

gRPCclient

```
using Grpc.Core;
using Mygrpcproto;
using System;

namespace gRPCclient
{
    class Program {
        static void Main(string[] args)
        {
            Console.WriteLine("Starting gRPC Client");
            MyData.info();
            Channel channel = new Channel("127.0.0.1:10000",
            ChannelCredentials.Insecure);
            var client = new MyGrpcSrv.MyGrpcSrvClient(channel);
            String str;
            int num1, num2;
            Console.Write("Enter number 1: ");
            str = Console.ReadLine();
            if (int.TryParse(str, out num1))
            {
                Console.Write("Enter number 2: ");
                str = Console.ReadLine();
                if (int.TryParse(str, out num2))
                {
                    var reply = client.addInt(new AddIntRequest
                    {
                        Num1 = num1,
                        Num2 = num2
                    });
                    Console.WriteLine("From server: " + reply.Comment + reply.Result);
                }
                else
                    Console.WriteLine("Wrong value!");
            }
            else
                Console.WriteLine("Wrong value!");
            Console.WriteLine("Stopping gRPC Client");
            Console.ReadKey(); //After we press any key, the program continues
            channel.ShutdownAsync().Wait();
        }
    }
}
```

```
// Include namespace system
using System;
using System.Net;

public class MyData
{
    public static void info()
    {
        var dt = DateTime.Now.ToString("yyyy/MM/dd");
        Console.WriteLine("yyyy/MM/dd " + dt);
        Console.WriteLine("Data:Alvaro Lopez Pereda and Alberto Hernandez Lado");
        Console.WriteLine("UserName: {0}", Environment.UserName);
        Console.WriteLine("Operating System Name:" + Environment.MachineName);
        String strHostName = string.Empty;
        IPHostEntry ipEntry = Dns.GetHostEntry(Dns.GetHostName());
        IPAddress[] addr = ipEntry.AddressList;

        for (int i = 0; i < addr.Length; i++)
        {
            Console.WriteLine("IP Address {0}: {1} ", i, addr[i].ToString());
        }
    }
}
```

gRPCserver

```
using Grpc.Core;
using Mygrpcproto;
using System;
using System.Threading.Tasks;

namespace gRPCserver{
    class MyGrpcSrvImpl : MyGrpcSrv.MyGrpcSrvBase {
        public override Task<AddIntReply> addInt(AddIntRequest req, ServerCallContext ctx) {
            string comment;
            int result = req.Num1 + req.Num2;
            if (result >= 0){
                comment = "+";
            }
            else if (result < 0){
                comment = "-";
            }
            else{
                comment = " ";
            }

            return Task.FromResult(new AddIntReply { Result = result, Comment = comment });
        }
    }
    class Program {
        const int port = 10000;
        static void Main(string[] args)
        {
            Console.WriteLine("Starting Hello gRPC server");
            Server myServer = new Server()
            { //now we define the services and ports of our server.
                Services = { MyGrpcSrv.BindService(new MyGrpcSrvImpl()) }, //The service we start is to generate
                this class, with all the methods.
                Ports = { new ServerPort("localhost", port, ServerCredentials.Insecure) } //define the port we will
                work on.
            };
            myServer.Start(); //the server starts
            Console.WriteLine("Hello gRPC server listening on port " + port);
            Console.WriteLine("Press any key to stop the server...");
            Console.ReadKey(); //After we press any key, the program continues
            myServer.ShutdownAsync().Wait(); //The server ends asynchronously, waiting for last process.
        }
    }
}
```

gRPCproto

```
syntax = "proto3";
package mygrpcproto;

//Service definition
service MyGrpcSrv{
    rpc addInt (AddIntRequest) returns (AddIntReply) {}
}

//The request message
message AddIntRequest {
    int32 num1 = 1;
    int32 num2 = 2;
}

//The response message
message AddIntReply {
    int32 result = 1;
    string comment = 2;
}
```

Test

```
C:\Users\alber\Desktop\Practice02-C\MygRPC\gRPCserver\bin\Debug\netcoreapp3.1\gRPCserver.exe
Starting Hello gRPC server
Hello gRPC server listening on port 10000
Press any key to stop the server...
_
```

```
Seleccionar C:\Users\alber\Desktop\Practice02-C\MygRPC\gRPCclient\bin\Debug\netcoreapp3.1\gRPCclient.exe
Starting gRPC Client
yyyy/MM/dd 2022/05/03
Data:Alvaro Lopez Pereda and Alberto Hernandez Lado
UserName: alber
Operating System Name:LAPTOP-NS016AQT
IP Address 0: fe80::119d:deb5:e878:537d%6
IP Address 1: fe80::5453:2522:8b7d:f674%8
IP Address 2: 192.168.56.1
IP Address 3: 10.1.28.22
Enter number 1: 45
Enter number 2: 82
From server: +127
Stopping gRPC Client
```

It executes the server side first and make a two numbers sum.