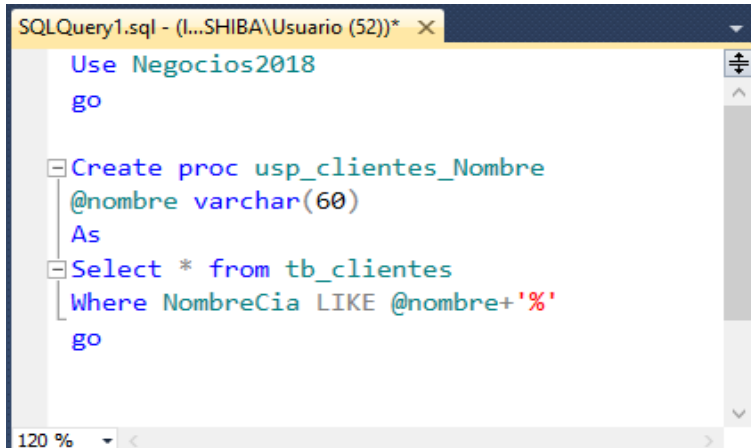


## Sesión 01: Consulta de Datos con parámetros

Implemente un proyecto ASP.NET MVC, diseñe una página que permita listar los registros de la tabla tb\_clientes de la base de datos Negocios2018, filtrando por las iniciales del campo NombreCia.

### Creando el procedimiento almacenado

En el administrador del SQL Server, defina el procedimiento almacenado usp\_clientes\_Nombre, tal como se muestra.

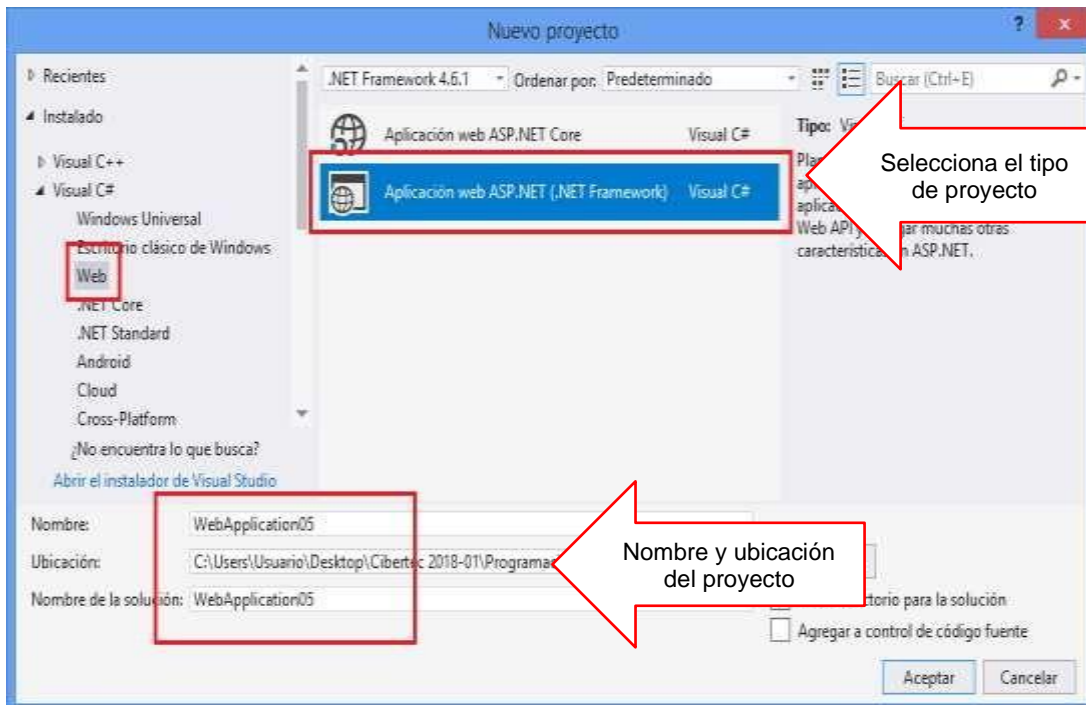


```
SQLQuery1.sql - (I...SHIBA\Usuario (52))* X
Use Negocios2018
go

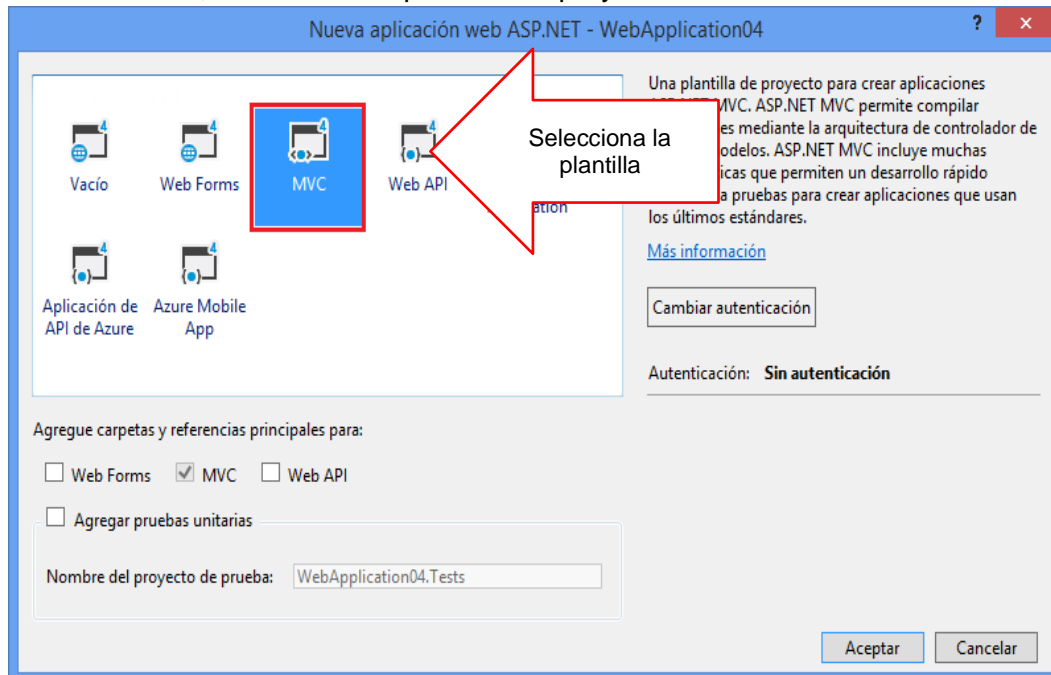
Create proc usp_clientes_Nombre
@nombre varchar(60)
As
Select * from tb_clientes
Where NombreCia LIKE @nombre+'%'
go
```

### Creando el proyecto

Iniciamos Visual Studio; seleccionar el proyecto Web; selecciona la plantilla Aplicación web ASP.NET (.NET Framework), asignar el nombre y ubicación del proyecto; y presionar el botón ACEPTAR

