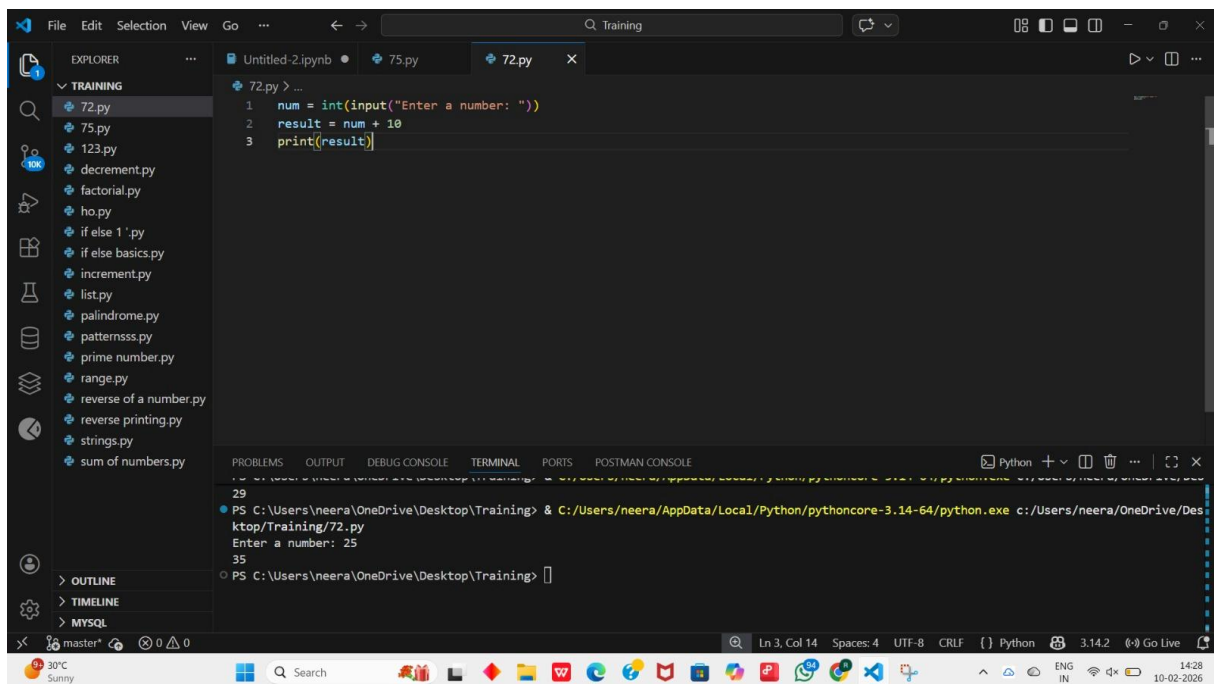
ROLLNO: 2303A51932

BATCH:30

TASK-1 PROMPT: num =

input("Enter a number: ") result

= num + 10 print(result) CODE:



The screenshot shows a Python IDE with a file explorer on the left containing various Python files. The main editor displays a file named '72.py' with the following code:

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

The terminal at the bottom shows the command prompt execution:

```
PS C:\Users\neera\OneDrive\Desktop\Training> & C:/Users/neera/AppData/Local/Python/pythoncore-3.14-64/python.exe c:/Users/neera/OneDrive/Desktop/Training/72.py
Enter a number: 25
35
PS C:\Users\neera\OneDrive\Desktop\Training>
```

OBSERVATION:

The program was executed successfully. It accepts a number from the user using the input function. The entered value is added to 10 and the result is displayed. The output verifies the execution of input handling and addition operation in the program.

TASK-2

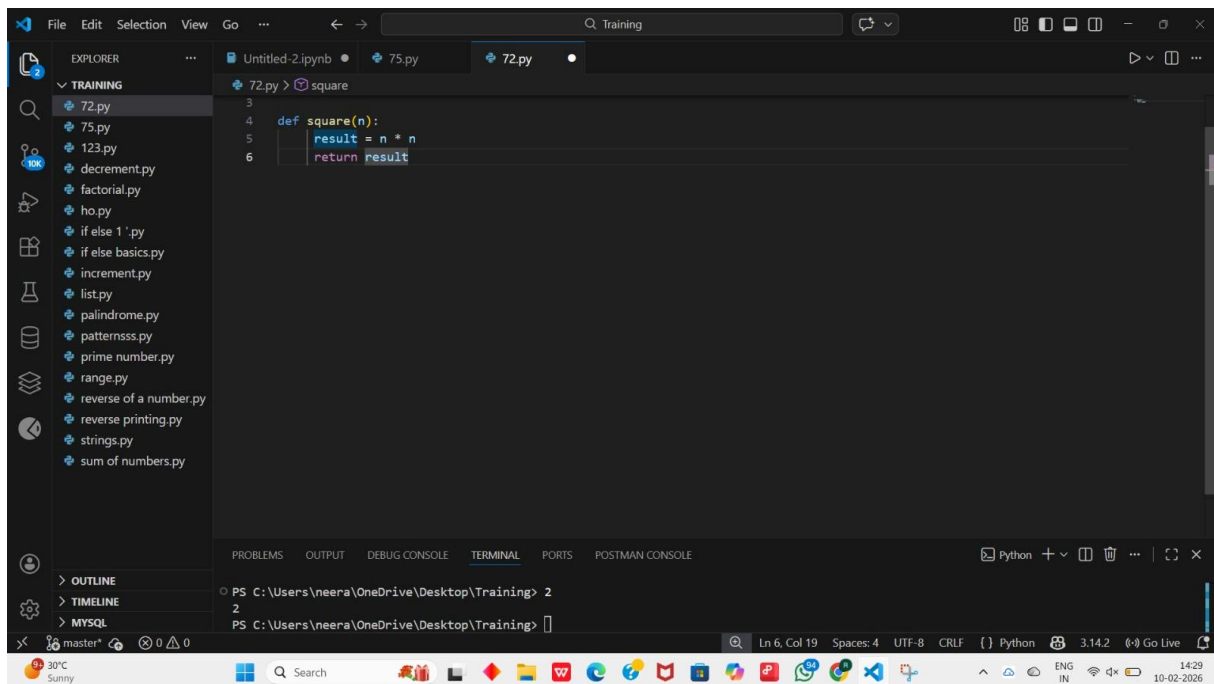
PROMPT:

