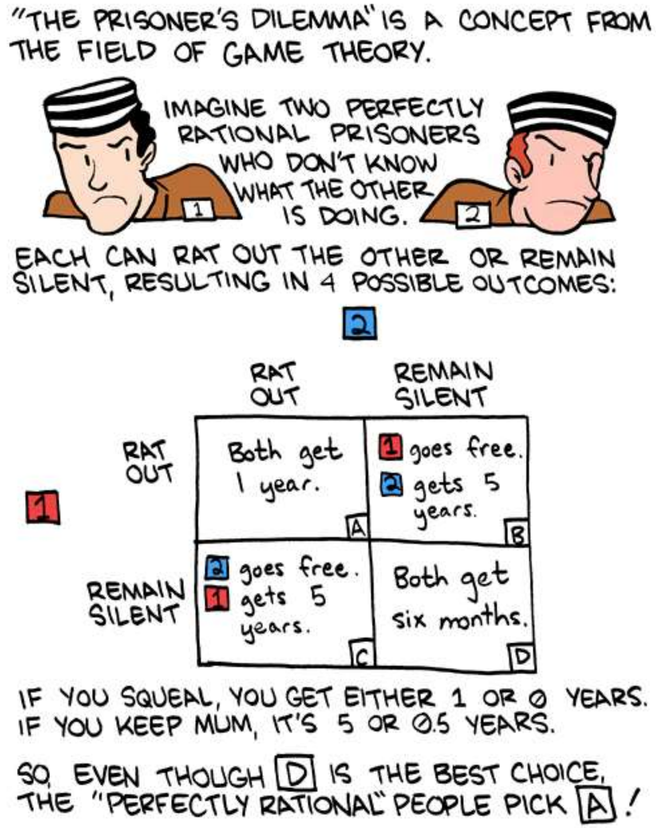
Nash Equilibrium



On the website of Princeton University, where Nash worked, the Nash equilibrium definition sounds like this:

Nash equilibrium ((game theory) a stable state of a system that involves several interacting participants in which no participant can gain by a change of strategy as long as all the other participants remain unchanged) [13].
Nash was the first to prove that such equilibria must exist for all finite games with any number of players. This was done in his 1950 dissertation on non-cooperative games. [14]

Today it is the most popular principle of rational behavior in the theory of non-cooperative games.

Suppose (S, H) is a noncooperative n-player game in normal form, where S is a set of pure strategies and H is a set of payoffs. When each player

$$i \in \{1, \ldots, n\}$$

chooses a strategy

$x_i \in S$

in the profile of strategies

$x = (x_1, \ldots, x_n)$

gets a prize of

$H_i(x)$

The payoff depends on the entire strategy profile: not only on the strategy

$\boldsymbol{x_i}$, but also on other people's strategies

$\boldsymbol{x_{-i}}$ that is, all strategies x_j for $j \neq i$.

The strategy profile

$x^* \in S$

is a Nash equilibrium if you change your strategy from $\boldsymbol{x_i^*}$ on $\boldsymbol{x_i}$ is not beneficial to any player $\boldsymbol{i}$, that is, for any $\boldsymbol{i}$

$H_i(x^*) \geqslant H_i(x_i, x^*_{-i})$.

The most popular example of explaining the Nash equilibrium is the prisoner's dilemma. players 1 and player 2

$i \in \{1, 2\}$

suspected of committing a serious crime, are isolated from each other in pre-trial detention. In the absence of direct evidence, the success or failure of the prosecution depends on recognition (strategy 1) or non-recognition (strategy 2)

of the bandits themselves. If both bandits confess (situation (1,1)), then they will

found guilty and sentenced to 8 years in prison.

For player1 - i=1 (see Figure 3).

$H_2(x)$ $H_1(x)$	*Player 2*	Recognition;	Non-recognition;
Player 1		1	2
Recognition;	1	**-8**	**0**
Non-recognition;	2	**-10**	**-1**

Figure 3. Matrix for i=1

For i=2 (see Figure 4).

