

Assignment 3.1

2303A510A1 - Bhanu

Batch - 14

Lab Experiment: Prompt Engineering – Improving Prompts and Context Management

Question 1: Zero-Shot Prompting (Palindrome Number Program)

Prompt:

Write a Python function that checks whether a given number is a palindrome and returns True or False.

Code & Output:

The screenshot shows a Jupyter Notebook cell with the following Python code:

```
1 #Generate a Python function that checks whether a given number is a palindrome
2 #and returns True or False
3 def is_palindrome_number(num):
4     return str(num) == str(num)[::-1]
5 num = int(input("Enter a number : "))
6 print(f"{num} : {is_palindrome_number(num)}")
7
```

Below the code cell, the terminal output shows the execution of the script and the user's input:

```
PS C:\Users\danda\OneDrive\Documents\3.2\AIA> & C:\Users\danda\AppData\Local\Programs\Python\Python313\python.exe
c:/Users/danda/OneDrive/Documents/3.2/AIA/lab-3.1.py
Enter a number : 23432
23432 : True
PS C:\Users\danda\OneDrive\Documents\3.2\AIA>
```

Explanation:

- Define a function called `is_palindrome_number` that takes a single argument `num`.
- Convert the number to a string using `str(num)`
- Reverse the string using `[::-1]`
- Compare the original string with the reversed string using `==`
- Return the result of the comparison
- Print the result of the function call

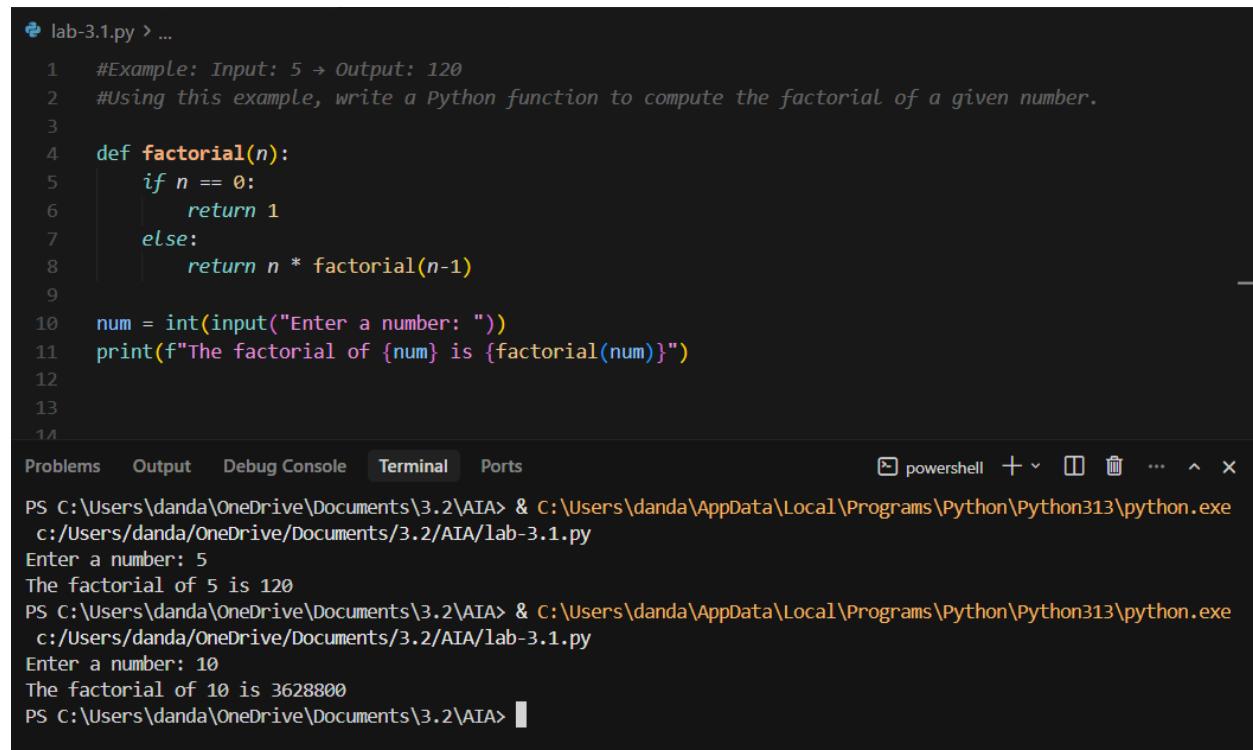
Question 2: One-Shot Prompting (Factorial Calculation)

Prompt:

Example: Input: 5 → Output: 120

Using this example, write a Python function to compute the factorial of a given number.