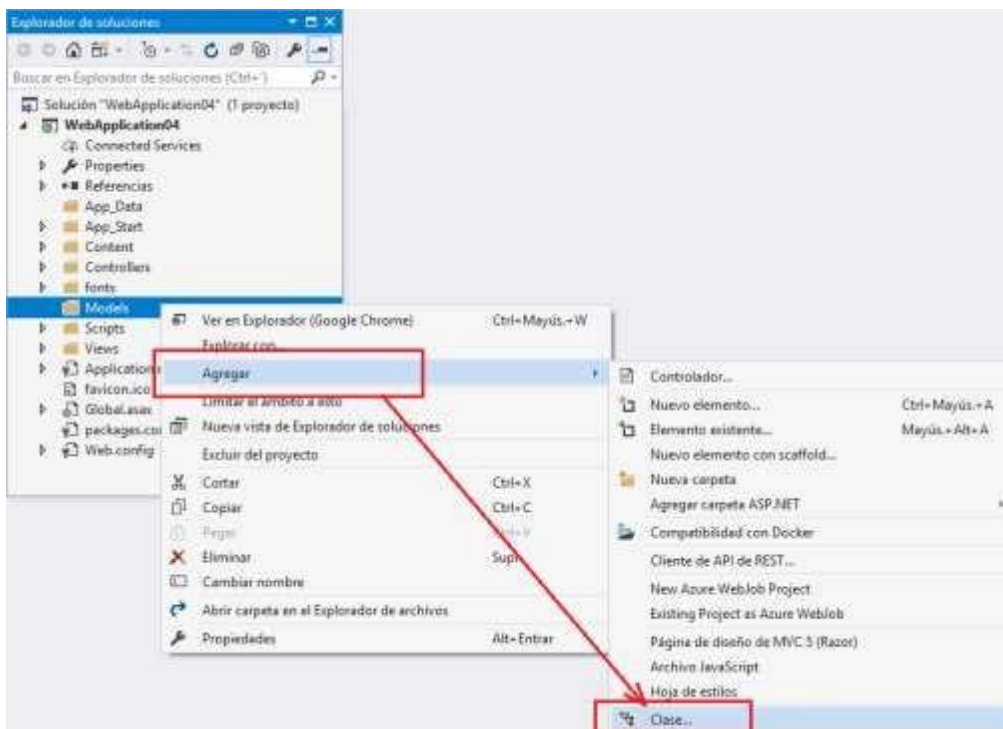


A continuación, seleccionar la plantilla del proyecto MVC.

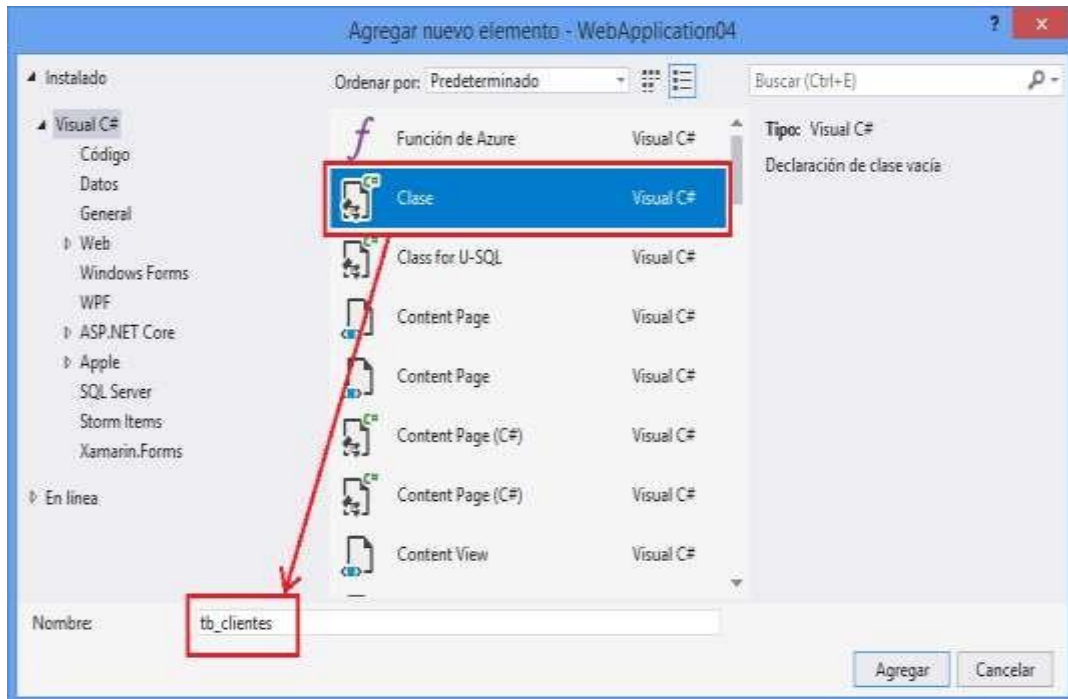


## Creando la Clase del Modelo

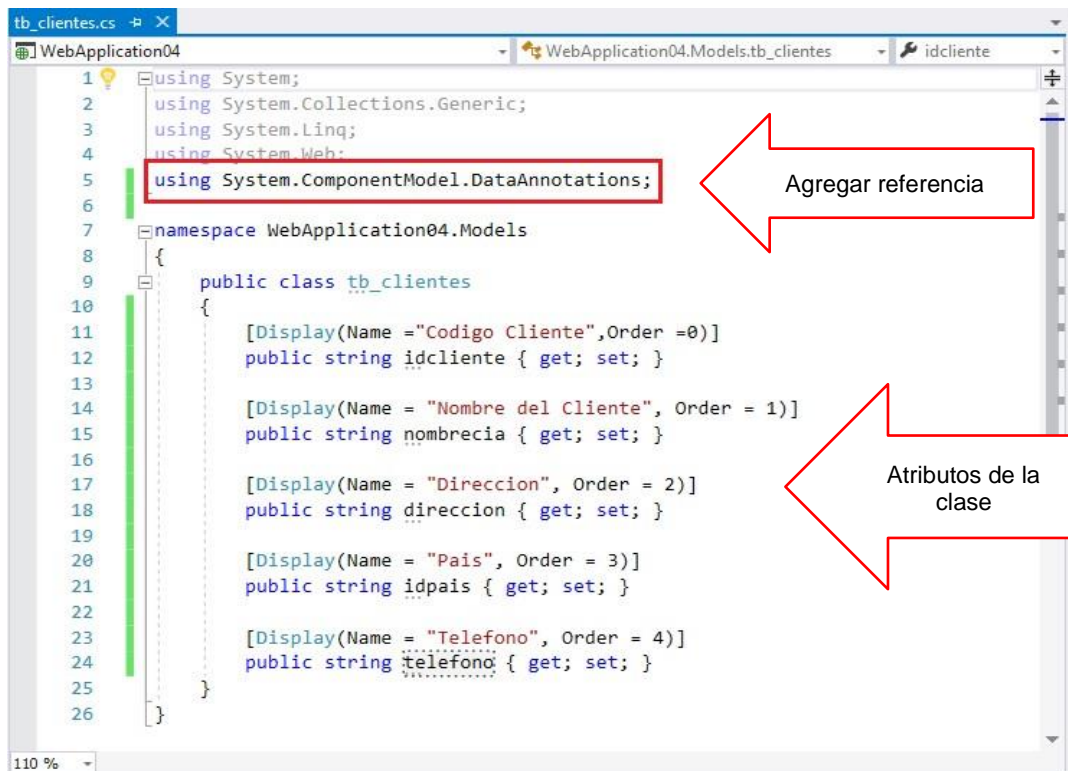
Para efectuar y realizar la consulta de los datos de la tabla tb\_clientes, agregamos, en la carpeta Models, una clase, tal como se muestra.



