

# Assessment Task: Instalment Payment Plan Generator – Level 1

## Objective:

The goal of this assessment is to develop a console application that generates an instalment payment plan based on user input. This task will test your ability to handle basic input/output operations, arithmetic calculations, and formatting output in Java.

## Task Description:

You are required to create a Java console application that prompts the user to enter two values:

- **Total Due Amount** (the total amount that needs to be paid).
- **Number of Months:** The number of months over which the total amount will be paid.

The application will then generate a table that displays the instalment number, the instalment amount, and the remaining balance after each instalment is paid, until the total amount reaches zero.

## Inputs:

- **Total Due Amount (int):** The total amount due that needs to be paid off through instalments.
- **Number of Months:** The number of months over which the total amount will be paid.

## Expected Outputs:

The program should output a neatly formatted table with three columns:

1. Instalment Number
2. Instalment Amount
3. Remaining Balance

The last instalment or the remaining balance may be adjusted to ensure the total due is paid off completely.

## Example:

Input:

...

Enter the Total Due Amount: 100000

Enter the Number of Months: 20

### Output:

...

Instalment Number	Instalment Amount	Remaining Balance
1	5000	95000
2	5000	90000
3	5000	85000
...		
19	5000	5000
20	5000	0

### Instructions:

1. Take input from the user.
2. Ensure that your program handles scenarios where the total amount is not perfectly divisible by the instalment amount. The final instalment should adjust to clear the remaining balance.
3. Format the output in a readable table format.

### Evaluation Criteria:

- **Correctness:** The program must correctly compute and display the schedule based on the inputs.
- **Code Quality:** Code readability, use of appropriate variable names, and comments explaining critical sections of the code.
- **Error Handling:** The program should handle possible logical errors such as zero or negative values for the due amount or instalment.

### Extra Marks:

- **User Experience:** Enhancements in user interaction, like repeating input prompt on invalid entries.
- **Efficiency:** Code optimizations for better performance.
- **Documentation:** Well-documented code with inline comments and a clear description of functions.

## Submission:

Submit a single .java file containing the source code with your name and date commented at the top. Ensure your program runs without errors before submission.

This task is designed to gauge your basic programming skills and understanding of language, which are crucial for the internship role at our organization. Good luck!