SCHOOL OF COMPUTER SCIENCE AND ARTIFICIAL INTELLIGENCE	Name:Alluri chaithra Id :2403A51105
AssignmentNumber:3.2(Present assignment number)/**24**(Total number of assignments)	

1

Task Description#1

- Ask AI to write a function to calculate compound interest, starting with only the function name. Then add a docstring, then input-output example

Expected Output#1

- Comparison of AI-generated code styles

Code And Output:

```
def compound_interest(principal, rate, time, n):
    """Calculates compound interest.

    Args:
      principal: The initial principal amount.
      rate: The annual interest rate (as a decimal).
      time: The number of years the money is invested for.
      n: The number of times that interest is compounded per year.

    Returns:
      The final amount after compounding.
    """
    amount = principal * (1 + rate / n)**(n * time)
    return amount

# Example usage:
principal = 1000
rate = 0.05  # 5% annual interest rate
time = 10    # 10 years
n = 4      # Compounded quarterly

final_amount = compound_interest(principal, rate,
print(f"The final amount after compounding is: {fi
```

```
The final amount after compounding is: 1643.62
```

give acode calculate compound interest with function

Certainly, I can provide a Python function to calculate compound interest.

I have provided the code to calculate compound interest using a function. Is there anything else you would like to do with this code or perhaps explore other financial calculations?

What can I help you build?

```
def compound_interest(principal, rate, time, n):
```

write aprogram in python calculate compound interest with function. Then add a docstring, then input-output

Ask or edit in context GPT-4.1

Accept Close

```
    """
    Calculate compound interest.

    Args:
        principal (float): The initial amount of money.
        rate (float): Annual interest rate (in percent).
        time (float): Number of years.
        n (int): Number of times interest applied per year.

    Returns:
        float: The compound interest earned.
    """
    amount = principal * (1 + rate / (100 * n)) ** (n * time)
    return amount - principal

# Input
p = float(input("Enter principal amount: "))
r = float(input("Enter annual interest rate (in %): "))
t = float(input("Enter time (in years): "))
n = int(input("Enter number of times interest applied per year: "))

# Output
ci = compound_interest(p, r, t, n)
print(f"Compound Interest: {ci:.2f}")
```

OUTPUT PROBLEMS DEBUG CONSOLE TERMINAL PORTS

```
Enter principal amount: 10000
Enter annual interest rate (in %): 10
Enter time (in years): 4
Enter number of times interest applied per year: 2
Compound Interest: 4774.55
```

Explanation:

write aprogram in python calculate compound interest with function. Then add a docstring, then input-output

/explain explain code line by line

Certainly! Here's a line-by-line explanation of your code:

```
def compound_interest(principal, rate, time, n):
```

- **Defines a function** named compound_interest with four parameters:

Ask or edit in context GPT-4.1

Accept Close

Task Description#2

- Do math stuff, then refine it to: # Write a function to calculate average, median, and mode of a list of numbers.

Expected Output#2

- AI-generated function evolves from unclear to accurate multi-statistical operat

give me python code to find mean median and mode of list os 10 numbers

Ask or edit in context

Accept Close

```
# List of 10 numbers
numbers = [12, 15, 21, 18, 15, 17, 19, 15, 20, 16]

# Calculate mean
mean = statistics.mean(numbers)

# Calculate median
median = statistics.median(numbers)

# Calculate mode
mode = statistics.mode(numbers)

print("Mean:", mean)
print("Median:", median)
print("Mode:", mode)
```

```
from statistics import mean, median, mode
```

Write a function to calculate average, median, and mode of a list of numbers.

Ask or edit in context GPT-4.1

Accept Close

```
def calculate_stats(numbers):
    """
    Calculate average, median, and mode of a list of numbers.

    Args:
        numbers (list of float/int): The list of numbers.

    Returns:
        dict: A dictionary with 'average', 'median', and 'mode'.
    """
    stats = {
        'average': mean(numbers),
        'median': median(numbers),
        'mode': mode(numbers)
    }
    return stats

# Example usage:
# nums = [1, 2, 2, 3, 4]
# result = calculate_stats(nums)
# print(result)
```

Explanation:

/explain × explain the code

This function, calculate_stats, computes three common statistics—average (mean), median, and mode—for a given list of numbers.

Step-by-step explanation:

1. **Function Definition**
   def calculate_stats(numbers):
   The function takes one argument, numbers, which should be a list of integers or floats.
2. **Docstring**

Ask or edit in context GPT-4.1

Close View in Chat

Task Description#3

- Provide multiple examples of input-output to the AI for convert_to_binary(num) function. Observe how AI uses few-shot prompting to generalize.

