

AI Assistant Coding

Assignment – 3.5

Name : G.Samshray

Batch : 23

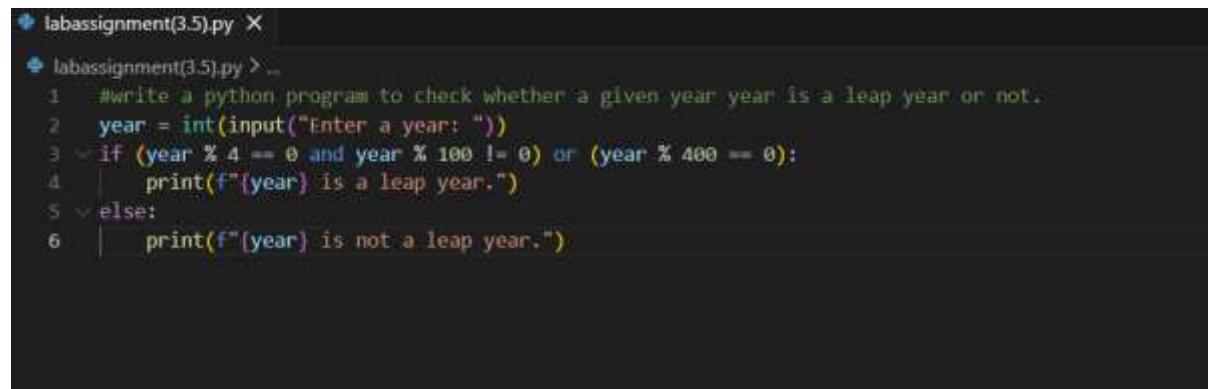
Ht.no : 2303A51670

Question 1: Zero-Shot Prompting (Leap Year Check)

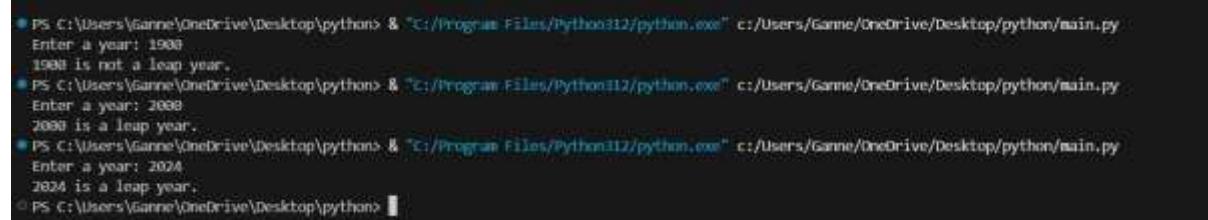
Write a zero-shot prompt to generate a Python function that checks whether a given year is a leap year.

Task:

- Record the AI-generated code.
- Test with years like 1900, 2000, 2024.
- Identify logical flaws or missing conditions.



```
# labassignment(3.5).py
# write a python program to check whether a given year year is a leap year or not.
year = int(input("Enter a year: "))
if (year % 4 == 0 and year % 100 != 0) or (year % 400 == 0):
    print(f"{year} is a leap year.")
else:
    print(f"{year} is not a leap year.")
```



```
PS C:\Users\Ganne\OneDrive\Desktop\python> & "C:/Program Files/Python312/python.exe" c:/Users/Ganne/OneDrive/Desktop/python/main.py
Enter a year: 1900
1900 is not a leap year.
PS C:\Users\Ganne\OneDrive\Desktop\python> & "C:/Program Files/Python312/python.exe" c:/Users/Ganne/OneDrive/Desktop/python/main.py
Enter a year: 2000
2000 is a leap year.
PS C:\Users\Ganne\OneDrive\Desktop\python> & "C:/Program Files/Python312/python.exe" c:/Users/Ganne/OneDrive/Desktop/python/main.py
Enter a year: 2024
2024 is a leap year.
PS C:\Users\Ganne\OneDrive\Desktop\python>
```

Question 2: One-Shot Prompting (GCD of Two Numbers)

Write a one-shot prompt with one example to generate a Python function that finds the Greatest Common Divisor (GCD) of two numbers.

