

Formularios

Para comenzar debemos definir para ello ¿Que es un formulario? los mismos son uno de los elementos básicos para capturar información, estos formularios existen de manera impresa, y de manera digital bien sea en aplicaciones de escritorio o aplicaciones en entorno web y en la actualidad también usados en Teléfonos móviles o Smartphone.

En el siguiente tutorial o guía básica de creación de formularios nos vamos a centrar en Formularios para uso en la web es decir un formulario Web, para ellos debemos utilizar un lenguaje especial llamado HTML, el cual en la actualidad marzo 2020 ha evolucionado a HTML5 trayendo consigo mejoras entre ellas validaciones, textos para cada tipo de datos entre otros que mencionaremos.

Vale la pena definir ciertos conceptos para tener claros entre ellos:

Para el diseño de Formularios

HTML: es un lenguaje de marcado que se utiliza para el desarrollo de páginas web de Internet. Se trata de la sigla que corresponden a HyperText Markup Language, es decir, Lenguaje de Marcas de Hipertexto, sólo sirve para indicar como va ordenado el contenido de una página web. Esto lo hace por medio de las marcas de hipertexto las cuales son etiquetas conocidas en inglés como tags. Cuando nos referimos a cómo va ordenado el contenido de la página, es que por medio de uso de etiquetas se define el encabezado, título y cuerpo de la página web, entre otros elementos como líneas, imágenes, textos, viñetas, tablas. HTML no permite acceder a datos de base de datos, pero permite crear los formularios elementos que permiten la captura de datos, el cual es el objetivo de esta guía.

HTML5: (HyperText Markup Language, versión 5) es la quinta revisión del lenguaje HTML. Esta nueva versión define los nuevos estándares de desarrollo web, rediseñando el código para resolver problemas y actualizándolo así a nuevas necesidades. No se limita solo a crear nuevas etiquetas o atributos, sino que incorpora muchas características nuevas y proporciona una plataforma de desarrollo de complejas aplicaciones web (mediante los APIs).

Para el proceso de Formularios

Existen una variedad de lenguajes que permiten la manipulación de los datos capturados por los formularios entre ellos podemos mencionar PHP, Java, ASP.net, C#, Ruby, Python entre otros. Para que tengamos una idea de cómo está formada la web existen **páginas estáticas** que solo muestran información, imágenes, textos, videos música entre otros y las **páginas dinámicas** las cuales permiten interactuar de una forma con los usuarios es aquí cuando entran en funcionamiento los formularios los cuales recogen los datos y son enviados a un Script realizado en un lenguaje de los mencionados anteriormente el cual permite procesar dichos datos y realiza una tarea específica como por ejemplo realizar una operación matemática, realizar una búsqueda en una base de datos.

Un ejemplo básico es el login(Formulario de Acceso o Formulario de Inicios de Sesión) del correo electrónico, el mismo cuando se ingresa a la URL solicita Usuario y Password, estos datos son introducidos luego se le da al botón de ingresar y es aquí cuando la clave y el Password son enviados a un proceso el cual se conecta con una base de datos y devuelve bien sea un mensaje de error indicando que el usuario No está registrado, el Password es invalido o simplemente son correctos los datos y es enviado a la bandeja de entrada o menú principal de la aplicación(Gestor de Correo en este caso).

En este guía paso a paso vamos hacer uso de PHP para realizar el proceso, y que es **PHP:** Lenguaje de programación interpretado es usado principalmente en interpretación del lado del servidor (server-side scripting), PHP es un Acrónimo recursivo que significa PHP Hypertext Pre-processor (inicialmente PHP Tools, o, Personal Home Page

Tools), el mismo es utilizado para el desarrollo de páginas web el cual tiene como objetivo hacer las páginas dinámicas las cuales permitan interactuar con bases de datos, directorios de discos, búsquedas o consultas, estas páginas dinámicas en la actualidad las podemos encontrar en los Bancos, páginas de e-Comer, páginas de consultas de datos por ejemplo: Consulta de Rif Seniat, consulta de Datos personales CNE, Consulta de cuenta de SSO, consulta de un status de un proceso Pasaporte entre miles de páginas web especializadas.

