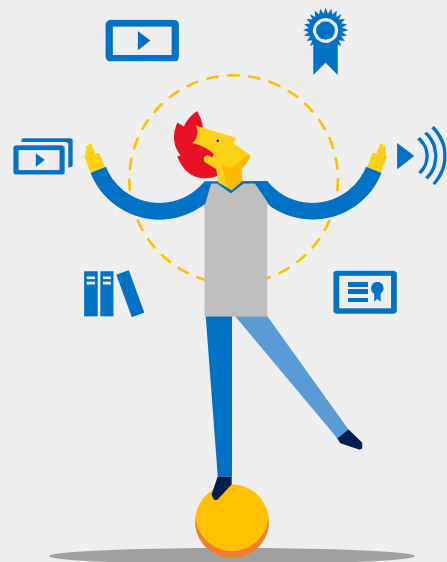


Microsoft Virtual Academy

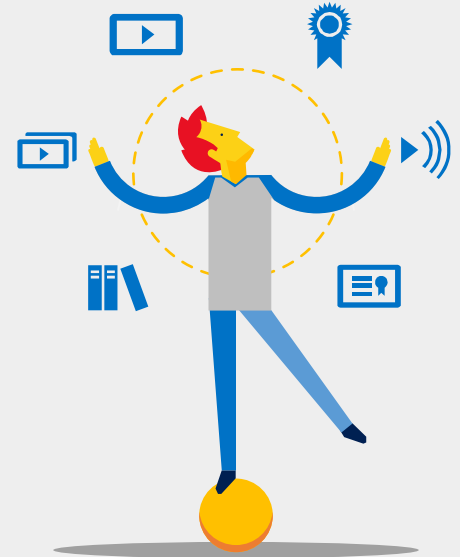
El Servidor Web Conceptos Avanzados

Uso del Framework Microsoft .NET
Aprendiendo a Programar. Capitulo 7. Tutorial



Objetivo

- Utilizaremos funciones del framework de .NET para enviar correos
- Ejercitaremos sobre diferentes características del mismo

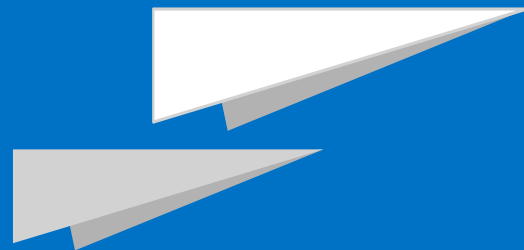


Conceptos previos para enviar un correo

Antes que nada debemos aclarar una obviedad.

Podemos enviar un correo desde nuestra aplicación, siempre y cuando tengamos acceso a un Servidor De Correo.

Por lo general tenemos acceso a uno, el que usa nuestra cuenta de correo



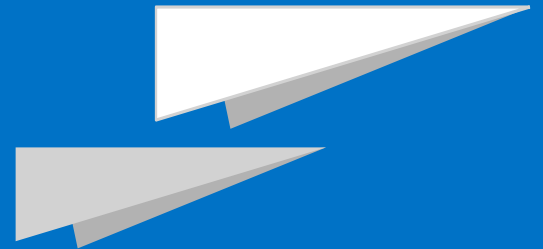
Conceptos previos para enviar un correo

Para identificar el servidor, debemos saber el nombre del SMTP

Coloca en el buscador web las siguientes claves

« SMTP Live Hotmail Configuration »

Si tienes otro correo que no sea Live Hotmail reemplaza el nombre que corresponda



Lo que estamos buscando

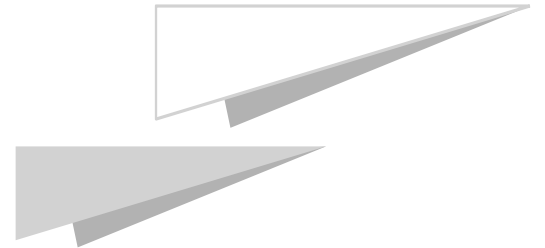
La configuración que estamos buscando es:

