

Conceptos.

Introducción.

Cuando un sitio web necesita identificar un usuario que visita un conjunto de páginas web puede emplear cookies y variables de sesión. Como veremos las cookies se almacenan en el cliente (navegador) y son enviadas al servidor cada vez que le solicitamos una página a dicho servidor. En cambio las variables de sesión se almacenan en el servidor.

Cookies

El protocolo http es un protocolo desconectado. El protocolo http por si mismo no provee al servidor Web si dos peticiones provienen del mismo navegador. Desde este punto de vista, para el Servidor Web cada petición de página proviene de un nuevo usuario. Esta característica puede ser un inconveniente para ciertos sitios Web.

Netscape introdujo las cookies en su primera versión de navegador. Desde entonces, el World Wide Web Consortium (W3C) ha sumado las cookies al estandar. Muchos navegadores ahora pueden utilizar las cookies. Las Cookies son pequeñas piezas de información que el servidor solicita que el navegador las registre en el equipo del cliente. De acuerdo a la especificación original de Netscape una cookie no puede contener más de 4 Kb (el tamaño se refiere a la combinación del nombre y valor de la cookie)

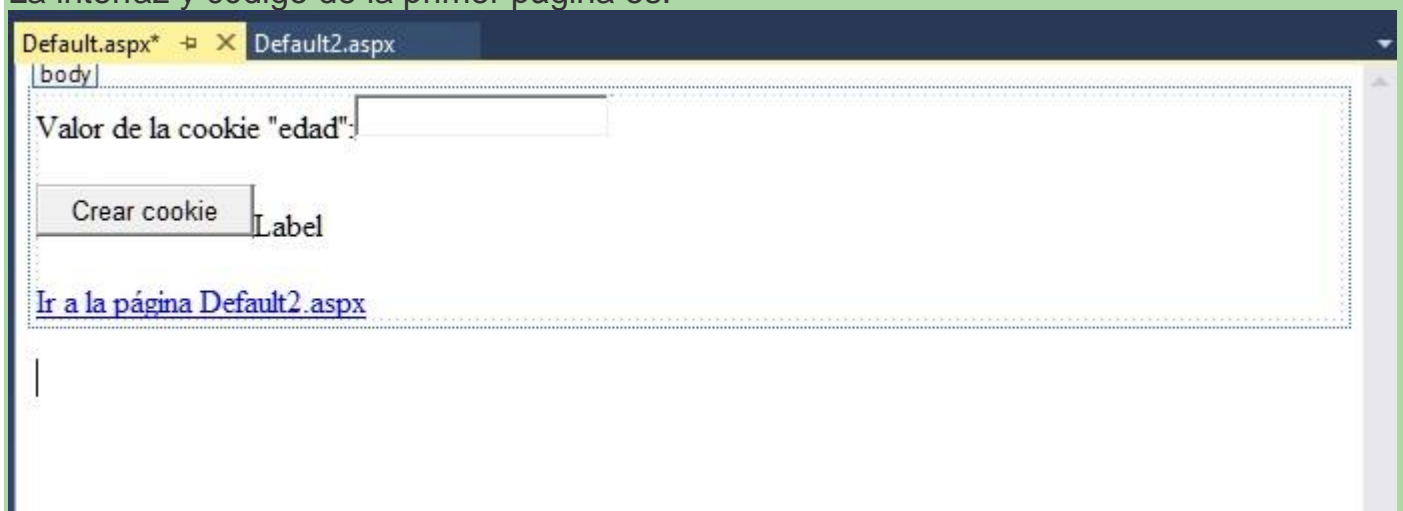
cookie persistente : Creación y acceso.

Una cookie persistente tiene una fecha de caducidad de la misma. Cuidado: No podemos garantizar que la cookie existirá por el tiempo que hemos fijado al crearla, ya que el navegador puede en cualquier momento borrarla o inclusive el usuario del equipo cliente puede manualmente borrar las cookies.

Crear un sitio web ejercicio019.

