Portafolio de Evidencias agosto – diciembre 2024

Estructuras de Datos

Nombre del Alumno:

  Tercer Semestre

Contenido

[Evaluación Diagnostica 2](#_Toc177975490)

[Semana 1- Intro a Herramientas para el Curso de Estructuras de datos 3](#_Toc177975491)

[Semana 2 - 4](#_Toc177975492)

# Evaluación Diagnostica

Instrucciones: Lee cada pregunta y selecciona la opción correcta.

¿Cuál de los siguientes es un principio fundamental de la programación orientada a objetos?

A) Encapsulación

B) Recursividad

C) Segmentación

D) Iteración

Respuesta Correcta: A) Encapsulación

En Programación Orientada a Objetos, ¿qué es una clase?

A) Un tipo de variable que almacena datos temporales

B) Una estructura de control de flujo

C) Una plantilla o modelo para crear objetos

D) Una función que realiza una tarea específica

Respuesta Correcta: C) Una plantilla o modelo para crear objetos

¿Qué significa herencia en el contexto de la Programación Orientada a Objetos?

A) La capacidad de una clase para derivar atributos y métodos de otra clase

B) La habilidad de crear múltiples instancias de una clase

C) La posibilidad de dividir un programa en múltiples subprogramas

D) El uso de variables globales dentro de una clase

Respuesta Correcta: A) La capacidad de una clase para derivar atributos y métodos de otra clase

¿Qué es un objeto en POO?

A) Un tipo de dato primitivo

B) Una función que encapsula código reutilizable

C) Una instancia de una clase que contiene atributos y métodos

D) Un bloque de código ejecutable de manera independiente

Respuesta Correcta: C) Una instancia de una clase que contiene atributos y métodos

¿Cuál de los siguientes conceptos describe el "polimorfismo" en la POO?

A) La capacidad de una función para realizar múltiples tareas

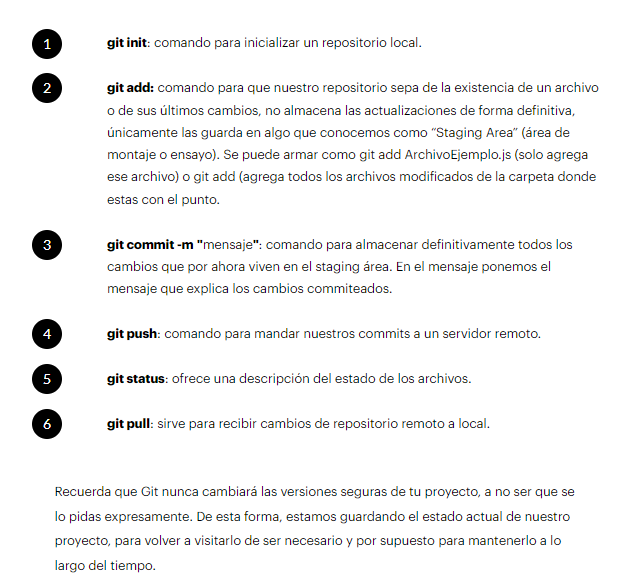
B) La habilidad de una función o método de adoptar diferentes comportamientos en función del contexto

C) La reutilización de variables en diferentes clases

D) La creación de múltiples objetos de una sola clase

Respuesta Correcta: B) La habilidad de una función o método de adoptar diferentes comportamientos en función del contexto

# Semana 1- Intro a Herramientas para el Curso de Estructuras de datos



# Semana 2 - Introducción a la terminal

Esta es la primera de nuestras video clases del PrepCourse, ¡esperamos que este despegue llegue bien alto! 🚀 En esta lección, veremos la introducción a la terminal. ¡Mira los videos a continuación!

**Comandos para otros sistemas operativos (Linux y Mac)**

**Linux:**

1. Presiona Ctrl + Alt + T para abrir una terminal.
2. Presiona en el menú de aplicaciones y busca “Terminal” o “Consola”.
3. Usa el atajo de teclado específico de tu entorno de escritorio (por ejemplo, Ctrl + Alt + T en GNOME y Unity, Super + T en XFCE).

**Mac:**

1. Abre la carpeta “Aplicaciones” y luego abre la carpeta “Utilidades”. Haz doble clic en “Terminal”.
2. Usa Spotlight para buscar “Terminal” y presiona Enter.
3. Presiona Cmd + Espacio para abrir Spotlight y escribe “Terminal”, luego presiona Enter.

