



UNIVERSIDAD AUTÓNOMA DE
SINALOA

Facultad de Informática Culiacán

Arreglos en Java

Instructor:

MC. Gerardo Gálvez Gámez

gerardo.galvez@uas.edu.mx



Diciembre de 2016



Competencias

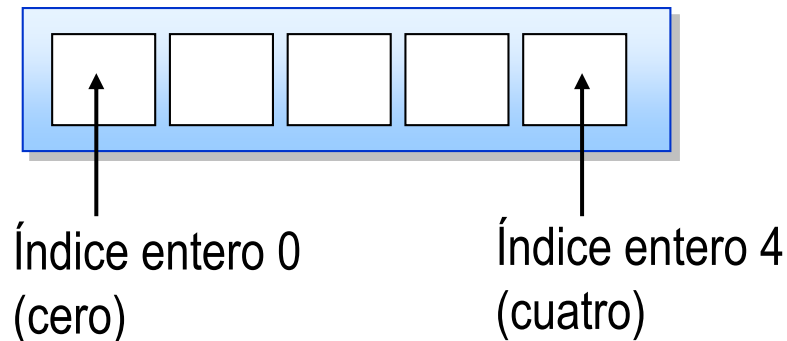
Al final de este módulo, los estudiantes serán capaces de:

- Describir los distintos tipos de Arreglos.
- Crear arreglos.
- Usar arreglos.

¿Qué es un Arreglo?

Una arreglo es una secuencia de elementos

- Todos los elementos de un arreglo son del mismo tipo
- Las estructuras pueden tener elementos de distintos tipos
- Se accede a elementos individuales usando índices enteros (acceso aleatorio)



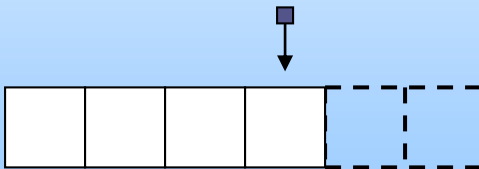
Tipos de arreglos en java

Notación para **declarar** arreglos, según su tipo:

```
long[] Arreglo;
```

Unidimensionales o Lineales

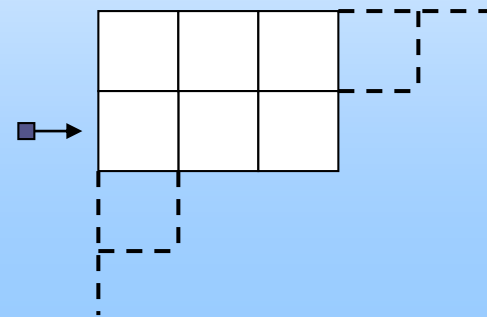
Un solo índice asociado con cada elemento



```
int[][] Matriz;
```