$H_2(x)$ $H_1(x)$	*Player 2*	Recognition;	Non-recognition;

Player 1		1	2
Recognition;	1	**-8**	**-10**
Non-recognition;	2	**0**	**-1**

Figure 4. Matrix for i=2

If none of them is not recognized (situation (2.2)), then on charges of the main

they will be acquitted of the crime, but the prosecutor will still succeed prove their guilt in some concomitant less serious a crime, such as carrying a weapon, as a result of which they will be sentenced to 1 year in prison.

$H_2(x)$ / $H_1(x)$	***Player 2***	Recognition;	Non-recognition;
Player 1		1	2
Recognition;	1	**-8** / **-8**	**-10** / **0**
Non-recognition;	2	**0** / **-10**	**-1** / **-1**

$H_2(x)$ / $H_1(x)$	***Player 2***	Recognition;	Non-recognition;
Player 1		1	2
Recognition;	1	**-8**	**-10**

		-8	**0**
Non-recognition;	2	**0** **-10**	**-1** **-1**

Figure 5. Matrix for both players i=1 and i=2

If finally confessed only one of them (situations (2,1) and (1,2)), then the admitted will be released (for helping the investigation), and the unrecognized person will sentenced to a maximum term of 10 years.

For player 1 (i=1) – see Fig. 6.

$H_1(x_1^*, x_2^*) >= H_1(x_1, x_2^*)$

$H_2(x)$ / $H_1(x)$	*Player 2*	Recognition;	Non-recognition;
Player 1		1	2
Recognition;	1	**-8*** ↓	**0**
*N*on-recognition;	2	**-10**	**-1**

Figure 16. Possible moves for Player 1

$H_2(x)$ $H_1(x)$	*Player 2*	Recognition;	Non-recognition;
Player 1		1	2
Recognition;	1	**-8*** →	**-10**
*N*on-recognition;	2	**0**	**-1**

Figure 7. Possible moves for Player 2

For player 2 i=2 – see Fig. 7.

$H_1(\boldsymbol{x_2}^*, \boldsymbol{x_1}^*) >= H_1(\boldsymbol{x_2}, \boldsymbol{x_1}^*)$

i.e.

$H_i(x_i^*, x_{-i}^*) \geqslant H_i(x_i, x_{-i}^*)$.