**¿Cómo cambiar la configuración de mi terminal?**

Como ya sabes, estaremos utilizando la terminal de **Git Bash** a lo largo del Prep Course. Uno de los beneficios que tiene el editor de texto Visual Studio Code es que nos permite integrar esta terminal allí mismo. Por defecto este editor de texto utiliza la terminal de **powershell**. ¡A continuación te compartimos cómo cambiar esto!

# Semana 3 – HTML

**Introducción**

En esta clase haremos una introducción al lenguaje de marcado HTML. Exploraremos su sintaxis y los distintos tipos de etiquetas que podremos encontrar. El objetivo es que al finalizar esta clase puedas construir una página web básica.

**Definición**

**HyperText Markup Language** (HTML) es un lenguaje de marcado (no de programación) utilizado para crear páginas web. Este tipo de código te permitirá crear una página web sencilla.

**¿Sabías que lo que consumes en internet está hecho con HTML?** Para revisarlo haz click derecho y busca la opción inspeccionar. Se abrirá una columna en la que podrás observar código. Este código es HTML😎 .

**📖 [DATO]:** los archivos HTML tienen extensión**.html**.

**Estructura de un archivo HTML**

En la siguiente imagen veremos cómo es la **estructura inicial** de un archivo HTML. Estas líneas de código irán creciendo a medida que construyamos nuestra página web.

**💡[TIP]:** la **indentación**(el espacio que colocamos al inicio de las líneas) es un recurso muy útil, ya que nos permite distinguir visualmente cómo es la estructura del archivo. Aunque no es obligatorio, todos los desarrolladores en el mundo lo utilizan.

La etiqueta **<html>**tiene un atributo**lang="es"**. Este atributo configura el idioma. Si fuese inglés habría que escribir **lang="en"**.

**¿Qué es un elemento?**

Todos los archivos HTML están conformados por distintos elementos HTML. Estos elementos son componentes individuales (conocidos como **etiquetas**) que nos permiten darle una estructura al archivo. Por ejemplo...

**Etiquetas**

Como hemos visto en la lección anterior, los elementos dentro de un archivo HTML se distinguen mediante **etiquetas**. Cada elemento tiene una etiqueta asociada.

**💡[TIP]:** las gran mayoría de las etiquetas cuentan con una de apretura y una de cierre: **<p> </p>**.

Por ejemplo, todo lo que escribamos dentro de una etiqueta**<p>** se transformará en texto plano dentro de nuestra página web.

# Semana 4 - CSS

**Introducción**

En esta clase haremos una introducción al lenguaje de hojas de estilo CSS. Este se aplica solamente luego de haber creado un archivo HTML (utilizaremos el que trabajaste en la clase de HTML). Aprenderemos su sintaxis y a darle estilos a nuestra página web.

**Definición**

CSS significa **Cascading Style Sheets**(hojas de estilo en cascada). Este permite a los desarrolladores web controlar el diseño y la apariencia de sus sitios web separando el contenido (como el texto y las imágenes) de la presentación (como los colores, las fuentes y el diseño).

**📖 [DATO]:** los archivos CSS tienen extensión**.css**.

**Estilos in-line**

**Todas las etiquetas** de HTML que hemos visto anteriormente, como por ejemplo la etiqueta **<p>** o la etiqueta **<h1>**, pueden recibir algo llamado atributo.

Te invitamos a que intentes utilizar estos estilos en tu archivo HTML.

Los **atributos** son propiedades o*"configuraciones"* que nos permitirán modificar ese elemento de alguna manera (por ejemplo, los estilos).

En el ejemplo de la imagen podemos ver como el atributo**style** seguido de una propiedad y su valor, nos permite cambiar el color del fondo y de un texto.

**Estilos en archivo**

**Archivo de CSS**

Como has visto en la lección anterior, podemos integrar fácilmente estilos de CSS dentro de las etiquetas de HTML a través del atributo **style**. Pero esta no es la única manera. De hecho, es más recomendado tener un archivo específicamente de estilos de CSS, ya que de esta forma el código estará mejor organizado.

**¿Cómo funciona?**

En las siguientes lecciones veremos las distintas formas que tenemos de escribir código CSS, pero a modo de introducción ahora veremos una de ellas. Utilizaremos un **selector**llamado **clase**, que nos permitirá elegir a qué elemento HTML queremos darle estilos.

El **selector**será una referencia que nos permitirá seleccionar un elemento HTML. La **propiedad**dirá qué característica queremos cambiar de ese elemento. Por último, el **valor**dirá cuál será el estilo que tendrá esa propiedad.

# Semana 5 - JS

Texto