Bidimensionales

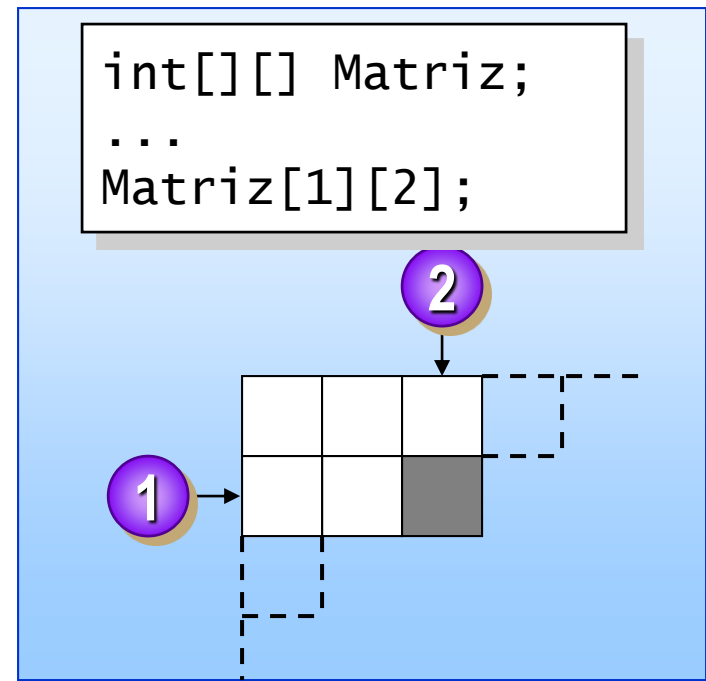
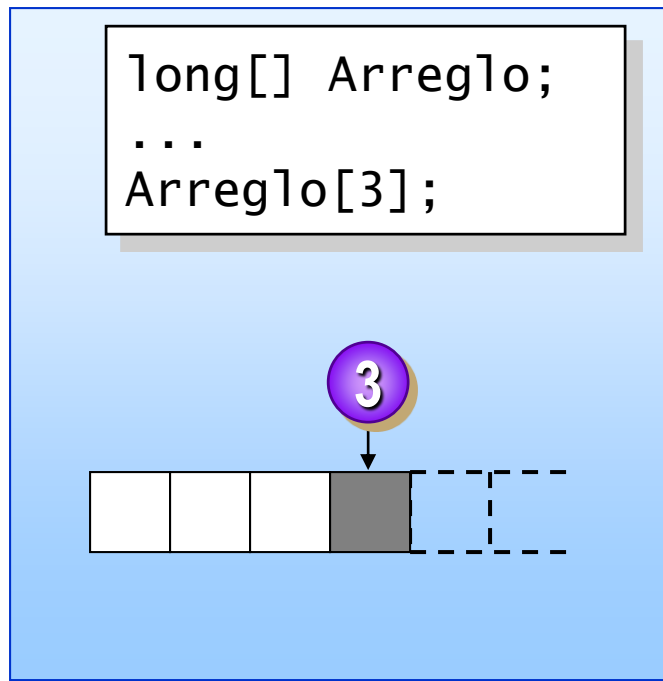
Dos o más índices asociados con cada elemento



Acceso a los elementos de un arreglo

Se indica un índice entero para cada dimensión

- Los índices se cuentan a partir de cero

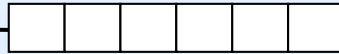




Conocer la longitud de un arreglo

Se comprueba con la propiedad **Length**

Arreglo



```
int filas = Arreglo.length;
```

Matriz



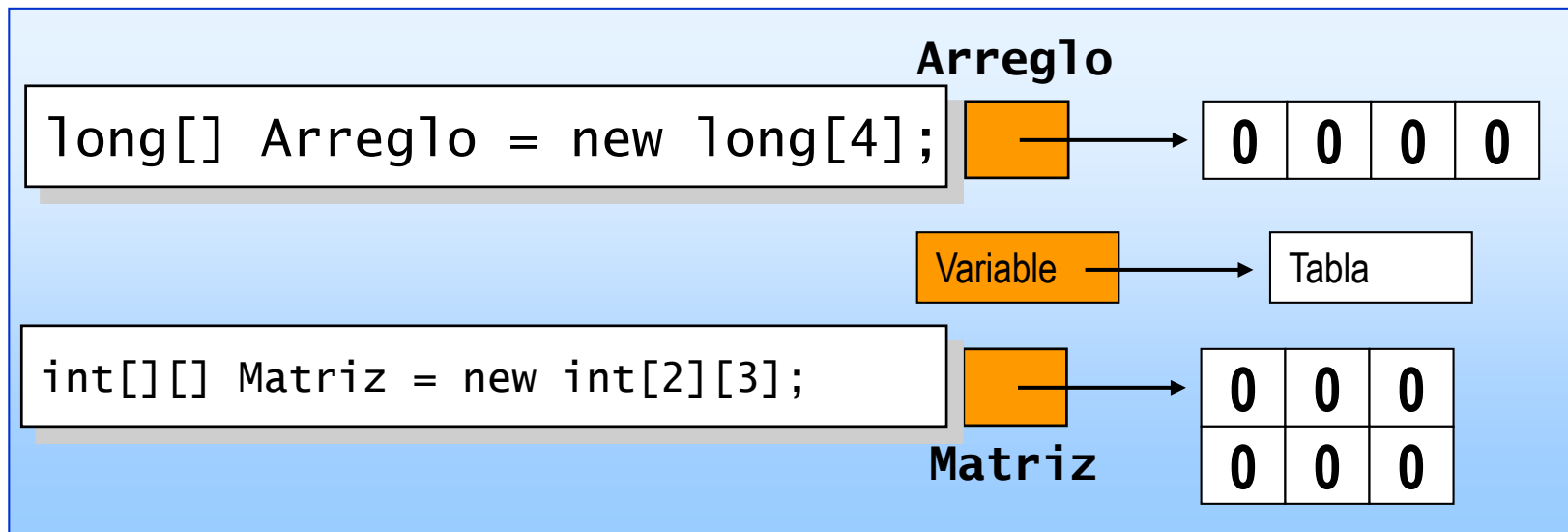
```
int Filas = matriz.length;
```

```
int Columnas = matriz[0].length;
```

