



PROCESO DE GESTIÓN DE FORMACIÓN PROFESIONAL INTEGRAL

FORMATO GUÍA DE APRENDIZAJE

IDENTIFICACIÓN DE LA GUÍA DE APRENDIZAJE

- **Denominación del Programa de Formación:** TGO Análisis y Desarrollo de Sistemas de Información
- **Código del Programa de Formación:** 228106 V102
- **Nombre del Proyecto:** Construcción de un sistema de información que cumpla con los requerimientos del cliente en procesos que se lleven a cabo en el sector productivo del departamento de Caldas
- **Fase del Proyecto:** ANÁLISIS
- **Actividad de Proyecto:** Seleccionar la alternativa de solución que cumpla con los requerimientos establecidos por el cliente
- **Competencia:** Analizar los requisitos del cliente para construir el sistema de información.
- **Resultados de Aprendizaje Alcanzar:** Interpretar el informe de requerimientos, para determinar las necesidades tecnológicas en el manejo de la información, de acuerdo con las normas y protocolos establecidos en la empresa
- **Duración de la Guía:** 10 horas

2. PRESENTACIÓN

Los arreglos son estructuras de datos homogéneas (todos sus datos son del mismo tipo) que permiten almacenar un determinado número de datos bajo un mismo identificador, para luego referirse a los mismos utilizando uno o más subíndices.

Los arreglos pueden pensarse como vectores, matrices, etc.. En esta guía los aprendices aprenderán a utilizar los diferentes tipos de arreglos en los ejercicios a realizar.



3. FORMULACIÓN DE LAS ACTIVIDADES DE APRENDIZAJE

- **Descripción de la(s) Actividad(es)**
 - **Actividades de aprendizaje**
 - Aplicar fundamentos de programación en el desarrollo de aplicaciones con programación estructurada.
 - Utilizar arreglos en DART.
 - Utilizar estructuras cíclicas en la solución de ejercicios algorítmicos con programación estructurada.
 - Reconocer la sintaxis propia del lenguaje de programación Dart para la construcción de aplicaciones tipo consola.
 - Construir estructuras secuenciales, condicionales y cíclicas empleando el lenguaje de programación Dart.
 - Emplear las convenciones de codificación propias del lenguaje de programación Dart para el desarrollo de aplicaciones de tipo consola
 - Utilizar estructuras de datos relacionadas con arreglos para resolver los problemas

Arreglo

Un *Arreglo* (en inglés Array) es una estructura de datos que almacena bajo el mismo nombre (variable) a una colección de datos del mismo tipo (int, float, double, String, etc.).

En **Dart**, un arreglo (conocido como lista) es una colección ordenada de elementos. Las listas en Dart son similares a los arrays en otros lenguajes de programación y pueden contener elementos del mismo tipo o de tipos diferentes. Dart proporciona una clase llamada List que permite trabajar con estas colecciones de elementos.

Las listas o arreglos son una parte fundamental del lenguaje Dart y en general de todos los lenguajes de programación y se usan ampliamente para manejar colecciones de datos de manera eficiente y conveniente.

Los arreglos se caracterizan por:

- Almacenan los elementos en posiciones contiguas de memoria
- Tienen un mismo nombre de variable que representa a todos los elementos. Para hacer referencia a esos elementos es necesario utilizar un índice que especifica el lugar que ocupa cada elemento dentro del archivo.

Los arreglos pueden ser:

1-Unidimensionales: solo tiene una sola dimensión, una fila y varias columnas (Vectores)

2-Bidimensionales: tablas o matrices.

3-Multidimensionales: de 3 o más dimensiones.