Haremos dos páginas, en una cargaremos el valor de la cookie y en la segunda la imprimiremos.

La interfaz y código de la primer página es:



The screenshot shows a web browser window with two tabs: 'Default.aspx*' and 'Default2.aspx'. The active tab is 'Default.aspx'. The page content is enclosed in a dashed border and contains the following elements:

- A text input field with the label 'Valor de la cookie "edad":'.
- A button labeled 'Crear cookie'.
- A link labeled 'Ir a la página Default2.aspx'.

Para el evento Clic del botón ?Crear cookie? escribimos el siguiente código:

```
using System;
```

Cookies y Session.

```
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;

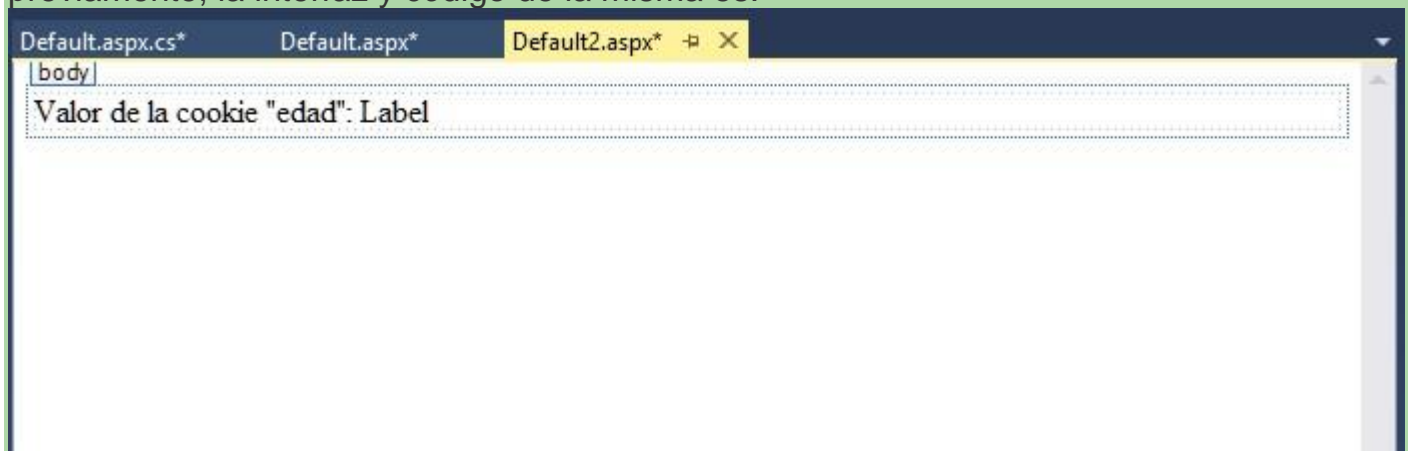
public partial class Default : System.Web.UI.Page
{
    protected void Page_Load(object sender, EventArgs e)
    {

    }

    protected void Button1_Click(object sender,
    EventArgs e)
    {
        HttpCookie cookie1 = new HttpCookie("edad",
    TextBox1.Text);
        cookie1.Expires = new DateTime(2018, 12, 25);
        Response.Cookies.Add(cookie1);
        Label1.Text = "Se creó la cookie";
    }
}
```

Creamos una cookie con el nombre ?edad? y el valor almacenado en el TextBox1. Fijamos como fecha de caducidad de la cookie el 25 de diciembre de 2018, por último llamamos al método add del objeto Cookies.

