|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | http://alimentos2014.epn.edu.ec/logo_epn_android.jpg | | **ESCUELA POLITÉCNICA NACIONAL ESCUELA DE FORMACIÓN DE TECNÓLOGOS** | | | | | |  |
|  |  |  |  |  |  |  |  |
|  | **GUIA DE PRÁCTICAS  LABORATORIO TALLER 2  No. Práctica: 2** | | | | | |  |
|  |  |  |  |  |  |  |  |  |  |
|  | **CARRERA:** |  | **ASA**  \_\_\_ **ASI** \_X\_ **EM** \_\_\_\_ **ET** \_\_\_ | | | |  |  |  |
|  | **ASIGNATURA:** | | Arquitectura Web | | **CÓDIGO:** | TSI463 | **GRUPO:** | GR1 |  |
|  | **FECHA:** |  | (04/11/15) | |  |  |  |  |  |
|  | **APELLIDOS Y NOMBRES :** | | Sánchez Arteaga Fredy Vicente | | | | | |  |
|  |  |  |  |  |  |  |  |  |  |
|  | **CÉDULA DE IDENTIDAD:** | | 1725632552 | |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  | **1. PROPÓSITO DE LA PRÁCTICA:**   * Introducción al lenguaje de programación HTML. | | | | | | | |  |
|  |  |
|  |  |
|  |  |  |  |  |  |  |  |  |  |
|  | **2. OBJETIVO GENERAL:**   * Comprender los tags básicos de una página web con HTML. | | | | | | | |  |
|  |  |
|  |  |
|  |  |  |  |  |  |  |  |  |  |
|  | **3. OBJETIVOS ESPECÍFICOS:**   * Comprender el uso y estructura de los tags de HTML. * Crear un sitio web informativo por medio de HTML. * Configurar el entorno de desarrollo. * Comprender los conceptos de lenguajes del lado del cliente y servidor (Back - End y Front - End) | | | | | | | |  |
|  |  |
|  |  |
|  |  |
|  |  |  |  |  |  |  |  |  |  |
|  | **4. DESCRIPCIÓN DE ACTIVIDADES Y PROCEDIMIENTO DE LA PRÁCTICA:**  Parte I:  1. Vamos a entender que es una arquitectura web por medio de los conceptos presentados en clases.  2. Vamos a comprender que son los lenguajes de lado del servidor y de cliente y en que se diferencian por medio de los conceptos.  3. Presentaremos ejemplos de varios lenguajes con sus respetivas características y sintaxis.  4. Nos vamos a centrar ahora en el lenguaje HTML que es un lenguaje traducido, no compilado, observaremos un poco de la historia del mismo.  5. Primero tenemos que saber que significa HTML (HyperText Markup Languaje).  6. Revisaremos la estructura de un archivo de HTML (nombre y extensión).  7. Por medio de un gráfico observamos cómo se despliega una página HTML en un navegador web.  8. Antes de empezar vamos a definir el contenido que vamos a tener en nuestro sitio web.  9. La estructura de un tag dentro de HTML es la siguiente: (tag de apertura) texto (tag de cierre siempre acompañado con un /)  10. Dentro de cloud9 vamos crear un nuevo archivo para ir escribiendo los tags que vamos a usar.  11. El primer tag que vamos a aprender es el <h1>, este nos permite modificar el formato del texto con el objetivo de mostrar jerarquía dentro de los contenidos, generalmente el h1 se usa para el título de la página o el nombre de la compañía, el h2 para el tema principal y del h3 hasta el h6 en adelante para organizar el contenido de la página.  12. El tag <p> se usa para escribir párrafos.  13. El tag <ul> nos permite crear listas no ordenadas, por ejemplo se usa en ítems que no tienen un orden específico como los ingredientes de una receta, para añadir un ítem dentro de la lista usaremos el tag <li>.  14. Con este tag observamos algo muy importante y es que tenemos tags que van dentro de otros por lo cual están embebidos en los mismos, al tag que contiene otros se le llama padre y a los que están contenidos se les llama hijos.  15. Algo importante que resalta en esta parte es la importancia de la indentación del código para poder entender la jerarquía del código.  16. No realizar la indentación del mismo puede causar que no entendamos el código y terminemos perdidos en el mismo.  17. El tag <ol> nos va a permitir realizar listas ordenadas usando también como elemento lo que este dentro de <li>, la diferencia es que esta será una lista ordenada ya que agregara una viñeta que muestre un orden especifico, esto será usado en elementos que necesitan seguir un orden especifico como son los pasos para la preparación de una receta.  18. Procedemos a envolver todo lo que hemos escrito en el documento dentro del tag <body> todo contenido que se va a mostrar en la página estará dentro del body, en realidad el body no despliega nada pero ayuda a mantener el contenido organizado.  