SMTP address server

SMTP username

SMTP port

SMTP TLS/SSL



Para live hotmail la configuración es

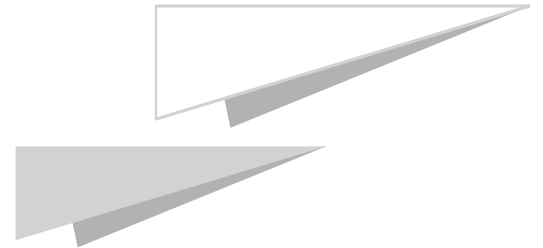
Si tienes una cuenta live, estos serían los valores:

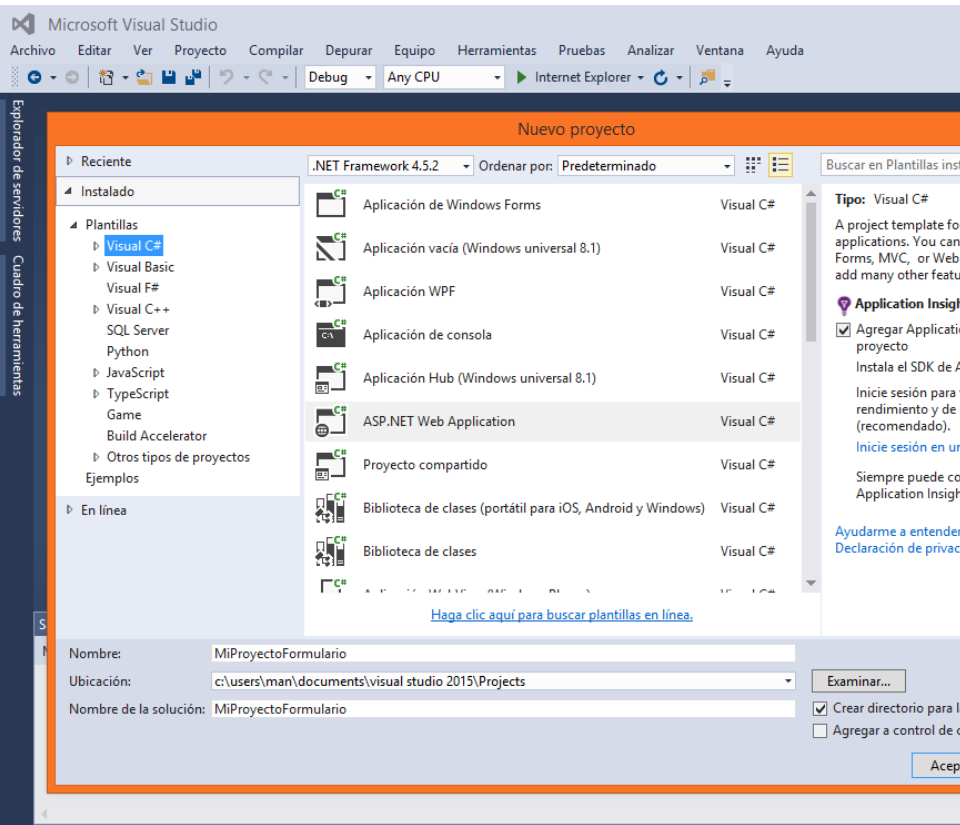
SMTP address server: smtp.live.com

SMTP username: tunombre@live.com

SMTP port: 587

SMTP TLS/SSL: si





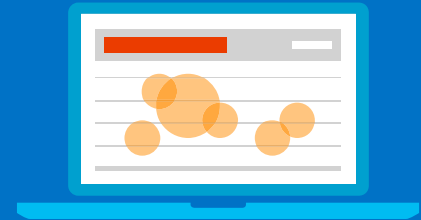
Creo un nuevo proyecto de tipo Visual C# - Aplicación web ASP.NET y utiliza la plantilla simple MVC