Asigne el nombre de la clase, la cual se llamará `tb_clientes`, tal como se muestra. Presiona el botón AGREGAR

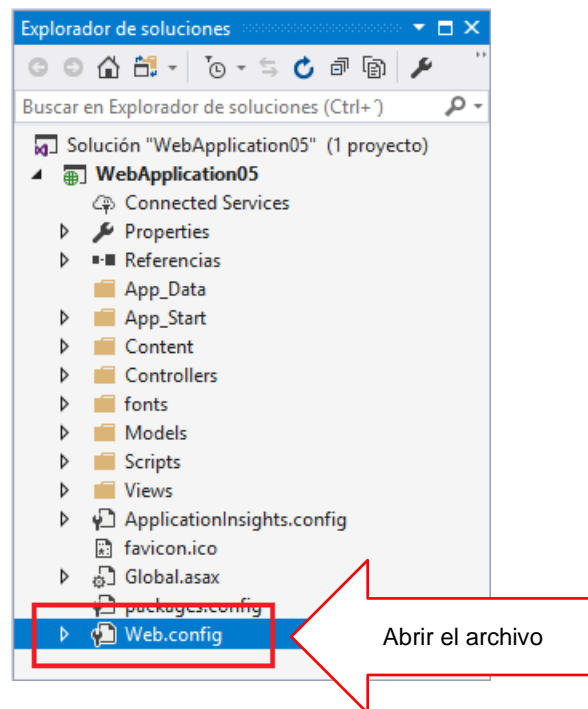


Para iniciar, agregamos la referencia `DataAnnotations`, luego defina los atributos de la clase, visualizando el texto y orden de cada uno de ellos [`Display (Name="texto", Order="n")`], tal como se muestra.



## Publicando la cadena de conexión

Abrir el archivo Web.config, tal como se muestra

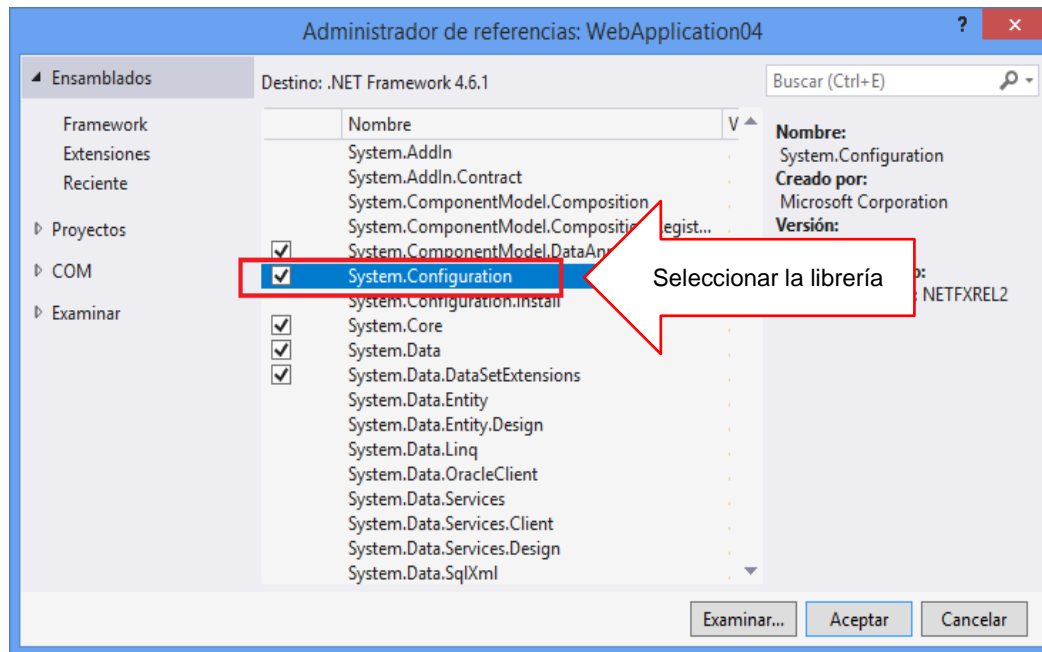


Defina la etiqueta <connectionStrings> para agregar una cadena de conexión a la base de datos Negocios2018, tal como se muestra.



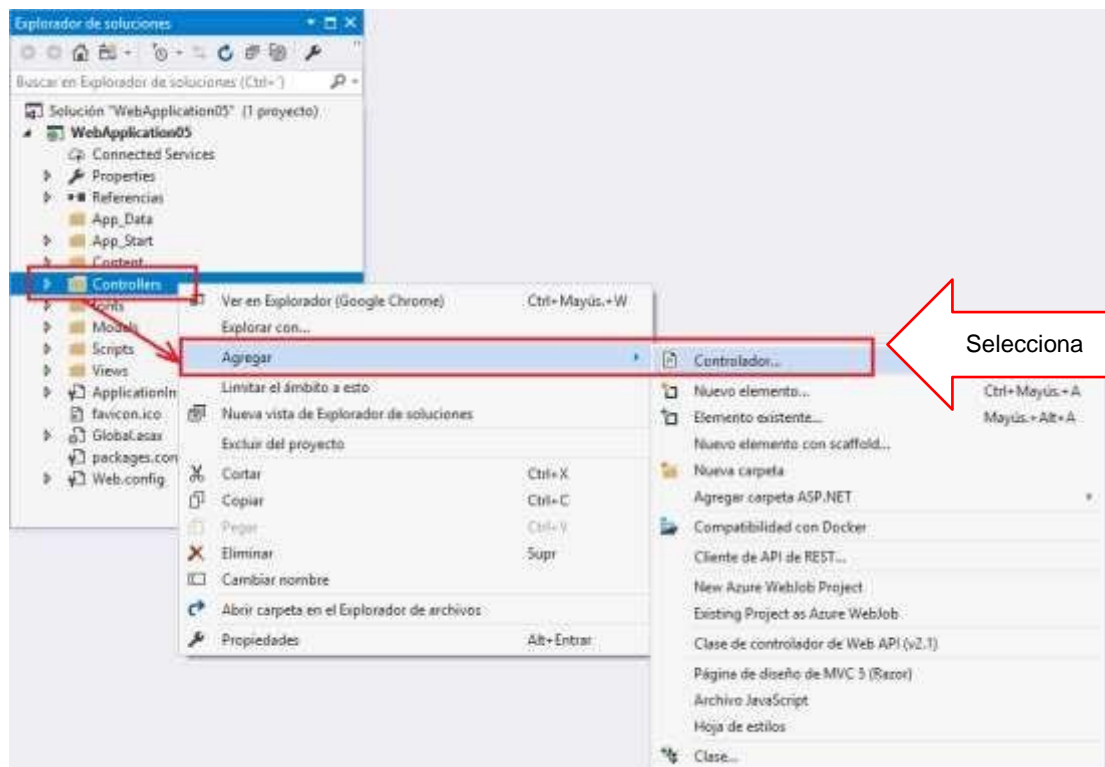
A continuación agregar la referencia al proyecto:

Selecciona desde la opción **Proyectos** → **Agregar Referencia**, selecciona la librería System.Configuration, tal como se muestra. Presiona el botón ACEPTAR.



## Creando un Controlador

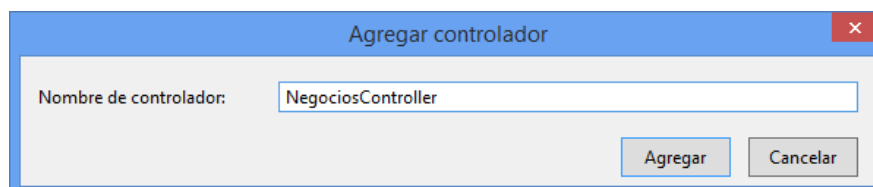
A continuación, creamos un controlador: Desde la carpeta **Controllers**, selecciona la opción **Agregar, Controlador...**, tal como se muestra.



