**Cách sử dụng Socket trong C#**

Socket là một giao diện lập trình mạng giúp ứng dụng có thể giao tiếp với nhau thông qua giao thức TCP hoặc UDP.

**1. Các bước sử dụng Socket trong C#**

**🔹 Server (Lắng nghe kết nối)**

1. **Tạo socket** với địa chỉ IP và cổng.
2. **Ràng buộc (bind) socket** với một địa chỉ IP cụ thể.
3. **Bắt đầu lắng nghe kết nối** từ client.
4. **Chấp nhận kết nối** khi có client yêu cầu.
5. **Gửi và nhận dữ liệu** qua kết nối.
6. **Đóng kết nối khi xong**.

**🔹 Client (Gửi yêu cầu)**

1. **Tạo socket**.
2. **Kết nối đến server** bằng địa chỉ IP và cổng.
3. **Gửi dữ liệu đến server**.
4. **Nhận dữ liệu phản hồi từ server**.
5. **Đóng kết nối**.

**2. Ví dụ Cụ Thể**

**📌 Server sử dụng TCP Socket**

using System;

using System.Net;

using System.Net.Sockets;

using System.Text;

using System.Threading;

class Server

{

static void Main()

{

// 1⃣ Tạo socket

Socket serverSocket = new Socket(AddressFamily.InterNetwork, SocketType.Stream, ProtocolType.Tcp);

// 2⃣ Ràng buộc socket vào địa chỉ và cổng

serverSocket.Bind(new IPEndPoint(IPAddress.Any, 12345));

// 3⃣ Lắng nghe kết nối từ client

serverSocket.Listen(5);

Console.WriteLine("🔹 Server đang chạy...");

while (true)

{

// 4⃣ Chấp nhận client mới

Socket clientSocket = serverSocket.Accept();

Console.WriteLine("✅ Client kết nối!");

// Xử lý client trên thread riêng

Thread clientThread = new Thread(() => HandleClient(clientSocket));

clientThread.Start();

}

}

static void HandleClient(Socket clientSocket)

{

byte[] buffer = new byte[1024];

int receivedBytes;

try

{

while ((receivedBytes = clientSocket.Receive(buffer)) > 0)

{

string message = Encoding.UTF8.GetString(buffer, 0, receivedBytes);

Console.WriteLine($"📩 Nhận: {message}");

// Gửi phản hồi

string response = $"Server nhận: {message}";

clientSocket.Send(Encoding.UTF8.GetBytes(response));

}

}

catch (Exception ex)

{

Console.WriteLine("⚠️ Lỗi: " + ex.Message);

}

finally

{

clientSocket.Close();

}

}

}

**📌 Client gửi dữ liệu đến Server**

using System;

using System.Net;

using System.Net.Sockets;

using System.Text;

class Client

{

static void Main()

{

try

{

// 1⃣ Tạo socket client

Socket clientSocket = new Socket(AddressFamily.InterNetwork, SocketType.Stream, ProtocolType.Tcp);

// 2⃣ Kết nối đến server

clientSocket.Connect(new IPEndPoint(IPAddress.Parse("127.0.0.1"), 12345));

Console.WriteLine("🟢 Kết nối thành công!");

while (true)

{

Console.Write("Nhập tin nhắn: ");

string input = Console.ReadLine();

if (input.ToUpper() == "EXIT") break;

// 3⃣ Gửi dữ liệu lên server

clientSocket.Send(Encoding.UTF8.GetBytes(input));

// 4⃣ Nhận phản hồi từ server

byte[] buffer = new byte[1024];

int receivedBytes = clientSocket.Receive(buffer);

string response = Encoding.UTF8.GetString(buffer, 0, receivedBytes);

Console.WriteLine("📩 Server: " + response);

}

clientSocket.Close();

}

catch (Exception ex)

{

Console.WriteLine("⚠️ Lỗi: " + ex.Message);

}

}

}

**3. Cách chạy chương trình**

1⃣ **Chạy Server trước**  
Mở terminal/cmd, biên dịch và chạy server:

dotnet run

2⃣ **Chạy Client**  
Mở terminal/cmd khác, biên dịch và chạy client:

dotnet run

3⃣ **Gửi tin nhắn từ Client** và nhận phản hồi từ Server.

**4. Mở rộng**

✅ **Xử lý nhiều client cùng lúc** → Dùng **MultiThread**.  
✅ **Gửi file, hình ảnh** → Dùng **byte array** thay vì chuỗi.  
✅ **Sử dụng UDP thay vì TCP** → Thay SocketType.Stream thành SocketType.Dgram.

📌 **Bạn có muốn mở rộng theo hướng nào không?** 🚀