Arreglos/Listas unidimensionales

Síntaxis, Declaración y Definición:

```
1 void main(List<String> args) {  
2     // Lista vacía  
3     List<int> numeros1 = [];  
4  
5     // Lista con elementos iniciales  
6     List<int> numeros2 = [1, 2, 3, 4, 5];  
7  
8     // Lista de tipo dinámico (puede contener cualquier tipo de elemento)  
9     List<dynamic> elementos = [1, 'dos', 3.0, true];  
10 }
```

```
1 // Lista con elementos iniciales  
2 List<int> numeros = [3, 7, 8, 4, 5, 9, 90, 12, 17, 1];
```

Con esta declaración estamos creando una colección de 10 números enteros

3	7	8	4	5	9	90	12	17	1
0	1	2	3	4	5	6	7	8	9

Por regla, todo arreglo o lista, comienza en la posición 0 y llega hasta n, accedemos a sus valores por medio del índice que se coloca en la posición del valor referido.

Acceso a Elementos:

Los elementos en una lista se pueden acceder mediante índices, donde el índice comienza en 0. Siempre que se quiera hacer referencia a uno de los elementos del arreglo se debe indicar la posición del subíndice.

Si que queremos escribir el valor de posición 6 tendremos que escribir:

```
1 print(numeros[6]); //El dato contenido en la posición 6 del vector es 90
```



```
1 List<String> frutas = ['manzana', 'banana', 'cereza'];
2 print(frutas[0]); // SALIDA: manzana
3 print(frutas[2]); // SALIDA: cereza
```

Modificar Elementos: se pueden modificar los elementos de una lista asignando un nuevo valor a una posición específica.

Para asignar valores a la matriz podemos, asignar a elementos de matriz individuales, utilizando el número de índice.

```
1 List<int> numeros = [1, 2, 3];
2 numeros[1] = 10;
3 print(numeros); // SALIDA: [1, 10, 3]
```

Métodos Comunes

Dart ofrece varios métodos útiles para manipular listas, como `add`, `remove`, `length`, entre otros.

```
1 List<int> numeros = [1, 2, 3];
2 numeros.add(4); // Agrega un elemento al final de la lista
3 print(numeros); // SALIDA: [1, 2, 3, 4]
4
5 numeros.remove(2); // Elimina el elemento 2
6 print(numeros); // SALIDA: [1, 3, 4]
7
8 print(numeros.length); // SALIDA: 3 (número de elementos en la lista)
```



Ejemplo de ingreso y visualización de valores en una lista:

```
1 List<int> listaNumeros = [];  
2     int num;  
3  
4     // Ingreso de números a la Lista  
5     for (int i = 0; i < 10; i++) {  
6         print("Digite número ${i + 1}");  
7         num = int.parse(stdin.readLineSync());  
8         listaNumeros.add(num); // Usamos add() para agregar elementos a la lista  
9     }  
10  
11     // Visualización de números del vector  
12     print('*' * 30); // Usamos '*' * 30 para imprimir 30 asteriscos  
13     for (int i = 0; i < 10; i++) {  
14         print("Número ${i+1}: ${listaNumeros[i]}");  
15     }
```



Arreglos/Listas de 2 dimensiones (Bidimensional)

También se les denomina matrices o tablas. Un arreglo o lista bidimensional es una tabla que ahora tiene filas y columnas, donde las filas indican la primera dimensión y las columnas la segunda dimensión.

En Dart, los arreglos de dos dimensiones se pueden crear utilizando listas anidadas. Una lista bidimensional es una lista que contiene otras listas como sus elementos.

La tabla que se muestra, nos representa un arreglo de 2 dimensiones con 3 filas y 4 columnas, el código para declarar este arreglo sería:

	0	1	2	3
0				
1				
2				

La referencia a un determinado elemento de la matriz, requiere el empleo de un primer subíndice que indica la fila y el segundo que indica la columna. Ambos subíndices deberán de ser de tipo entero.

