

Clase 3 - No Presencial. Por Guillermo Guastavino

Abrir y transferir a otras páginas

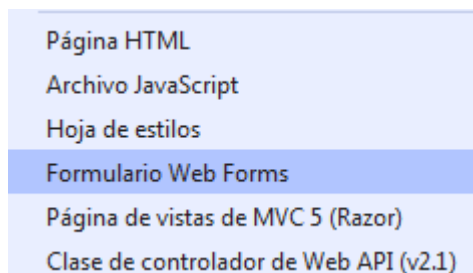
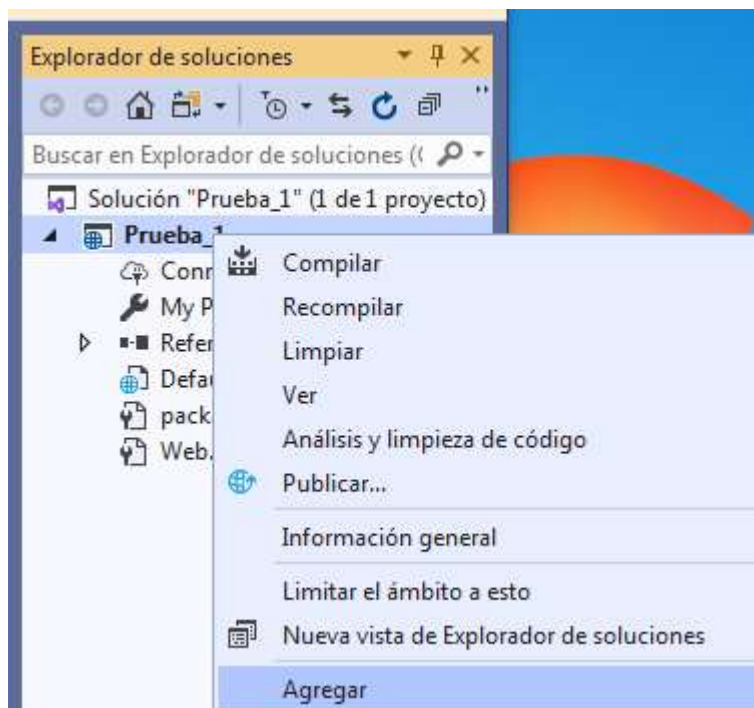
Bienvenidos a la tercera clase. Hoy vamos a ver cómo movernos entre páginas html y páginas ASP .NET y la próxima clase veremos cómo NO movernos entre páginas ASP .NET, aumentando mucho la performance, con un método que inventé para mis proyectos y que vengo usando hace años. También vamos a empezar a ver cómo mejorar la apariencia de la aplicación, pero eso me queda para la próxima. Quería poner todo ahora, pero se me hizo muy larga la clase explicando cosas muy importantes y de base, así que encararemos todo eso a partir de la clase 4.

Saltando al vacío... digo... a otra página ASP .NET

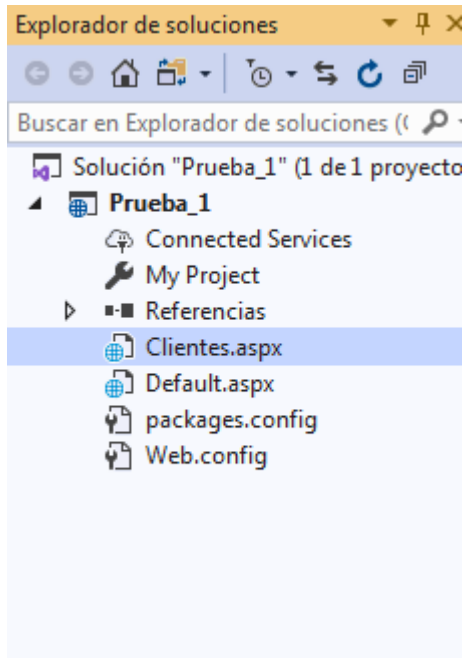
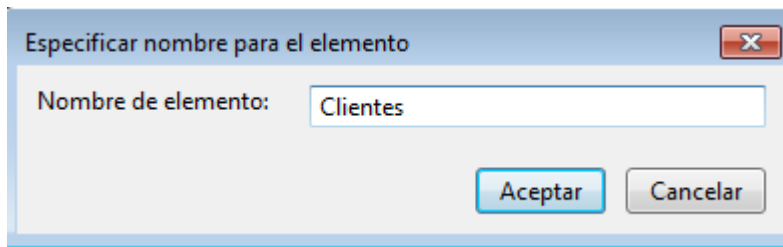
Seguro te parece exagerado o raro lo que escribí de título, pero tiene mucho de cierto..., lo vemos en un rato.

