VIT Vidyalankar Institute of Technology Accredited A+ by NAAC	DEPARTMENT OF COMPUTER ENGINEERING

Experiment No. 02

Semester	B.E. Semester VIII – Computer Engineering
Subject	Distributed Computing Lab
Subject Professor In-charge	Dr. Umesh Kulkarni
Assisting Professor	Prof. Prakash Parmar
Academic Year	2024-25

Student Name	Deep Salunkhe
Roll Number	21102A0014

Title: Concepts of Operating Systems in Distributed Computing

---

Explanation:

The objective of this lab was to implement a distributed application using socket programming in Java. The application consists of a **server** that calculates the factorial of an integer received from the **client** and sends the result back. This exercise helped demonstrate the fundamental principles of client-server communication, socket programming, and distributed system design.

Concepts Covered:

1. **Distributed Computing**: Distributed computing involves distributing tasks across multiple computers that communicate with each other over a network. In this lab, a client sends a request to the server for a factorial calculation, and the server computes the result and returns it. This interaction between the client and server is a fundamental example of distributed computing, where each component (client and server) operates on separate systems (in this case, they could even be on different machines, though here they are assumed to be local).
2. **Client-Server Architecture**: A **client-server architecture** is a common design pattern in distributed systems. In this model, the **server** provides a service or resource, while the **client** accesses this service. In the application implemented in this lab:
    - The **server** listens on a specific port for incoming client requests, processes them, and sends back a response.
    - The **client** sends an integer to the server, requests the factorial calculation, and receives the computed result.

3. **Socket Programming**: Socket programming allows communication between two computers over a network by using a combination of IP addresses and ports. In this lab, we used **Java Sockets** to facilitate the client-server communication:
    - The **ServerSocket** class in Java is used to create a server-side socket that listens for incoming client connections.
    - The **Socket** class is used on the client side to establish a connection with the server.

    Sockets provide a way for applications to exchange data over a network, making them a core tool for implementing distributed applications.

4. **Factorial Calculation**: The server in this lab receives an integer from the client, calculates its **factorial**, and sends the result back. A factorial of a number nnn is the product of all positive integers less than or equal to nnn. It is defined as:

    n!=n×(n−1)×(n−2)×⋯×1n! = n \times (n-1) \times (n-2) \times \dots \times 1n!=n×(n−1)×(n−2)×⋯×1

    For example:

    - 5!=5×4×3×2×1=1205! = 5 \times 4 \times 3 \times 2 \times 1 = 1205!=5×4×3×2×1=120
    - 3!=3×2×1=63! = 3 \times 2 \times 1 = 63!=3×2×1=6

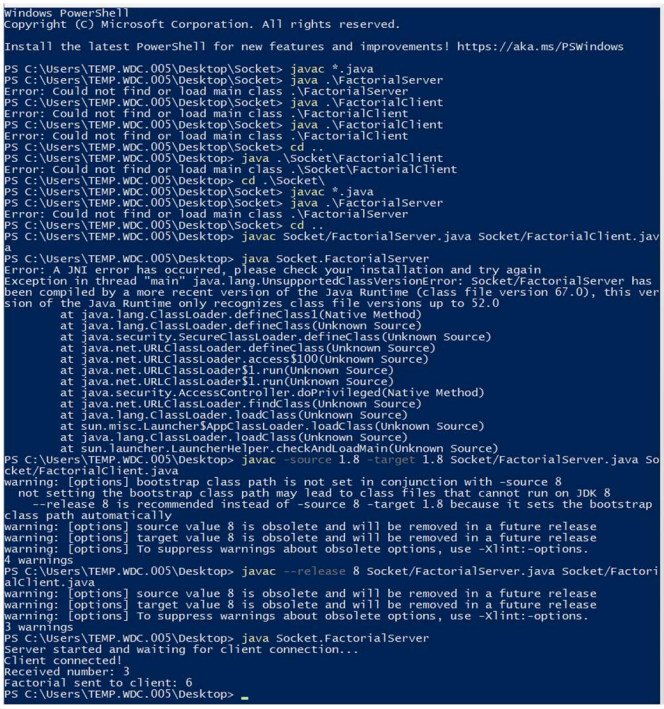
    In the code, a method (`calculateFactorial`) implements this logic using a simple loop to multiply numbers from 1 to the given integer.