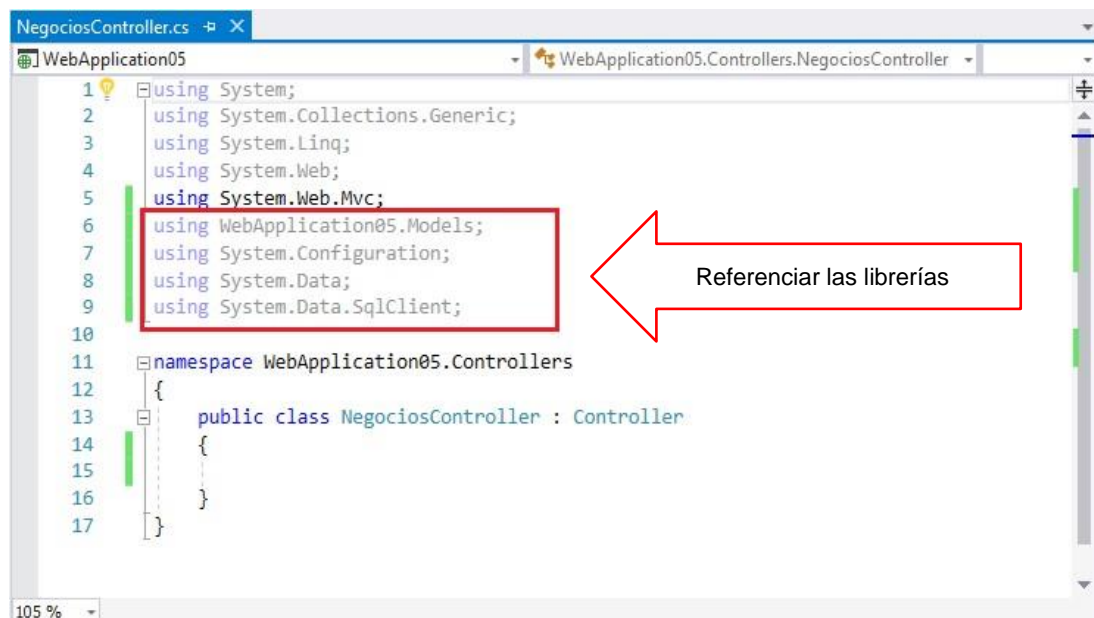
Selecciona el scaffold, Controlador de MVC5: en blanco, presiona el botón AGREGAR



Asigne el nombre del controlador: NegociosController, tal como se muestra



Referenciar la carpeta **Models**, la librería **Data.SqlClient** y el **System.Configuration** tal como se muestra.



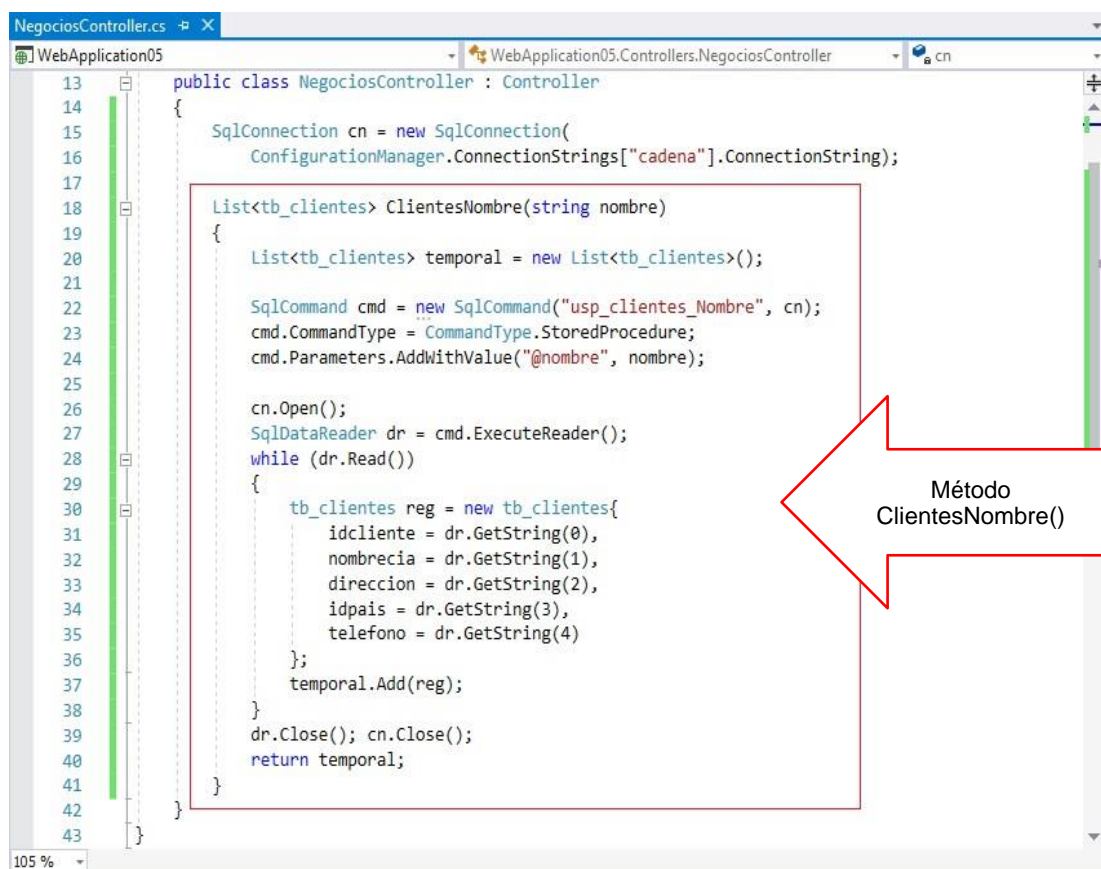


Dentro del controlador, defina la conexión a la base de datos Negocios2018 instanciando el **SqlConnection**, utilice el **ConfigurationManager**.

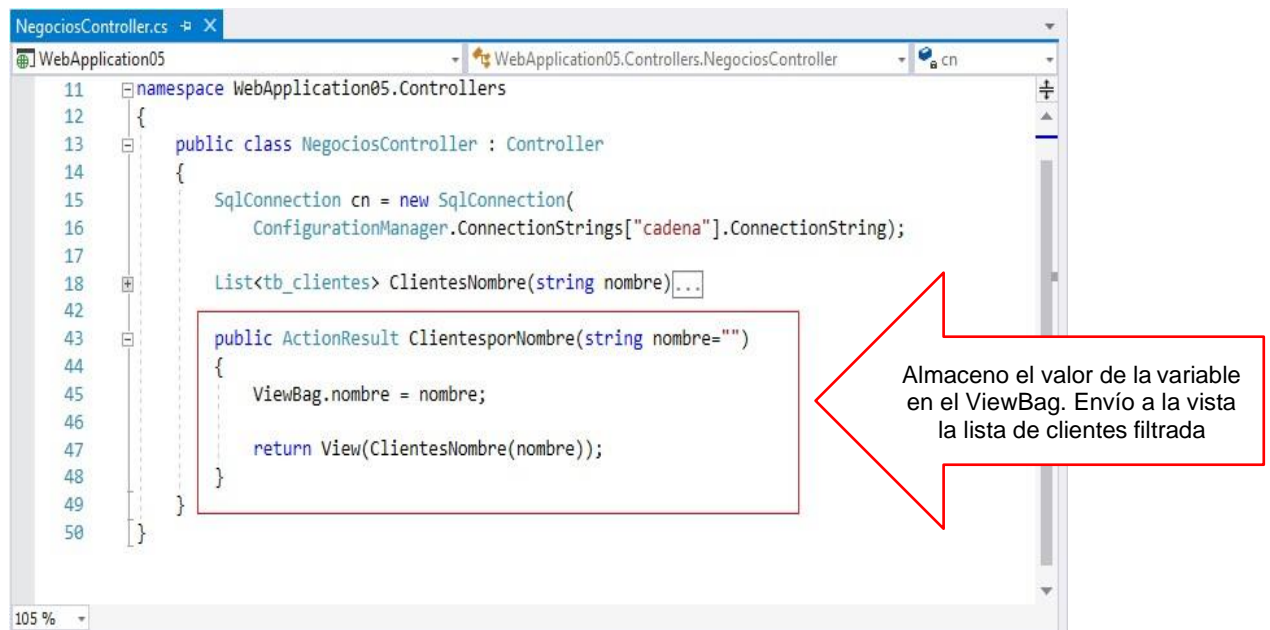


A continuación codifique la lista llamada ClientesNombre:

1. Ejecuta el procedimiento almacenado usp\_clientes\_Nombre a través del SqlCommand, pasando el valor de su parámetro @nombre. El resultado se almacena en el SqlDataReader.
2. Para guardar los resultados en el temporal se realiza a través de un bucle (mientras se pueda leer: dr.Read()), vamos almacenando cada registro en el temporal.
3. Cerrar los objetos y enviar el objeto lista llamada temporal.



