
PROJECT PERSONAL INFO

PREPARED BY
Sana Fatima

sana.faati@gmail.com

PREPARED TO
A&D Tech

ad.techinnov25@gmail.com

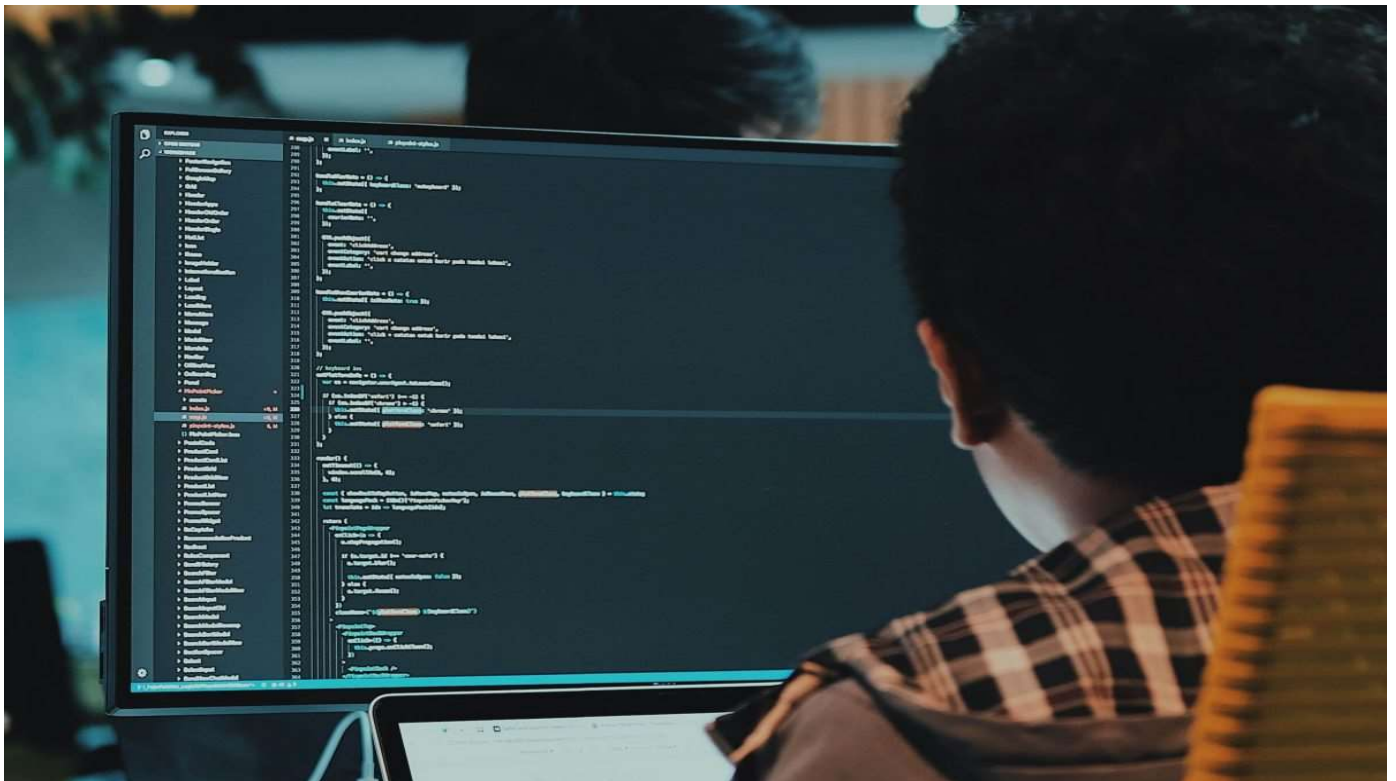


TABLE OF CONTENTS

Problem Statement



Solution Overview



Challenges and Resolution



Instructions



Problem Statement

Implementing the Program

Prompt the user to enter their name (string), age (integer), and salary (float). Store each input in a variable with the appropriate data type. Output the information back to the user in a sentence like "Hello, [name], you are [age] years old and your salary is \$[salary]."

SOLUTION OVERVIEW

The solution overview for this program is:

1. Create a program that gathers personal information from the user (name, age, and salary).
2. Use appropriate data types for each input (string, integer, and float).
3. Validate user input to ensure age is a positive integer and salary is not negative.
4. Display the entered information in a formatted sentence.
5. Allow the user to repeat the data entry process multiple times or choose to exit the program.

The program will:

- Prompt the user for input
- Store the input in variables
- Validate the input
- Display the output
- Repeat the process until the user chooses to exit

CHALLENGES AND RESOLUTIONS

Challenges:

1. Validating user input to ensure it matches the expected data type (e.g., age must be an integer, salary must be a non-negative number).
2. Repeating the data entry process until the user chooses to exit.
3. Displaying the output in a formatted sentence.

Resolutions:

1. Use a while loop to repeat the data entry process until the user chooses to exit.
2. Use condition statement to handle the valid inputs for age and salary.

INSTRUCTIONS

1. Create a new project in your IDE and name it "Personal Info".
2. Define variables to store the user's name (string), age (integer), and salary (float).
3. Use a while loop to repeatedly prompt the user for input until they choose to exit.
4. Inside the loop:
 - a. Prompt the user to enter their name and store it in the name variable.
 - b. Prompt the user to enter their age and store it in the age variable. Validate that the age is a positive integer.
 - c. Prompt the user to enter their salary and store it in the salary variable. Validate that the salary is a non-negative number.
 - d. Display the entered information in a formatted sentence.
5. Ask the user if they want to enter another set of data. If they respond with "y", continue the loop. If they respond with "n", exit the loop.
6. Use appropriate input validation and error handling techniques to ensure that the program handles invalid input gracefully.
7. Test the program with different inputs to ensure it works as expected.