Vamos a seguir exprimiendo un poco más a Prueba_1, así que abrílo.

Ahora vas a agregar otro **web form** a tu proyecto.

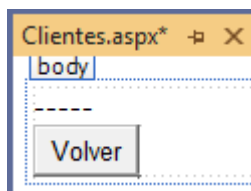


Cambiale ahora el nombre... hacerlo después es engorroso. Ponele “Clientes”.

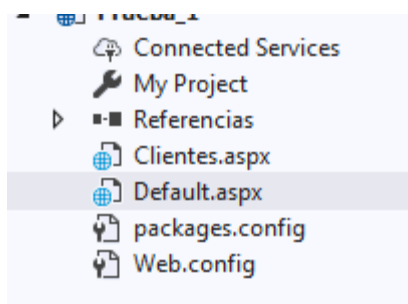


Ahora tenés un nuevo formulario web **Clientes.aspx**.

En **Diseño** de Clientes.aspx, agregá una label llamada **ICliente** (acostumbro poner adelante del nombre una letra minúscula que identifica que tipo de objeto es, en este caso la letra L minúscula por label, aclaro para los “nuevos” y para que no parezca un 1 –uno-, yo al menos los veo igual en este documento..). Ponele “-----” en text. También un botón “**bVolver**” con text: “**Volver**”.



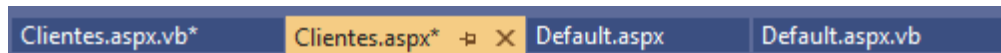
Para volver a Default.aspx, simplemente **hacé click en Default.aspx** del Explorador de Soluciones



Y luego elegís si querés Diseño o Código



Como con aplicaciones de Windows, no recomiendo que uses los tabs de arriba...



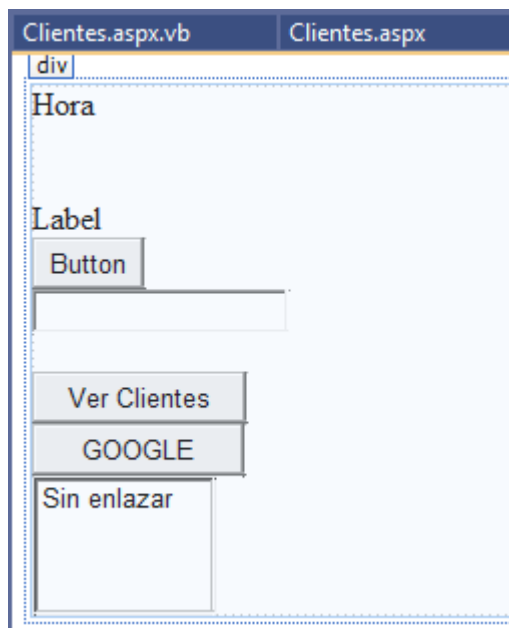
Porque, como ves, vas a tener 2 tabs por cada web form, con estos dos es manejable y podés verlos bien y usarlos, pero si tuvieras 20, 50 o más... cada uno tendría un milímetro de ancho...

Una cosa **importante**, que tal vez ya notaste antes... cada vez que ejecutás **se guarda el fuente**... o sea... tené cuidado con las “pruebas”, porque ya perdiste el código “bueno”. Los * indican que todavía no se guardó, pero al ejecutar desaparecen...

Si vas a hacer una prueba grande y que no sabés si va a servir o no; mejor copiá la carpeta del proyecto en otro lado antes.

Cosa importante 2... y una trampa en la que caigo siempre... es que **antes de ejecutar**, debés estar viendo (tener seleccionado), al web form de inicio (default.aspx), porque si no, inicia en el que estás (por ejemplo en Clientes)... y por supuesto que no anda como esperarías, y perdiste el largooooooooooooo tiempo que tarda en compilar los proyectos grandes. Me olvido siempre de esto, porque vos estás trabajando en el web form nuevo cuando querés ejecutar y manoteás directo el Play... Probá ejecutar estando en clientes.aspx...