def square(n):

 result = n * n

return result

CODE:



The screenshot shows a Python IDE with a file explorer on the left containing a folder named 'TRAINING' with several Python files. The main editor window displays a file named '72.py' with a function definition:

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

 The bottom panel shows a terminal with the command `PS C:\Users\neera\OneDrive\Desktop\Training> 2` and the output `2`. The status bar at the bottom indicates the file is at line 6, column 19, with 4 spaces, using UTF-8 encoding and CRLF line endings. The system tray shows a temperature of 30°C and the date 10-02-2026.

OBSERVATION:

The function `square(n)` was executed successfully. It accepts a number as an argument, calculates its square by multiplying the number with itself, and returns the result. The function works correctly and produces the expected output.

TASK-3

PROMPT:

`numbers = [10, 20, 30]` for `i` in

`range(0, len(numbers))`:

`print(numbers[i])` CODE:

The screenshot shows a Python IDE with a file explorer on the left containing a folder named 'TRAINING'. Inside this folder, there are several Python files, including '72.py', '75.py', '123.py', 'decrement.py', 'factorial.py', 'ho.py', 'if else 1'.py', 'if else basics.py', 'increment.py', 'list.py', 'palindrome.py', 'patternsss.py', 'prime number.py', 'range.py', 'reverse of a number.py', 'reverse printing.py', 'strings.py', and 'sum of numbers.py'. The main editor window displays the code in '72.py':

```
7
8 numbers = [10, 20, 30]
9 for i in range(0, len(numbers)+1):
10     print(numbers[i])
```

Below the editor, the terminal window shows the command prompt output:

```
PS C:\Users\neera\OneDrive\Desktop\Training> & C:/Users/neera/AppData/Local/Python/pythoncore-3.14-64/python.exe c:/Users/neera/OneDrive/Desktop/Training/72.py
20
20
20
20
PS C:\Users\neera\OneDrive\Desktop\Training>
```

The status bar at the bottom indicates the current line is 10, column 20, with 4 spaces, UTF-8 encoding, CRLF line endings, and Python 3.14.2 interpreter.

OBSERVATION:

The program was executed successfully. A list of numbers was created and a for loop was used to iterate through each element using index values. Each number in the list was printed as output. The results confirm that the loop and list operations are functioning correctly.

TASK-4

PROMPT:

total = 60 if

True:

```
print(total)
```

CODE:

```
13 total = 0
14 total += numbers[i]
15
16 print(total)
```

```
PS C:\Users\neera\OneDrive\Desktop\Training> & C:/Users/neera/AppData/Local/Python/pythoncore-3.14-64/python.exe c:/Users/neera/OneDrive/Desktop/Training/72.py
10
20
30
60
PS C:\Users\neera\OneDrive\Desktop\Training>
```

OBSERVATION:

The program was executed successfully. A variable was assigned a value and an if condition was used to check the statement. Since the condition is True, the value of the variable was printed. The output confirms the correct working of conditional statements in the program.

TASK-5

PROMPT: marks

= 85 if marks >=

90: grade = "A"

elif marks >= 80:

grade = "C" else:

grade = "B"

print(grade) CODE:

The screenshot shows the Visual Studio Code interface. The Explorer panel on the left lists files in a folder named 'TRAINING', including '72.py', '75.py', '123.py', and several other Python files. The main editor window displays the code in '72.py':

```
16 print(total)
17
18
19 marks = 85
20 if marks >= 90:
21     grade = "A"
22 elif marks >= 80:
23     grade = "B"
24 else:
25     grade = "C"
26 print(grade)
```

Below the editor, the TERMINAL panel shows the command prompt output:

```
PS C:\Users\neera\OneDrive\Desktop\Training> & C:/Users/neera/AppData/Local/Python/pythoncore-3.14-64/python.exe c:/Users/neera/OneDrive/Desktop/Training/72.py
10
20
30
60
B
PS C:\Users\neera\OneDrive\Desktop\Training>
```

The status bar at the bottom indicates the file is '72.py' at line 26, column 1, using UTF-8 encoding and Python 3.14.2.

OBSERVATION:

The program was executed successfully. It assigns marks to a variable and uses conditional statements to determine the grade. Based on the given condition, the appropriate grade is selected and displayed. The output confirms the correct working of the if-elif-else control structure.