Crea los siguientes secciones

Crea una carpeta dentro de views llamada Formularios

Crea una vista dentro de Formularios llamado Contacto

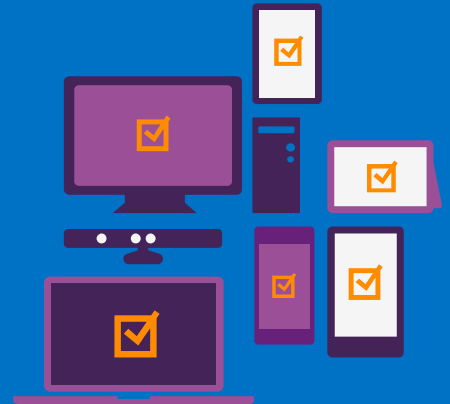
Edita Contacto.cshtml y crea elementos html como el formulario visto en el capítulo III, con nombre, apellido y dirección de email



Continua con

Crea un controlador llamado Formas
Dentro de el, crea un punto de entrada
llamado Contacto y retorna la vista
`/Views/Formularios/Contacto.cshtml`
Prueba la aplicación y asegúrate que muestre
el formulario.

Recuerda que el punto de entrada es
`/Formas/Contacto`



Editando formulario .cshtml

```
<form id="miformulario" method="post" action="/Forma/GuardarContacto">  
  <fieldset>  
    <label for="apellido">Apellido</label>  
    <input type="text" id="apellido" name="apellido" />  
  </fieldset>  
</fieldset>
```

Editando el formulario llamado contacto.cshtml y escribiendo en la sección action lo siguiente

Continúa con

Vuelve al controlador denominado Formas y crea otro punto de entrada, esta vez, para recibir los datos del formulario.

Denomínalo GuardarContacto.

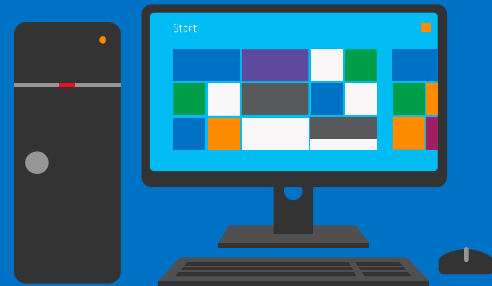
Hasta aquí el ejercicio es el mismo que en el capítulo anterior



Vamos a enviar un correo

Para esto, escribimos la función de envío de correo en el método GuardarContacto

Utilizamos la función de envío de mail provista en el framework de .NET, solo tenemos que tener presente los parámetros que buscamos con anterioridad



FormasController.cs

```
namespace MiProyectoFormulario.Controllers
{
    public class FormasController : Controller
    {
        // GET: Formas
        public ActionResult Contacto()
        {
            return View("/Views/Formularios/Contacto.cshtml");
        }

        public ActionResult GuardarContacto(string nombre, string apellido, string email, string mensaje)
        {
            string nombreCompleto = nombre.ToUpper() + " " + apellido.ToUpper();

            // Armo el mensaje
            System.Net.Mail.MailMessage mailMessage = new System.Net.Mail.MailMessage();
            mailMessage.To.Add(email);
            mailMessage.Subject = "Tu solicitud ha sido procesada";
            mailMessage.Body = "Hemos recibido tu consulta. Responderemos a la brevedad.";
            mailMessage.IsBodyHtml = true;

            // me conecto al servidor smtp
            System.Net.Mail.SmtpClient smtp = new System.Net.Mail.SmtpClient();
            smtp.Host = "smtp.live.com";
            smtp.Port = 587;
            smtp.Credentials = new System.Net.NetworkCredential("miusuario@live.com", "micontraseña@live");
            smtp.EnableSsl = true;

            // envío el mensaje
            smtp.Send(mailMessage);

            ViewBag.nombreentero = nombreCompleto;
            ViewBag.texto = "Te responderemos a la brevedad a " + email;
            return View("/Views/Formularios/ContactoRespuesta.cshtml");
        }
    }
}
```