Code & Output :



```

lab-3.1.py > ...
1 #Example: Input: 5 → Output: 120
2 #Using this example, write a Python function to compute the factorial of a given number.
3
4 def factorial(n):
5     if n == 0:
6         return 1
7     else:
8         return n * factorial(n-1)
9
10 num = int(input("Enter a number: "))
11 print(f"The factorial of {num} is {factorial(num)}")
12
13
14
Problems    Output    Debug Console    Terminal    Ports
PS C:\Users\danda\OneDrive\Documents\3.2\AIA> & C:\Users\danda\AppData\Local\Programs\Python\Python313\python.exe
c:/Users/danda/OneDrive/Documents/3.2/AIA/lab-3.1.py
Enter a number: 5
The factorial of 5 is 120
PS C:\Users\danda\OneDrive\Documents\3.2\AIA> & C:\Users\danda\AppData\Local\Programs\Python\Python313\python.exe
c:/Users/danda/OneDrive/Documents/3.2/AIA/lab-3.1.py
Enter a number: 10
The factorial of 10 is 3628800
PS C:\Users\danda\OneDrive\Documents\3.2\AIA>

```

Explanation:

- Define a function called `factorial` that takes a single argument `n`.
- If `n` is 0, return 1

- Else, return $n * \text{factorial}(n-1)$
 - Print the result of the function call

Question 3: Few-Shot Prompting (Armstrong Number Check)

Prompt:

Write a Python function to check whether a number is an Armstrong number.

Code & Output:

Explanation:

- Define a function called `is_armstrong_number` that takes a single argument `num`.
 - Initialize a variable called `sum` to 0.
 - Initialize a variable called `temp` to `num`.
 - Initialize a variable called `count` to the length of the number.

- While temp is greater than 0, do the following:
- Get the last digit of temp using temp % 10.
- Add the cube of the last digit to sum.
- Divide temp by 10 to remove the last digit.
- Return True if sum is equal to num, otherwise return False.
- Print the result of the function call.

Question 4: Context-Managed Prompting (Optimized Number Classification)

Prompt:

Write an optimized Python program that classifies a number as prime, composite, or neither. Ensure proper input validation and efficient logic.

Code & Output:

```

lab-3.1.py > ...
1  # Write an optimized Python program that classifies a number as prime, composite, or neither.
2  # Ensure proper input validation and efficient logic.
3  def classify_number(num):
4      if num <= 1:
5          return "Neither"
6      for i in range(2, int(num**0.5) + 1):
7          if num % i == 0:
8              return "Composite"
9      return "Prime"
10 num = int(input("Enter a number: "))
11 print(f"{num} is {classify_number(num)}")

Problems   Output   Debug Console   Terminal   Ports   powershell   +   ×
PS C:\Users\danda\OneDrive\Documents\3.2\AIA> & C:\Users\danda\AppData\Local\Programs\Python\Python313\python.exe
c:/Users/danda/OneDrive/Documents/3.2/AIA/lab-3.1.py
Enter a number: 5
5 is Prime
PS C:\Users\danda\OneDrive\Documents\3.2\AIA> & C:\Users\danda\AppData\Local\Programs\Python\Python313\python.exe
c:/Users/danda/OneDrive/Documents/3.2/AIA/lab-3.1.py
Enter a number: 6
6 is Composite
PS C:\Users\danda\OneDrive\Documents\3.2\AIA> & C:\Users\danda\AppData\Local\Programs\Python\Python313\python.exe
c:/Users/danda/OneDrive/Documents/3.2/AIA/lab-3.1.py
Enter a number: 1
1 is Neither
PS C:\Users\danda\OneDrive\Documents\3.2\AIA>

```

Explanation:

- Define a function called classify_number that takes a single argument num.

- If num is less than or equal to 1, return "Neither".
- For i in range (2, int(num**0.5) + 1), do the following:
- If num is divisible by i, return "Composite".
- Return "Prime".
- Print the result of the function call.

Question 5: Zero-Shot Prompting (Perfect Number Check)

Prompt:

Write a Python function to check whether a given number is a perfect number.

Code & Output:

```
lab-3.1.py > ...
1  #Write a Python function to check whether a given number is a perfect number
2  def is_perfect_number(num):
3      sum = 0
4      for i in range(1, num):
5          if num % i == 0:
6              sum += i
7      return sum == num
8  num = int(input("Enter a number: "))
9  if is_perfect_number(num):
10     print(f"{num} is a perfect number")
11 else:
12     print(f"{num} is not a perfect number")

Problems    Output    Debug Console    Terminal    Ports
PS C:\Users\danda\OneDrive\Documents\3.2\AIA> & C:\Users\danda\AppData\Local\Programs\Python\Python313\python.exe
c:/Users/danda/OneDrive/Documents/3.2/AIA/lab-3.1.py
Enter a number: 6
6 is a perfect number
PS C:\Users\danda\OneDrive\Documents\3.2\AIA> & C:\Users\danda\AppData\Local\Programs\Python\Python313\python.exe
c:/Users/danda/OneDrive/Documents/3.2/AIA/lab-3.1.py
Enter a number: 28
28 is a perfect number
PS C:\Users\danda\OneDrive\Documents\3.2\AIA> & C:\Users\danda\AppData\Local\Programs\Python\Python313\python.exe
c:/Users/danda/OneDrive/Documents/3.2/AIA/lab-3.1.py
Enter a number: 54
54 is not a perfect number
PS C:\Users\danda\OneDrive\Documents\3.2\AIA>
```