19. El tag <head> lo ubicaremos arriba de donde empieza el <body>, este contiene datos que no se ven en la página como CSS, JavaScript, etc, estos los veremos más adelante.  20. Una vez completo esto lo envolveremos dentro del tag principal <html>.  21. Un tag que ubicaremos fuera del <html> es el <!DOCTYPE html> el cual le dice al navegador que versión de html usa, si bien este tag es opcional es bueno usarlo ya que ayuda al navegador a tomar decisiones acerca de cómo desplegar su página de la mejor manera.  22. Una vez terminado esto vamos a guardar el archivo con el nombre de index.html y vamos a ejecutar desde el cloud9 para poder ver lo que hemos realizado.  23. Añadiremos ahora enlaces para darle movilidad a la página.  24. Comprendamos que es lo que pasa cuando escribimos un url dentro del navegador, primero tenemos siempre un protocolo luego el nombre del servidor y al final la ubicación del archivo.  25. Luego se realiza la petición al servidor y se obtiene el archivo.html que se ha solicitado, aun no se ha cargado nada en el navegador.  26. Cuando el servidor responde a la solicitud con el documento y el navegador lo despliega podemos ver la página solicitada.  27. Comprendamos ahora que pasa cuando hacemos clic en un link dentro del navegador, primero vamos a ver qué hace una petición de una nueva página, el servidor responde con el nuevo archivo para ser desplegado en el navegador y vemos que el URL cambia a la nueva dirección a la cual realizamos la petición.  28. Como hacemos para que esta magia suceda, para esto vamos a usar el tag <a>, lo que este dentro de este tag se va a entender como un enlace.  29. Cada tag tiene varios atributos que nos permiten modificar su funcionalidad, estos los usaremos conforme avancemos con la materia.  30. El primer atributo que usaremos es el “href” este atributo contiene la dirección a la cual vamos a enlazar al texto contenido entre el tag <a>.  31. Tenemos 2 opciones, que la pagina a la cual enlazamos reemplace la página actual y la otra es que se abra en una nueva ventana, para esto vamos a usar otro atributo llamado “target” dentro de este vamos a poner “\_blank” si necesitamos que se abra en una nueva ventana y vamos a poner “\_self” si la idea es que reemplace la página actual, por defecto si no ubicamos el target va a sobre escribir la página actual.  32. Vamos a realizar 2 enlaces distintos y ver qué pasa probando lo visto anteriormente.  33. Dentro del href podemos ubicar 2 tipos de direcciones que son absolutas y relativas.  34. Una dirección absoluta contiene la dirección de un servidor específico.  35. Una dirección relativa no especifica ningún servidor, por lo tanto el navegador asume que el contenido se va a encontrar en el mismo servidor en el cual está la página que contiene el enlace.  Parte II  1. Crearemos varios archivos html con contenidos distintos para terminar con un sitio web básico.  2. Crearemos una página llamada receta.html la cual va a contener el menú puesto en la página principal o el index.html y además de eso tendremos ahí el listado de las recetas que vamos a desplegar.  3. Crearemos otro archivo que se llamara sugerencias.html en donde vamos a poner el mismo menú del index.html y vamos a poner un párrafo explicativo de a que se refieren las sugerencias.  4. Crearemos un archivo que va a contener el mismo menú del index.html y este a diferencia del resto va a mostrar la receta puesta dentro del archivo recetas.html y aquí vamos a tener el nombre de la receta, los ingredientes y los pasos de la preparación todo organizado con listas ordenadas o no ordenadas y con sus respectivos “headings”.  5. Tendremos que crear una página por cada receta que vayamos a poner.  6. Finalmente añadiremos los links a cada una de las páginas creadas anteriormente para que desde nuestra página principal podamos observar cómo van pasando de página a página creando así nuestro primer sitio web.  7. Además ubicaremos enlaces externos a páginas colgadas en internet para los ingredientes para que la persona pueda saber que son y donde comprarlos.  Deber:  • Crear 4 páginas nuevas cada una para una nueva receta de cualquier comida con sus respectivos ingredientes e instrucciones de preparación.  • Enlazarlas al sitio web que tenemos hasta el momento usando enlaces. | | | | | | | |  |