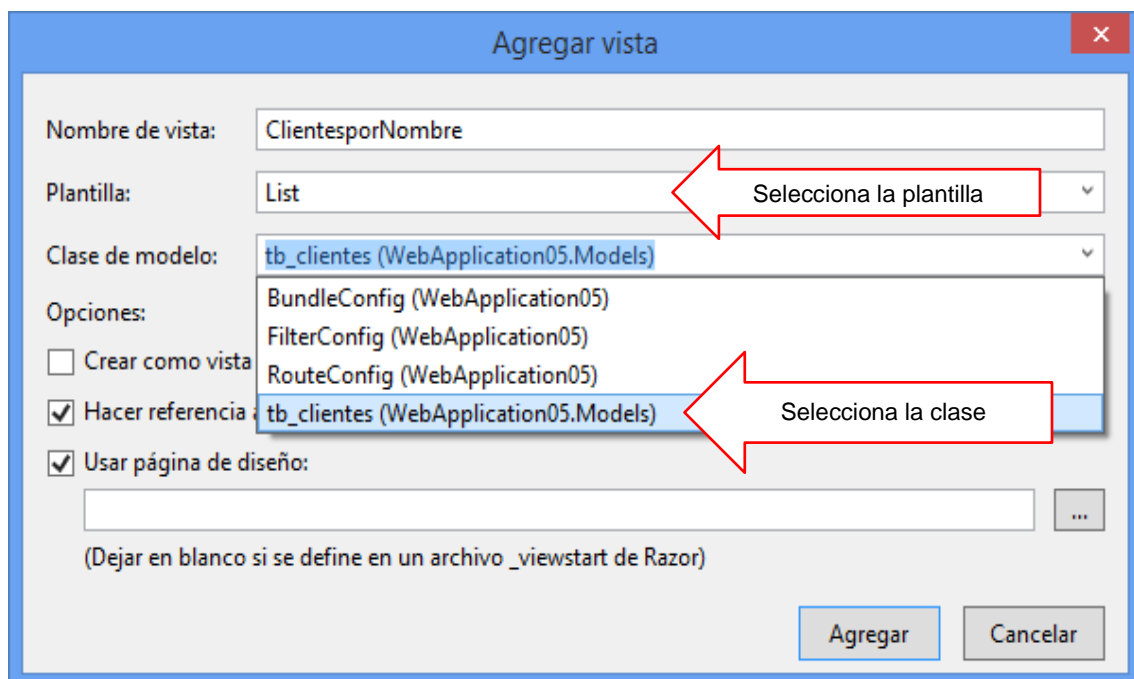
Defina el ActionResult **CientesporNombre**(string **nombre**), el cual enviará a la Vista el resultado del método **CientesNombre**(). Inicialice el parámetro **nombre**, tal como se muestra.



## Agregando la Vista.

En el ActionResult, agrega la vista.

En dicha ventana, selecciona la **plantilla**, la cual será **List**; y la **clase de modelo** la cual es **tb\_clientes**, tal como se muestra





Vista generada por la plantilla **List**, tal como se muestra

```
1 @model IEnumerable<WebApplication05.Models.tb_clientes>
2
3 @{ ViewBag.Title = "ClientesporNombre"; }
4
5 <h2>Listado de Clientes por Nombre</h2>
6
7 <p><a href="#">Create New</a></p>
8
9 <table class="table">
10     <tr>
11         <th>@Html.DisplayNameFor(model => model.idcliente)</th>
12         <th>@Html.DisplayNameFor(model => model.nombrecia)</th>
13         <th>@Html.DisplayNameFor(model => model.direccion)</th>
14         <th>@Html.DisplayNameFor(model => model.idpais)</th>
15         <th>@Html.DisplayNameFor(model => model.telefono)</th>
16     </tr>
17
18     @foreach (var item in Model) {
19     <tr>
20         <td>@Html.DisplayFor(modelItem => item.idcliente)</td>
21         <td>@Html.DisplayFor(modelItem => item.nombrecia)</td>
22         <td>@Html.DisplayFor(modelItem => item.direccion)</td>
23         <td>@Html.DisplayFor(modelItem => item.idpais)</td>
24         <td>@Html.DisplayFor(modelItem => item.telefono)</td>
25
26         <td>
27             <a href="#">Edit</a> |
28             <a href="#">Details</a> |
29             <a href="#">Delete</a>
30         </td>
31     </tr>
32     }
```

Recibe la lista de clientes

Cabecera de la lista

Registros que se visualizaran en la Vista

Modifica la Vista, agregando un formulario, tal como se muestra

```
1 @model IEnumerable<WebApplication05.Models.tb_clientes>
2
3 @{ ViewBag.Title = "ClientesporNombre"; }
4
5 <h2>Listado de Clientes por Nombre</h2>
6
7 @using (Html.BeginForm()) {
8     <span>Ingrese el Nombre del cliente</span>
9     <input type="text" name="nombre" value="@ViewBag.nombre" />
10    <input type="submit" value="Consulta" />
11 }
12
13 <table class="table">
14     <tr>
15         <th>@Html.DisplayNameFor(model => model.idcliente)</th>
16         <th>@Html.DisplayNameFor(model => model.nombrecia)</th>
17         <th>@Html.DisplayNameFor(model => model.direccion)</th>
18         <th>@Html.DisplayNameFor(model => model.idpais)</th>
19         <th>@Html.DisplayNameFor(model => model.telefono)</th>
20     </tr>
21
22     @foreach (var item in Model) {
23     <tr>
24         <td>@Html.DisplayFor(modelItem => item.idcliente)</td>
25         <td>@Html.DisplayFor(modelItem => item.nombrecia)</td>
26         <td>@Html.DisplayFor(modelItem => item.direccion)</td>
27         <td>@Html.DisplayFor(modelItem => item.idpais)</td>
28         <td>@Html.DisplayFor(modelItem => item.telefono)</td>
29
30         <td>
31             <a href="#">Edit</a> |
32             <a href="#">Details</a> |
33             <a href="#">Delete</a>
34         </td>
35     </tr>
36     }
```

Agrega un Formulario para ingresar el valor para el parámetro "nombre"

Ejecutamos la Vista, ingresa las iniciales en el campo text (nombre), al presionar el botón Consulta (submit) visualizamos los registros de la tabla tb\_clientes



The screenshot shows a web browser window with the address bar displaying 'localhost:56013/Negocios/CientesporNombre'. The page has a dark navigation bar with links: 'Nombre de aplicación', 'Inicio', 'Acerca de', and 'Contacto'. Below the navigation bar, the main heading is 'Listado de Clientes por Nombre'. There is a search form with the placeholder text 'Ingrese el Nombre del cliente' and a 'Consulta' button. The search input contains the letter 'A'. Below the search form is a table with five columns: 'Codigo Cliente', 'Nombre del Cliente', 'Direccion', 'Pais', and 'Telefono'. The table contains four rows of data.

