

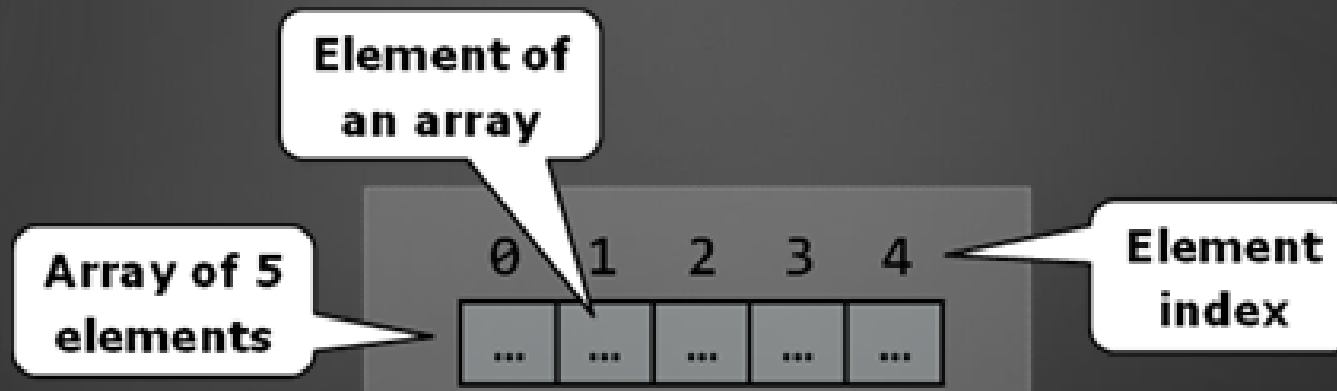


Arreglos Unidimensionales

ING. EDGAR GERARDO SALINAS GURRIÓN

¿Qué es un arreglo?

- ▶ Es una simple secuencia de objetos o datos primitivos que son todos del mismo tipo.
 - ▶ Están empaquetados juntos bajo un mismo identificador.



Declaración de un arreglo

- ▶ Para declarar un arreglo se debe de especificar primero el **tipo de datos** que almacenará seguido de **corchetes vacíos** y luego el **identificador**.
- ▶ Ejemplos:

The diagram shows two lines of code: `int [] numeros;` and `string [] args;`. A callout box labeled 'Tipo de Dato' points to the `int` in the first line. Another callout box labeled 'Identificador' points to the `numeros` in the first line.

```
int [] numeros;  
string [] args;
```

Creación de Arreglos

- ▶ Como todos los objetos para crearlos es necesario llamar a un constructor, la manera de hacerlo es:

- ▶ `new tipoDeDato[tamaño];`

- ▶ Ejemplos:

```
int [] numeros = new int [6];  
numeros;
```

**Tamaño
del arreglo**

Creación e Inicialización de Arreglos

- ▶ Existe otra manera de crear un arreglo y además inicializarlo con valores definidos
 - ▶ Para hacerlo simplemente se pasa una lista de elementos entre llaves y separado cada elemento con comas, entonces se creará un arreglo de tamaño igual al numero de elementos que se pasaron en la lista
- ▶ Ejemplo:

```
int [] numeros = {8,2,6,4,9,5};  
numeros;
```

**Lista de
elementos**

Acceder al arreglo

- Para acceder a un elemento de los contenidos en el arreglo es necesario indicar la posición que queremos acceder, siendo 0 la posición del primer elemento:

```
int [] numeros = {8,2,6,4,9,5};  
numeros[3]=10;
```

- El arreglo quedaría así:

```
{8,2,6,10,9,5}
```

Arreglos como parámetros

- ▶ Para definir que una función va a recibir un arreglo como parámetro solo es necesario indicar el tipo de datos que va a contener el arreglo seguido de corchetes y luego el identificador
- ▶ Ejemplo:

```
public void hola(int[] numeros){  
    System.out.println(numeros[0]);  
}
```

- ▶ Se imprimirá el primer elemento del arreglo.

Regresar un arreglo

- ▶ Si queremos que un método regrese un arreglo entonces se debe de especificar el tipo de dato que contiene el arreglo y luego los corchetes vacíos.

- ▶ Ejemplo:

```
public String[] regresa()
```

Tipo de dato
del arreglo

- ▶ Este método regresa un arreglo de Strings.

