MODEL BINDING

MODEL BINDING (ASP.NET MVC)

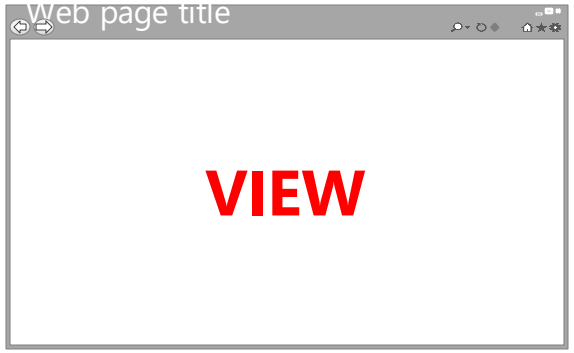
Model Binding (vinculación del modelo) es el mecanismo por el que MVC permite enviar un objeto a la View desde un controlador, y viceversa.

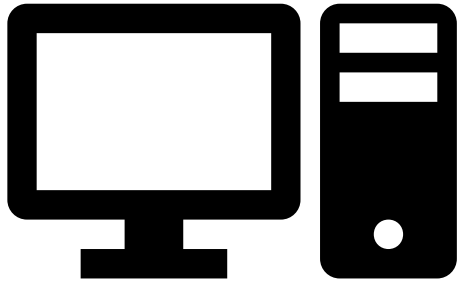
MVC realiza un mapeo de los valores de la solicitud HTTP convirtiéndolos en un objeto con esos atributos.

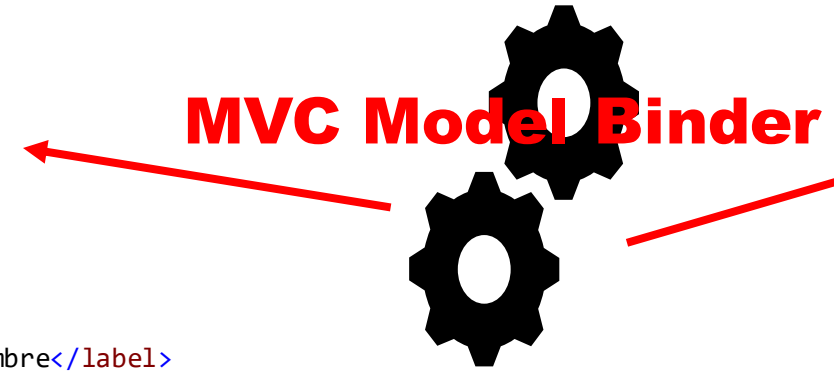
MVC realiza un mapeo de los atributos de un objeto a los elementos editables o seleccionables de un formulario HTML .

El mapeo se realiza utilizando la propiedad ***name*** de las etiquetas HTML y los nombres de los parámetros de un método, o los atributos de un objeto.

Ciclo del model binding

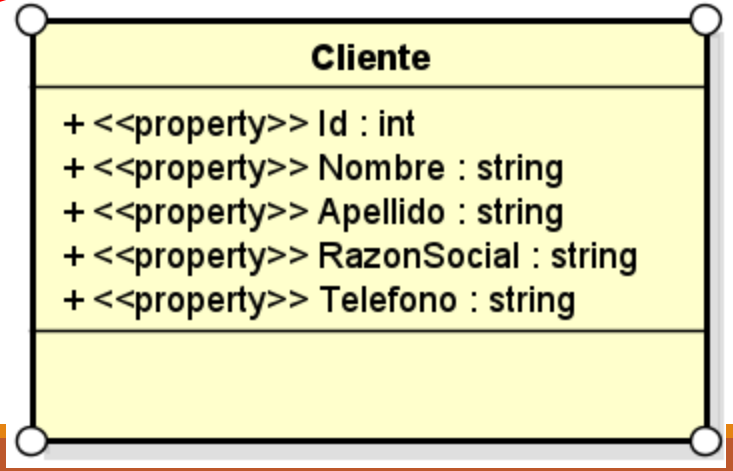







```
@model DominioDistribuidora.EntidadesNegocio.Cliente
<form action="clientes/create" method="post">
    <div class="form-horizontal">
        <h4>Ingresar un cliente</h4>

        <div class="form-group">
            <label class = "control-label col-md-2">Nombre</label>
            <div class="col-md-10">
                <input type="text" name="nombre"  class = "form-control" />
                        ..........
</form>
```

```
// POST: Clientes/Create
[HttpPost]
public ActionResult Create(Cliente unCliente)
{
        .....
}
```



