Table of Contents

## 0.1 ClassDef Main

# 1 Class: Main

## 1.1 Overview

The Main class serves as a container for several static utility methods and acts as the main entry point for the Java application.

## 1.2 Description

The Main class is designed as a utility class with static methods, meaning it does not need to be instantiated to be used. It contains methods for basic arithmetic, a recursive calculation, and console output. The class also includes the main method, which is the standard entry point for execution in a Java program.

The class defines the following static methods:

* **add(int a, int b)**: A simple function that accepts two integers, a and b, as input and returns their sum.
* **factorial(int n)**: This method calculates the factorial of a given integer n using recursion. The base case for the recursion is when n is less than or equal to 1, at which point the function returns 1. For any n greater than 1, it returns n multiplied by the factorial of n - 1.
* **greet(String name)**: A void method that takes a single String argument, name, and prints a formatted greeting message “Hello, [name]!” to the standard console output.

The main method demonstrates the usage of these utility functions by calling each one with sample values and printing the results to the console.

// Example of calling the methods  
int sum = Main.add(5, 10);  
int fact = Main.factorial(5);  
Main.greet("Prateek");

## 1.3 Usage Notes

* All methods in this class are static, so they should be called directly on the class itself (e.g., Main.add(5, 3)) rather than on an instance of the class.
* The factorial method is implemented recursively. Using a very large number as input may result in a StackOverflowError.
* The greet method does not return any value; its sole purpose is to print a message to the console.

## 1.4 Example

The following code demonstrates how the methods within the Main class are executed from the main method.

public static void main(String[] args) {  
 System.out.println("Sum: " + add(5, 10));  
 System.out.println("Factorial: " + factorial(5));  
 greet("Prateek");  
}

**Output:**

Sum: 15  
Factorial: 120  
Hello, Prateek!