En Default.aspx, agregá un botón **bClientes** con text “**Ver Clientes**”, otro **bGoogle** con text “**GOOGLE**”, y un listbox “**Lista**”



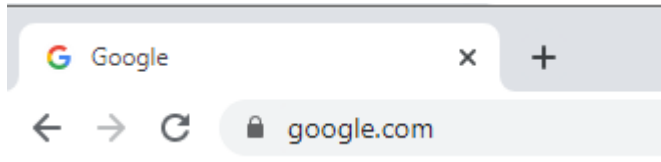
Hay dos maneras de **re direccionar** a otra página, una es desde el cliente (sin seguridad) y otra es desde el servidor.

Veamos primero como podés abrir un sitio web normal. Se hace con **Response.Redirect (url)**. Abrí el evento click del botón GOOGLE:

0 referencias

```
Protected Sub bGoogle_Click(sender As Object, e As EventArgs) Handles bGoogle.Click  
    Response.Redirect("https://www.google.com/")  
End Sub
```

Ejecutá, y apretá el botón de Google



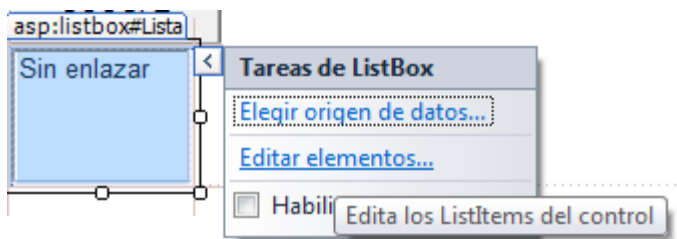
Notá que ya no estás en tu proyecto, sino que **redireccionaste** a la página de Google; para volver debés apretar la flechita ←

Te acordás del Page Load real, que ponía la hora con el cartel?, si mirás con atención, vas a ver que se ejecuta y ves al cartel un segundito, porque estás cargando la página nuevamente, así que hay Page Load... Atención con esto...

Un redirect es mejor que abrir otra ventana o pop up, porque es muy común (yo lo hago), bloquear los pop ups para que no te salgan con una oferta de inodoros con envío a domicilio por la cuarentena... Entonces, descartados los pop ups, salvo que estés haciendo una aplicación para empleados o clientes avisados de la empresa, a los que les obligás a habilitar los pop ups para que ande tu aplicación... Como ves, suena mal, así que lo descartamos como dijimos.

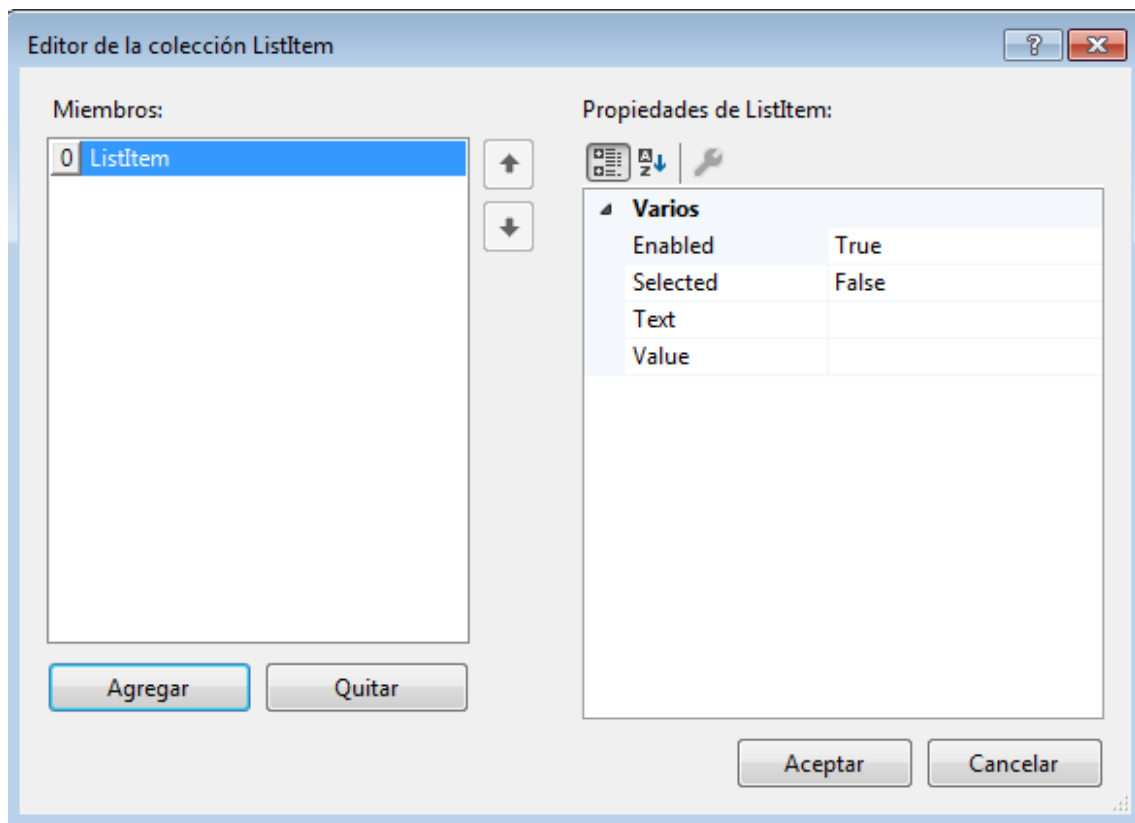
Aprovechemos para ver otro objeto “archiconocido” que (ya lo ves venir...) no anda igual que en Windows..., el **listbox**; y todos los que heredan de este.