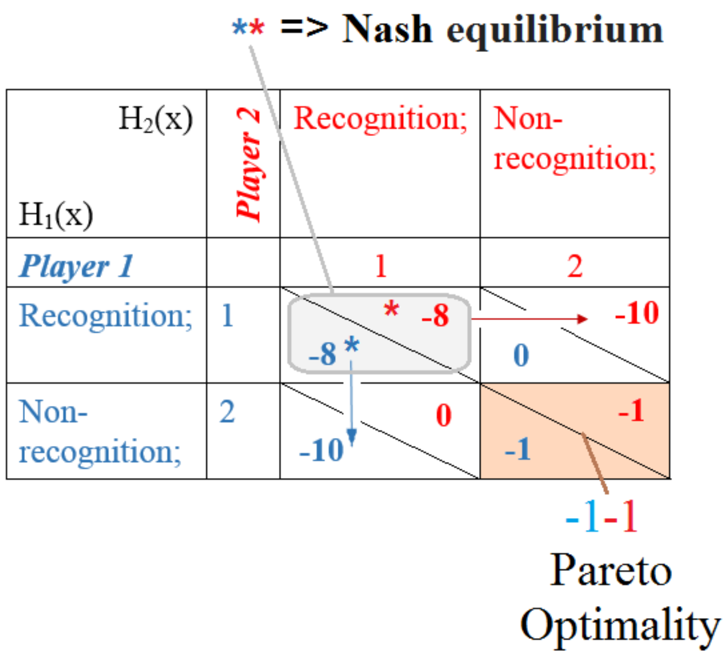
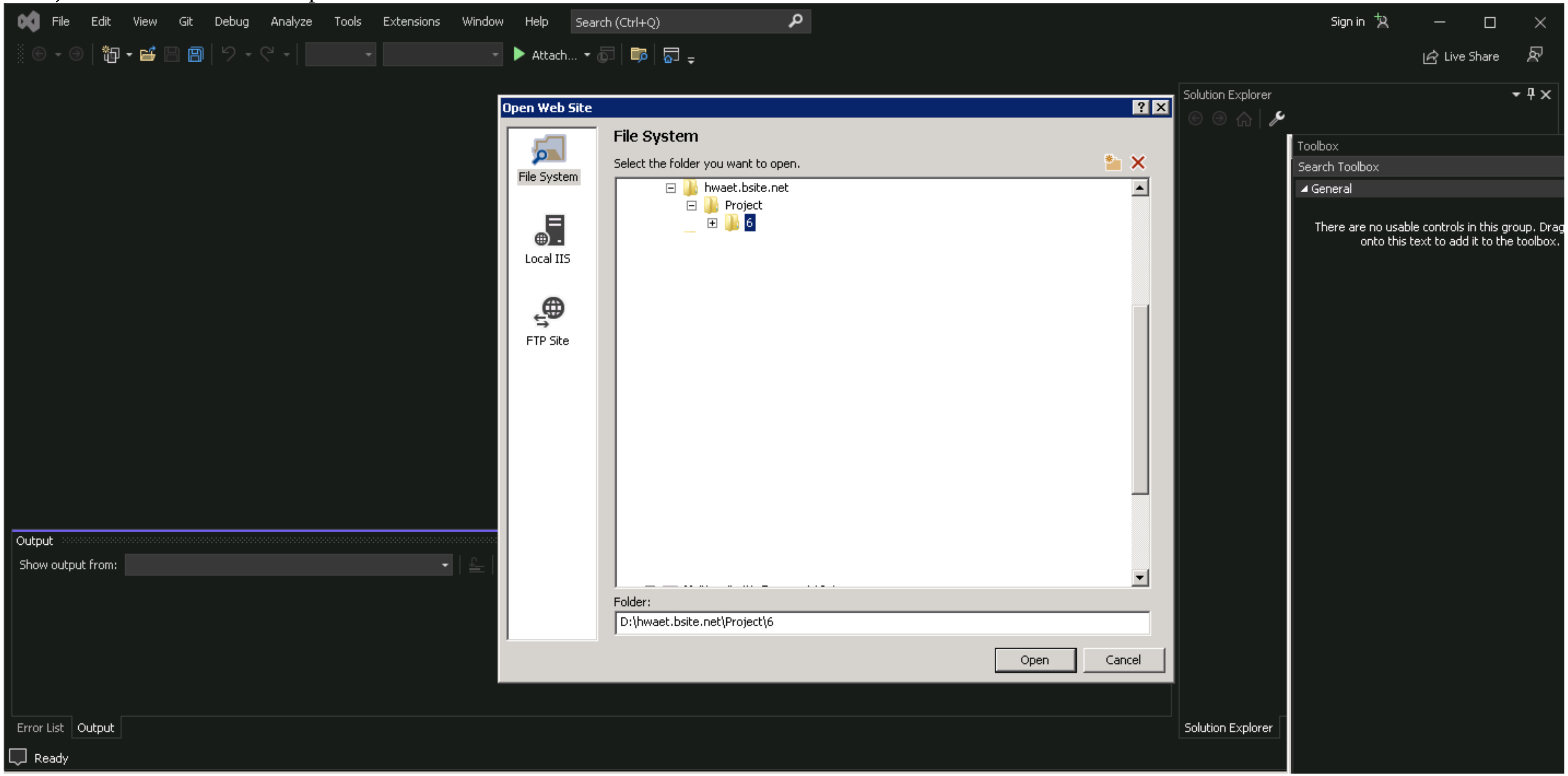