La segunda página tiene por objetivo recuperar la cookie en caso que se haya creado previamente, la interfaz y código de la misma es:



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

Cookies y Session.

```
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;

public partial class Default2 : System.Web.UI.Page
{
    protected void Page_Load(object sender, EventArgs e)
    {
        if (Request.Cookies["edad"] == null)
        {
            Label1.Text = "No existe la cookie edad";
        }
        else
        {
            Label1.Text =
this.Request.Cookies["edad"].Value;
        }
    }
}
```

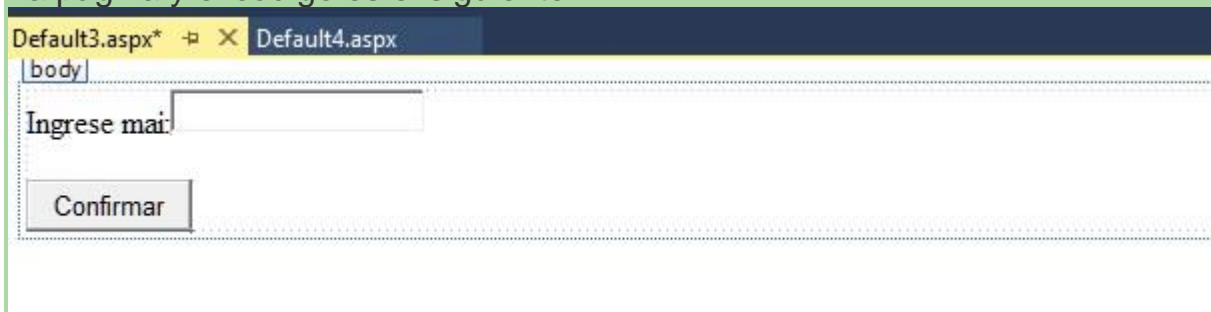
En el evento Load de la página verificamos si existe la cookie, en caso de existir la recuperamos de la propiedad Cookies del objeto Request.

1 ? Mostrar el último mail ingresado en un control TextBox.

Agregar una tercer y cuarta página a nuestro sitio web.

El objeto del siguiente ejemplo es la creación de una página que solicite el ingreso del mail de una persona, si en otro momento ya lo había ingresado mostrarlo precargado en el control TextBox.

Emplearemos una cookie persistente para almacenar el mail ingresado. La página y el código es el siguiente:

The screenshot shows a web browser window with two tabs: 'Default3.aspx*' and 'Default4.aspx'. The active tab displays a web form. At the top, there is a label 'Ingrese mail:' followed by a text input field. Below the input field is a button labeled 'Confirmar'. The form is enclosed in a simple border.

Luego codificamos para el evento clic del botón y para el evento Page_Load:

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

Cookies y Session.

```
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;

public partial class Default3 : System.Web.UI.Page
{
    protected void Page_Load(object sender, EventArgs e)
    {
        if (this.IsPostBack == false)
        {
            if (Request.Cookies["mail"] != null)
            {
                TextBox1.Text =
Request.Cookies["mail"].Value;
            }
        }

        protected void Button1_Click(object sender,
EventArgs e)
        {
            HttpCookie cookie1 = new HttpCookie("mail",
TextBox1.Text);
            cookie1.Expires = new DateTime(2018, 12, 25);
            Response.Cookies.Add(cookie1);
            Response.Redirect("Default4.aspx");
        }
    }
}
```

En el método Load de la página verificamos si es la primera vez que pedimos la página (es decir que no se recarga por presionar el botón) y si existe la cookie, en dicho caso cargamos el contenido de la propiedad Text del control TextBox.

Luego de ingresar un mail en el control TextBox y presionar el botón ?Confirmar? se crea una cookie llamada ?mail? y almacena el valor ingresado en el control TextBox. Seguidamente redirecciona a la página ?Default4.aspx?. Si Luego volvemos a ejecutar la página Default3.aspx veremos que el control TextBox aparece inicializado con el último mail ingresado (Aunque apaguemos y prendamos la máquina el último mail aparecerá dentro del control)

Como podemos ver una cookie es muy útil si queremos almacenar datos de configuración de un sitio para cada visitante.

Cookies y Session.

cookie de sesión : Creación y acceso.