Vamos a hacer que el listbox Lista, tenga una lista de tus sitios “favoritos”.



Tocás primero la flechita de arriba a la derecha del listbox, y luego “Editar elementos”.

Al tocar :



Bueno... hay 2 cosas para escribir en vez de una. Tenemos Text y Value. Poné en el Text Clarín (podés poner el que quieras...., pero necesitamos al menos 3 para la prueba, y se me ocurrieron Clarín, La Nación e Infobae.). Cuando terminás de escribir en el text, hacé click en value, y vas a notar que te repite Clarín. En este punto, a pesar de tener 2 textos, funcionaría similar a Windows, ya que muestra (text) lo mismo que devuelve (value).

▲ Varios	
Enabled	True
Selected	False
Text	Clarín
Value	Clarín

Entonces, en el value, poné la url de Clarín

▲ Varios	
Enabled	True
Selected	False
Text	Clarín
Value	https://www.clarin.com/

Con esto te muestra Clarín en la lista, pero al apretarlo, devuelve la url en el value, que es lo que vamos a utilizar.

Tocá Agregar de nuevo, y hacé lo mismo con La Nación y con Infobae (o las urls que quieras).

Miembros:

0	Clarín
1	La Nación

Propiedades de La Nación:

Varios	
Enabled	True
Selected	False
Text	La Nación
Value	https://www.lanacion.com.ar/

Miembros:

0	Clarín
1	La Nación
2	Infobae

Propiedades de Infobae:

Varios	
Enabled	True
Selected	False
Text	Infobae
Value	https://www.infobae.com/

Si querés que alguno de los 3 esté seleccionado al empezar, ponelo el Selected en true

Miembros:

0	Clarín
1	La Nación
2	Infobae

Propiedades de Infobae:

Varios	
Enabled	True
Selected	True
Text	True
Value	False

Al terminar:

Y acá volvemos al viejo tema del autopostback, el listbox lo tiene por defecto en false, así que por más que apretemos en la lista no pasa nada, si no apretamos también un botón.. Ponelo en true

(ID)	Lista
AccessKey	
AppendDataBound	False
AutoPostBack	True

Ahora necesitás que haga el redirect usando el value, cuando hagas clic en aún medio. Doble clic sobre Lista:

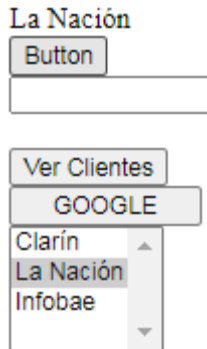
```
Protected Sub Lista_SelectedIndexChanged(sender As Object, e As EventArgs) Handles Lista.SelectedIndexChanged
    ' ...
End Sub
```

Lista.SelectedValue devuelve el value (la url) del ítem que seleccionaste.
 Lista.SelectedItem.Text devuelve el text (el diario digital) del ítem seleccionado.

```
Protected Sub Lista_SelectedIndexChanged(sender As
    Label1.Text = Lista.SelectedItem.Text
End Sub
```

Probá, label1 debe mostrar el text del Diario Digital que elegiste.