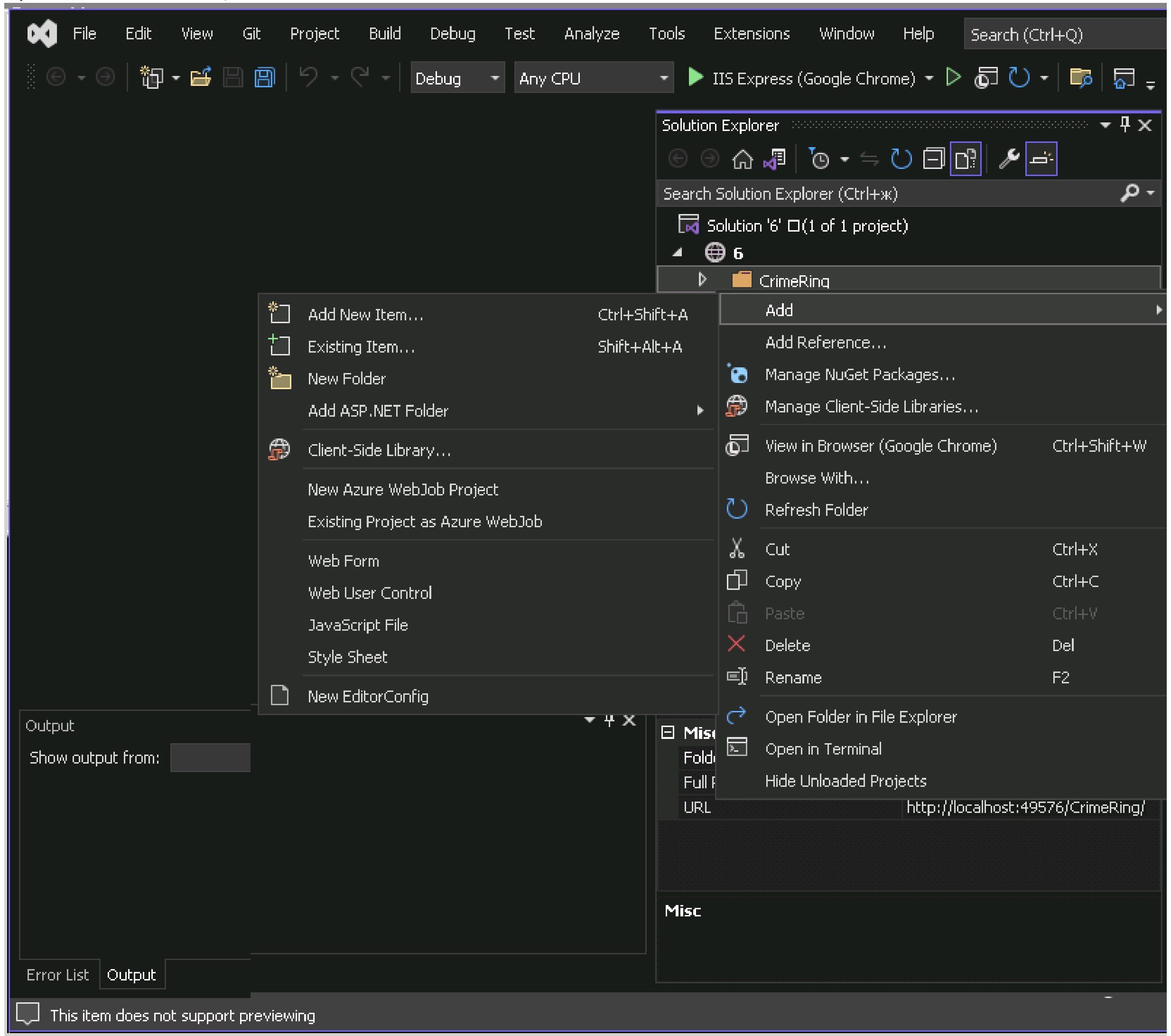
Figure 8. Nash equilibrium vs Pareto optimality