Explorador de soluciones

Buscar en Explorador de soluciones (Ctrl+)

Solución 'MiProyectoFormulario' (1 proyecto)

MiProyectoFormulario

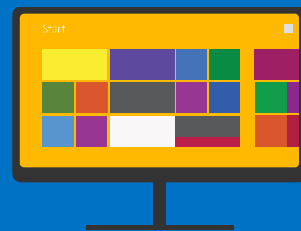
- Properties
- Referencias
- App_Data
- App_Start
- Areas
- Content
- Controllers
- fonts
- Models
- Providers
- Results
- Scripts
- Views
- ApplicationInsights.config
- favicon.ico
- Global.asax
- packages.config
- Project_Readme.html
- Startup.cs
- Web.config

Explorador de soluciones Team Explorer Vista de clases

Por ultimo, y como dice el controlador

Crea una vista, en la carpeta Formularios
Que se llame ContactoRespuesta
Informa en esta vista que haz enviado un
correo.

Podrías en el método mensaje.To.Add,
agregar a más de un destinatario al mismo
tiempo del mismo correo.



ctoFormulario - Microsoft Visual Studio

r Ver Proyecto Compilar

Depurar Equipo Herramientas Pruebas Analizar Ventana Ayuda

trroller.cs ✕
ctoFormulario

namespace MiProyectoFormul

{

public class FormasCon

{

// GET: Formas

public ActionResult

{

return View("/

}

public ActionResult

{

string nombreC

// Armo el men

System.Net.Mai

mailMessage.To

mailMessage.Su

mailMessage.Body = "Hemos recibido tu consulta. Responderemos a la brevedad.";

mailMessage.IsBodyHtml = true;

- Ventanas
- Gráficos
- Mostrar herramientas de diagnóstico Ctrl+Alt+F2
- Iniciar depuración F5**
- Iniciar sin depurar Ctrl+F5
- Iniciar herramientas de diagnóstico sin depuración... Alt+F2
- Asociar al proceso... Ctrl+Alt+P
- Otros destinos de depuración
- Generador de perfiles
- Paso a paso por instrucciones F11
- Paso a paso por procedimientos F10
- Alternar puntos de interrupción F9
- Nuevo punto de interrupción
- Eliminar todos los puntos de interrupción Ctrl+Mayús.+F9
- Opciones...
- Propiedades de MiProyectoFormulario...