6:12:10 P.M.



Ahora probamos el value:

```
Protected Sub Lista_SelectedIndexChanged(sender As
    Label1.Text = Lista.SelectedItem.Text
    Response.Redirect(Lista.SelectedValue)
End Sub
```

Ejecutá. Vas a notar que al volver de un diario, no ves (en realidad no alcanzás a verlo nunca) a la label1 poniendo el nombre del diario. Por qué?..... pensá....

Porque como vimos antes, se ejecuta el Page Load real, y todo vuelve al principio...bueno, no tanto... probá escribir algo en el textbox, luego abrí un diario y volvé... vas a ver que sigue estando lo que escribiste; entonces no es que se “resetea”, sólo se ejecuta el PageLoad con lo que hay adentro, y como no hay ninguna sentencia que borre lo que hay en el textbox, este retiene su contenido.

Hay un efecto no deseado en la prueba anterior... Infobae está selected=true, y aparece seleccionado, pero si le hacés clic no pasa nada... en cambio a los otros sí; y si le sacás el selected=true y lo dejás false, anda todo bien, aunque nada esté seleccionado al principio. Este problema salió en el foro, y en vez de quitar el pedacito en el texto de la clase, lo dejé para explorarlo también. Veamos que pasa...

El problema es este... Infobae está seleccionada desde el mismo diseño, por lo tanto al ejecutar sigue seleccionada, y vos cuando le hacés clic el item **no cambia**, porque estás apretando el mismo índice de la colección que ya está seleccionada. Entonces si no cambia el índice, o sea está seleccionado 1, y apretás 1, entonces no se ejecuta el evento Lista_SelectedIndexChanged. Si no se ejecuta el evento, tampoco va a redireccionar a Infobae. Cuando vos le sacás el selected=true, al iniciar, el índice es -1, o sea ni 0, ni 1,

ni 2 que son los tres diarios, así que basta que aprietes cualquiera para que de -1 pase al que elegiste y salte el evento `SelectedIndexChanged` y redireccione al elegido.

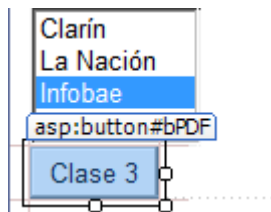
Con el `selected=true` en `infobae`, al elegir La Nación, cambia el índice de 2 (`Infobae`) a 1 (la Nación), redirecciona a la nación, y vos podrías entonces apretar `infobae` al volver, pero está el problema que al volver se recarga la página, e `Infobae` vuelve a estar seleccionada, entonces "no cambia".

El que esté `selected=true` un ítem sirve sólo si fuese para poder elegir un valor por defecto, para que el cliente no lo deje vacío, es preferible que haya algo que nada, siempre que (en contra de mi ejemplo), justo haya un redireccionamiento y luego un `PageLoad...` Por ejemplo, en una lista de tipos de documento, conviene que tengas en `true` a "DNI", así siempre hay algo marcado, y como sólo vamos a leer el `selectedItem`, para ver que tiene, no hay problema. En definitiva, úsalo sólo para estos casos.

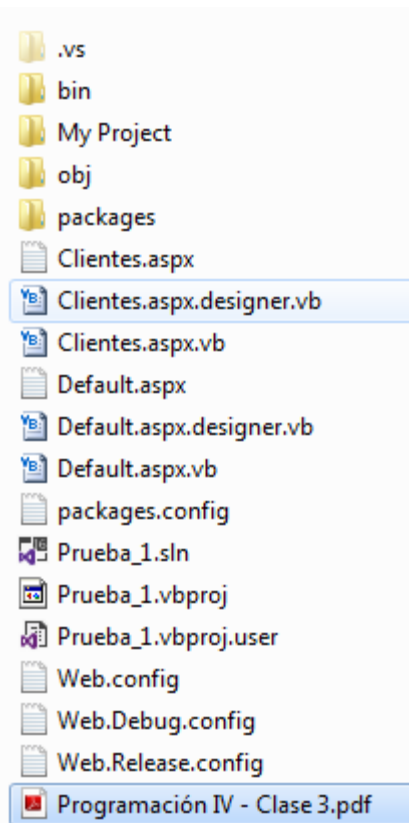
Abrir un PDF

Siguiendo con el `response`, vamos a ver ahora como poder abrir un pdf (el cliente tiene que tener instalado un programa o aplicación que pueda abrirlo).

