



Actividad - Unidad 1: Fundamentos e introducción a tecnologías Web

Introducción a las tecnologías web

Unidad 1: Introducción: Cómo funciona Internet y la Web

1. Definición de Internet:

- ¿Qué es Internet y cuál es su importancia en la infraestructura de la Web?*

Internet es una red global de computadoras interconectadas que permite la comunicación y el intercambio de datos a través de protocolos (como TCP/IP). Es fundamental para la infraestructura técnica de la Web porque proporciona la conectividad para que los usuarios accedan a sitios web y servicios en línea.

- Describe brevemente la evolución de ARPANET a Internet.*

ARPANET fue la primera red de computadoras, desarrollada por el Departamento de Defensa de los Estados Unidos. Su propósito inicial era permitir la comunicación entre diferentes universidades y centros de investigación. Con el tiempo, ARPANET evolucionó y se expandió, adoptando el protocolo TCP/IP, que permitió la creación de redes interconectadas más grandes. Eventualmente, ARPANET se convirtió en Internet, una red global abierta.

2. Protocolos de Comunicación:

- Explica la función del protocolo TCP/IP en Internet.*

TCP/IP es un conjunto de protocolos de comunicación que permite la transferencia de datos a través de redes de computadoras. TCP

(Transmission Control Protocol) se encarga de dividir los datos en paquetes y asegurarse de que se entreguen correctamente, mientras que IP (Internet Protocol) maneja el direccionamiento y la entrega de estos paquetes.

- b. *¿Qué es una dirección IP y cuál es la diferencia entre una IP pública y una privada?*

Una dirección IP es un identificador único asignado a cada dispositivo conectado a una red para permitir su identificación y comunicación. Las direcciones IP públicas son únicas en Internet y se utilizan para identificar dispositivos accesibles desde fuera de una red local. Las direcciones IP privadas son utilizadas dentro de redes locales y no son únicas a nivel global, lo que significa que no pueden ser utilizadas para acceder directamente a través de Internet.

3. Infraestructura de Internet:

- a. *¿Qué elementos componen la infraestructura de comunicación de Internet?*

Servidores, routers, switches, centros de datos, y los enlaces de comunicación como cables de fibra óptica, cables submarinos, antenas y satélites.

- b. *Menciona y explica brevemente el rol de los satélites, antenas y cables submarinos en Internet.*

- **Satélites:** Permiten la comunicación y transmisión de datos en lugares donde las infraestructuras terrestres no son posibles.
- **Antenas:** Se usan para transmitir y recibir señales de datos inalámbricas, especialmente en redes móviles y Wi-Fi.
- **Cables submarinos:** Son los principales medios de transmisión de datos a nivel global, ya que conectan diferentes continentes y permiten la rápida transferencia de grandes cantidades de datos.

4. Conceptos Básicos de la Web:

- a. *Define y explica la importancia de los siguientes términos: HTML, URL, y HTTP/HTTPS.*
- **HTML (HyperText Markup Language):** Es el lenguaje de marcado para crear y diseñar páginas web. Define la estructura del

contenido utilizando elementos como títulos, párrafos, enlaces, imágenes, etc, a través de etiquetas.

- **URL (Uniform Resource Locator):** Es la dirección que se utiliza para acceder a recursos específicos en la web, como páginas, imágenes, videos, etc.
- **HTTP (HyperText Transfer Protocol) y HTTPS (HTTP Secure):** HTTP es el protocolo usado para la transferencia de datos en la web. HTTPS es una versión segura de HTTP que usa encriptación para proteger la comunicación entre el navegador del usuario y el servidor web.

b. *¿Cuál es la relación entre Internet y la Web?*

Internet es la infraestructura física y de red que conecta computadoras y otros dispositivos a nivel mundial. La Web es un servicio que opera sobre Internet, proporcionando una forma de acceder y compartir información a través de páginas web y aplicaciones.

