Table of Contents

Write a program that takes an integer N as input and determines whether it falls within a specific range, such as 1 to 100.

Program In C

C

```c
#include <stdio.h>

int main() {
    int N;
    int lowerLimit = 1;
    int upperLimit = 100;

    // Getting input from the user
    printf("Enter an integer: ");
    scanf("%d", &N);

    // Checking if N falls within the specified range
    if (N >= lowerLimit && N <= upperLimit) {
        printf("%d falls within the range %d to %d.\n", N, lowerLimit,
upperLimit);
    } else {
        printf("%d does not fall within the range %d to %d.\n", N,
```

```
lowerLimit, upperLimit);
    }

    return 0;
}
```

Explanation:

1. The program starts by including the necessary header file stdio.h, which provides input/output functions like printf and scanf.
2. In the main function, we declare the variable N to store the input integer and the variables lowerLimit and upperLimit to represent the range.
3. The lowerLimit and upperLimit variables are assigned the values of the specific range, which in this case is 1 to 100.
4. The printf function is used to prompt the user to enter an integer.
5. The scanf function is used to read the integer entered by the user and store it in the variable N.
6. We then use an if statement to check if the integer N falls within the specified range. This is done by checking if N is greater than or equal to lowerLimit and less than or equal to upperLimit.
7. If N falls within the range, we print that it falls within the specified range using the printf function.
8. If N does not fall within the range, we print that it does not fall within the specified range using the printf function.
9. Finally, we return 0 to indicate successful execution of the program.

```
Enter an integer: 4
4 falls within the range 1 to 100.
```

Program In Java

```java
Java
import java.util.Scanner;

public class NumberRange {
    public static void main(String[] args) {
        int N;
        int lowerLimit = 1;
        int upperLimit = 100;

        // Getting input from the user
        Scanner scanner = new Scanner(System.in);
        System.out.print("Enter an integer: ");
        N = scanner.nextInt();

        // Checking if N falls within the specified range
```

```
        if (N >= lowerLimit && N <= upperLimit) {
            System.out.printf("%d falls within the range %d to %d.\n",
N, lowerLimit, upperLimit);
        } else {
            System.out.printf("%d does not fall within the range %d to
%d.\n", N, lowerLimit, upperLimit);
        }
    }
}
```

Explanation:

1. The program starts by importing the java.util.Scanner class, which allows us to read input from the user.

2. The NumberRange class is defined, which contains the main method where the program execution begins.

3. Inside the main method, three integer variables are declared: N to store the user input, lowerLimit to represent the lower limit of the range, and upperLimit to represent the upper limit of the range.

4. A Scanner object named scanner is created to read input from the user.

5. The program prompts the user to enter an integer using System.out.print("Enter an integer: ").

6. The nextInt() method of the Scanner class is used to read an integer value entered by the user, and the value is assigned to the variable N.

7. The program checks if the value of N falls within the specified range using an if statement. The condition N >= lowerLimit && N <= upperLimit checks if N is greater than or equal to lowerLimit and less than or equal to upperLimit.

8. If the condition is true, meaning N falls within the range, the program prints the message "%d falls within the range %d to %d.\n" using System.out.printf(). The

placeholders %d are replaced with the corresponding values of N, lowerLimit, and upperLimit.

9. If the condition is false, meaning N does not fall within the range, the program prints the message "%d does not fall within the range %d to %d.\n" using System.out.printf(). Again, the placeholders %d are replaced with the corresponding values of N, lowerLimit, and upperLimit.

10. The program execution ends, and the program terminates.

`Java Output`

```
Enter an integer: 104
104 does not fall within the range 1 to 100.
```

Related posts:

1. Programming examples
2. Program to prints even numbers from 1 to 20
3. Program to calculate sum of all numbers from 1 to 100.
4. Program to get factorial of a number
5. Program to get Fibonacci sequence
6. Program to checks if number is prime
7. Program to get multiplication table
8. Program to find largest element in an array
9. Program to prints reverse of a string
10. Program to calculates sum of all elements in a list
11. Program determines integer is positive, negative, or zero
12. Program to find largest among three numbers using conditional statements.
13. Program determines it is a leap year or not

14. Program to determines even or odd
15. Program to calculate student exam grade
16. Program determines character is a vowel or consonant
17. Program to determines product is positive or negative
18. Program to determine divisible by both 5 and 7
19. Program to determines equilateral, isosceles, or scalene triangle
20. Function to calculate the factorial
21. Write a function to detect palindromes in strings
22. Write a function to find the greatest common divisor of two numbers
23. Program to calculate the area of different geometric shapes
24. try-catch block in C++