Excepcion

- ▶ Si tenemos un arreglo de tamaño n , la posición máxima a la que podemos acceder es $n-1$.
- ▶ Por lo tanto si intentamos acceder a cualquier posición mayor o igual a n , la instrucción arrojará la excepción *ArrayIndexOutOfBoundsException*

Recorrer un arreglo

- Comúnmente se utiliza el siguiente código para recorrer un arreglo :

```
for(int i = 0; i< arreglo.length; i++){  
    arreglo[i]; //Poner la operacion a realizar  
}
```



Arreglos Multidimensionales

ING. EDGAR GERARDO SALINAS GURRIÓN

Arreglo multidimensional

- Java no soporta los arreglos multidimensionales pero en su lugar soporta los **arreglos de arreglos** lo que en la practica nos sirve para lo mismo.

	Column 0	Column 1	Column 2	Column 3
Row 0	a[0][0]	a[0][1]	a[0][2]	a[0][3]
Row 1	a[1][0]	a[1][1]	a[1][2]	a[1][3]
Row 2	a[2][0]	a[2][1]	a[2][2]	a[2][3]

Arreglos Bidimensionales

- ▶ Los arreglos bidimensionales comúnmente son utilizados para representar tablas que están compuestos por **filas** y **columnas**.
- ▶ El primer identificador es para las filas y el segundo para las columnas.

```
arreglo[fila][columna];
```

Inicialización

- ▶ Al igual que los arreglos unidimensionales los arreglos multidimensionales pueden inicializarse en la declaración de manera que se anidan los arreglos.
- ▶ Ejemplo:

```
int[] [] b = {{1,2},{3,4}};
```

	[][0]	[][1]
[0][]	1	2
[1][]	3	4

Diferente longitud de filas

- La manera en que Java representa los arreglos multidimensionales es muy flexible, lo que permite tener filas de diferentes longitudes.

```
int[][] b = {{1,2},{3,4,5}};
```

1	2	
3	4	5

Creación con *new*

- ▶ Al igual que los arreglos unidimensionales es posible definir la longitud del arreglo sin tener que poner cuales son los elementos que van a ir en el arreglo.

Ejemplos

► Ejemplo 1:

```
int[][] b;  
b= new int[2][3];|
```



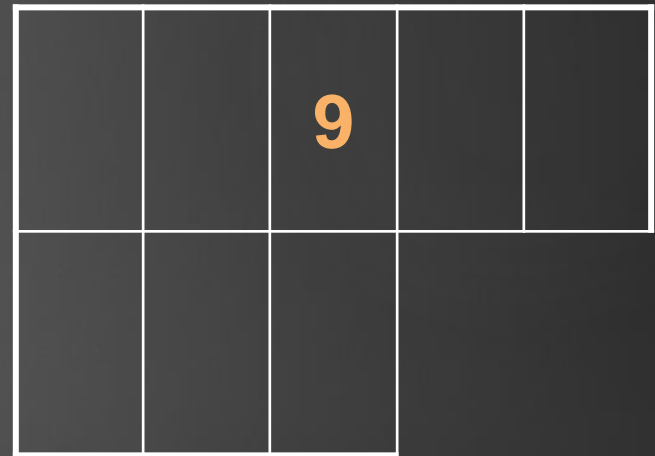
► Ejemplo 2:

```
int[][] c;  
c=new int[2][ ];  
c[0]=new int[5];  
c[1]=new int[3];
```



Acceder a un elemento del arreglo

```
int[ ][ ] c;  
c=new int[2][ ];  
  
c[0]=new int[5];  
c[1]=new int[3];  
c[0][2]=9;
```



Recorrer un arreglo bidimensional

```
for(int i=0; i<arreglo.length; i++){  
    for(int j=0; j<arreglo[ i ].length; j++){  
        arreglo[ i ][ j ];  
    }  
}
```

Actividad

Implementar los siguientes métodos en la clase `PruebaArreglos`:

- ▶ `int suma(int[] arreglo)`
 - ▶ `int maximo(int[] arreglo)`
 - ▶ `int minimo(int[] arreglo)`
 - ▶ `double promedio(int[] arreglo)`
- ▶ Hacer un método llamado *imprime* que reciba un arreglo bidimensional de Strings y que imprima en pantalla los elementos del arreglo.

"cat"	"dog"	"lizard"
"bear"	"shark"	"snake"



cat dog lizard
bear shark snake

Actividad

- ▶ Escribir el método `main` el cual va a recibir parámetros de la línea de comandos.
- ▶ Los 2 primeros parámetros van a indicar el numero de filas y columnas y los demás parámetros van a indicar las palabras que contendrá el arreglo.

```
d:\>java EjemploArreglos 2 3 A B C D E F
A B C
D E F
d:\>
```