Que necesitamos para realizar estos formularios:

Servidor Web: aunque para trabajar con formularios Web no es necesario un servidor web para el caso de esta guía vamos a necesitar uno, ya que vamos a realizar el proceso el cual va a estar en PHP y necesitamos un compilador de este. En internet podemos encontrar muchos softwares que permiten crear un servidor Web para entornos de desarrollo inclusive de Producción, entre ellos XAMPP, WAMP, LAMPP, Easy Php.

Editor de Texto: necesitaremos un simple editor de texto donde podremos escribir código ya que tanto el lenguaje de HTML, HTML5 y PHP solo necesitamos editar código. Dentro de los editores son miles podemos mencionar Block de Notas, el editor de texto por defecto de Windows sin embargo no es un editor profesional para desarrolladores entonces podemos usar [Notepad++](#), [Sublime Text](#), [Visual Studio Code](#), [NetBeans](#), DreamWevear, [Atom](#), [Vim](#) entre otros de tipo profesional cuya diferencia es la facilidad de editar código incluso traen plugin que con solo hacer una combinación de teclas tenemos un formulario y unos Controles de Texto, Enumeración de líneas esto es una característica muy importante de los editores, Corrección de errores, ayudas visuales y propuestas de métodos, coloración de Sintaxis este permite distinguir entre instrucciones del lenguaje e instrucciones definidas por el usuario, reconocimiento del lenguaje de programación esta característica importante ya que permiten trabajar con múltiples lenguajes.

Algunas interrogantes antes de continuar

¿Se necesita internet para crear un formulario Web, página web o una Aplicación web?

No, con solo tener un editor de texto básico como el Notepad de Windows o cualquier editor de texto plano.

¿Se necesita acceso a Internet para tener un servidor web?

No, se pueden descargar software para la creación de servidores Locales o servidores de desarrollo los cuales se crean en cualquier PC o Laptop y una vez terminado el desarrollo o página se hace un Upload al sitio web el cual si requiriese internet para acceder.

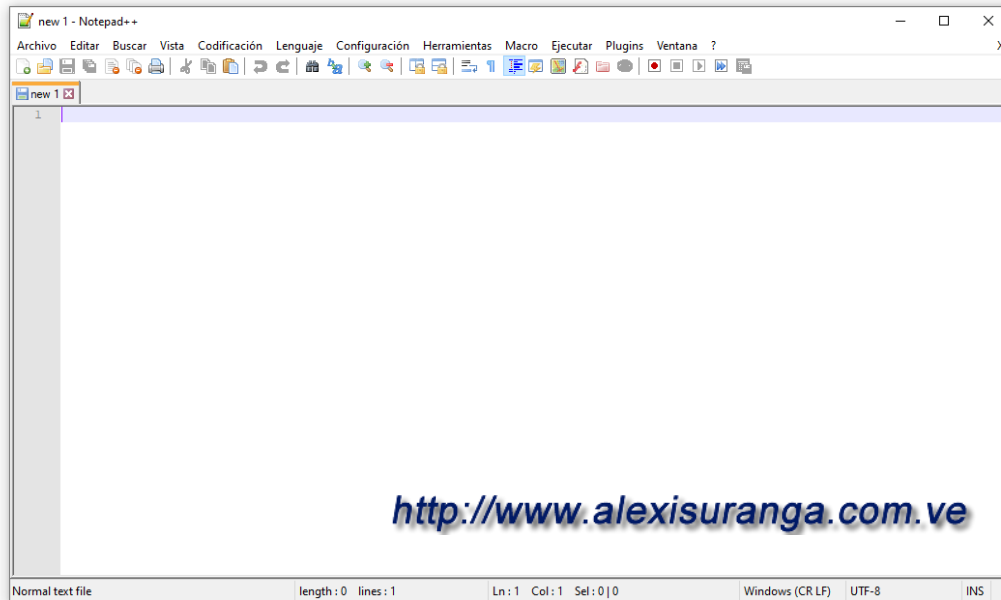
¿Los servidores Web para desarrollo son Pagos?