CODE:

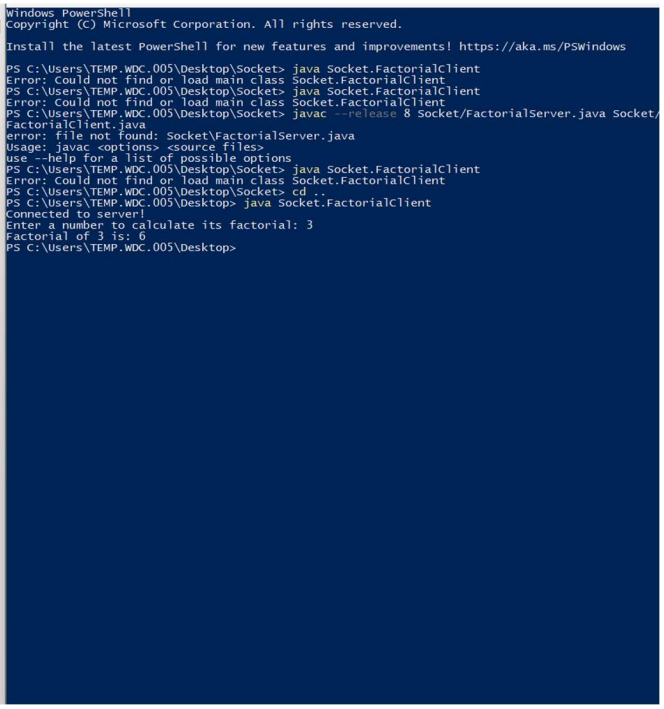
```
package Socket;
import java.io.*;
import java.net.*;

public class FactorialClient {
    public static void main(String[] args) {
        try {
            // Connect to the server running on localhost and port 1234
            Socket socket = new Socket("localhost", 1234);
            System.out.println("Connected to server!");

            // Create input and output streams for communication
            BufferedReader input = new BufferedReader(new InputStreamReader(System.in));
            PrintWriter output = new PrintWriter(socket.getOutputStream(), true);
            BufferedReader serverInput = new BufferedReader(new InputStreamReader(socket.getInputStream()));

            // Ask user for an integer input
            System.out.print("Enter a number to calculate its factorial: ");
            int number = Integer.parseInt(input.readLine());

            // Send the number to the server
            output.println(number);

            // Receive the factorial result from the server
            String result = serverInput.readLine();
            System.out.println("Factorial of " + number + " is: " + result);

            // Close the streams and socket
            input.close();
            output.close();
            serverInput.close();
            socket.close();

        } catch (IOException e) {
            e.printStackTrace();
        }
    }
}
```

```java
package Socket;
import java.io.*;
import java.net.*;


public class FactorialServer {
    public static void main(String[] args) {
        try {
            // Create a server socket that listens on port 1234
            ServerSocket serverSocket = new ServerSocket(1234);
            System.out.println("Server started and waiting for client connection...");

            // Accept a client connection
            Socket clientSocket = serverSocket.accept();
            System.out.println("Client connected!");

            // Create input and output streams for communication
            BufferedReader input = new BufferedReader(new InputStreamReader(clientSocket.getInputStream()));
            PrintWriter output = new PrintWriter(clientSocket.getOutputStream(), true);

            // Read the integer sent by the client
            String inputValue = input.readLine();
            int number = Integer.parseInt(inputValue);
            System.out.println("Received number: " + number);

            // Calculate factorial
            long factorial = calculateFactorial(number);

            // Send the result back to the client
            output.println(factorial);
            System.out.println("Factorial sent to client: " + factorial);

            // Close the connection
            input.close();
            output.close();
            clientSocket.close();
            serverSocket.close();

        } catch (IOException e) {
            e.printStackTrace();
        }
    }

    // Method to calculate factorial
    private static long calculateFactorial(int number) {
        long result = 1;
        for (int i = 1; i <= number; i++) {
            result *= i;
        }
        return result;
    }
}
```

