**OBSERVACIONES IMPORTANTE ANTES DE INICIAR:**

Cada trabajo o actividad académica que sea desarrollado por usted como evidencia de su proceso de aprendizaje, debe estar 100% libre de frases copiadas desde cualquier fuente original (documento, libro, página web, proyecto de código fuente, video tutorial, etc) de información.

En ningún momento se le está pidiendo que haga las cosas de 0 y olvide por completo los conocimientos sobre los cuales se basa algún tema de esta asignatura, por el contrario, lo estamos animando a que lea, investigue, comprenda, deduzca, analice, sintetice, concluya, interprete, argumente, domine, practique, critique, niegue o afirme con bases sólidas basadas en la lectura y el estudio autónomo y autodidacta.

Tenga muy presente que mientras estudias esta profesión y durante toda tu vida, siempre seras evaluado y medido de acuerdo por un docente, por un instructor, por tu equipo de trabajo, por tus jefes, por la sociedad o por tus clientes.

Antes de realizar cualquier acción indebida en el desarrollo de esta y cualquier otra actividad académica, recuerda que actualmente existen muchas herramientas modernas y avanzadas que existen analizar, buscar, identificar, rastrear y comparar la idoneidad y originalidad de cada contenidos digital (texto, imágenes, archivos, etc), con especial profundidad en archivos de documentos académico o que contienen fuentes de programas informáticos, las cuales permiten detectar plagio de una manera tremendamente simple, rápida y automatizada.

Evita que el pensamiento del mínimo esfuerzo y el facilismo invada tu mente y tus acciones, ya que este destruirá tus sueños de crecimiento y éxito profesional.

**OBSERVACIONES**:

##### **1.** **ACTIVIDAD GRUPAL (GRUPOS DE 3 PERSONAS)**

1. CADA INTEGRANTE (SIN EXCEPCIÓN ALGUNA) DEBE SUBIR LA ACTIVIDAD DESDE SU RESPECTIVA CUENTA EN SIMA-PESAD
2. SOLO SE EVALUARÁ Y CALIFICARÁ A LOS ESTUDIANTES QUE HAYAN SUBIDO LA ACTIVIDAD, LOS ALUMNOS QUE NO ENTREGUEN LA ACTIVIDAD SERÁN EVALUADOS CON NOTA 1.0 AUNQUE APAREZCAN EN LA LISTA DE INTEGRANTES DEL TRABAJO.

FORMATO DE ENTREGA:

Un archivo comprimido en formato ZIP (NO USAR OTRO FORMATO)

○ El archivo comprimido debe contener como mínimo los siguientes archivos.

■ Documento en formato que contiene el desarrollo y explicación detallada del desarrollo de la actividad. En formato WORD (NO USAR OTRO FORMATO)

■ Adicionalmente, cada estudiante debe entregar un documento en word que contenga una explicación descriptiva y sustentación conceptual sobre el desarrollo de cada ítem.

■ El desarrollo y entrega del trabajo debe contener todos los ítem típicos y característicos de un trabajo académico.

1. Presentación
2. Tabla de contenido
3. Introducción
4. Objetivos
5. Justificación
6. Desarrollo
7. Síntesis y argumentación individual (es decir, una por cada integrante del grupo)
8. Bibliografía

**Normas APA para la elaboración de esta actividad**

<https://www.colconectada.com/normas-apa/>

**DESARROLLO DE LA ACTIVIDAD:**

Esta actividad evaluativa tiene como objetivo evidenciar el nivel de estudio y práctica que cada integrante del grupo ha tenido sobre los fundamentos de Arreglos y Algoritmos de Ordenamiento.

Como equipo de trabajo deben desarrollar y documentar los siguientes puntos:

1. Mostrar un ejemplo de uso de arreglos en java así:
   1. Declarar un arreglo por cada uno de los tipos de datos primitivos (byte, Byte, char, Character, short, Short, int, Integer, long, Long, float, Float, double, Double, Object, String.
   2. Definir cada uno de los arreglos anteriores con tamaño de 5 elementos
   3. Asignar valores a cada arreglo usando la notación de posiciones o índices
   4. Mostrar los valor es cada Arreglo
   5. Declarar y definir otro arreglo de tipo String en una variable llamada dataStructs, e inicializarlo con los siguientes datos, usando la sintaxis de llaves.
      1. Listas
      2. Colas
      3. Pilas
      4. Mapas
      5. Conjuntos
   6. Declarar y definir otro arreglo de tipo String en una variable llamada características, con 5 elementos.
   7. Ingresar los siguientes datos por teclado:
      1. Simples, Circulares y Enlazadas
      2. Primero en entrar, Primero en Salir. Último en entrar, Primero en salir
      3. Primero en entrar, Último en salir
      4. Parejas de clave y valor
      5. Elementos no repetidos
   8. Mostrar los elementos del arreglo dataStructs con su equivalente elemento del arreglo características.
   9. Declarar un arreglo de enteros en una variable llamada ventas,
   10. Inicializar el arreglo ventas con una cantidad aleatoria de elementos.
   11. Llenar el arreglo ventas con valores aleatorios entre 1 y 1000
   12. Mostrar la cantidad de ventas
   13. Mostrar el valor de cada una de las ventas
   14. Mostrar el total de ventas
   15. Mostrar el promedio de las ventas.
   16. Diseñar una función que permita buscar la venta más alta e invocar la función para probarla.
   17. Diseñar una función que permita buscar la venta más baja e invocar la función para probarla.
   18. Diseñar una función que permita obtener el total de las ventas e invocar la función para probarla.
   19. Diseñar una función que permita obtener el promedio de las ventas e invocar la función para probarla.
   20. Diseñar una función que devuelva la venta cuyo valor se acerque más al valor que le sigue al promedio e invocar la función para probarla.
   21. Diseñar una función que permita ordenar las ventas de forma descendente e invocar la función para probarla.
   22. Diseñar una función que permita ordenamiento que permita ordenar las ventas de forma ascendente e invocar la función para probarla.
   23. Diseñar una función que permite desordenar las ventas e invocar la función para probarla.
   24. Diseñar una función que permita ordenar las ventas primero las partes y luego en impares e invocar la función para probarla.
   25. Declarar e inicializar una matriz de 5 x 8 de tipo Object en una variable llamada lenguajes
   26. A la primera línea de la matriz se de debe asignar los siguientes valores en ese orden:

[NOMBRE, AÑO, AUTOR, DETALLES, FRAMEWORKS]

* 1. Realizar una función llamada **infoLenguajes** que permita llenar la matriz con datos por teclado a partir de la segunda fila.
  2. Realizar una función que permita mostrar los datos de la matriz según la siguiente logia:
     1. La función de debe recibir como parámetro un arreglo con 5 elementos de tipo boolean
     2. Al invocar la función se debe mostrar solo la información de la matriz de acuerdo al valor (false o true) que tenga el arreglo que recibe como parámetro. Por ejemplo:

boolean **info**[] = **{**true**,** false**,** true**,** true**,** false**}**;

**infoLenguajes**(info)

Debe mostrar los nombres de los lenguajes, los autores, los frameworks