|  |  |
|  |  |
|  |  | | **ESCUELA POLITÉCNICA NACIONAL ESCUELA DE FORMACIÓN DE TECNÓLOGOS** | | | | | |  |
|  |  |  |  |  |  |  |  |
|  | **http://alimentos2014.epn.edu.ec/logo_epn_android.jpg5. TÉCNICAS E INSTRUMENTOS APLICADOS:**   * Blog de notas. * Cloud9. | | | | | | | |  |
|  |  |
|  |  |
|  |  |
|  |  |  |  |  |  |  |  |  |  |
|  | **6. RESULTADOS**  1. Conceptos presentados en clases.   * **Arquitectura web.-** Nos referimos a la disciplina y arte encargada del estudio, análisis, organización, disposición y estructuración de la información en espacios de información, en este caso específicamente, Páginas Web. * **Modelos de arquitectura:**   + - **Modelo CGI:** Mantiene la persistencia, negocio y presentación mezclados.     - **Modelo JSP y Servlets:** Separación de responsabilidades, donde los JSPs llevan la lógica de presentación y los Beans asumen la responsabilidad del negocio y datos.     - **Modelo Vista Controlador:** Incorpora el diseño MVC siendo que:       * Controlador: Navegación.       * Negocio y datos: Beans       * Presentación: JSPs.   2. Lenguajes de lado del servidor y de cliente y en que se diferencian por medio de los conceptos.  **Lenguajes del lado del servidor:** Utilizan sus propios recursos y solamente muestran los resultados al cliente.  http://www.adelat.org/media/docum/nuke_publico/images/php.gifhttp://www.adelat.org/media/docum/nuke_publico/images/asp.gifhttp://www.adelat.org/media/docum/nuke_publico/images/xml.gif  **Lenguajes de lado del cliente:** Utilizan nuestros recursos se ejecutan en nuestro computador.  http://www.adelat.org/media/docum/nuke_publico/images/html.gifhttp://www.adelat.org/media/docum/nuke_publico/images/javascript.gifhttp://www.adelat.org/media/docum/nuke_publico/images/java.gif  3. Lenguajes con sus respetivas características y sintaxis.   * **Lenguaje HTML**   + Permite la escritura de hipertexto.   + Despliegue rápido.   + Lo admiten todos los exploradores.   + Lenguaje estático.   + Interpretación depende del navegador. * **Lenguaje JavaScript**   + Lenguaje interpretado que no requiere compilación.   + Similar a java aunque no mantiene la orientación a objetos.   + Lenguaje seguro y fiable.   + Código visible para cualquier usuario.      * **Lenguaje PHP**   + No necesita ser compilado para ser ejecutado.   + Para su funcionamiento necesita tener instalado   Apache o IIS con librerías PHP.   * + Mantiene clase y herencia.   + Capacidad de expandirse usan módulos.      * **Lenguaje ASP.net**   + Desarrollo en C#, Visual Basic, J#   + Completamente orientado a objetos.   + Mayor consumo de recursos.   + Controles de usuario personalizados. * **Lenguaje Python**   + Lenguaje más limpio para programar.   + Libre y de fuente abierta.   + Multiplataforma.   + Lentitud por ser lenguaje interpretado.   4. Lenguaje HTML  **HTML (HyperText Markup Languaje).**  El HTML se escribe en forma de «etiquetas», rodeadas por [corchetes angulares](https://es.wikipedia.org/wiki/Par%C3%A9ntesis) (<,>, /). El HTML también puede describir, hasta un cierto punto, la apariencia de un documento, y puede incluir o hacer referencia a un tipo de programa llamado [script](https://es.wikipedia.org/wiki/Script), el cual puede afectar el comportamiento de [navegadores web](https://es.wikipedia.org/wiki/Navegador_web) y otros procesadores de HTML   * Hace referencia al marcado para la elaboración de [páginas web](https://es.wikipedia.org/wiki/P%C3%A1gina_web). * Lo admiten todos los navegadores. * Extension .htm o .html   7. Navegador web respuesta a petición de mostrar una pagina.  http://memo-dev.com/blog/wp-content/uploads/2014/01/Funcionamiento-Servidor-Web.png  8. Dentro de cloud9 vamos crear un nuevo archivo para ir escribiendo los tags que vamos a usar.    9. El primer tag que vamos a aprender es el <h1>, este nos permite modificar el formato del texto con el objetivo de mostrar jerarquía dentro de los contenidos, generalmente el h1 se usa para el título de la página o el nombre de la compañía, el h2 para el tema principal y del h3 hasta el h6 en adelante para organizar el contenido de la página.    10. El tag <p> se usa para escribir párrafos.      