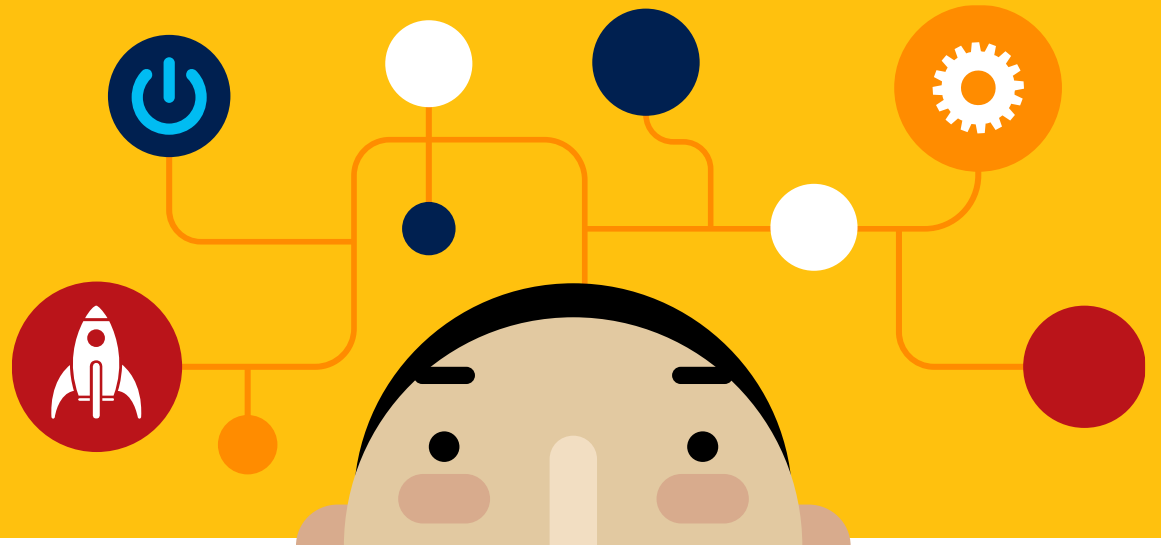
Ejecuta la aplicación

Prueba el circuito.
Inicia desde
/Forma/Contacto
Y verifica como funciona
cuando haces clic en Enviar

Continua ejercitando

Agrega validadores con JQuery Validator al formulario, para evitar que envíes datos vacíos.

El más importante, la dirección de mail no debería estar vacía



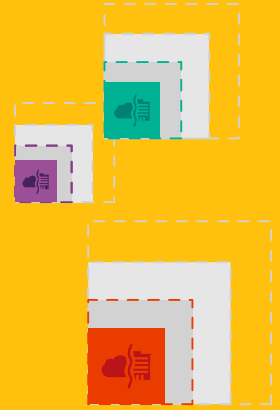
Otro ejercicio

En el mismo proyecto, crea una vista en la carpeta formularios que reciba dos números.

La estructura es similar al formulario anterior, pero solo con dos campos.

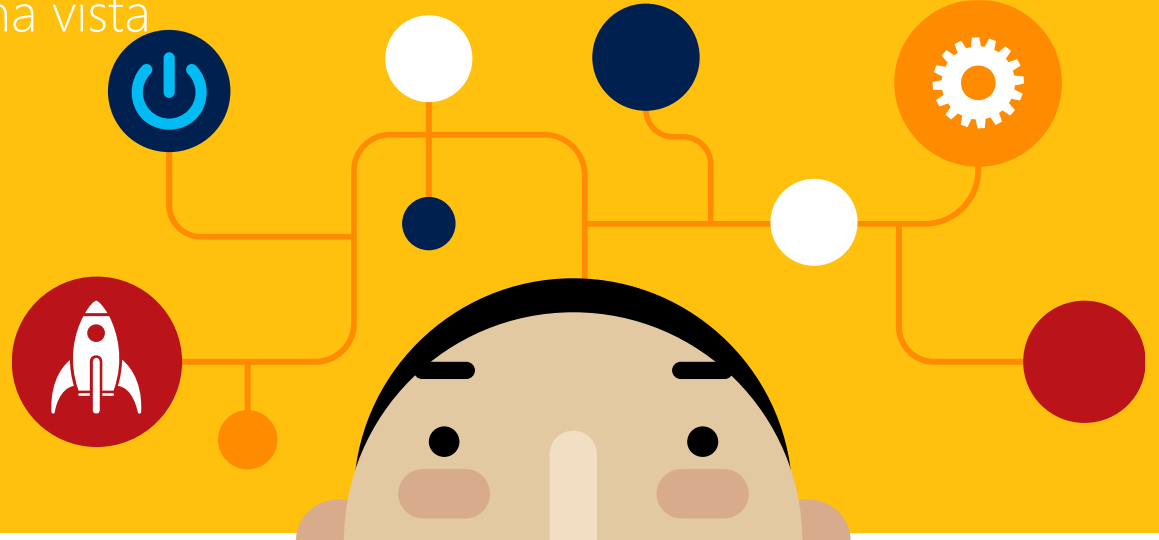
Incluye el botón enviar

En el valor action del formulario coloca Formas/Raices, este método calculará la raíz cuadrada