La diferencia de una cookie de sesión con una persistente es que las cookies de sesión permanecen mientras no cerramos la instancia del navegador, luego el código para crear una cookie de sesión es similar a las cookies persistentes con la salvedad que no debemos especificar fecha de expiración:

```
HttpCookie cookie1=new  
HttpCookie("edad",this.TextBox1.Text);  
Response.Cookies.Add(cookie1);  
Label1.Text = "Se creó la cookie";
```

Variables de Sesión

Una variable de sesión permite preservar el valor de una variable a través de una serie de páginas. Una variable de sesión se utiliza normalmente para almacenar una preferencia del usuario, un carrito de compras, información de seguridad del usuario, nombres de usuarios, password, etc. Las variables de sesión son almacenadas durante el tiempo que el usuario visita el sitio Web.

Cuando el servidor detecta que el usuario no hace más peticiones de páginas, la información almacenada en las variables de sesión es automáticamente destruida (por defecto está configurado para que la destrucción de dichas variables suceda luego de 20 minutos de inactividad)

Podemos modificar el tiempo de vida de las variables de sesión inicializando la propiedad Timeout del objeto Session que tiene toda página (el valor que se asigna representa minutos)

```
Session.Timeout = 10;
```

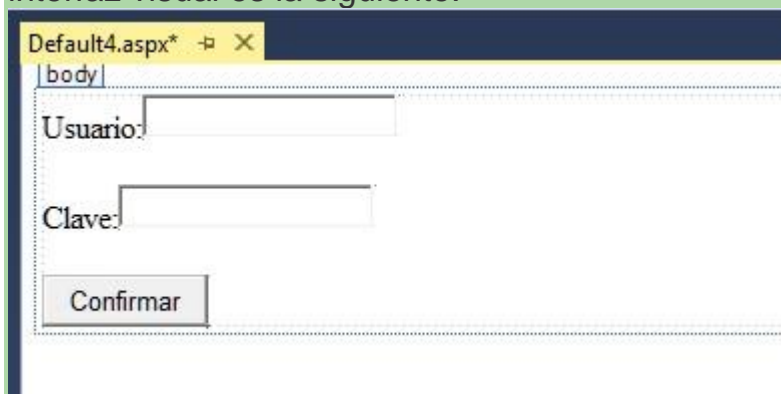
Los datos que podemos almacenar en variables de sesión pueden ser de cualquier tipo: string, int, ArrayList, etc.

Para ilustrar el uso de variables de sesión haremos una serie de páginas donde en la primera ingresaremos el nombre de usuario y clave, en la segunda los listaremos, y en esta dispondremos un hipervínculo para dirigirnos a una tercera página, donde mostraremos nuevamente el contenido de las variables de sesión.

Crear las páginas Default5.aspx y Default6.aspx.

(Default4.aspx):

Esta página solicita la carga de los dos datos y redirecciona a la segunda página, la interfaz visual es la siguiente:

The image shows a web browser window with a single tab titled 'Default4.aspx'. The page content includes a label 'body' at the top. Below it, there are two text input fields. The first is labeled 'Usuario:' and the second is labeled 'Clave:'. At the bottom of the form, there is a button labeled 'Confirmar'.

El código para el evento Click del botón confirmar es el siguiente:

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;

