**RARPServer.java:**

import java.io.\*;

import java.net.\*;

import java.util.\*;

public class RARPServer {

public static void main(String[] args) {

try {

Map<String, String> rarpTable = new HashMap<>();

rarpTable.put("AA:BB:CC:DD:EE:01", "192.168.1.2");

rarpTable.put("AA:BB:CC:DD:EE:02", "192.168.1.3");

rarpTable.put("AA:BB:CC:DD:EE:03", "192.168.1.4");

ServerSocket serverSocket = new ServerSocket(6789);

System.out.println("RARP Server is running...");

while (true) {

Socket clientSocket = serverSocket.accept();

System.out.println("Client connected: " + clientSocket);

BufferedReader in = new BufferedReader(new InputStreamReader(clientSocket.getInputStream()));

PrintWriter out = new PrintWriter(clientSocket.getOutputStream(), true);

String macRequest = in.readLine();

System.out.println("Request for MAC: " + macRequest);

String ip = rarpTable.getOrDefault(macRequest, "IP Address Not Found");

out.println(ip);

clientSocket.close();

}

} catch (IOException e) {

e.printStackTrace();

}

}

}

**RARPClient.java**

import java.io.\*;

import java.net.\*;

public class RARPClient {

public static void main(String[] args) {

try {

Socket socket = new Socket("localhost", 6789);

BufferedReader userIn = new BufferedReader(new InputStreamReader(System.in));

BufferedReader in = new BufferedReader(new InputStreamReader(socket.getInputStream()));

PrintWriter out = new PrintWriter(socket.getOutputStream(), true);

System.out.print("Enter MAC Address to resolve: ");

String mac = userIn.readLine();

out.println(mac);

String response = in.readLine();

System.out.println("IP Address: " + response);

socket.close();

} catch (IOException e) {

e.printStackTrace();

}

}

}

**A computer screen with white text

AI-generated content may be incorrect.OUTPUT:**

**A screen shot of a computer

AI-generated content may be incorrect.**