No, son totalmente gratuitos, se descargan libremente de internet, ahora para publicar la página o proyecto final si requiere pagar un alojamiento.

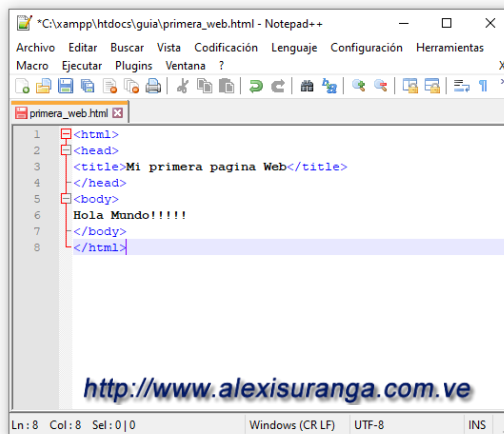
¿Los editores de desarrollo de páginas Web profesionales son Pagos?

No, la mayoría son de descarga gratuita (Notepad++, Visual Studio Code), sin embargo, hay editores pagos como DreamWevear, y otros de descarga gratuita con opción a una versión paga con otras características no disponibles en las versiones gratis.

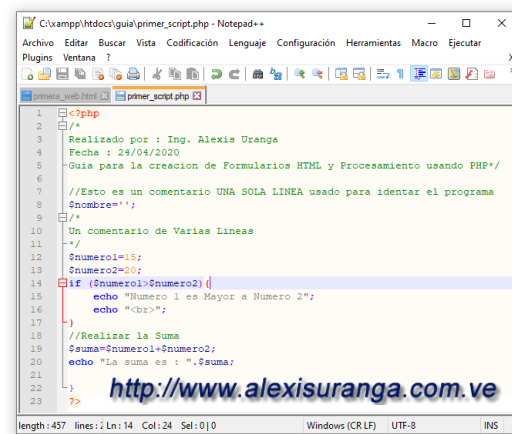
Vista previa de Notepad++



Mostrando Código de HTML

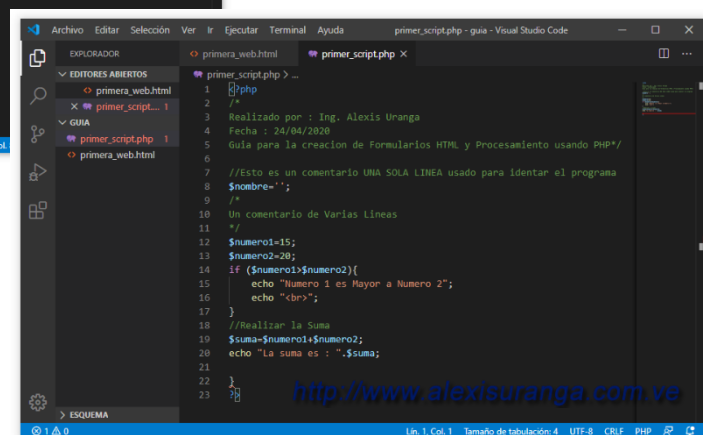
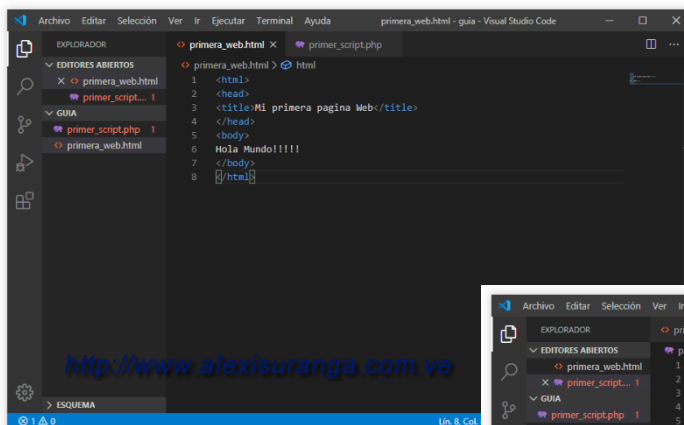
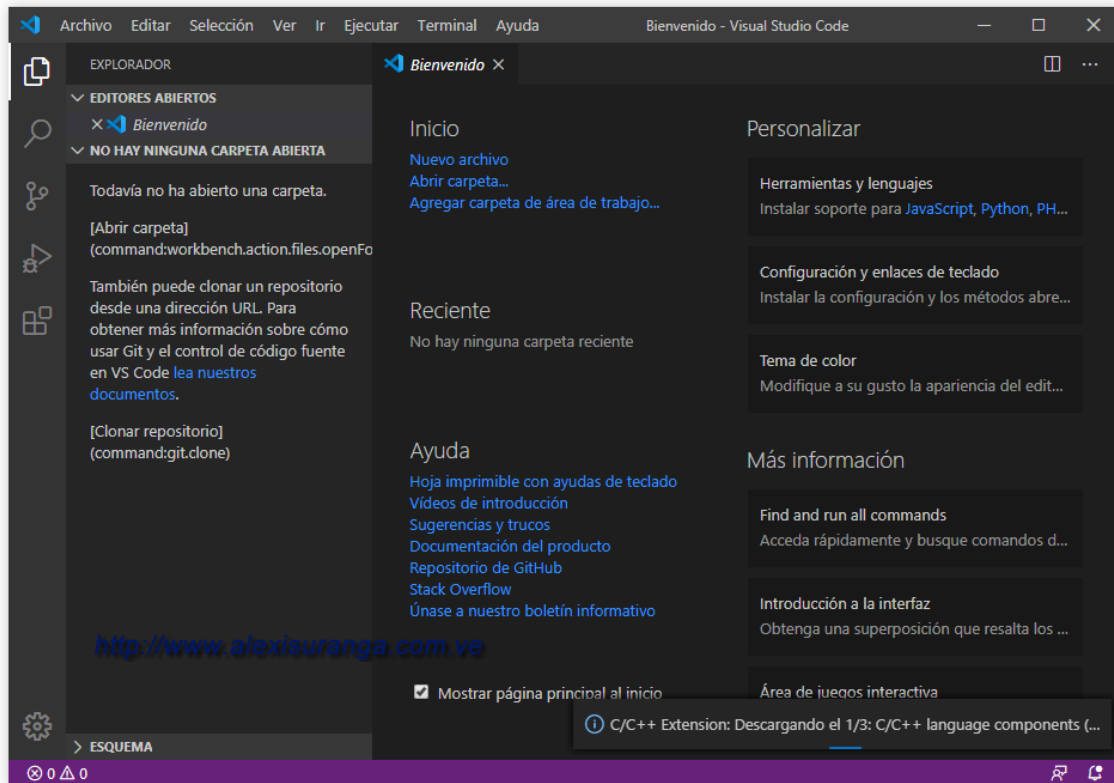


Mostrando código PHP



Nótese la enumeración de líneas y la coloración de sintaxis, los mismos ayudan al momento de corregir errores.

Vista previa de Visual Studio Code



Controles o elementos de los formularios.

Para el diseño de formularios tenemos varios controles los cuales tienen un objetivo el cual guarda estrecha relación con los tipos de datos que vaya a capturar:

Creando un formulario

El lenguaje HTML es un lenguaje que se trabaja a base de etiquetas es por ello por lo que siempre encontramos etiquetas como por ejemplo `<html>` la cual indica el inicio de código HTML y esta etiqueta debe finalizar en la última línea de la página `</html>` el carácter slash (/) indica el fin de la etiqueta.

`<head>` permite definir el inicio del encabezado donde colocamos las etiquetas meta e title, algunos scripts en JavaScript, llamados a archivos .Js, definición de css o llamado a archivos css, al igual que el HTML debe finalizar `</head>`

`<title>` permite definir el título de la página, igualmente finaliza en `</title>`

`<body>` el cuerpo de la página todo lo que se coloque en esta área se mostrara al usuario, debe finalizar antes del HTML, el mismo termina en `</body>`

`<form>` permite definir el formulario, al igual que las etiquetas anteriores debe finalizar en `</form>`

Algunas propiedades de la etiqueta `<form>`:

- **Name:** es un identificador (no necesariamente único) para los elementos de formulario.
- **Id:** es un identificador único (debería serlo siempre) para cualquier etiqueta HTML.
 - Se recomienda que los atributos Name e id sean iguales
- **Method:** permite especificar que método se va a utilizar, entre los métodos POST el cual permite enviar los datos ocultos al usuario como se dice por ahí bajo cuerda, el método GET el cual envía los datos, pero los mismos son mostrados en la URL del navegador.