Unidad 2: HTML5

1. Estructura de HTML:

a. *¿Qué es HTML y cuál es su propósito principal en la Web?*

HTML es el lenguaje de marcado utilizado para estructurar y presentar contenido en la Web. Su propósito principal es definir la estructura y el formato de una página web, incluyendo texto, imágenes, enlaces, formularios y otros elementos multimedia.

b. *Escribe una estructura básica de un documento HTML y explica cada una de sus partes principales.*

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="UTF-8">
    <title>Mi página web</title>
    <link rel="stylesheet" href="style.css">
  </head>
  <body>
    <header>
```

```

<h1>Bienvenidos a mi página web</h1>
</header>

<nav>
  <ul>
    <li><a href="#inicio">Inicio</a></li>
    <li><a href="#sobre-mi">Sobre mí</a></li>
    <li><a href="#contacto">Contacto</a></li>
  </ul>
</nav>

<main>
  <section id="inicio">
    <h2>Inicio</h2>
    <p>Contenido de la sección de inicio.</p>
  </section>
</main>

<footer>
  <p>© 2024 Mi Página Web</p>
</footer>

</body>
</html>

```

- `<!DOCTYPE html>`: Declara el tipo de documento como HTML5.
- `<html>`: Elemento raíz que contiene todo el contenido de la página.
- `<head>`: Contiene metadatos sobre la página, como el juego de caracteres (`<meta charset="UTF-8">`), el título (`<title>Mi página web</title>`) y enlaces a hojas de estilo externas (`<link rel="stylesheet" href="style.css">`).
 - `<title>`: Define el título de la página que se muestra en la pestaña del navegador.
 - `<link>`: Vincula una hoja de estilo externa.
- `<body>`: Contiene el contenido principal que el usuario va a ver de la página web, como texto, imágenes y otros elementos.
- `<header>`, `<nav>`, `<main>`, `<section>`, `<footer>`: Elementos semánticos que organizan el contenido de la página.

2. Etiquetas HTML:

a. *¿Qué son las etiquetas en HTML y cómo se utilizan?*

Las etiquetas en HTML son elementos que se utilizan para definir la estructura y el contenido de una página web. Cada etiqueta tiene un nombre y puede contener atributos. Las etiquetas suelen venir en pares de apertura y cierre, aunque algunas son autosuficientes.

b. *Menciona y describe brevemente al menos cinco etiquetas comunes en HTML.*

- `<p></p>` : Define un párrafo de texto.
- `<a>` : Define un enlace a otra página o recurso.
- `` : Inserta una imagen.
- `<div></div>` : Define una división o sección en la página.
- `<h1></h1>` a `<h6></h6>` : Definen encabezados, con `<h1>` siendo el más importante y `<h6>` el menos. (visualmente van de un tamaño más grande a uno más pequeño)

3. Atributos HTML:

a. *¿Qué son los atributos en HTML y para qué se utilizan?*

Los atributos en HTML proporcionan información adicional sobre los elementos. Se escriben dentro de la etiqueta de apertura y afectan cómo se muestra o se comporta el elemento. Por ejemplo, el atributo `href` en un enlace (`<a>`) especifica la URL a la que se dirige el enlace.

b. *Da un ejemplo de una etiqueta HTML con atributos y explica su función.*

```
<a href="https://www.google.com" target="_blank" rel="noopener">Enlace</a>
```

- `<a>` : Define un enlace.
- `href` : Especifica la URL a la que se dirige el enlace.
- `target="_blank"` : Abre el enlace en una nueva pestaña.
- `rel="noopener"` : Aumenta la seguridad y el rendimiento al abrir el enlace en una nueva pestaña.

Unidad 3: CSS3

1. Definición y Uso de CSS:

a. *¿Qué es CSS y cuál es su papel en el diseño de páginas web?*

CSS (Cascading Style Sheets) es un lenguaje de hojas de estilo utilizado para describir la presentación y el diseño de un documento HTML. CSS permite controlar el aspecto visual de una página web, incluyendo colores, fuentes, espaciado y diseño.

b. *Explica cómo CSS puede cambiar la apariencia de una página HTML.*

CSS cambia la apariencia de una página HTML definiendo estilos para los elementos. Por ejemplo, puede modificar el color del texto, el tamaño de las fuentes, el espaciado entre elementos, el fondo, los bordes. CSS permite separar el contenido de la presentación, lo que facilita la creación de diseños consistentes y atractivos.

2. Selectores y Propiedades CSS:

a. *¿Qué es un selector en CSS y cuáles son los tipos más comunes?*

Un selector en CSS es una parte del código que se utiliza para seleccionar los elementos HTML a los que se aplicarán las reglas de estilo. Los tipos más comunes incluyen:

- **Selectores de elementos:** Seleccionan todos los elementos de un tipo dado, por ejemplo, `p` para todos los párrafos.
- **Selectores de clases:** Seleccionan elementos con una clase específica, por ejemplo, `.mi-clase`.
- **Selectores de ID:** Seleccionan un elemento con un ID específico, por ejemplo, `#mi-id`.
- **Selectores de atributos:** Seleccionan elementos basados en un atributo específico, por ejemplo, `a[href="https://www.ejemplo.com"]`.
- **Selectores de pseudoclases:** Seleccionan elementos en un estado específico, por ejemplo, `a:hover`.

b. *Menciona y describe cinco propiedades de CSS con ejemplos de uso.*

- `color` : Define el color del texto.