Codigo Cliente	Nombre del Cliente	Direccion	Pais	Telefono
ALFKI	Alfreds Futterkista	Obere Str. 57	002	030-0074321
ANATR	Ana Trujillo Emparedados y helados	Avda. de la Constitucion 2222	005	(5) 555-4729
ANTON	Antonio Moreno Taquería	Mataderos 2312	007	(5) 555-3932
AROUT	Around the Horn	120 Hanover Sq	004	(71) 555-7788

## Sesión 02: Consulta de Datos con parámetros

Diseña una página que permita listar los registros de la tabla tb\_pedidoscabe (idpedido, FechaPedido, NombreCia (tb\_clientes), direccionDestinatario de la base de datos Negocios2018, filtrando por un determinado año del campo FechaPedido.

### Creando el procedimiento almacenado

En el administrador del SQL Server, defina el procedimiento almacenado usp\_pedido\_Año, tal como se muestra

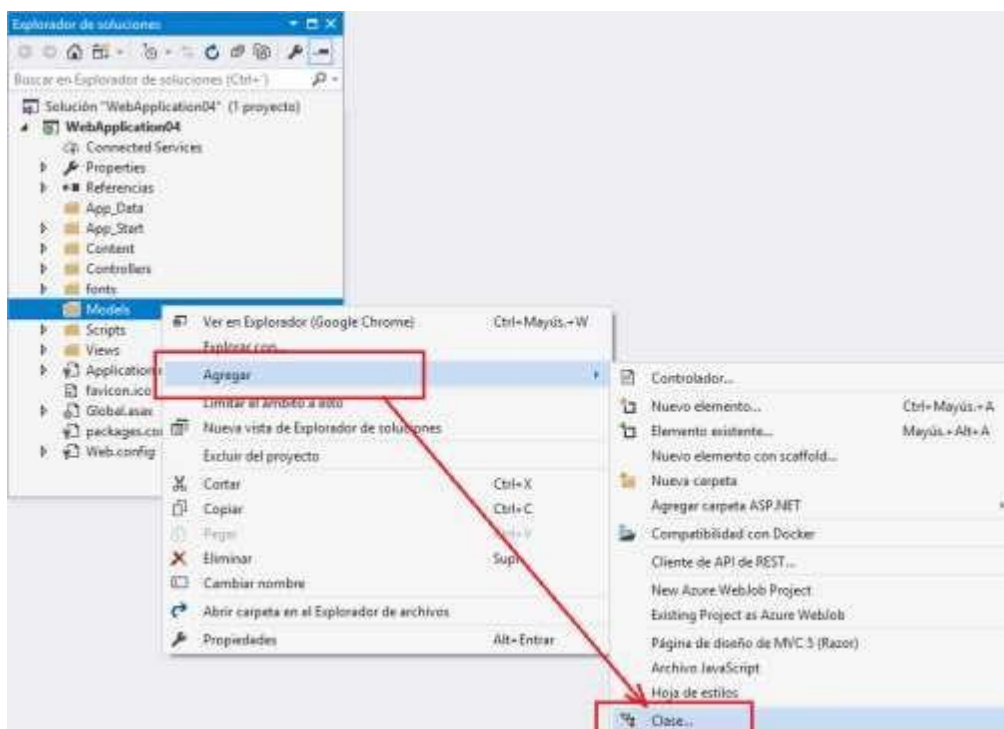


```
SQLQuery1.sql - (\\...SHIBA\\Usuario (52))* x
Use Negocios2018
go

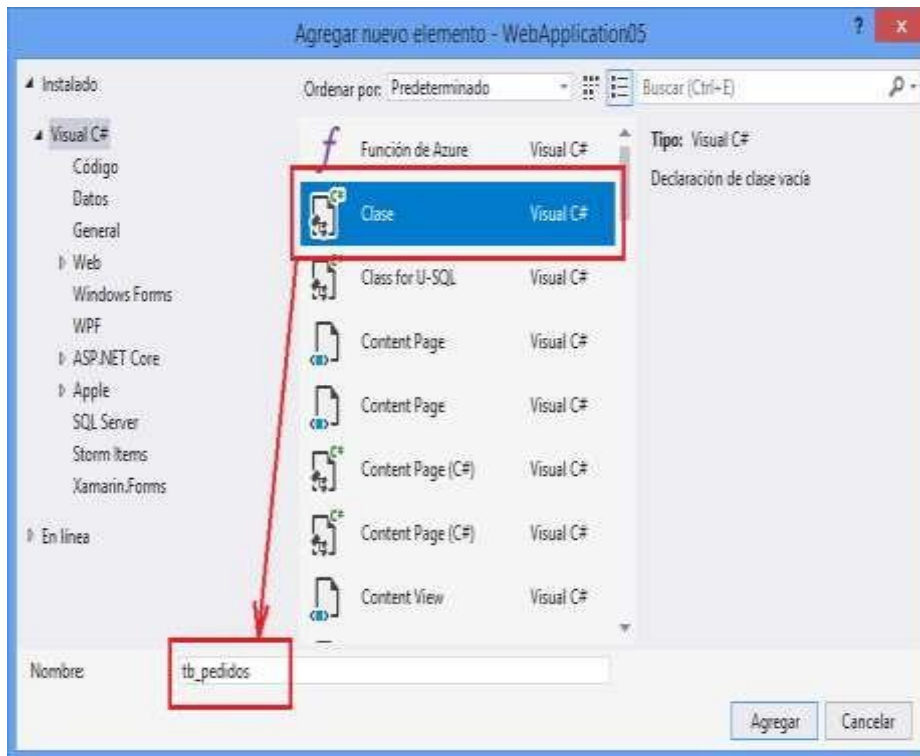
Create proc usp_pedido_Año
@y int
As
Select IdPedido, FechaPedido, NombreCia, DireccionDestinatario
from tb_pedidoscabe p inner join tb_clientes c
on p.IdCliente=c.IdCliente
Where Year(FechaPedido)=@y
go
```

### Creando la Clase del Modelo

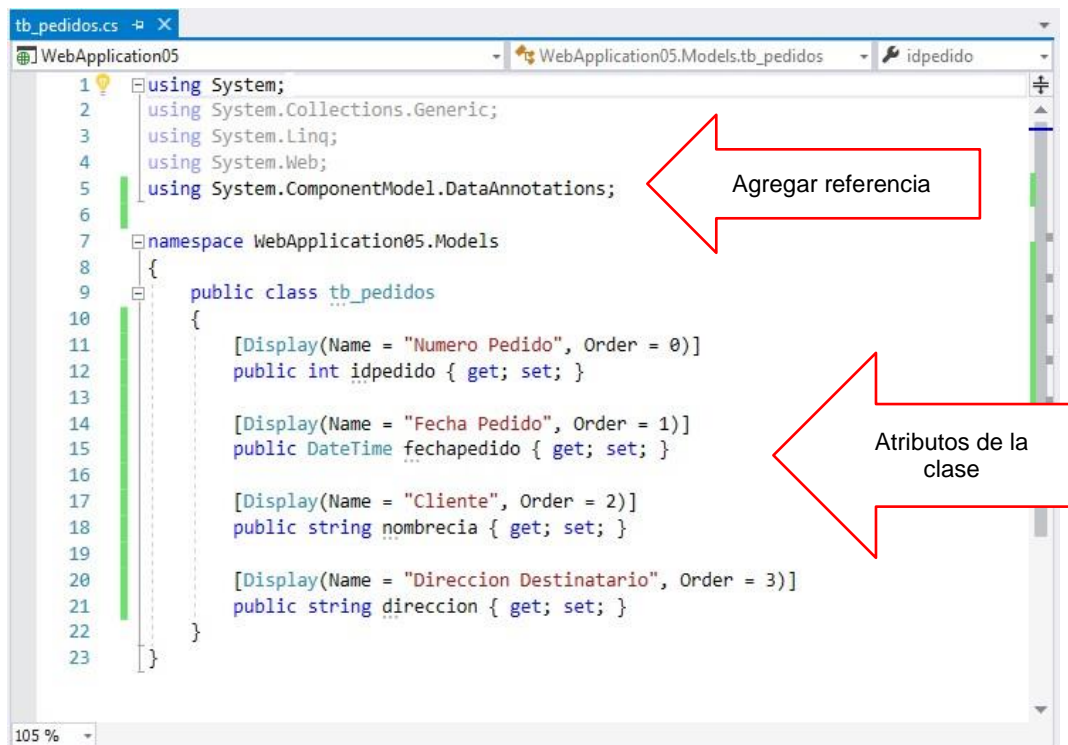
Para efectuar y realizar la consulta de los datos de la tabla tb\_pedidos, agregamos, en la carpeta Models, una clase, tal como se muestra.