Para el siguiente ejemplo se muestra como seria la URL enviada con GET:

`http://localhost/guia/proceso_form_base.php?cedula=14123456&nombre=Pedro+Perez&direccion=La+concordia&fechanac=01%2F01%2F2000&email=pedroperez35%40gmail.com&button=Enviar`

Para el siguiente ejemplo se muestra como seria la URL enviada con POST:

http://localhost/guia/proceso_form_base.php

- **Action:** permite especificar el nombre del archivo que debe recibir los datos procesar el mismo puede ser `proceso_form_base.php` si es lenguaje php, `proceso_form_base.asp` si es asp, `proceso_form_base.jsp` entre otros.

Ejemplo de la definición del formulario completo:

```
<form id="form1" name="form1" method="POST" action="proceso_form_base.php">
```

Aquí elementos que contiene el formulario.

```
</form>
```

Nota: existen otras propiedades que mas adelante se especificaran, tales como class(Para darle un toque de diseño a los formularios) eventos tales como onSubmit, onBlur(para realizar procesos via Ajax) entre otros.

Objetos contenidos en el formulario

<label>permite indicar una etiqueta para un campo de texto, la misma debe cerrar con **</label>**

<input/> esta permite crear un objeto de texto, Campo de texto, Text Box entre otros nombres que se le conoce, su función es capturar datos permiten que el usuario edite texto en ellos, a diferencia de las demás etiquetas finaliza en **/>**, entre **<input** y **/>** se colocan las propiedades.

```
<input type="text" name="cedula" id="cedula" size="12" maxlength="12" />
```

Algunos Atributos:

- **Type:** El atributo type de la etiqueta **<input>** indica el tipo de control de que se trata,
 - En HTML 4.01, los tipos de **<input>** eran: botón para enviar submit, botón para reiniciar reset, texto text, contraseña password, archivo file, imagen image y oculto hidden.
 - En HTML 5 se añadieron los tipos de **<input>**: botón button, número number, búsqueda search, teléfono tel, dirección web url, dirección de correo email, fecha date, hora time, color color y rango range.
 - En HTML 5.1 se añadieron los tipos de **<input>**: mes month, semana week y fecha y hora local datetime-local.
- **Size:** tamaño visual.
- **Maxlength:** tamaño en caracteres.

<input/> de tipo submit, esta etiqueta permite crear un botón de envió, por lo general los botones básicos de un formulario son 2, Submit cuya función es enviar el formulario hacia el archivo de proceso, Reset el cual permite resetear o limpiar los textos u otros objetos del formulario funciones preestablecidas solo con colocar Submit para enviar o Reset para cancelar. A continuación, un ejemplo de ambos botones:

```
<input name="button" type="submit" id="buttonenviar" value="Enviar" />
```

```
<input name="button" type="reset" id="buttoncancelar" value="Cancelar" />
```


Para capturar los valores enviados por el formulario se debe tener en cuenta si el envío fue realizado con POST o con GET, en el caso de que sea POST entonces la captura tendrá la siguiente sintaxis

```
$variable = $_POST['nombre_objeto']
```

En el caso de GET:

```
$variable = $_GET['nombre_objeto']
```

Si se desconoce el envío de la misma o el proceso recibe el envío en ocasiones por POST y en otras GET entonces se puede utilizar `$variable = $_REQUEST['nombre_objeto']`, el cual captura los datos sin importar que sea GET o POST.

\$variable: es el nombre de la variable que almacenara el valor enviado desde el formulario.

Nombre_objeto: es el nombre especificado en Name del objeto definido en el form.