Output:



```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows

PS C:\Users\TEMP.WDC.005\Desktop\Socket> javac *.java
PS C:\Users\TEMP.WDC.005\Desktop\Socket> java .\FactorialServer
Error: Could not find or load main class .\FactorialServer
PS C:\Users\TEMP.WDC.005\Desktop\Socket> java .\FactorialClient
Error: Could not find or load main class .\FactorialClient
PS C:\Users\TEMP.WDC.005\Desktop\Socket> java .\FactorialClient
Error: Could not find or load main class .\FactorialClient
PS C:\Users\TEMP.WDC.005\Desktop\Socket> cd ..
PS C:\Users\TEMP.WDC.005\Desktop> java .\Socket\FactorialClient
Error: Could not find or load main class .\Socket\FactorialClient
PS C:\Users\TEMP.WDC.005\Desktop> cd .\Socket\
PS C:\Users\TEMP.WDC.005\Desktop\Socket> javac *.java
PS C:\Users\TEMP.WDC.005\Desktop\Socket> java .\FactorialServer
Error: Could not find or load main class .\FactorialServer
PS C:\Users\TEMP.WDC.005\Desktop\Socket> cd ..
PS C:\Users\TEMP.WDC.005\Desktop> javac Socket/FactorialServer.java Socket/FactorialClient.jav
a
PS C:\Users\TEMP.WDC.005\Desktop> java Socket.FactorialServer
Error: A JNI error has occurred, please check your installation and try again
Exception in thread "main" java.lang.UnsupportedClassVersionError: Socket/FactorialServer has
been compiled by a more recent version of the Java Runtime (class file version 67.0), this ver
sion of the Java Runtime only recognizes class file versions up to 52.0
        at java.lang.ClassLoader.defineClass1(Native Method)
        at java.lang.ClassLoader.defineClass(Unknown Source)
        at java.security.SecureClassLoader.defineClass(Unknown Source)
        at java.net.URLClassLoader.defineClass(Unknown Source)
        at java.net.URLClassLoader.access$100(Unknown Source)
        at java.net.URLClassLoader$1.run(Unknown Source)
        at java.net.URLClassLoader$1.run(Unknown Source)
        at java.security.AccessController.doPrivileged(Native Method)
        at java.net.URLClassLoader.findClass(Unknown Source)
        at java.lang.ClassLoader.loadClass(Unknown Source)
        at sun.misc.Launcher$AppClassLoader.loadClass(Unknown Source)
        at java.lang.ClassLoader.loadClass(Unknown Source)
        at sun.launcher.LauncherHelper.checkAndLoadMain(Unknown Source)
PS C:\Users\TEMP.WDC.005\Desktop> javac -source 1.8 -target 1.8 Socket/FactorialServer.java So
cket/FactorialClient.java
warning: [options] bootstrap class path is not set in conjunction with -source 8
  not setting the bootstrap class path may lead to class files that cannot run on JDK 8
    --release 8 is recommended instead of -source 8 -target 1.8 because it sets the bootstrap
class path automatically
warning: [options] source value 8 is obsolete and will be removed in a future release
warning: [options] target value 8 is obsolete and will be removed in a future release
warning: [options] To suppress warnings about obsolete options, use -Xlint:-options.
4 warnings
PS C:\Users\TEMP.WDC.005\Desktop> javac --release 8 Socket/FactorialServer.java Socket/Factori
alClient.java
warning: [options] source value 8 is obsolete and will be removed in a future release
warning: [options] target value 8 is obsolete and will be removed in a future release
warning: [options] To suppress warnings about obsolete options, use -Xlint:-options.
3 warnings
PS C:\Users\TEMP.WDC.005\Desktop> java Socket.FactorialServer
Server started and waiting for client connection...
Client connected!
Received number: 3
Factorial sent to client: 6
PS C:\Users\TEMP.WDC.005\Desktop>
```



```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows

PS C:\Users\TEMP.WDC.005\Desktop\Socket> java Socket.FactorialClient
Error: Could not find or load main class Socket.FactorialClient
PS C:\Users\TEMP.WDC.005\Desktop\Socket> java Socket.FactorialClient
Error: Could not find or load main class Socket.FactorialClient
PS C:\Users\TEMP.WDC.005\Desktop\Socket> javac --release 8 Socket/FactorialServer.java Socket/
FactorialClient.java
error: file not found: Socket\FactorialServer.java
Usage: javac <options> <source files>
use --help for a list of possible options
PS C:\Users\TEMP.WDC.005\Desktop\Socket> java Socket.FactorialClient
Error: Could not find or load main class Socket.FactorialClient
PS C:\Users\TEMP.WDC.005\Desktop\Socket> cd ..
PS C:\Users\TEMP.WDC.005\Desktop> java Socket.FactorialClient
Connected to server!
Enter a number to calculate its factorial: 3
Factorial of 3 is: 6
PS C:\Users\TEMP.WDC.005\Desktop>
```