Por ejemplo si quisiéramos guardar el valor de 30 en la fila 4 columna 3 el código en PSeInt sería:

```
numeros[3,2] = 30;
```

Declaración vacía y luego Inicialización

```
1 List<List<int>> matriz = [];  
2  
3 // Inicialización con valores  
4 for (int i = 0; i < 3; i++) { // FILAS  
5     matriz.add([]);  
6     for (int j = 0; j < 3; j++) { // COLUMNAS  
7         matriz[i].add(0); // Inicializa con 0  
8     }  
9 }
```



Declaración con Valores Iniciales:

```
1 List<List<int>> matriz = [  
2     [1, 2, 3],  
3     [4, 5, 6],  
4     [7, 8, 9]  
5 ];
```

Acceso y Modificación de Elementos

Acceder y modificar elementos en una lista bidimensional es similar a hacerlo en una lista unidimensional, pero se requieren **dos índices**: uno para la fila y otro para la columna.

```
1 List<List<int>> matriz = [  
2     [1, 2, 3],  
3     [4, 5, 6],  
4     [7, 8, 9]  
5 ];  
6  
7 // Acceso a elementos  
8 print(matriz[0][0]); // SALIDA: 1  
9 print(matriz[1][2]); // SALIDA: 6  
10  
11 // Modificación de elementos  
12 matriz[2][1] = 10;  
13 print(matriz[2][1]); // SALIDA: 10
```

Ejemplo de ingreso y visualización de elementos al arreglo multidimensional

A continuación un ejemplo completo donde se crea una matriz de 3x3 a partir de la entrada del usuario y luego se imprime.



Ejemplo Dart:

```
1 List<List<int>> matriz = [];
2
3 // Lectura de la matriz desde el usuario
4 for (int i = 0; i < 3; i++) {
5     List<int> fila = [];
6     for (int j = 0; j < 3; j++) {
7         print('Digite el elemento [$i][$j]:');
8         int num = int.parse(stdin.readLineSync());
9         fila.add(num);
10    }
11    matriz.add(fila);
12 }
13
14 // Visualización de la matriz
15 print('Matriz:');
16 for (int i = 0; i < 3; i++) {
17     for (int j = 0; j < 3; j++) {
18         stdout.write('${matriz[i][j]} ');
19     }
20     print('\n');
21 }
```

Ejemplo C#

```
for (int f = 0; f < num.GetLength(0); f++)
{
    for (int c = 0; c < num.GetLength(1); c++)
    {
        Console.WriteLine("Ingrese número:");
        num[f, c] = int.Parse(Console.ReadLine());
    }
}

for (int f = 0; f < num.GetLength(0); f++)
{
    for (int c = 0; c < num.GetLength(1); c++)
    {
        Console.WriteLine(num[f, c]);
    }
}
```




Actividad 01

Realizar los siguientes ejercicios en **DART** de Arreglos Unidimensionales y Bidimensionales

1. Arreglos Unidimensionales

1. Calcular el promedio de 50 valores almacenados en un vector. Determinar además cuantos son mayores que el promedio, imprimir el promedio, el número de datos mayores que el promedio y una lista de valores mayores que el promedio.
2. Llenar dos vectores A y B de 10 elementos cada uno, sumar el elemento uno del vector A con el elemento uno del vector B y así sucesivamente hasta 10, almacenar el resultado en un vector C, e imprimir el vector resultante.
3. Almacenar 15 números en un vector, imprimir cuantos son ceros, cuántos son negativos, cuantos positivos. Imprimir además la suma de los negativos, la suma de los ceros y la suma de los positivos.
4. Almacenar 8 números en un vector, almacenarlos en otro vector en orden inverso al vector original e imprimir el vector resultante.
5. Diseñe un algoritmo que lea un número cualquiera y lo busque en el vector X, el cual tiene almacenados 12 elementos. Escribir la posición donde se encuentra almacenado el número en el vector o el mensaje "NO" si no lo encuentra.
6. Diseñe un algoritmo que lea dos vectores A y B de 7 elementos cada uno y multiplique el primer elemento de A con el último elemento de B y luego el segundo elemento de A por el sexto elemento de B y así sucesivamente hasta llegar al séptimo elemento de A por el primer elemento de B. El resultado de la multiplicación almacenarlo en un vector C. Mostrar el resultado.
7. Diseñar un algoritmo que lea una palabra o una frase en un arreglo de caracteres y determinar si esa palabra o frase es palíndroma.
8. Consultar el método de ordenación por burbuja y ordenar un arreglo entero de 10 elementos.
9. Una agencia de seguros desea obtener un reporte al final del día de sus n empleados para determinar cuál fue el empleado que obtuvo el mayor sueldo en base a sus ventas y comisiones, se registrará el nombre del empleado la edad y el sueldo que obtuvo al final del día. Desarrollar un programa que pida al usuario el nombre de los n empleados, su edad y el sueldo para generar un reporte que lo muestre en pantalla de la siguiente manera: el nombre del empleado, edad, sueldo, el empleado con mayor sueldo, el sueldo del empleado que gana más y su edad.
Debe declarar un arreglo de enteros para la edad.