Si en el formulario se definió el input de la siguiente manera:

```
<form id="form1" name="form1" method="POST" action="proceso_form_base.php">
    <input type="text" name="cedula" id="cedula" size="12" maxlength="12" />
</form>
```

Para recoger los datos seria: `$cedula = $_POST['cedula']` ó `$cedula = $_REQUEST['cedula']`

Ejemplo del Script de Proceso en PHP

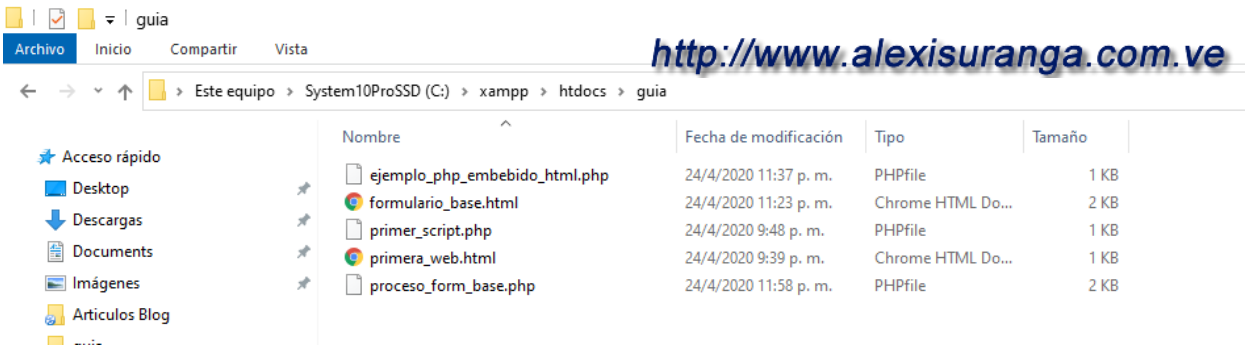
```

1  <?php
2  //Proceso de formulario
3
4  /*Recoger los datos enviados del formulario
5     Sintaxis : $variable=POST['nombre_objeto'];
6     Donde $variable : es cualquier nombre de variable este nombre debe guardar cierta relacion con el valor a almacenar.
7     $_POST : Arreglo que contiene los valores enviados, depende del metodo seleccionado en el Formulario, puede
8               ser POST este envia los datos oculto al usuario.
9     $_GET  : Arreglo que contiene los valores enviados, envia los datos en la url del navegador se ve la url de
10              la siguiente manera :
11              http://localhost/guia/proceso_form_base.php?cedula=14123456&nombre=Pedro+Perez&direccion=La+concordia
12              &fechanac=01%2F01%2F2000&email=pedroperez35%40gmail.com&button=Enviar
13 'nombre_objeto' : es el nombre especificado en la propiedad name del objeto cuando se define el objeto.
14 Conclusión : si se envia con el Metodo POST se debe recibir con el metodo POST[], si se envia con GET[] se debe recibir
15 usado GET[].
16 */
17
18 /*Recoger los datos enviados del Formulario*/
19 $cedula = $_POST['cedula'];
20 $nombre = $_POST['nombre'];
21 $direccion = $_POST['direccion'];
22 $fechanac = $_POST['fechanac'];
23 $email = $_POST['email'];
24
25 /*Proceso: el proceso puede ser variado, una operacion matematica, un calculo de sueldo base comision deduccion
26 un acceso a base de datos, consultar si la cedula esta registrada, actualizar datos, comparacion de datos, para
27 este ejemplo solo mostraremos mos datos editados.*/
28 echo "Los datos enviados del formularios son : <br>";
29 echo "Cédula : " . $cedula;
30 echo "Nombre : " . $nombre;
31 echo "Direccion : " . $direccion;
32 echo "Fecha Nac. : " . $fechanac;
33 echo "Email : " . $email;
34 //Fin del proceso
35 ?>

```

<http://www.alexisuranga.com.ve>

Para poner en ejecución el formulario y el proceso debemos de tener configurado nuestro servidor, para este caso XAMPP y guardar los archivos de formulario_base.html y proceso_form_base.php en una carpeta en el directorio de publicación el cual es la carpeta htdocs, ejemplo si esta guardado en C:/xampp/htdocs/guía, véase [Como crear un servidor de desarrollo usando XAMPP](#).



