**Assigment 2 :**

This code represents a client-server application for a basic calculator using CORBA (Common Object Request Broker Architecture) in Java. CORBA allows objects to communicate with each other across a network, regardless of the programming language or hardware platforms they are implemented on. The application consists of a client-side file (`CalcClient.java`) and a server-side file (`CalcServer.java`).

The `CalcClient.java` file:

- This file contains the client-side logic for interacting with the calculator server.

- It imports various packages and classes required for CORBA, input/output operations, and exceptions.

- The main logic is implemented inside the `main` method.

- The code creates and initializes the ORB (Object Request Broker) using the `ORB.init` method.

- It retrieves the root naming context and the object reference for the calculator server from the naming service.

- The code displays a menu to the user with options for different calculator operations: sum, subtraction, multiplication, division, and exit.

- Based on the user's choice, the corresponding calculator operation is invoked on the server-side object (`calcImpl`) obtained from the object reference.

- The result is displayed to the user.

- The program continues to display the menu until the user chooses to exit.

The `CalcServer.java` file:

- This file contains the server-side logic for the calculator.

- It imports various packages and classes required for CORBA, the Portable Object Adapter (POA), and properties.

- The `CalcImpl` class implements the `Calc` interface generated from the IDL (Interface Definition Language).

- The `CalcImpl` class provides implementations for the calculator operations: sum, subtraction, multiplication, and division.

- The division operation checks if the divisor is zero and throws a custom `DivisionByZero` exception if it is.

- The `CalcServer` class contains the main logic for the server.

- It creates and initializes the ORB.

- It retrieves the reference to the root POA (Portable Object Adapter) and activates the POA manager.

- An instance of `CalcImpl` is created and registered with the ORB.

- An object reference (`ref`) is obtained from the servant and narrowed to the `Calc` interface.

- The root naming context is retrieved, and the object reference is bound to a name in the naming context.

- The server is set to wait for invocations from clients using `orb.run()`.

In summary, this code establishes communication between a client and server using CORBA for a basic calculator application. The client interacts with the server by invoking calculator operations, and the server performs the requested operations and returns the results to the client.

CORBA (Common Object Request Broker Architecture) is a middleware technology and architecture developed by the Object Management Group (OMG). It is a standard that enables interoperability and communication between distributed objects implemented in different programming languages and running on different platforms.