Debe declarar un arreglo de reales para el sueldo.

Debe declarar un arreglo de cadena para el nombre



2. Arreglos Bidimensionales

1. Hacer un algoritmo que almacene números en una matriz de 5×6 . Imprimir la suma de los números almacenados en la matriz.
2. Hacer un algoritmo que llene una matriz de 10×10 y determine la posición [fila, columna] del número mayor almacenado en la matriz. Los números son diferentes.
3. La compañía manufacturera Monte Real fábrica 5 artículos diferentes y se trabajan tres turnos por día, la compañía necesita obtener un reporte al final del día del total de la producción por artículo y el total de la producción por turno. Desarrollar un programa que pida al usuario el nombre de cada artículo y la producción que se hizo en cada uno de los tres turnos del día y genere un reporte al final del día mostrado en pantalla el total de la producción por artículo, el total de la producción por turno y el artículo con mayor producción.

Articulo \ Turno	Turno 1	Turno 2	Turno 3
Articulo 1	30	40	20
Articulo 2	10	12	15
Articulo 3	8	10	7
Articulo 4	25	30	30
Articulo 5	12	20	10

4. La escuela Ignacio Barrera, desea que le desarrolles un juego para sus alumnos de primaria. El juego se llama "cuadro mágico", un cuadro mágico es aquel en que la suma de cada fila, la suma de cada columna y la suma de las diagonales es exactamente igual. El cuadro mágico es de dos dimensiones. Desarrollar un programa que pida al usuario las dimensiones deseadas para el cuadro mágico (un arreglo de dos dimensiones) en forma de cuadro y luego le pida los datos para llenar el arreglo. Después el programa debe calcular la suma de cada fila, de cada columna y de cada diagonal del cuadro, si todas las sumas son iguales entonces le deberás de indicar por medio de un mensaje "Felicidades creaste un cuadro mágico", de lo contrario le deberás de decir el siguiente mensaje: "Modifica los números en el cuadro para que lo hagas mágico".
5. Hacer un algoritmo que solicite los valores para llenar dos matrices de 3×3 y las multiplique, almacenando el resultado en otra matriz de 3×3 .
Se debe consultar el método para multiplicar matrices.

Como resultado de la ejecución del programa, además del mensaje correspondiente, deberás mostrar en pantalla el arreglo de entrada (el cuadro), la suma de cada renglón y la suma de cada columna.