```
p {  
    color: blue;  
}
```

- `background-color`: Define el color de fondo de un elemento.

```
div {  
    background-color: yellow;  
}
```

- `font-size`: Define el tamaño de la fuente.

```
h1 {  
    font-size: 24px;  
}
```

- `margin`: Define el espacio exterior alrededor de un elemento.

```
.contenedor {  
    margin: 10px;  
}
```

- `padding`: Define el espacio interior de un elemento.

```
.caja {  
    padding: 20px;  
}
```

3. Diseño Responsivo:

- a. ¿Qué es el diseño responsive y por qué es importante en el desarrollo web moderno?

El diseño responsive es un enfoque de diseño web que permite que las páginas web se adapten a diferentes tamaños de pantalla y dispositivos. Es importante porque asegura que los usuarios tengan una experiencia de navegación más óptima, independientemente del dispositivo que utilicen, ya sea una computadora de escritorio, una tableta o un teléfono móvil.

b. *Explica cómo se puede lograr un diseño responsive utilizando CSS.*

El diseño responsive se puede lograr utilizando CSS con técnicas como:

- **Media Queries:** Permiten aplicar estilos específicos para diferentes tamaños de pantalla.

```
@media (max-width: 600px) {  
    .contenedor {  
        flex-direction: column;  
    }  
}
```

- **Unidades flexibles:** Utilizar unidades relativas como `%`, `em`, `rem` y `vw/vh` en lugar de unidades absolutas como `px`.
- **Diseño de cuadrícula y Flexbox:** Utilizar estas técnicas de diseño para crear estructuras de página flexibles que se adapten a diferentes tamaños de pantalla.

Unidad 4: JavaScript

1. Introducción a JavaScript:

a. *¿Qué es JavaScript y cuál es su propósito principal en el desarrollo web?*

JavaScript es un lenguaje de programación que se utiliza para crear contenido interactivo en las páginas web. Su propósito principal es permitir la manipulación dinámica del contenido del DOM (Document Object Model), lo que permite actualizar el contenido de la página, responder a eventos de usuario, validar formularios, entre otras funcionalidades.

b. *Explica la diferencia entre JavaScript y otros lenguajes de programación como Python o Java.*

JavaScript es un lenguaje de programación de alto nivel y basado en eventos, principalmente utilizado para el desarrollo web. Es interpretado y se ejecuta en el navegador del cliente, lo que lo hace ideal para la interacción en tiempo real en páginas web. Python y Java, aunque también son lenguajes de alto nivel, suelen ser utilizados en

diferentes contextos, como desarrollo de aplicaciones de servidor, ciencia de datos, aplicaciones de escritorio, etc. Además, JavaScript tiene una sintaxis basada en prototipos, mientras que Python y Java son orientados a objetos de clase.

2. Sintaxis Básica de JavaScript:

- a. Escribe un ejemplo simple de un script en JavaScript que muestre un mensaje en la consola del navegador.

```
console.log("¡Hola, mundo!");
```

- b. ¿Qué son las variables en JavaScript y cómo se declaran?

Las variables en JavaScript son contenedores que almacenan datos. Se pueden declarar utilizando `var`, `let`, o `const`. `let` y `const` fueron introducidos en ES6 y se prefieren sobre `var` debido a su alcance más limitado.

```
let nombre = "Juan";
const edad = 30;
```

3. Eventos en JavaScript:

- a. ¿Qué es un evento en JavaScript y cómo se manejan?

Un evento en JavaScript es una acción o suceso que ocurre en la página web, como un clic de un usuario, una tecla presionada o la carga de la página. Los eventos se manejan mediante el uso de manejadores de eventos (event listeners), que son funciones que responden a esos eventos.

- b. Da un ejemplo de cómo utilizar un evento para cambiar el contenido de una página web.

HTML:

```
<button id="miBoton">Haz clic aquí</button>
<p id="miParrafo">Texto original.</p>
```

JAVASCRIPT:

```
document.getElementById("miBoton").addEventListener("click", function() {
    document.getElementById("miParrafo").textContent =
    "¡Hiciste clic!";
});
```

En este ejemplo, al hacer clic en el botón con el ID "miBoton", el contenido del párrafo con el ID "miParrafo" se cambia a "¡Hiciste clic!".