Expected Output#3

- Enhanced AI output with clearer prompts

Code:

```
def int_to_binary(n):
```

give python code to convert an integer to binary code

Ask or edit in context

Accept Close

```
    return bin(n)[2:]

# Example usage
number = 42
binary_code = int_to_binary(number)
print(f"Binary of {number} is {binary_code}")
```

z-1 / ai / assign_3_2.py / convert_to_binary

```
1
```

convert_to_binary(num) function

Ask or edit in context

Accept Close

```
2  def convert_to_binary(num):
3      """
4      Converts an integer to its binary representation as a string.
5      Args:
6          num (int): The integer to convert.
7      Returns:
8          str: Binary representation of num.
9      """
10     if not isinstance(num, int):
11         raise ValueError("Input must be an integer.")
12     return bin(num)[2:]
```

Explanation:

/explain give full explanation of code

Certainly! Based on your selection, it looks like you have a variable or function named binary_code in your Python file. However, you haven't provided the full code for how the integer is converted to binary. I'll explain the typical approach for converting an integer to binary in Python, and what binary_code might represent.

Typical Python Code to Convert Integer to Binary

```python
number = 10
binary_code = bin(number)[2:]
print(binary code)  # Output: '1010'
```

Ask or edit in context GPT-4.1

Accept Close

Task Description#4

- Create an user interface for an hotel to generate bill based on customer requirements

Expected Output#4

- Consistent functions with shared logic

```python
import tkinter as tk
```

convert_to_binary(num) function

• Create an user interface for an hotel to generate bill based on customer requirements

