

Find the missing properties from a class by comparing with other class

Published 4/15/2013 by [Raghav Khunger](#)

Lets say we have two classes Person1 and Person2. Below is the code for it:

```
public class Person1
{
    public int ID { get; set; }
    public string Name { get; set; }
    public string FirstName { get; set; }
    public string Age { get; set; }
}

public class Person2
{
    public string Name { get; set; }
    public string FirstName { get; set; }
}
```

Our goal is to find out the properties which are present in "Person1" class and missing in "Person2" class. It can be achieved via "[Reflection](#)". Below is a sample console program to do the same:

```
using System;
using System.Collections.Generic;
using System.Linq;

public class Program
{
    private static void Main(string[] args)
    {
        // We need to find missing properties in target class.
        Type source= typeof(Person1);
        Type target = typeof(Person2);

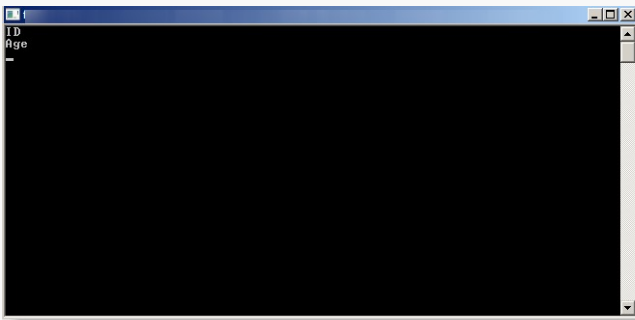
        List<string> properties = source.GetProperties()
                                     .Select(x => x.Name)
                                     .Except(target.GetProperties().Select(y => y.Name))
                                     .ToList();

        foreach (string property in properties)
        {
            Console.WriteLine(property);
        }

        Console.ReadLine();
    }
}
```

```
}
```

After running the above program the output will be:



tags : .net, c#