Creación de un arreglo

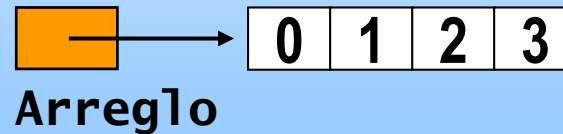
¡Declarar una variable de arreglo no es lo mismo que crear un arreglo!

- Para crear el arreglo explícitamente hay que usar **new**
- El valor implícito por defecto de los elementos de un arreglo es cero

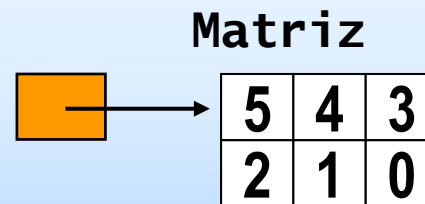


Inicialización de los elementos de un arreglo

```
long[] Arreglo = {0, 1, 2, 3};
```



```
int[][] Matriz = {  
    {5, 4, 3},  
    {2, 1, 0}  
};
```





Creación de un arreglo de tamaño calculado

```
class Example {  
    static void main( ) {  
        int[] Arreglo = CrearArreglo();  
        ...  
    }  
    static int[] CrearArreglo(){  
        int Tamaño;  
        Scanner Leer= new Scanner(System.in);  
        System.out.print("Tamaño del arreglo:");  
        Tamaño=Leer.nextInt();  
        int[] Arreglo = new int[Tamaño];  
        return Arreglo;  
    }  
}
```



Devolución de Arreglos desde métodos

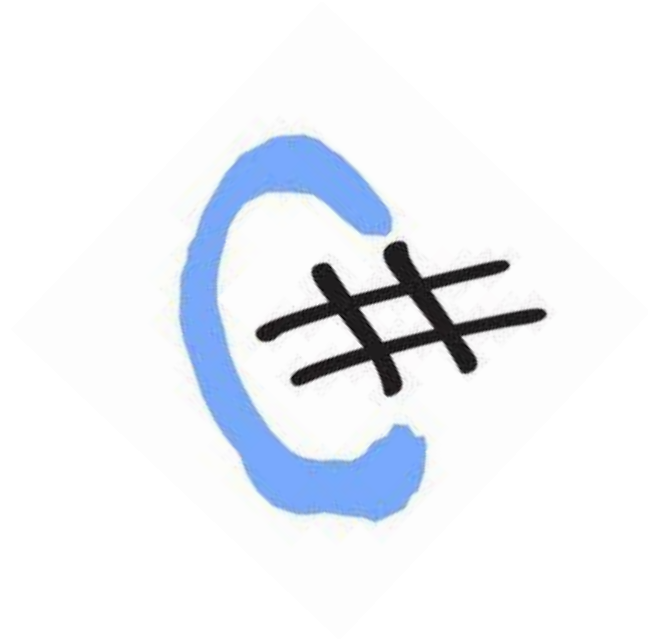
```
class Example {  
    static void main( ) {  
        int[] Arreglo = CrearArreglo(42);  
        ...  
    }  
    static int[] CrearArreglo(int Tamaño) {  
        int[] Arreglo = new int[Tamaño];  
        return Arreglo;  
    }  
}
```