Agregá el botón "bPDF" que diga "Clase 3"



Copió este pdf, el de la clase, “Programación IV - Clase 3.pdf” en la misma carpeta de esta aplicación (así cuando lo subas a un hosting, va a levantarlo no importa dónde esté, ya que lo busca en la misma ruta en la que está la aplicación web).

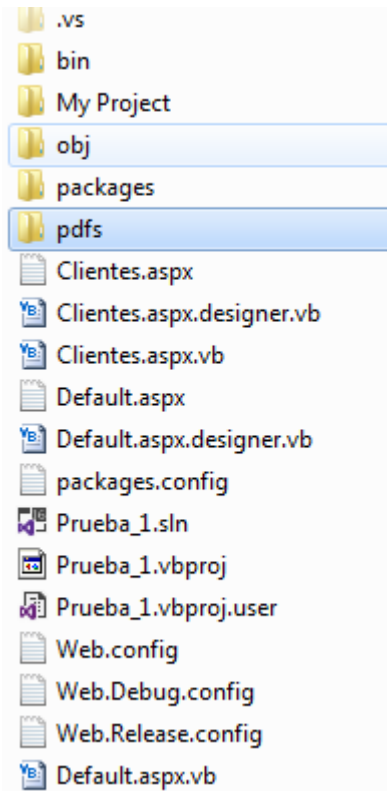


Doble clic en el botón bPDF

Creamos una variable nombrePDF. Luego le decimos al response que vamos a ejecutar un PDF y le pasamos el nombre del archivo. Lo va a ir a buscar a la misma ruta de la aplicación

```
Protected Sub bPDF_Click(sender As Object, e As EventArgs) Handles bPDF.Click
    Dim nombrePDF As String = "Programación IV - Clase 3.pdf"
    Response.Clear()
    Response.ContentType = "application/pdf"
    Response.AddHeader("Content-disposition", "attachment; filename=" & nombrePDF)
    Response.Redirect(nombrePDF)
    Response.Flush()
    Response.Close()
End Sub
```

Pero si tenés muchos pdfs tal vez quieras tenerlos en una carpeta. Creá la carpeta “pdfs” y colocá adentro el pdf de la clase



Ahora sólo necesitamos indicarle que desde la ruta de la aplicación, debe subir una carpeta más, y lo hacés directamente en la variable nombrePDF

```
Protected Sub bPDF_Click(sender As Object, e As EventArgs) Handles bPDF.Click
    Dim nombrePDF As String = "pdfs/Programación IV - Clase 3.pdf"
    Response.Clear()
    Response.ContentType = "application/pdf"
    Response.AddHeader("Content-disposition", "attachment; filename=" & nombrePDF)
    Response.Redirect(nombrePDF)
    Response.Flush()
    Response.Close()
End Sub
```

Por último, vaciamos el response.

Bueno, al fin llegamos a título “Saltando al vacío...”, vamos a transferir la ejecución (server.transfer) a clientes.aspx

En el botón “bClientes”:

```
Protected Sub bClientes_Click(sender As Object, e As EventArgs) Handles bClientes.Click
    Server.Transfer("Clientes.aspx")
End Sub
```

Y para poder volver desde Clientes... en el botón bVolver:

```
Protected Sub bVolver_Click(sender As Object, e As EventArgs) Handles bVolver.Click
    Server.Transfer("Default.aspx")
End Sub
```

Probalo....

Posiblemente pienses que anda muy bien... pero tiene un pero...

ASP .NET trabaja sin estado (**stateless**), significa que no guarda información de los estados anteriores. Para el servidor, cada vez que cambiás de web form (entre otras cosas...), “**se olvida**” de dónde vino, y para cuando abre clientes, no sabe que viniste desde default, ni siquiera que estuviste usando la aplicación por horas... estilo “sólo vive el momento”.

Un ejemplo:

Estás en la page Menu, elegís Clientes. Se abre Clientes, te muestra una lista de clientes y te permite buscar uno. Buscás Gonzalez, y te aparecen 5..., en una grilla elegís a Gonzalez Juan, con legajo 1500. Ahora apretás el botón VENDER, porque vas a venderle cosas a Gonzalez. Te abre la página de ventas, dónde muestra la lista de todos los productos, para que elijas los que le vas a vender, te tomás tu tiempo y elegís 2, el 250 y el 740, ahora apretás FACTURAR.