Example:

Input: 12, 18 → Output: 6

Task:

- Compare with a zero-shot solution.
- Analyze algorithm efficiency.

```
#generate a python program to find the gcd of two numbers :12 and 18
def gcd(a, b):
    while b:
        a, b = b, a % b
    return a
num1 = 12
num2 = 18
print(f"The GCD of {num1} and {num2} is {gcd(num1, num2)}")
```

```
PS C:\Users\Ganne\OneDrive\Desktop\python> & "C:/Program Files/Python312/python.exe" c:/Users/Ganne/OneDrive/Desktop/python/main.py
The GCD of 12 and 18 is 6
PS C:\Users\Ganne\OneDrive\Desktop\python> []
```

Question 3: Few-Shot Prompting (LCM Calculation)

Write a few-shot prompt with multiple examples to generate a Python function that computes the Least Common Multiple (LCM).

Examples:

- Input: 4, 6 → Output: 12
- Input: 5, 10 → Output: 10
- Input: 7, 3 → Output: 21

Task:

- Examine how examples guide formula selection.
- Test edge cases.

```
21  #generate a python code to find LCM of two numbers
22
23  examples:
24  input:4 and 6
25  output:12
26  input:5 and 10
27  output:10
28  input:7 and 3
29  output:21
30
31  def lcm(a, b):
32      def gcd(x, y):
33          while y:
34              x, y = y, x % y
35          return x
36      return abs(a * b) // gcd(a, b)
37  num1 = int(input("Enter first number: "))
38  num2 = int(input("Enter second number: "))
39  print(f"The LCM of {num1} and {num2} is {lcm(num1, num2)})"
40
```

```
● PS C:\Users\Ganne\OneDrive\Desktop\python> & "C:/Program Files/Python312/python.exe" c:/Users/Ganne/OneDrive/Desktop/python/main.py
Enter first number: 4
Enter second number: 6
The LCM of 4 and 6 is 12
● PS C:\Users\Ganne\OneDrive\Desktop\python> & "C:/Program Files/Python312/python.exe" c:/Users/Ganne/OneDrive/Desktop/python/main.py
Enter first number: 5
Enter second number: 10
The LCM of 5 and 10 is 10
● PS C:\Users\Ganne\OneDrive\Desktop\python> & "C:/Program Files/Python312/python.exe" c:/Users/Ganne/OneDrive/Desktop/python/main.py
Enter first number: 7
Enter second number: 3
The LCM of 7 and 3 is 21
● PS C:\Users\Ganne\OneDrive\Desktop\python>
```

Question 4: Zero-Shot Prompting (Binary to Decimal Conversion)

Write a zero-shot prompt to generate a Python function that converts a binary number to decimal.

Task:

- Test with valid and invalid binary inputs.
- Identify missing validation logic.

```
27 #generate a python code that converts a binary number into decimal.
28
29 def binary_to_decimal(binary_str):
30     decimal_number = 0
31     binary_str = binary_str[::-1] # Reverse the string for easier calculation
32     for index, digit in enumerate(binary_str):
33         if digit == '1':
34             decimal_number += 2 ** index
35
36     return decimal_number
37 binary_input = input("Enter a binary number: ")
38 decimal_output = binary_to_decimal(binary_input)
39 print(f"The decimal equivalent of binary {binary_input} is {decimal_output}")
```

```
PS C:\Users\Ganne\OneDrive\Desktop\python> & "C:/Program Files/Python312/python.exe" c:/Users/Ganne/OneDrive/Desktop/python/main.py
Enter a binary number: 101
Enter a binary number: 101
The decimal equivalent of binary 101 is 5
The decimal equivalent of binary 101 is 5
PS C:\Users\Ganne\OneDrive\Desktop\python> & "C:/Program Files/Python312/python.exe" c:/users/Ganne/OneDrive/Desktop/python/main.py
Enter a binary number: 100
The decimal equivalent of binary 100 is 4
PS C:\Users\Ganne\OneDrive\Desktop\python>
```

Question 5: One-Shot Prompting (Decimal to Binary Conversion)

Write a one-shot prompt with an example to generate a Python function that converts a decimal number to binary.