Paso de Arreglos como parámetros

```
class Example2 {
    static void main( ) {
        int[] Arreglo = {10, 9, 8, 7};
        Metodo(Arreglo);
        System.out.print(Arreglo[0]);
    }

    /*******método*****
    static void Metodo(int[] pArreglo) {
        pArreglo[0]++;
    }
}
```





Problema #1

- Codificar el siguiente pseudocódigo, que permite:
 - Leer 10 números enteros proporcionados por el usuario, para:
 - Imprimir aquellos números que son superiores a la media de los números proporcionados.



Propuesta Algoritmo Modular

//Objetivo: Leer valores numéricos, guardarlos en un arreglo e imprimir aquellos que son superiores a la media.

//Programador: MC. Gálvez

//Fecha: ____de Noviembre de 2015



INICIO

//Definición de Constantes y Variables Globales

PRINCIPAL ()

INICIO

//Definición de Constantes y Variables Locales

CONST ENTERO Tamaño=10

ENTERO Numeros[Tamaño]

REAL Media=0

LecturaDatos(Numeros,Tamaño)

Media= CalcularMedia(Numeros,Tamaño)

ImprimirNumerosSuperiores(Numeros,Tamaño,Media)

FIN



Propuesta Algoritmo Modular

SINVALOR LecturaDatos(ENTERO Numeros[], ENTERO Tamaño)

INICIO

ENTERO Indice

DESDE(Indice=0;Indice<Tamaño,Indice=Indice+1)

IMPRIMIR "Proporciona el valor de la celda [", Indice+1, "]:"

LEER Numeros[Indice]

FIN_DESDE

IMPRIMIR "Fin de lectura..."

FIN

REAL CalcularMedia (ENTERO Numeros[], ENTERO Tamaño)

INICIO

ENTERO Indice,Suma=0,Media

DESDE(Indice=0;Indice<Tamaño,Indice=Indice+1)

Suma=Suma+ Numeros[Indice]

FIN_DESDE

Media=Suma/Tamaño

FIN





Propuesta Algoritmo Modular

```
SINVALOR   ImprimirNumerosSuperiores  (ENTERO  Numeros[],  
      ENTERO Tamaño, REAL Media)  
INICIO  
    ENTERO Indice  
    DESDE(Indice=0;Indice<Tamaño,Indice=Indice+1)  
      SI(Numero[Indice]>Media)  
        IMPRIMIR  Numeros[Indice]  
      FIN_SI  
    FIN_DESDE  
    IMPRIMIR "Fin de Impresión..."  
FIN
```





Problema #2

- Codificar al lenguaje el pseudocódigo Obtenido en la materia de algoritmia, que:
 - Permitía leer 10 valores enteros y guardarlos en un arreglo, luego
 - a) Imprimirlos en forma inversa a como fueron proporcionados.
 - b) Imprimir el mayor
 - c) Imprimir el menor
 - d) Determinar si un elemento X se encuentra en el arreglo.



Problema #3

- Codificar el pseudocódigo Obtenido en la materia de algoritmia que permitía:
 - Leer y guardar datos numéricos en un arreglo bidimensional de 2 x 3, y determinar:
 - Cual es el número mayor
 - Cual es el promedio
 - Cuantos números son mayores al promedio
 - Imprimir los valores por renglón
 - Imprimir los valores por columna



Preguntas?