En la page FACTURAR, le vas a facturar a.... a quién???...no sé....le ibas a facturar estos productos.... Eh... cuáles?... Eso es lo que “piensa” el servidor, no tiene idea de lo que fuiste haciendo antes. Cuando elegiste a Gonzalez, y lo guardaste (más bien guardaste el legajo 1500 en una variable), pensando que al ir a ventas ibas a tener a la variable “viva”... bueno, no... se murió abandonada...lo mismo pasó con los productos 250 y 740 que elegiste, al llegar a Facturar, **ya no existían esas variables**. Eso es importante, no es que las variables perdieron su valor, directamente NO existen en el otro web, y aún si las hubieras declarado también en ventas y en facturación, esas variables son OTRAS variables. Cada vez que pasaste perdiste todo. Inclusive, si al llegar a ventas, hubieras vuelto a clientes, hubiera aparecido la lista completa de clientes, y sin rastro alguno del Gonzalez que elegiste... todo de nuevo, y para nada. Por todo esto vino el título de “saltando al vacío”, porque al ir a otro web form sin haber tomado medidas, es como saltar al vacío sin paracaídas.

Para poder saber el legajo estando en ventas, necesitás antes habérselo pasado antes... Acá hay varios métodos. El primero son las cookies, que es un archivo de datos creado en el servidor y enviado al cliente, en dónde reside (por ejemplo las famosas y super molestas cookies que te dejan las páginas webs cuando las visitás, y quedás marcado para siempre que te gustan los corpiños puntiagudos o los inodoros japoneses, sólo porque tuviste la desgracia de tocar el link incorrecto).

Las cookies son muy fáciles de modificar, porque están en tu pc, no en el servidor. Si yo veo una cookie que dice 1501 (suponiendo que sea yo), eso quiere decir que debe haber posiblemente un cliente 1500, la cambio, y de golpe para la web soy Gonzalez...entonces compro algo a nombre de Gonzalez y que me lo manden, no se... cualquier cosa es posible...

Lo mismo pasa con las variables PHP que se pasan en la url estilo www.comprotodo.com/?legajo=1500 (es un ejemplo burdo pero que sirve para contar lo que pasa), cambiemos entonces la url para que diga otro número... Muchas veces esto está oculto, pero oculto por el código html, que también está en mi pc, y lo veo y lo cambio facilísimo. Entonces me queda encriptar para que diga (como debiste ver muchas veces) algo estilo: www.comprotodo.com/?l=dkla3454c564afg422aJJ3421331SDD

Aún así el programador está exponiendo información interna y muy sensible de la aplicación a cualquiera que anda por allí aburrido por la cuarentena o con ganas de conseguir algo gratis, y te lo hackean horriblemente.

Lo descartamos también...

El mecanismo que queda es la creación de variables de sesión "session". Estas variables se crean en el servidor, durante el período de tu sesión, y se libera el recurso luego. Como buscamos que haya millones de personas usando la aplicación, estas variables deben ser muy miserables para cada uno.

Session("legajo")=1500 así de simple creo una variable de sesión llamada Legajo. Estas variables soportan distintos tipos de datos.

Para recuperar el valor en ventas hacemos:

```
Dim legajo as integer=Session("legajo")
```

Lo loco del asunto es que, si quiero volver a clientes desde ventas, para hacer otra cosa con Gonzalez, por ejemplo verle la cuenta corriente; debo pasarle a clientes NUEVAMENTE la variable de session o cerciorarme de que aún esté activa, y en clientes tomar el legajo, buscar a Gonzalez de nuevo... Entonces, en clientes busco a Gonzalez (primera búsqueda en la base de datos), obtengo 1500, se lo paso a Ventas. En ventas no sé quién es el cliente, así que leo la variable, busco a ver quién es el 1500 y me entero que es Gonzalez (segunda búsqueda en la base de datos de la misma información). Ahora en ventas elijo los productos 250 y 740. Para ir a Facturar, debo pasarle el legajo y los 2 productos en 3 variables de sesión. En Facturar, nuevamente no sé quién es el cliente ni que compró, así que hay que buscar por 3ra vez a Gonzalez, y por segunda vez a los productos. Terminada la facturación, vuelve a ventas, pasándole los 3 datos. En ventas, por si quiere comprar otra cosa, debo buscar a Gonzalez por 4ta vez. Vuelvo a clientes pasándole el legajo. En Clientes para continuar en Gonzalez por si hay que hacer otra cosa como ver su Cuenta Corriente; lo leo por 5ta vez!!!!. Para evitar todo esto, y lograr exactamente lo mismo pero leyendo a Gonzalez UNA SOLA VEZ, y a los productos UNA SOLA VEZ, la clase que viene les explico mi método, que es mi figurita difícil con la que me diferencio en las aplicaciones, porque, siguiendo el ejemplo anterior, evitando el lío de las re lecturas, la aplicación sería más de 5 veces más rápida.