11. El tag <ul> crea listas no ordenadas, por ejemplo se usa en ítems que no tienen un orden específico como los ingredientes de una receta, para añadir un ítem dentro de la lista usaremos el tag <li>.    **Es importante la identación del código para poder entender la jerarquía del código.**  12. El tag <ol> nos va a permitir realizar listas ordenadas usando elemento lo que este dentro de <li>, la diferencia es que esta será una lista ordenada ya que agregara una viñeta que muestre un orden especifico, esto será usado en elementos que necesitan seguir un orden especifico como son los pasos para la preparación de una receta.      13. Todo el código identado se encuentra dentro del <body>, que es lo que se mostrara en pantalla.  En el <head> contiene datos que no se ven en la página como CSS, JavaScript, etc.  Todo este código se encuentra dentro del tag principal<html>  Para indicar la versión que se usa al navegador se utiliza el tag <! DOCTYPE html>.  Realizado el código Guardaremos con el nombre index.html.  **La estructura de un tag dentro de HTML es la siguiente: (tag de apertura) texto (tag de cierre siempre acompañado con un /)**  Las imágenes representan la implementación del código HTML y los resultados que este muestra con la utilización de los tags correspondientes.    14. Ahora añadiremos ahora enlaces para darle movilidad a las páginas.  http://recursostic.educacion.es/observatorio/web/images/upload/jmaa0052/01-1.JPGEstructura del URL:  15. Como hacemos para que se muestre la página con un clic; vamos a usar el tag <a>, lo que este dentro de este tag se va a entender como un enlace. El tag tiene varios atributos:   * El “href” este atributo contiene la dirección a la cual vamos a enlazar al texto contenido entre el tag <a>.   Y luego tenemos dos opciones, que la pagina a la cual enlazamos reemplace la página actual usamos “\_self” y la otra es que se abra en una nueva ventana, para esto usamos “\_blank” sin olvidar que para las dos circunstancias necesitamos del “target.  Dentro del href podemos ubicar 2 tipos de direcciones que son absolutas y relativas.  **Dirección absoluta:** Contiene la dirección de un servidor específico.  **Dirección relativa:** Una dirección relativa no especifica ningún servidor, por lo tanto el navegador asume que el contenido se va a encontrar en el mismo servidor en el cual está la página que contiene el enlace.  16. Codificación de cada una de las páginas que permitan realizar los enlaces correspondientes.  Resultados al realizar los enlaces con el uso del tag <a> usando “\_self” o “\_blank”:    Parte II  Creación de páginas adicionales para determinar un sitio web básico.   1. Creamos receta.html codificamos y visualizamos sus resultados.      1. Creamos pastelmagico.html donde mostrara la receta detallada. Y además realizamos un enlace con una página Wikipedia para mostrar formación adicional en otra ventana. Estos son enlaces externos.          1. Creamos sugerencias.html codificamos con un párrafo indicativo y visualizamos resultados.     Deber:  Creación de 4 páginas web de comida.    Página principal de todas las recetas a las que se les agrego las 4 recetas.       1. galletas.html implementación y visualización correcta.     Resultados visualización.   1. pay.html implementación y visualización correcta.     Resultados Visualización.   1. frambuesa.html implementación y visualización correcta.   Resultados Visualización.   1. platanos.html implementación y visualización correcta.     Resultados Visualización. | | | | | | | |  |
|  |  |
|  |  |
|  |  |
|  |  |  |  |  |  |  |  |  |  |
|  | **7. CONCLUSIONES**     * Se entiende los conceptos de la arquitectura de las páginas web así también el código de implementación HTML * Todas las páginas web son creadas con lenguaje HTML y se las conecta por medio de enlaces. * Todas las páginas fueron creadas en el ambiente de cloud9. | | | | | | | |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |  |  |  |  |  |  |  |  |
|  | **8. BIBLIOGRAFÍA REFERENCIAL:**  [1] c9.io | | | | | | | |  |
|  |  |
|  |  |
|  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  | **Fredy** | **Sánchez A.** |  |  |  |  |
|  | **FIRMA DEL ESTUDIANTE** | | | | | | | |  |