Para ejecutar el formulario debemos de ir a nuestro navegador web y editar la url perteneciente a http://localhost/guia/formulario_base.html

Y se mostrara el formulario definido, pero corriendo es decir ejecutándose en el navegador web.

Vista del Formulario en ejecución.

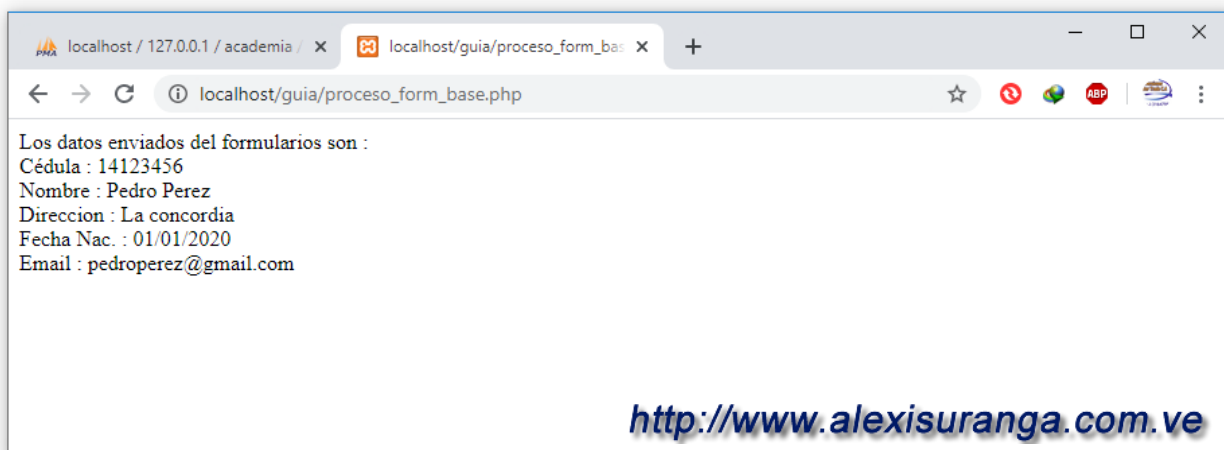


Formulario con los datos editados.



A screenshot of a web browser window. The address bar shows 'localhost/guia/formulario_base.html'. The form contains the following data: Cédula: 14123456, Nombre: Pedro Perez, Dirección: La concordia, Fecha Nac.: 01/01/2020, Email: pedroperez@gmail.com. There are 'Enviar' and 'Cancelar' buttons at the bottom left. The URL 'http://www.alexisuranga.com.ve' is displayed at the bottom right.

Procesando el formulario: esto se logra haciendo click en el botón de enviar.



A screenshot of a web browser window. The address bar shows 'localhost/guia/proceso_form_base.php'. The page displays the following text: 'Los datos enviados del formularios son :', 'Cédula : 14123456', 'Nombre : Pedro Perez', 'Direccion : La concordia', 'Fecha Nac. : 01/01/2020', 'Email : pedroperez@gmail.com'. The URL 'http://www.alexisuranga.com.ve' is displayed at the bottom right.

Para este ejemplo se usó un formulario básico sin ningún adorno, pero se puede mejorar haciendo uso de css y de Framework especializados el cual permite dar ese toque de diseño el cual los formularios sean llamativos y estilizados haciendo que nuestras aplicaciones estén acorde a la era actual.

Att.

Ing. Alexis Uranga

+58 (424) 520.65.39 / +58 (416) 451.52.24

Desarrollo de Portales Web, Aplicaciones Móviles, Cableado Estructurado, Redes en General Cámaras de Seguridad, Consultoría en el Área de Informática, Docente Universitario.

Empresa: mastertradeca.com | Email: info@mastertradeca.com

Web : alexisuranga.com.ve | Twitter : [@alexisuranga](https://twitter.com/alexisuranga) | Facebook : [alexis.uranga](https://www.facebook.com/alexis.uranga)

Instagram : [alexisuranga.com.ve](https://www.instagram.com/alexisuranga.com.ve) | LinkedIn : [alexis-uranga](https://www.linkedin.com/company/alexis-uranga)