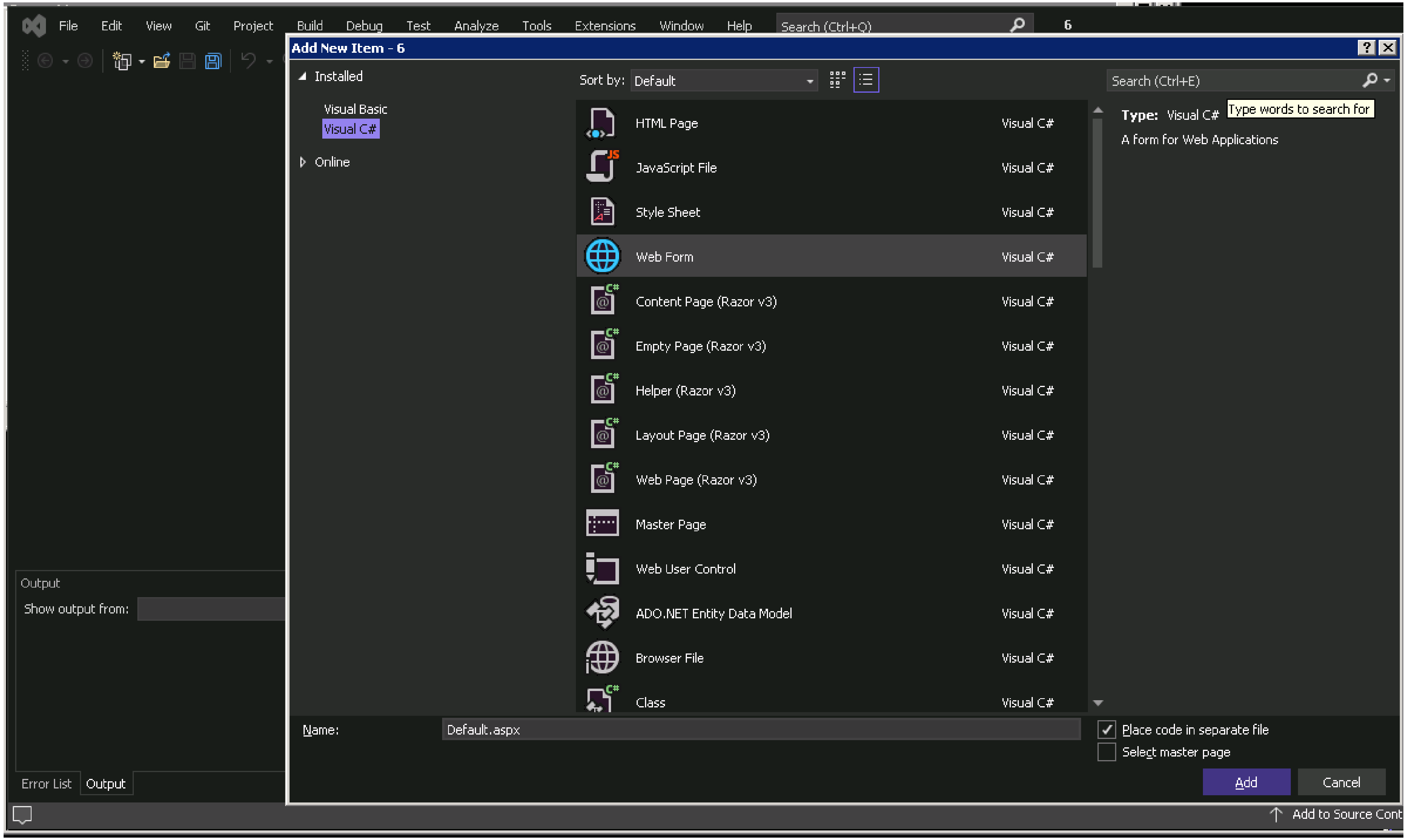
Steps.