Solicitud (Request) HTTP

El navegador envía la solicitud HTTP con los elementos de formulario que tengan seteada la propiedad name (ejemplo con POST):

nombre=Ana
apellido=García
RazonSocial=abc
Telefono=091919191

El ruteador de MVC detecta que debe invocar el método Create

El model binder de MVC detecta que el método Create recibe un objeto Cliente. Lo instancia, llena sus atributos (properties públicas) con los valores correspondientes a los elementos del mismo nombre (que tenían la misma propiedad ***name)*** y lo pasa como parámetro.

Respuesta (Response) HTTP

El método de acción retorna la view pasándole un cliente con sus atributos en un estado determinado (con determinados valores):

return View (unCliente) ;

MVC crea una instancia de la clase View, y le setea como property Model el objeto pasado como parámetro a su constructor.

MVC genera la respuesta HTTP llena el atributo HTML ***VALUE*** con los valores de la property del objeto Model - que es una referencia al objeto Cliente, en nuestro ejemplo - siempre y cuando el atributo Name del input HTML coincida con el nombre de la property pública.

El navegador recibe la respuesta y despliega los valores en los atributos value de las etiquetas HTML correlativas por name.

Model

Es una property del objeto View

Razor (el motor de vistas) la puede utilizar

Para utilizarla en la vista usamos @Model

Para que la vista sea fuertemente tipada (podemos acceder a los atributos públicos, y eventualmente a los métodos de Model).

Debemos declarar en la vista el tipo del modelo:

```
@model EntidadesNegocio.Cliente
@model IEnumerable< EntidadesNegocio.Cliente>
```

Model puede ser cualquier tipo de objeto (incluso un Int32, Decimal, etc.)

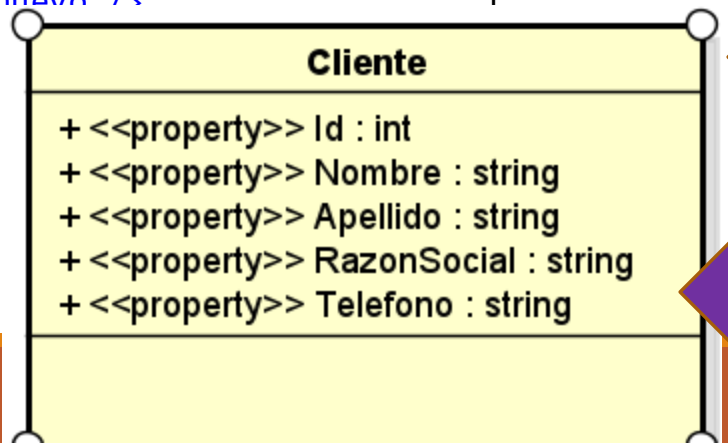
El Model Binding en código

EN LA VIEW:

```
@model DominioDistribuidora.EntidadesNegocio.Cliente

<form action="/clientes/create" method="post">

    Nombre: <input type="text" name="nombre"/>

    Apellido: <input type="text" name="apellido" />

    Razón social: <input type="text" name="RazonSocial" />

   Teléfono: <input type="text" name="telefono" />

   <input type="submit" value="Crear nuevo" />

</form>
```

EN EL CONTROLADOR CLIENTES:

```
[HttpPost]
 public ActionResult Create(Cliente unCliente)
{
     if (ModelState.IsValid) {
          if (repo.Add(unCliente))
               return RedirectToAction("Index");
          }
     return View(unCliente);
}
```



Debe tener properties públicas

Debe tener constructor sin parámetros

Los nombres de las properties deben coincidir con el name de los elementos HTML correspondientes