Example:

Input: 10 → Output: 1010

Task:

- Compare clarity with zero-shot output.
- Analyze handling of zero and negative numbers.

The screenshot shows a Jupyter Notebook interface. The code cell contains the following Python script:

```
41: # generate a python program to convert decimal number to binary number.
42: def decimal_to_binary(decimal_number):
43:     if decimal_number == 0:
44:         return "0"
45:     binary_str = ""
46:     while decimal_number > 0:
47:         binary_str = str(decimal_number % 2) + binary_str
48:         decimal_number //= 2
49:     return binary_str
50:
51: decimal_input = int(input("Enter a decimal number: "))
52: binary_output = decimal_to_binary(decimal_input)
53: print(f"The binary equivalent of decimal {decimal_input} is {binary_output}")
```

Below the code cell, the Jupyter interface shows tabs for PROBLEMS, DEBUG CONSOLE, OUTPUT, TERMINAL, and PORTS. The OUTPUT tab is selected, displaying the terminal session:

```
PROBLEMS DEBUG CONSOLE OUTPUT TERMINAL PORTS

● PS C:\Users\Ganne\OneDrive\Desktop\python> & "C:/Program Files/Python312/python.exe" c:/Users/Ganne/OneDrive/Desktop/python/main.py
Enter a decimal number: 1810
The binary equivalent of decimal 1810 is 1111110010
● PS C:\Users\Ganne\OneDrive\Desktop\python> & "C:/Program Files/Python312/python.exe" c:/Users/Ganne/OneDrive/Desktop/python/main.py
Enter a decimal number: 10
The binary equivalent of decimal 10 is 1010
PS C:\Users\Ganne\OneDrive\Desktop\python>
```



```
● PS C:\Users\Ganne\OneDrive\Desktop\python> & "C:/Program Files/Python312/python.exe" c:/Users/Ganne/OneDrive/Desktop/python/main.py
Enter a number: 18
18 is a Harshad number.
● PS C:\Users\Ganne\OneDrive\Desktop\python> & "C:/Program Files/Python312/python.exe" c:/Users/Ganne/OneDrive/Desktop/python/main.py
Enter a number: 45
45 is a Harshad number.
● PS C:\Users\Ganne\OneDrive\Desktop\python> & "C:/Program Files/Python312/python.exe" c:/Users/Ganne/OneDrive/Desktop/python/main.py
Enter a number: 98
98 is not a Harshad number.
● PS C:\Users\Ganne\OneDrive\Desktop\python>
```

Question 6: Few-Shot Prompting (Harshad Number Check)

Write a few-shot prompt to generate a Python function that checks whether a number is a Harshad (Niven) number.

Examples:

- Input: 18 → Output: Harshad Number
- Input: 21 → Output: Harshad Number
- Input: 19 → Output: Not a Harshad Number

Task:

- Test boundary conditions.
- Evaluate robustness

```
54 #generate a python code to find whether the given number is harshad number or not
55 """
56 examples
57 Input: 18
58 Output: Harshad Number
59
60 Input: 21
61 Output: Harshad Number
62
63 Input: 19
64 Output: Not a Harshad Number"""
65 num = int(input("Enter a number: "))
66 sum_of_digits = sum(int(digit) for digit in str(num))
67 if num % sum_of_digits == 0:
68     print(f"{num} is a Harshad Number")
69 else:
70     print(f"{num} is Not a Harshad Number")
```

```
● PS C:\Users\Ganne\OneDrive\Desktop\python> & "C:/Program Files/Python312/python.exe" c:/users/Ganne/OneDrive/Desktop/python/main.py
Enter a number: 18
18 is a Harshad number.
● PS C:\Users\Ganne\OneDrive\Desktop\python> & "C:/Program Files/Python312/python.exe" c:/users/Ganne/OneDrive/Desktop/python/main.py
Enter a number: 45
45 is a Harshad number.
● PS C:\Users\Ganne\OneDrive\Desktop\python> & "C:/Program Files/Python312/python.exe" c:/users/Ganne/OneDrive/Desktop/python/main.py
Enter a number: 98
98 is not a Harshad number.
○ PS C:\Users\Ganne\OneDrive\Desktop\python>
```