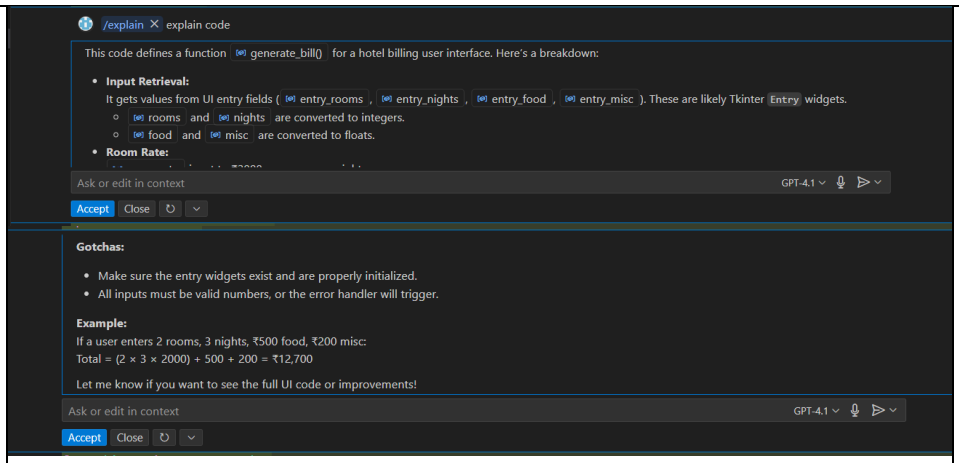
Ask or edit in context

Accept Close

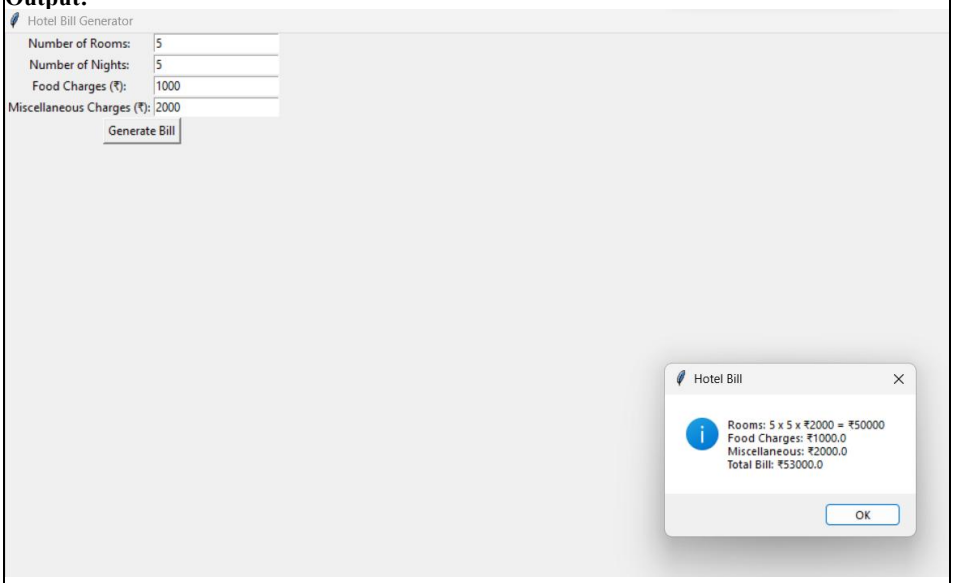
```python
from tkinter import messagebox


def generate_bill():
    try:
        rooms = int(entry_rooms.get())
        nights = int(entry_nights.get())
        food = float(entry_food.get())
        misc = float(entry_misc.get())
        room_rate = 2000  # Example rate per room per night

        total = (rooms * nights * room_rate) + food + misc
        bill_text = f"Rooms: {rooms} x {nights} x ₹{room_rate} = ₹{rooms * nights * room_rate}\n"
        bill_text += f"Food Charges: ₹{food}\n"
        bill_text += f"Miscellaneous: ₹{misc}\n"
        bill_text += f"Total Bill: ₹{total}"
        messagebox.showinfo("Hotel Bill", bill_text)
    except ValueError:
        messagebox.showerror("Input Error", "Please enter valid numbers.")

root = tk.Tk()
root.title("Hotel Bill Generator")
```

```python
        messagebox.showerror("Input Error", "Please enter valid numbers.")

root = tk.Tk()
root.title("Hotel Bill Generator")

tk.Label(root, text="Number of Rooms:").grid(row=0, column=0)
entry_rooms = tk.Entry(root)
entry_rooms.grid(row=0, column=1)

tk.Label(root, text="Number of Nights:").grid(row=1, column=0)
entry_nights = tk.Entry(root)
entry_nights.grid(row=1, column=1)

tk.Label(root, text="Food Charges (₹):").grid(row=2, column=0)
entry_food = tk.Entry(root)
entry_food.grid(row=2, column=1)

tk.Label(root, text="Miscellaneous Charges (₹):").grid(row=3, column=0)
entry_misc = tk.Entry(root)
entry_misc.grid(row=3, column=1)

tk.Button(root, text="Generate Bill", command=generate_bill).grid(row=4, column=0, columnspan=2)

root.mainloop()
```

Explanation:



Output:



Task Description#5

- Analyzing Prompt Specificity: Improving Temperature Conversion Function with Clear Instructions

Expected Output#5

- Code quality difference analysis for various prompts

Code:

```python
def celsius_to_fahrenheit(celsius):
```

give code to convert temperature using functions

Ask or edit in context GPT-4.1

Accept Close

```python
    return (celsius * 9/5) + 32

def fahrenheit_to_celsius(fahrenheit):
    return (fahrenheit - 32) * 5/9

def main():
    temp = float(input("Enter temperature value: "))
    unit = input("Enter unit (C for Celsius, F for Fahrenheit): ").strip().upper()
    if unit == 'C':
        print(f"{temp}°C = {celsius_to_fahrenheit(temp):.2f}°F")
    elif unit == 'F':
        print(f"{temp}°F = {fahrenheit_to_celsius(temp):.2f}°C")
    else:
        print("Invalid unit entered.")

if __name__ == "__main__":
    main()
```

Explanation:

/explain × explain the whole code

This Python code is a simple temperature converter using functions. Here's a breakdown:

1. Function Definitions

- **celsius_to_fahrenheit(celsius):** Converts Celsius to Fahrenheit using the formula: `(C × 9/5) + 32`
- **fahrenheit_to_celsius(fahrenheit):** Converts Fahrenheit to Celsius using the formula: `(F - 32) × 5/9`

Ask or edit in context

Accept Close

Output:

```
PS C:\Users\allur\Desktop\B.Tech> python -u "c:\Users\a
Enter temperature value: 30
Enter unit (C for Celsius, F for Fahrenheit): C
30.0°C = 86.00°F
PS C:\Users\allur\Desktop\B.Tech>
```

Note: Report should be submitted a word document for all tasks in a single document with prompts, comments & code explanation, and output and if required, screenshots

Evaluation Criteria:

Criteria	Max Marks
Task#1	0.5
Task#2	0.5
Task #3	0.5
Task #4	0.5
Task #5	0.5
Total	**2.5 Marks**