Asigne el nombre de la clase, la cual se llamará tb\_pedidos, tal como se muestra. Presiona el botón AGREGAR



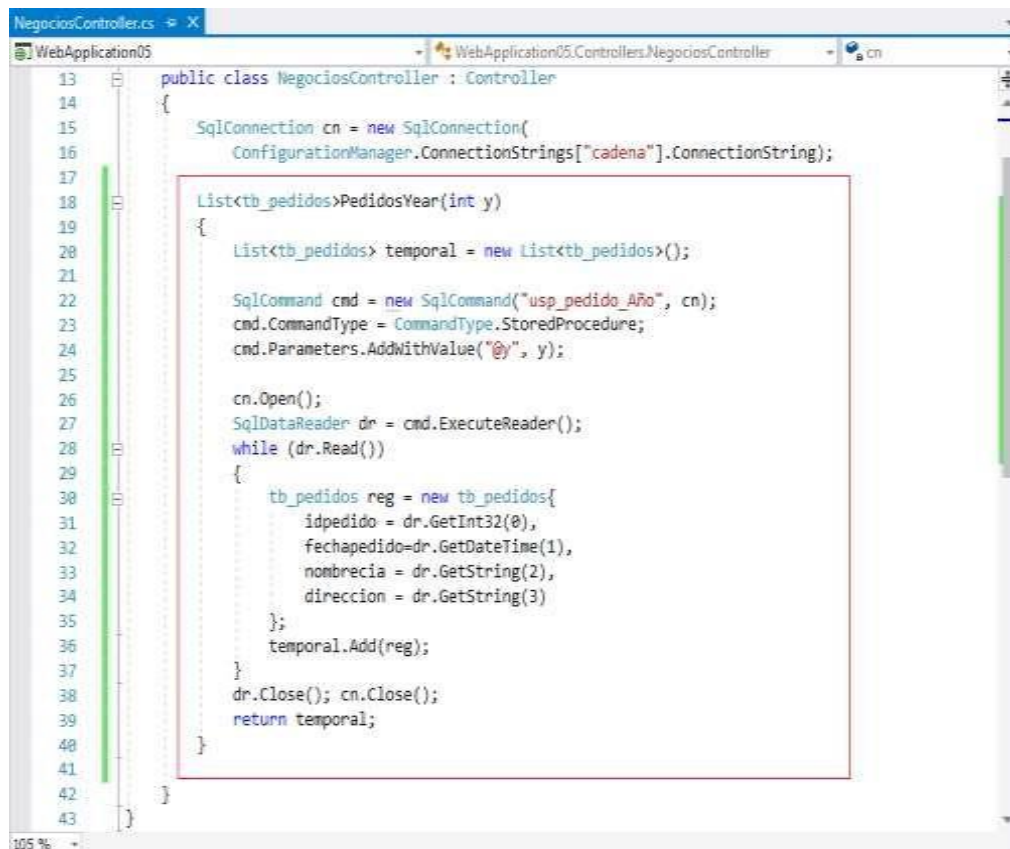
Para iniciar, agregamos la referencia DataAnnotations, luego defina los atributos de la clase, visualizando el texto y orden de cada uno de ellos [Display (Name="texto", Order="n")], tal como se muestra.



## Trabajando con el ActionResult del Controlador

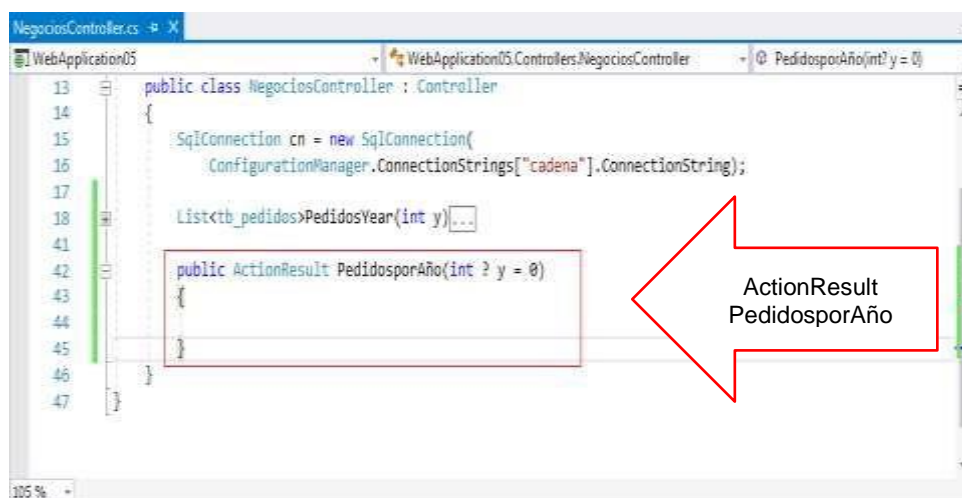
En el Controlador **NegociosController**, defina la lista llamada **PedidosYear**:

1. Defina el parámetro y (int y)
2. Ejecuta el procedimiento almacenado usp\_pedido\_Year a través del SqlCommand, pasando el valor de su parámetro @nombre.
3. El resultado se almacena en el SqlDataReader.
2. Para guardar los resultados en el temporal se realiza a través de un bucle (mientras se pueda leer: dr.Read()), vamos almacenando cada registro en el temporal.
3. Cerrar los objetos y enviar el objeto lista llamada temporal



```
13 public class NegociosController : Controller
14 {
15     SqlConnection cn = new SqlConnection(
16         ConfigurationManager.ConnectionStrings["cadena"].ConnectionString);
17
18     List<tb_pedidos> PedidosYear(int y)
19     {
20         List<tb_pedidos> temporal = new List<tb_pedidos>();
21
22         SqlCommand cmd = new SqlCommand("usp_pedido_Año", cn);
23         cmd.CommandType = CommandType.StoredProcedure;
24         cmd.Parameters.AddWithValue("@y", y);
25
26         cn.Open();
27         SqlDataReader dr = cmd.ExecuteReader();
28         while (dr.Read())
29         {
30             tb_pedidos reg = new tb_pedidos{
31                 idpedido = dr.GetInt32(0),
32                 fechapedido=dr.GetDateTime(1),
33                 nombrecia = dr.GetString(2),
34                 direccion = dr.GetString(3)
35             };
36             temporal.Add(reg);
37         }
38         dr.Close(); cn.Close();
39         return temporal;
40     }
41 }
42
43 }
```

