

Quiz 4 solution

Write your name on the exam. Write something for every question. Students who do not write something for everything lose out over students who write down wild guesses. You will get some points if you attempt a solution but nothing for a blank sheet of paper. Write something down, even wild guesses. Problems take long to read but can be answered concisely.

Problem	Maximum	Score
1	12	
2	8	
Total	20	

Problem 1 – Simple ASP.Net web page

Write a web page that plays the guessing game. The user will have to enter a number and submit it by the push of a button. The page will have to tell the user whether the number the user submitted is larger, smaller or exactly equal to the secret number. When the user first loads the page, it should pick the secret number randomly using the `MakeUpNumber()` method. Fill in the `.aspx` and the code behind file below. Page layout is not important. Hint: the `Session` object is used to store session state and it has a “read-write” indexer of type: `public object this[string name] { get; set; }`.

```
...
<head runat="server">
    <title>Guessing game</title>
</head>
<body>
    <form id="form1" runat="server">
        Enter your guess:<asp:TextBox ID="guessTxt" runat="server">
        </asp:TextBox><br />
        <asp:Button ID="Button1" runat="server" Text="Submit guess" />
    </form>
    <asp:Label ID="outputlbl" runat="server"></asp:Label>
</body>
```

```
...
public partial class GuessingGame : System.Web.UI.Page
{
    protected UInt32 MakeUpNumber(){...}
    protected UInt32 tonumber(string s){
        return Convert.ToUInt32(s);
    }
    protected void Page_Load(object sender, EventArgs e)
    {
        if (! IsPostBack)
        {
            Session["secret"]=MakeUpNumber();
        } else {
            UInt32 guess = tonumber(guessTxt.Text);
            if(guess<Convert.ToUInt32(Session["secret"])){
                outputlbl.Text = "My secret number is larger";
            } else if(guess>Convert.ToUInt32(Session["secret"])){
                outputlbl.Text = "My secret number is smaller";
            } else {
                outputlbl.Text = "You guessed right!";
            }
        }
    }
}
```

```
}
```

Problem 2 – Questions

a) (5 points) Briefly describe the role of VIEWSTATE and why it is needed.

VIEWSTATE is a hidden control put by ASP.NET in web pages and its role is to save the state of other controls on the page. Since all the objects used internally by ASP.NET to represent controls are discarded after the page request is served, the applications that need to keep the state of these controls between postbacks cannot rely on these objects. VIEWSTATE is a method for sending that information to the client which sends it back to the server with the next postback.

b) (3 points) Any site can be implemented without using a master page. Why are master pages still useful?

By concentrating the common elements from the individual pages in a single master page, the task of writing the pages becomes less repetitive and this can lead to a reduction in the number of mistakes. Furthermore, if the common elements change it is much easier to update the master page as opposed to updating many individual pages.