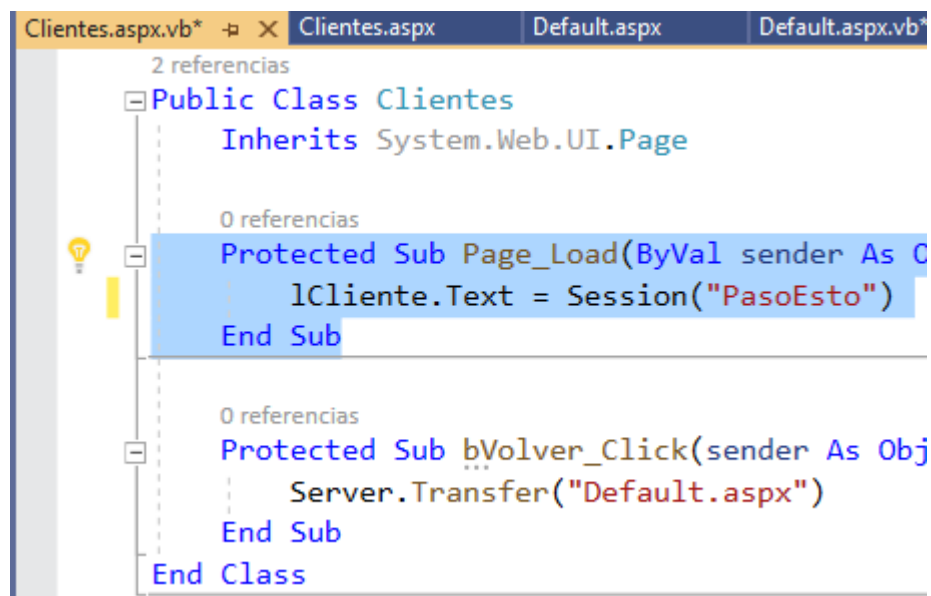
Vamos ahora a probar el paso por Session.

En Default.aspx, en el evento clic del botón bClientes, antes del server.transfer (o sea, debo pasar el valor **antes** de ir al otro web), ponemos:

```
Protected Sub bClientes_Click(sender As Object, e As EventArgs) Handles bClientes.Click
    Session("PasoEsto") = TextBox1.Text.Trim
    Server.Transfer("Clientes.aspx")
End Sub
```

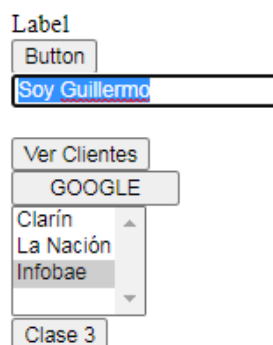
Con esto creás una variable de sesión llamada “PasoEsto” (tené cuidado con las mayúsculas y las minúsculas), y le cargamos lo que vas a escribir en el textbox.

Ahora andá a Clientes.aspx, al PageLoad, en dónde vamos a leer la variable “PasoEsto” y le vamos a cargar su valor en la label lCliente, así podés comprobar que se pasó el valor.



Volvé a Default y ejecutá. Escribí algo en el textbox y luego apretá el botón “Ver Clientes”

9:38:34 A.M.



En Clientes:

Soy Guillermo

[Volver](#)

Prestá atención a las urls, en ningún momento hay variables tipo “?PasoEsto=” ni nada... se pasan los valores en el servidor, y desde el cliente no se ve.

Bueno... terminamos por hoy, espero que les haya gustado y parecido interesante. La clase que viene vemos eso que les prometí para evitar “cargar a Gonzalez 5 veces” y comenzamos con las mejoras gráficas.

Nos leemos en el foro!