Explanation:

- Define a function called `is_perfect_number` that takes a single argument `num`.
- Initialize a variable called `sum` to 0.
- For `i` in `range (1, num)`, do the following:

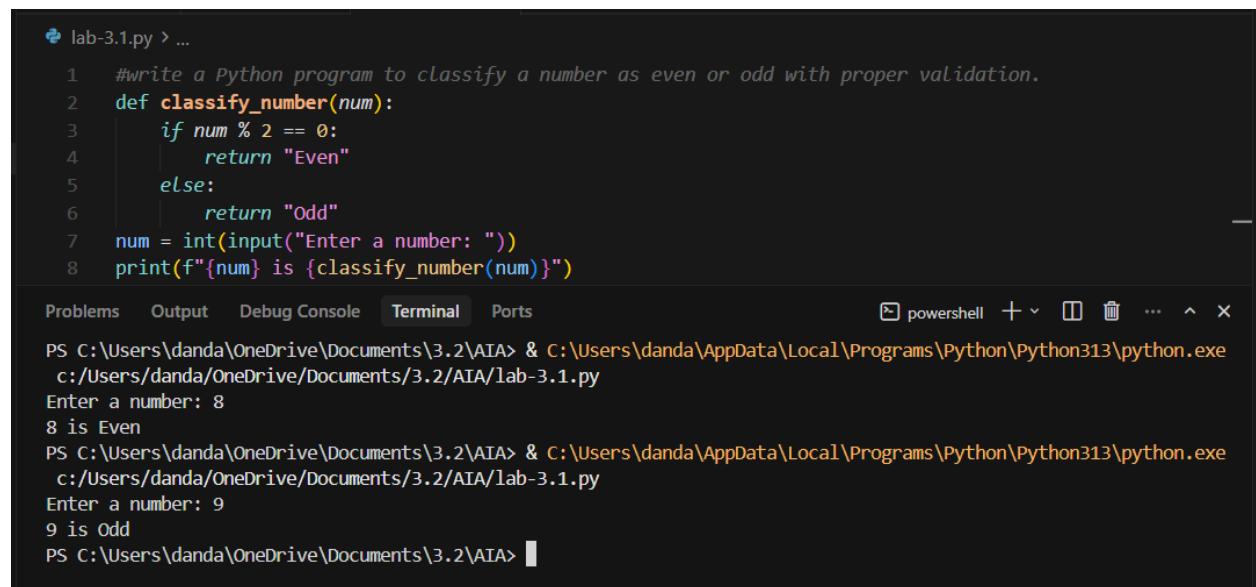
- If num is divisible by i, add i to sum.
- Return True if sum is equal to num, otherwise return False.
- Print the result of the function call.

Question 6: Few-Shot Prompting (Even or Odd Classification with Validation)

Prompt:

Write a Python program to classify a number as even or odd with proper validation.

Code & Output:



```

lab-3.1.py > ...
1  #write a Python program to classify a number as even or odd with proper validation.
2  def classify_number(num):
3      if num % 2 == 0:
4          return "Even"
5      else:
6          return "Odd"
7  num = int(input("Enter a number: "))
8  print(f"{num} is {classify_number(num)}")

Problems   Output   Debug Console   Terminal   Ports
PS C:\Users\danda\OneDrive\Documents\3.2\AIA> & C:\Users\danda\AppData\Local\Programs\Python\Python313\python.exe
c:/Users/danda/OneDrive/Documents/3.2/AIA/lab-3.1.py
Enter a number: 8
8 is Even
PS C:\Users\danda\OneDrive\Documents\3.2\AIA> & C:\Users\danda\AppData\Local\Programs\Python\Python313\python.exe
c:/Users/danda/OneDrive/Documents/3.2/AIA/lab-3.1.py
Enter a number: 9
9 is Odd
PS C:\Users\danda\OneDrive\Documents\3.2\AIA>

```

Explanation:

- Define a function called `classify_number` that takes a single argument `num`.
- If `num` is divisible by 2, return "Even".
- Else, return "Odd".
- Print the result of the function call.
- Print the number and the result of the function call.