0) Create folder for next lab
1) From Visual Studio Open WebSite from that folder



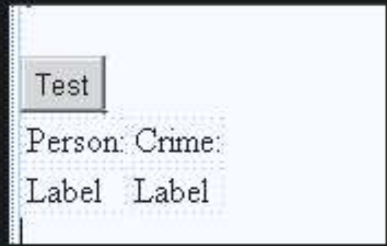
2) Add Default.aspx form



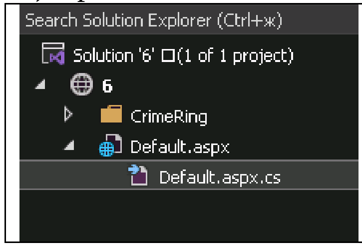

File
Edit
View
Git
Project
Build
Debug
Test
Analyze
Tools
Extensions
Window
Help
Search (Ctrl+Q)
6
Add New Item - 6
Installed
Visual Basic
Visual C#
Online
Sort by:
Default
Search (Ctrl+E)
Type: Visual C#
Type words to search for
A form for Web Applications
HTML Page Visual C#
JavaScript File Visual C#
Style Sheet Visual C#
Web Form Visual C#
Content Page (Razor v3) Visual C#
Empty Page (Razor v3) Visual C#
Helper (Razor v3) Visual C#
Layout Page (Razor v3) Visual C#
Web Page (Razor v3) Visual C#
Master Page Visual C#
Web User Control Visual C#
ADO.NET Entity Data Model Visual C#
Browser File Visual C#
Class Visual C#
Output
Show output from:
Name:
Default.aspx
Place code in separate file
Select master page
Add
Cancel
Error List
Output
Add to Source Cont

3) Add a button and 2 labels to the form

```
<asp:Button ID="Button1" runat="server" Text="Test" OnClick="Button1_Click" /><br />
<table>
    <tr>
        <td>
            Person:
        </td>
        <td>
            Crime:
        </td>
    </tr>
    <tr>
        <td>
            <asp:Label ID="LabelPerson" runat="server" Text="Label"></asp:Label>
        </td>
        <td>
            <asp:Label ID="LabelCrime" runat="server" Text="Label"></asp:Label>
        </td>
    </tr>
</table>
```



4) Open the file with C# code and add the definition of the Person class there




```
class Person
{

}
```

5) Add a method AdmitGuilty() to the class - which randomly gives the values true and false (50% by 50%)

```
class Person
{
static internal Random r = new Random();

 internal bool AdmitGuilty()
   {
      int x = r.Next(0, 100);
      if (x > 50) return true;
      return false;
   }

}
```

6) Adding the Crime class - for a person who decides to become outlaw person

```
class Crime
{
   internal Person person;
   internal bool m;              // status as a man of honor
   internal bool penito;

   internal Crime(Person p)
   {
      person = p;
   }

   public void Delete()
   {
      this.m = false;   //deprivation of the status of a person of honor and the acquisition of the status of penito
      this.penito = true;   // acquisition of the status of penito
   }
```

https://www.youtube.com/watch?v=8q7ZoB654P0&t=296s

https://www.youtube.com/watch?v=8q7ZoB654P0&t=347s

```
   internal bool RatOut()    //analog AdmitGuilty()
   {
      int x = (new Random()).Next(0, 100);//r.Next(0, 100);
```

```
        if (m)
        {
            if (x > 90) return true;   // "People of honor" (members of the crime ring) have a higher recognition threshold - 90% versus 50% of a common person
        }
        return false;
    }

}
```

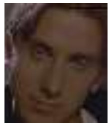
7) Add a class CrimeRing to creation a criminal community (anthropological association)

```
class CrimeRing     //class Crime Ring
{
    private Crime[] gang = new Crime[10];
    public Crime this[int i]
    {
        get
        {
            return gang[i];
        }
        set
        {
            gang[i] = value;
            gang[i]..m = true;
        }
    }
}


    protected void Button1_Click(object sender, EventArgs e)
    {
        Person orange = new Person();
        Person pink = new Person();

        CrimeRing camorra = new CrimeRing();  // Reservoir Dogs - https://youtu.be/1mxQaEZKbP0?t=92


        camorra[0] = new Crime(orange);
```



```
    camorra[1] = new Crime(pink);
```

```
    //

    // Assessing the probability of recognizing usual people
    LabelPerson.Text = " " + orange.AdmitGuilty() + "<br>" + pink.AdmitGuilty();

    // Assessing the probability of recognizing members of the crime ring "man of honor"
       LabelCrime.Text = " " + camorra[0].RatOut() +"<br>"+ camorra[1]. RatOut();
}
```