- **Ambiente Requerido**

Ambiente de SISTEMAS con conexión eléctrica e internet

- **Materiales**

- Computadores (30)
- Sillas (3)
- Televisor (1)
- Resma tamaño carta (1)
- Marcadores (3)
- Lápiz (1)



- Lapicero (1)

4. ACTIVIDADES DE EVALUACIÓN

Evidencias de Aprendizaje	Criterios de Evaluación	Técnicas e Instrumentos de Evaluación
Evidencias de Conocimiento: Evidencias de Desempeño: Asistencia y participación activa en las diferentes actividades propuestas Evidencias de Producto: Respuestas y procedimiento de los talleres realizados	Crea la base de datos en el motor de base de datos seleccionado, siguiendo especificaciones técnicas del informe, según normas y protocolos de la empresa.	Observación: EXC_D_01 Valoración del Producto: EXC_P_01

5. GLOSARIO DE TÉRMINOS

Sistema de información: es un conjunto de elementos orientados al tratamiento y administración de datos e información, organizados y listos para su uso posterior, generados para cubrir una necesidad o un objetivo

Sistema operativo – Es un conjunto de programas que sirven para manejar un ordenador.

Software - El conjunto de programas, procedimientos y documentación asociado a un sistema informático.

Javascript: es un lenguaje de programación del lado del cliente que se utiliza con frecuencia en diseño WEB para generar efectos más complejos que no se puedan alcanzar usando HTML.

HTML: Siglas de las palabras inglesas: Hypertext Markup Language. Es decir, lenguaje de marcado de hipertexto. Lenguaje informático para crear páginas web. Conjunto de etiquetas o instrucciones que permiten estructurar el contenido de una web e incluir los hipervínculos o enlaces a otras páginas. Este lenguaje lo inventó en 1991 el Doctor Berners-Lee del CERN en Suiza.

HTTPS: Siglas de las palabras inglesas: HyperText Transfer Protocol Secure o versión segura del protocolo HTTP. Es el protocolo empleado para la transferencia de ficheros HTML cifrados que puedan contener información confidencial.



HTTP: siglas de las palabras inglesas: Hypertext Transfer Protocol. A saber en español: Protocolo de Transmisión de Hipertexto. Protocolo estándar de transferencia de hipertexto. Es decir: el protocolo de comunicaciones en el que está basado la Word Wide Web.

Script: es un archivo de órdenes o archivo de procesamiento por lotes. Es un programa usualmente simple, que por lo regular se almacena en un archivo de texto plano.

MySQL: es un sistema de gestión de bases de datos de código abierto que, junto con PHP, permite darle a las páginas web cierto dinamismo, es decir, disponer de manera adecuada los datos solicitados por los navegadores. Es un sistema multiplataforma y su uso está tan extendido en las bases de datos que podría considerarse un estandar.

SEO (Search Engine Optimisation) Optimización en buscadores: técnica utilizada para asegurar que una página Web es compatible con los motores de búsqueda y así tener la posibilidad de aparecer en las posiciones más altas en los resultados de búsqueda.

Diseño web adaptable (responsive web design): se llama así al diseño web de aquellas páginas que se adaptan al tamaño de la pantalla o ventana en que se despliegan, por medio del uso de, idealmente, un solo documento HTML y un solo documento CSS. Esto permite hacer una sola página web para smartphones, phablets, tablets y PC.

Diagrama o Modelo Entidad Relación (DER): denominado por sus siglas en inglés, E-R "Entity relationship", o del español DER "Diagrama de Entidad Relación") es una herramienta para el modelado de datos que permite representar las entidades relevantes de un sistema de información así como sus interrelaciones y propiedades

Bases de Datos (BD): es un banco de información que contienen datos relativos a diversas temáticas y categorizados de distinta manera, pero que comparten entre sí algún tipo de vínculo o relación que busca ordenarlos y clasificarlos en conjunto.

6. REFERENTES BIBLIOGRÁFICOS

- Documentos técnicos relacionados en la plataforma
- <https://www.php.net>
- <http://formaentic.weebly.com/>

7. CONTROL DEL DOCUMENTO

	Nombre	Cargo	Dependencia	Fecha
Autor (es)	Julian Salazar Pineda	Instructor	Centro de Procesos Industriales y Construcción	2 de Mayo de 2024

8. CONTROL DE CAMBIOS (diligenciar únicamente si realiza ajustes a la guía)



	Nombre	Cargo	Dependencia	Fecha	Razón del Cambio
Autor (es)					