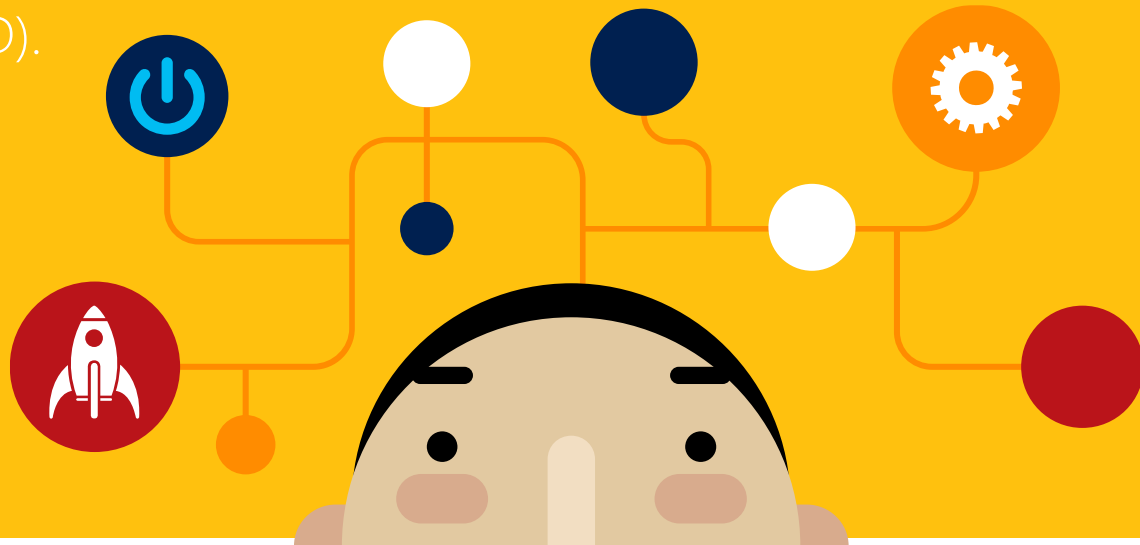
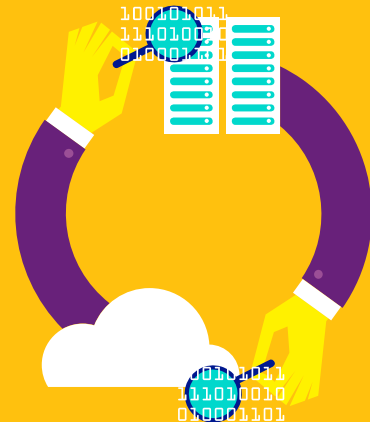
Otro ejercicio

Para esto utilizaremos otra función del Framework .NET, la clase MATH, que posee un sinfín de operaciones matemáticas
En el caso de la raíz cuadrada el método es `Math.Sqrt(tu numero)`
Devuelve el resultado en una vista