A continuación, defina el **ActionResult PedidosporAño**, tal como se muestra



```
13 public class NegociosController : Controller
14 {
15     SqlConnection cn = new SqlConnection(
16         ConfigurationManager.ConnectionStrings["cadena"].ConnectionString);
17
18     List<tb_pedidos> PedidosYear(int y)...
```

```
41
42     public ActionResult PedidosporAño(int? y = 0)
43     {
44
45     }
46
47 }
```

ActionResult  
PedidosporAño



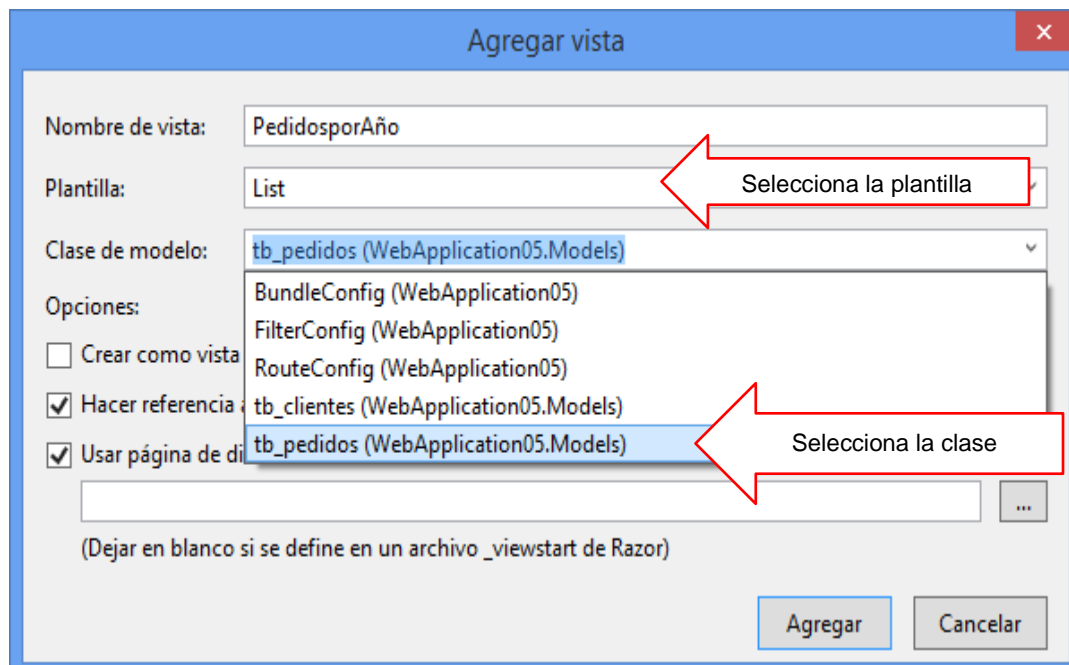
Defina en el ActionResult el **ViewBag.y** donde almaceno el valor y; retornando a la Vista el resultado del método PedidosYear filtrando con el valor del parámetro.



## Agregando la Vista.

En el ActionResult, agrega la vista.

En dicha ventana, selecciona la **plantilla**, la cual será **List**; y la **clase de modelo** la cual es **tb\_pedidos**, tal como se muestra





Vista generada por la plantilla **List**, tal como se muestra

```
1 @model IEnumerable<WebApplication05.Models.tb_pedidos>
2
3 @{ ViewBag.Title = "PedidosporAño"; }
4
5 <h2>Listado de Pedidos por Año</h2>
6
7 <p>@Html.ActionLink("Create New", "Create")</p>
8 <table class="table">
9     <tr>
10         <th>@Html.DisplayNameFor(model => model.idpedido)</th>
11         <th>@Html.DisplayNameFor(model => model.fechapedido)</th>
12         <th>@Html.DisplayNameFor(model => model.nombrecia)</th>
13         <th>@Html.DisplayNameFor(model => model.direccion)</th>
14     </tr>
15
16     @foreach (var item in Model) {
17     <tr>
18         <td>@Html.DisplayFor(modelItem => item.idpedido)</td>
19         <td>@Html.DisplayFor(modelItem => item.fechapedido)</td>
20         <td>@Html.DisplayFor(modelItem => item.nombrecia)</td>
21         <td>@Html.DisplayFor(modelItem => item.direccion)</td>
22         <td>
23             @Html.ActionLink("Edit", "Edit", new { /* id=item.PrimaryKey */ }) |
24             @Html.ActionLink("Details", "Details", new { /* id=item.PrimaryKey */ }) |
25             @Html.ActionLink("Delete", "Delete", new { /* id=item.PrimaryKey */ })
26         </td>
27     </tr>
28 }
29 </table>
```

Recibe la lista de pedidos

Cabecera de la lista

Registros que se visualizaran en la Vista

Modifica la Vista, agregando un formulario, tal como se muestra

```
1 @model IEnumerable<WebApplication05.Models.tb_pedidos>
2
3 @{ ViewBag.Title = "PedidosporAño"; }
4
5 <h2>Listado de Pedidos por Año</h2>
6
7 @using (Html.BeginForm())
8 {
9     <span>Ingrese el Año del Pedido</span>
10     <input type="text" name="y" value="@ViewBag.y" />
11     <input type="submit" value="Consulta" />
12 }
13 <table class="table">
14     <tr>
15         <th>@Html.DisplayNameFor(model => model.idpedido)</th>
16         <th>@Html.DisplayNameFor(model => model.fechapedido)</th>
17         <th>@Html.DisplayNameFor(model => model.nombrecia)</th>
18         <th>@Html.DisplayNameFor(model => model.direccion)</th>
19     </tr>
20
21     @foreach (var item in Model) {
22     <tr>
23         <td>@Html.DisplayFor(modelItem => item.idpedido)</td>
24         <td>@Html.DisplayFor(modelItem => item.fechapedido)</td>
25         <td>@Html.DisplayFor(modelItem => item.nombrecia)</td>
26         <td>@Html.DisplayFor(modelItem => item.direccion)</td>
27     </tr>
28 }
29 </table>
```

Formulario donde envío el valor del año al ActionResult

Ejecutamos la Vista, ingresa el año en el campo text (y), al presionar el botón Consulta (submit) visualizamos los registros de la tabla tb\_pedidos

The screenshot shows a web browser window with two tabs: 'ClientesporNombre - Mi aplic...' and 'PedidosporAño - Mi aplic...'. The address bar shows 'localhost:56013/ Negocios/PedidosporAño'. The page has a dark navigation bar with links: 'Nombre de aplicación', 'Inicio', 'Acerca de', and 'Contacto'. The main content area is titled 'Listado de Pedidos por Año'. Below the title, there is a form with a text input 'Ingrese el Año del Pedido' containing '2014' and a 'Consulta' button. Below the form is a table with four columns: 'Numero Pedido', 'Fecha Pedido', 'Cliente', and 'Direccion Destinatario'. The table contains five rows of data. At the bottom left, there is a copyright notice: '© 2018 - Mi aplicación ASP.NET'.

Numero Pedido	Fecha Pedido	Cliente	Direccion Destinatario
11074	16/01/2014 12:00:00 a.m.	Simons bistro	Vinblast 34
11075	16/03/2014 12:00:00 a.m.	Richter Supermarkt	Starnweg 5
11076	02/04/2014 12:00:00 a.m.	Bon app	12, rue des Bouchers
11077	19/04/2014 12:00:00 a.m.	Rattlesnake Canyon Grocery	2617 Milton Dr.
11078	26/04/2014 12:00:00 a.m.	Bon app	2617 Milton Dr.