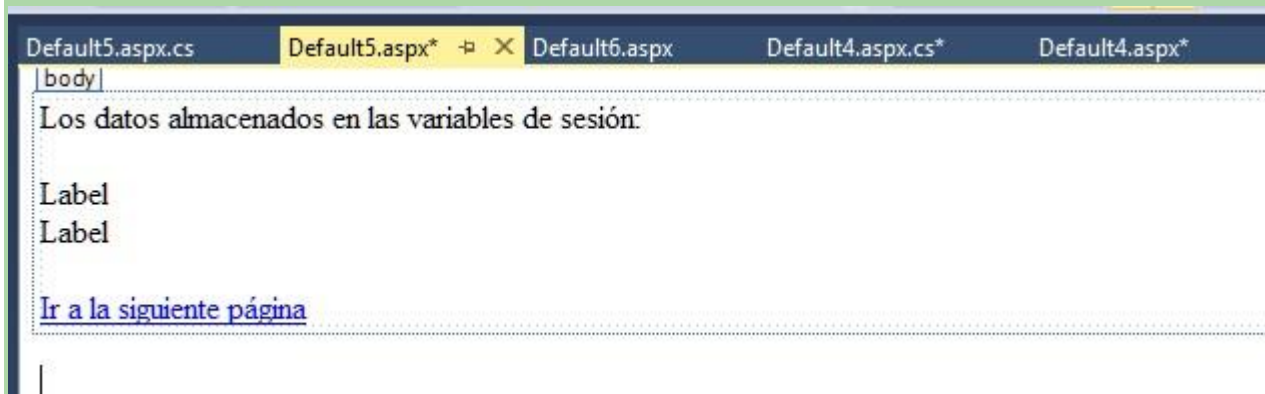
public partial class Default4 : System.Web.UI.Page
{
    protected void Page_Load(object sender, EventArgs e)
    {

    }

    protected void Button1_Click(object sender,
EventArgs e)
    {
        //Almacenamos las dos variables de sesion
        Session["usuario"] = TextBox1.Text;
        Session["clave"] = TextBox2.Text;
        //Redireccionamos a la siguiente pagina
        Response.Redirect("Default5.aspx");
    }
}
```

Default5.aspx:

Tiene por objetivo mostrar los contenidos de las dos variables de sesión. Además hay un hipervínculo (objeto de la clase HyperLink) que llama a la siguiente página. La interfaz visual es:



El código fuente de esta página es la siguiente:

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

Cookies y Session.

```
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;

public partial class Default4 : System.Web.UI.Page
{
    protected void Page_Load(object sender, EventArgs e)
    {

    }

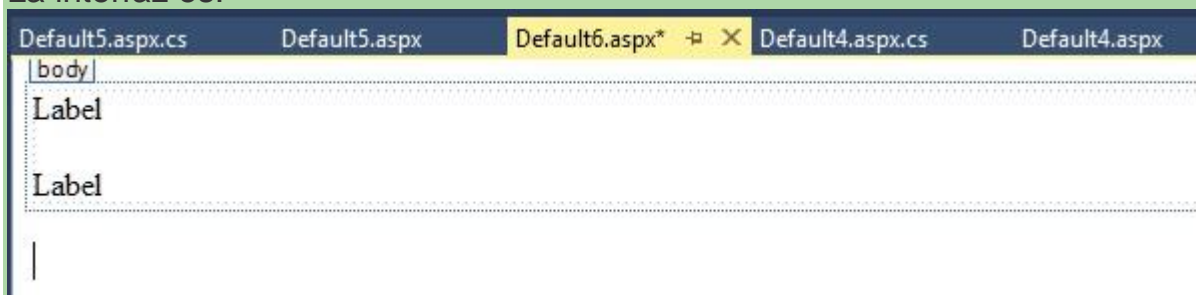
    protected void Button1_Click(object sender,
    EventArgs e)
    {
        //Almacenamos las dos variables de sesion
        Session["usuario"] = TextBox1.Text;
        Session["clave"] = TextBox2.Text;
        //Redireccionamos a la siguiente pagina
        Response.Redirect("Default5.aspx");
    }
}
```

Es decir inicializamos las Label con los contenidos de las variables de sesión que están almacenadas en memoria y administradas por el servidor Web.

Default6.aspx:

Por último esta tercer página tiene por objetivo mostrar nuevamente el contenido de las variables de sesión.

La interfaz es:



Y el código fuente del evento Load de la página es:

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

Cookies y Session.

```
using System.Web.UI;
using System.Web.UI.WebControls;

public partial class Default6 : System.Web.UI.Page
{
    protected void Page_Load(object sender, EventArgs e)
    {
        Label1.Text =
this.Session["usuario"].ToString();
        Label2.Text = this.Session["clave"].ToString();
    }
}
```

Las variables de sesión se verifican su existencia igual que las cookies.