Sigue explorando

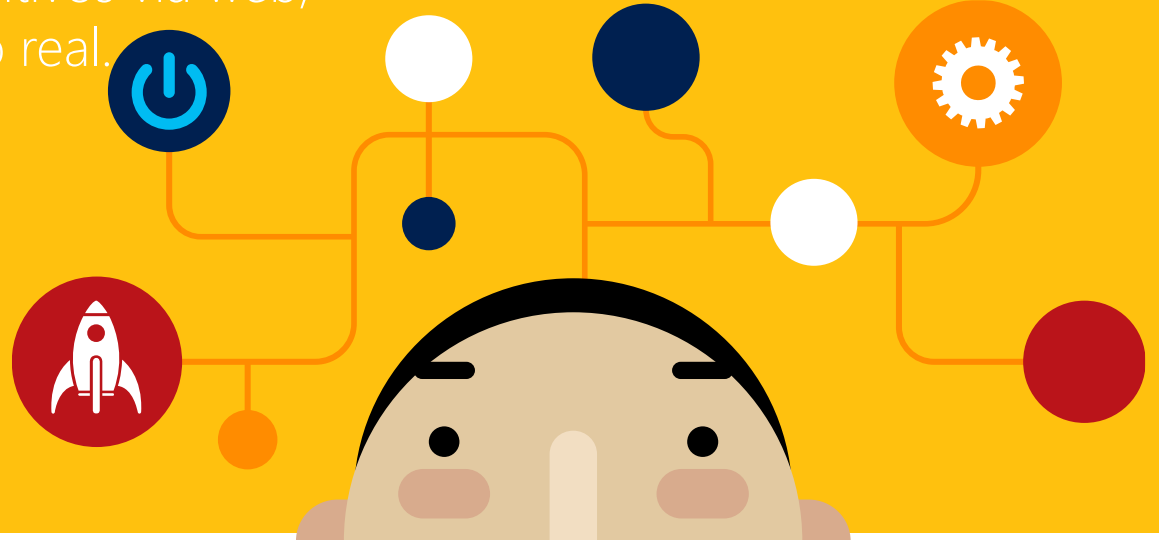
En la medida que puedas explora con más detalles el framework .NET
La clase `System.Math` te permite efectuar cálculos matemáticos avanzados de geometría (muy útil para 3D).



Sigue explorando

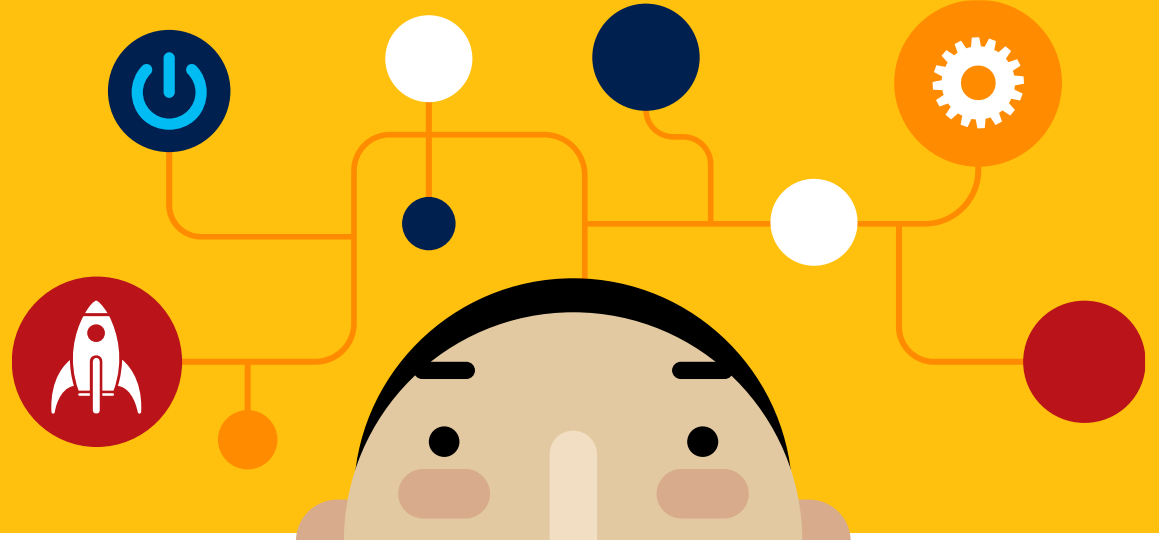
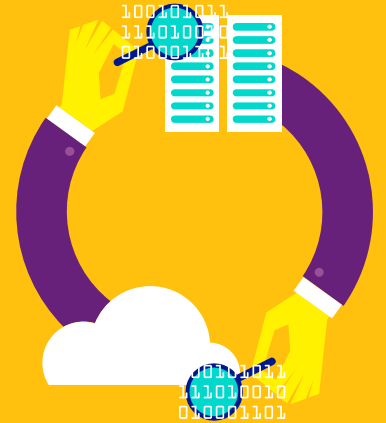
La clase `System.Text` te permite manipular textos

La clase `System.Web.Sockets` te permite comunicaciones en dos vías, ideal para transferir datos entre dispositivos vía web, ideal para juegos en tiempo real.



