

Best Practices for Assignment <3>: Palindrome

Summary of Learning:

- **Logic and Iteration:** This program taught learners the importance of iterative processes to solve problems, especially in scenarios involving number manipulation.
- **Variable Handling:** Learners grasped the concept of using variables to store and manipulate data. In this program, `input`, `output` were used to manage the number being checked and its reversed form.
- **Conditional Statements:** The use of an `if-else` statement was demonstrated to make decisions based on the comparison between the original and reversed numbers, determining whether the input is a palindrome.
- **Problem-Solving Skills:** Learners gained problem-solving skills by analyzing and understanding the logic behind palindrome detection, including the reversal of numbers.

The Palindrome program served as an introductory exercise in programming logic, variable usage, and conditional statements, which are fundamental building blocks for more complex programming challenges in Java.

Alternative Ways to Complete the Assignment:

- **Using while loop** to reverse the digits of the number. The loop continues until the `input` becomes zero, and the reversed number is calculated accordingly.
- **Using a calling method** that takes an integer as input. The main method calls this function to execute the function.

Best Practices:

- **Input Validation:** Check if the input number is non-negative before proceeding. Negative numbers and non-integer inputs like decimal numbers cannot be palindromes because they are considered as invalid inputs, so returning false immediately for negative and non-integer numbers is a good practice.
- **Use Meaningful Variable Names:** Use meaningful variable names like `input` and `output` to make your code more readable and maintainable.
- **Handle Special Cases:** Depending on your specific requirements, you might want to handle special case like single-digit number. Single-digit numbers are considered palindromes.