DNS Server side

```
import java.net.*;
import java.util.HashMap;

public class DNSServer {
   private static HashMap<String, String> dnsTable = new HashMap<>();

   public static void main(String[] args) throws Exception {
      dunsTable.put("example.com", "192.168.1.1");

      DatagramSocket socket = new DatagramSocket(53); // DNS port
      byte[] receiveData = new byte[1024];

      while (true) {
         DatagramPacket receivePacket = new DatagramPacket(receiveData,
receiveData.length);
         socket.receive(receivePacket);

         String query = new String(receivePacket.getData(), 0, receivePacket.getLength());
         System.out.println("Received query: " + query);

         InetAddress clientAddress = receivePacket.getAddress();
         int clientPort = receivePacket.getPort();

         String responseIP = dnsTable.getOrDefault(query, "Not Found");
         byte[] responseData = responseIP.getBytes();

         DatagramPacket sendPacket = new DatagramPacket(responseData,
responseData.length, clientAddress, clientPort);
         socket.send(sendPacket);
      }
   }
}
```

DNS Client side

```
import java.net.*;

public class DNSClient {
   public static void main(String[] args) throws Exception {
      DatagramSocket socket = new DatagramSocket();

      byte[] sendData = "example.com".getBytes();
      InetAddress serverAddress = InetAddress.getByName("localhost");
```

```
        DatagramPacket sendPacket = new DatagramPacket(sendData, sendData.length,
serverAddress, 53); // DNS port
        socket.send(sendPacket);

        byte[] receiveData = new byte[1024];
        DatagramPacket receivePacket = new DatagramPacket(receiveData,
receiveData.length);
        socket.receive(receivePacket);

        String responseIP = new String(receivePacket.getData(), 0, receivePacket.getLength());
        System.out.println("Resolved IP: " + responseIP);

        socket.close();
    }
}
```