

2303a52362

Batch: 38

Task-1: Progressive Prompting for Calculator Design

Prompt 1 (calculator)

```
def calculator(a, b):  
    result = a + b  
    print("Result:", result)  
  
calculator(10, 5)  
  
Result: 15
```

Prompt 2 (With Operation Comments)

```
def calculator(a, b, op):  
    if op == '+':  
        result = a + b  
    elif op == '-':  
        result = a - b  
    elif op == '*':  
        result = a * b  
    elif op == '/':  
        result = a / b  
  
    print("Result:", result)  
  
calculator(10, 5, '+')  
calculator(10, 5, '-')  
calculator(10, 5, '*')  
calculator(10, 5, '/')
```

```
Result: 15
Result: 5
Result: 50
Result: 2.0
```

Prompt 3 (With Usage Examples & Error Handling)

```
def calculator(a, b, op):
    if op == '+':
        print("Result:", a + b)
    elif op == '-':
        print("Result:", a - b)
    elif op == '*':
        print("Result:", a * b)
    elif op == '/':
        if b != 0:
            print("Result:", a / b)
        else:
            print("Error: Division by zero")
    else:
        print("Error: Invalid operator")

calculator(10, 5, '+')
calculator(10, 5, '-')
calculator(10, 5, '*')
calculator(10, 0, '/')
calculator(10, 5, '%')

Result: 15
Result: 5
Result: 50
Error: Division by zero
Error: Invalid operator
```

Task-2: Refining Prompts for Sorting Logic

Prompt 1 (Vague Prompt)

```
marks = [65, 85, 45, 90, 70]
marks.sort()
print("Sorted Marks:", marks)
```

```
Sorted Marks: [45, 65, 70, 85, 90]
```

Prompt 2 (Specify Sorting Order)

```
marks = [65, 85, 45, 90, 70]
marks.sort()
print("Ascending Order:", marks)
```

```
Ascending Order: [45, 65, 70, 85, 90]
```

Prompt 3 (Add Constraint – No Built-in Sort)

```
def sort_marks(marks):
    n = len(marks)
    for i in range(n):
        for j in range(i + 1, n):
            if marks[i] > marks[j]:
                marks[i], marks[j] = marks[j], marks[i]
    print("Ascending Order:", marks)
```

```
marks = [65, 85, 45, 90, 70]
sort_marks(marks)
```

```
Ascending Order: [45, 65, 70, 85, 90]
```

Prompt 4 (Refined Prompt with Descending Order)

```
def sort_marks_desc(marks):
    n = len(marks)
    for i in range(n):
        for j in range(i + 1, n):
            if marks[i] < marks[j]:
                marks[i], marks[j] = marks[j], marks[i]
    print("Descending Order:", marks)
```

```
marks = [65, 85, 45, 90, 70]
sort_marks_desc(marks)
```

```
Descending Order: [90, 85, 70, 65, 45]
```

Task-3: Few-Shot Prompting for Prime Number Validation

Prompt 1 (Basic / No Examples)

```
def is_prime(n):  
    for i in range(2, n):  
        if n % i == 0:  
            print(n, "is not prime")  
            return  
    print(n, "is prime")  
  
is_prime(7)
```

Prompt 2 (Refined Prompt with Condition)

```
def is_prime(n):  
    if n <= 1:  
        print(n, "is not prime")  
        return  
    for i in range(2, n):  
        if n % i == 0:  
            print(n, "is not prime")  
            return  
    print(n, "is prime")  
  
is_prime(1)  
is_prime(9)  
is_prime(11)
```

Prompt 3 (Few-Shot Prompting with Input-Output Examples)

```
def is_prime(n):  
    if n <= 1:  
        print(n, "is not prime")  
        return  
    for i in range(2, int(n ** 0.5) + 1):  
        if n % i == 0:  
            print(n, "is not prime")  
            return  
    print(n, "is prime")
```

```
is_prime(2)
is_prime(4)
is_prime(17)
is_prime(1)
```

Task-4: Prompt-Guided UI Design for Student Grading System

#Prompt 1 (Basic UI Prompt)

```
m1 = float(input("Enter marks 1: "))
m2 = float(input("Enter marks 2: "))
m3 = float(input("Enter marks 3: "))

Enter marks 1: 4
Enter marks 2: 4
Enter marks 3: 4
```

Prompt 2 – Calculate Total

```
total = m1 + m2 + m3
print("Total Marks:", total)

Total Marks: 12.0
```

Prompt 3 – Calculate Percentage

```
percentage = total / 3
print("Percentage:", percentage)

Percentage: 4.0
```

Prompt 4 – Assign and Display Grade

```
if percentage >= 90:
    grade = "A"
elif percentage >= 75:
    grade = "B"
elif percentage >= 60:
    grade = "C"
else:
```

```
grade = "Fail"
print("Grade:", grade)
Grade: Fail
```

Task-5: Analyzing Prompt Specificity in Unit Conversion Functions

Prompt 1 – Vague Prompt

```
km = float(input("Enter kilometers: "))
miles = km * 0.6
print("Miles:", miles)
Enter kilometers: 60
Miles: 36.0
```

Prompt 2 – More Specific Prompt

```
km = float(input("Enter kilometers: "))
miles = km * 0.621371
print("Miles:", miles)
Enter kilometers: 77
Miles: 47.845567
```

Prompt 3 – Add Reverse Conversion

```
km = float(input("Enter kilometers: "))
miles = float(input("Enter miles: "))

print("Km to Miles:", km * 0.621371)
print("Miles to Km:", miles / 0.621371)
```

Prompt 4 – Improve Accuracy and Clarity

```
km = float(input("Enter kilometers: "))
miles = float(input("Enter miles: "))
```

```
km_to_miles = km * 0.621371
miles_to_km = miles / 0.621371

print("Kilometers to Miles:", round(km_to_miles, 3))
print("Miles to Kilometers:", round(miles_to_km, 3))

Enter kilometers: 44
Enter miles: 55
Kilometers to Miles: 27.34
Miles to Kilometers: 88.514
```