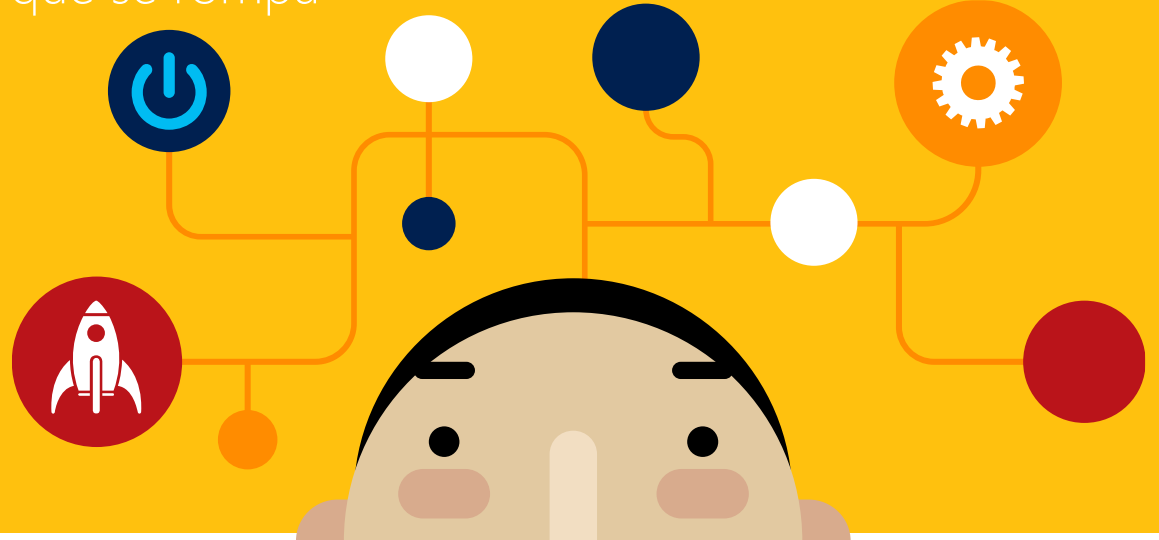
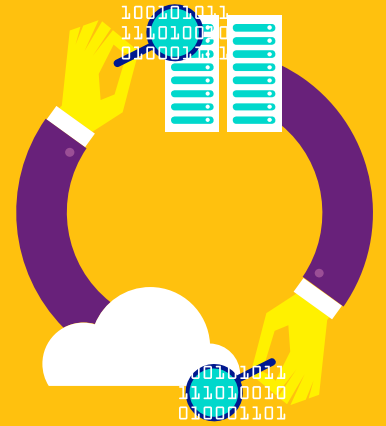
Sigue explorando

Si quieres usar los puertos serie o usb lo puedes hacer desde la clase System.IO (aunque no es un tipo de aplicación Web), con lo cual podrías controlar maquinaria y motores (robótica)



Sigue explorando

Al punto que te encuentras, puedes utilizar el controlador para codificar lo que precisas e ir descubriendo características del lenguaje. Anímate a probar y no te preocupes por equivocarte, no hay riesgo que se rompa nada.





© 2016 Microsoft Corporation. All rights reserved. The text in this document is available under the Creative Commons Attribution 3.0 License, additional terms may apply. All other content contained in this document (including, without limitation, trademarks, logos, images, etc.) are not included within the Creative Commons license grant. This document does not provide you with any legal rights to any intellectual property in any Microsoft product. You may copy and use this document for your internal, reference purposes.

This document is provided "as-is." Information and views expressed in this document, including URL and other Internet Web site references, may change without notice. You bear the risk of using it. Some examples are for illustration only and are fictitious. No real association is intended or inferred. Microsoft makes no warranties, express or implied, with respect to the information provided here.