

Lab Assignment 6.3 – AI Assisted Coding

Question 1: Student Class

Algorithm:

```
.lab6.3.py > ...
1 # Student Class Example
2
3 class Student:
4     def __init__(self, name, roll_number, branch):
5         self.name = name
6         self.roll_number = roll_number
7         self.branch = branch
8
9     def display_details(self):
10        print("Student Name:", self.name)
11        print("Roll Number:", self.roll_number)
12        print("Branch:", self.branch)
13
14
15 # Creating an object
16 student1 = Student("Mani", "23CS001", "CSE")
17 student1.display_details()
18
```

The screenshot shows a terminal window with the following content:

```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS Python + ▾ □
● PS C:\Users\porika manikanta\OneDrive\Desktop\ai assistant> & "C:/Program Files/Python311/python.exe" "manikanta/OneDrive/Desktop/ai assistant/lab6.3.py"
Student Name: Mani
Roll Number: 23CS001
Branch: CSE
○ PS C:\Users\porika manikanta\OneDrive\Desktop\ai assistant>
```

1. Define a Student class
2. Initialize attributes using constructor
3. Create display method
4. Create object and display details

Pseudo Code:

START

Define class Student

Initialize name, roll, branch

Display details

END

Question 2: Multiples Using Loops

Algorithm:

1. Take a number
2. Use loop to print first 10 multiples

Pseudo Code:

```
START
FOR i from 1 to 10
Print num * i
END
```

```
lab 4.3.py > ...
1 def is_leap_year(year):
2     if (year % 4 == 0 and year % 100 != 0) or (year % 400 == 0):
3         return True
4     else:
5         return False
6
7
8 if __name__ == "__main__":
9     year = int(input("Enter a year: "))
10
11     if is_leap_year(year):
12         print(f"{year} is a leap year.")
13     else:
14         print(f"{year} is not a leap year.")

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS +
```

PS C:\Users\porika manikanta\OneDrive\Desktop\ai assistant> & "C:/Program Files/Python311/python.exe" "c:/Users/porika manikanta/OneDrive/Desktop/ai assistant/lab 4.3.py"
Enter a year: 2026
2026 is not a leap year.
○ PS C:\Users\porika manikanta\OneDrive\Desktop\ai assistant>

Question 3: Age Classification

Algorithm:

1. Take age as input
2. Use if-elif-else to classify

Pseudo Code:

```
START
IF age < 13 print Child
ELSE IF age < 20 print Teenager
ELSE IF age < 60 print Adult
ELSE print Senior
END
```

```
.lab6.3.py > ...
39
40     def classify_age(age):
41         if age < 13:
42             return "Child"
43         elif age < 20:
44             return "Teenager"
45         elif age < 60:
46             return "Adult"
47         else:
48             return "Senior"
49
50
51 print(classify_age(10))
52 print(classify_age(17))
53 print(classify_age(25))
54 print(classify_age(65))
55
56
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS Python

```
PS C:\Users\porika manikanta\OneDrive\Desktop\ai assistant> & "C:/Program Files/Python311/python manikanta/OneDrive/Desktop/ai assistant/lab6.3.py"
Child
Teenager
Adult
Senior
PS C:\Users\porika manikanta\OneDrive\Desktop\ai assistant>
```

Question 4: Sum of First n Numbers

Algorithm:

1. Initialize sum
2. Use loop/formula to compute sum

Pseudo Code:

START

sum = n*(n+1)/2

Print sum

END

```
.lab6.3.py > ...
55
56 # Sum using for loop
57 def sum_to_n_for(n):
58     total = 0
59     for i in range(1, n + 1):
60         total += i
61     return total
62
63
64 print(sum_to_n_for(10))
65
66
67 # Sum using while loop
68 def sum_to_n_while(n):
69     total = 0
70     i = 1
71     while i <= n:
72         total += i
73         i += 1
74     return total
75
76
77 print(sum_to_n_while(10))

PROBLEMS    OUTPUT    DEBUG CONSOLE    TERMINAL    PORTS    Python
```

PS C:\Users\porika manikanta\OneDrive\Desktop\ai assistant> & "C:/Program Files/Python311/p
manikanta/OneDrive/Desktop/ai assistant/lab6.3.py"
55
55
55
PS C:\Users\porika manikanta\OneDrive\Desktop\ai assistant>

Question 5: Bank Account Class

Algorithm:

1. Define BankAccount class 2. Implement
deposit, withdraw, balance methods

Pseudo Code:

START

Create class BankAccount

Deposit, Withdraw, Check balance

END

```
lab6.3.py > ...
86
87 # Bank Account Class
88 class BankAccount:
89     def __init__(self, account_holder, balance=0):
90         self.account_holder = account_holder
91         self.balance = balance
92
93     def deposit(self, amount):
94         if amount > 0:
95             self.balance += amount
96             print(f"Deposited ₹{amount}")
97         else:
98             print("Invalid deposit amount")
99
100    def withdraw(self, amount):
101        if amount <= self.balance:
102            self.balance -= amount
103            print(f"Withdrawn ₹{amount}")
104        else:
105            print("Insufficient balance")
106
107    def check_balance(self):
108        print(f"Current Balance: ₹{self.balance}")

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS Python
```

PS C:\Users\porika manikanta\OneDrive\Desktop\ai assistant> & "C:/Program Files/Python311/py manikanta/OneDrive/Desktop/ai assistant/lab6.3.py"
Current Balance: ₹1000
Deposited ₹500
Withdrawn ₹300
Current Balance: ₹1200
PS C:\Users\porika manikanta\OneDrive\Desktop\ai assistant>