###### FdA - Tecnología Multimedial 1

### Arreglos (Array)

#### 1. Bibliografía

* **Processing | Un lenguaje al alcance de todos**
  + Unidad 29 - Datos: Arrays (Página 151)
  + Unidad 30 - Imagen: Animación (Página 160)
* **Processing: Guía Rápida.**
  + Arrays (Página 7)

#### 2. Conceptos básicos:

* Al igual que las variables, un Arreglo (Array en inglés) es un espacio en memoria al que puedo acceder durante la ejecución de mi aplicación para almacenar o consultar un dato.
* A diferencia de las variables, un Arreglo permite almacenar VARIOS valores.
* Los datos dentro de un Arreglo se organizan en casillas numeradas.
* Aclaraciones:
  + Es común que en otros lenguajes esas casillas también puedan tener un nombre, pero nosotros vamos a referirnos a cada casilla por su índice numérico.
  + En algunos casos puede resultar mejor utilizar ArrayList, pero no será lo que veremos en esta sección.

3. Algunos usos comunes con Arreglos:

La posibilidad de guardar un conjunto de elementos se aplica a casos como los siguientes:

* Cadenas de texto para nombrar elementos, por ejemplo "Lunes", "Martes", "Miércoles", "Jueves", "Viernes", "Sábado", "Domingo"; o los párrafos de una historia contada en secuencia.
* Valores numéricos, por ejemplo tamaños, coordenadas, colores, etc.
* Una secuencia de imágenes para poder generar una animación.
* La secuencia de notas musicales de una melodía.
* En casos más complejos también puedo almacenar las cartas de una baraja, mis números al tirar los dados, caminos posibles de un laberinto, las posiciones de mis enemigos, etc!

#### 4. Procedimientos/implementaciones:

* Recorrer el conjunto de datos almacenados para mostrar su valor o aplicar sus valores como propiedades de elementos. Para esto, vamos a notar que se lleva muy bien con ciclo for() y otras estructuras iterativas.
* Posibilidad de seleccionar un elemento en particular de todo el conjunto. También puede ser uno consecutivo por cada fotograma o cada cierto tiempo.
* Realizar cálculos con todos los datos almacenados. Por ejemplo, encontrar el mayor o el menor; sumar el acumulado, calcular el promedio, ordenar o incluso desordenar, preguntar si existe un elemento, etc.
* Operaciones con casillas, como agregarle o quitarle casillas, o unir un Arreglo con otro, tomar un subconjunto de elementos, etc.

#### 5. Sintaxis

1. DECLARACIÓN
2. CONSTRUCCIÓN DE LAS CASILLAS
3. ACCESO A CADA CASILLA PARA ASIGNAR
4. ACCESO A CADA CASILLA PARA CONSULTAR

//declaración:  
int[] miArreglo = new int[ *5* ];

int[] Crea un espacio en memoria con un arreglo de tipo enteros.

miArreglo El nombre de ese espacio en memoria. Usaremos ese nombre para acceder en el resto del programa.

new Es la instrucción para construir las casillas.

int[*5*] El tipo de dato y la cantidad de casillas creadas. En este ejemplo, 5 casillas int.

De ahora en adelante, usaremos los corchetes para indicar el número de casilla a la que queremos acceder, ya sea para guardar o leer un valor.

//asignación:  
miArreglo[2] = 125;

//invocación:  
println( miArreglo[2] );

#### 6. Estructura de un Arreglo:

* La primera casilla de un Arreglo es el índice 0.
* El índice de la última casilla es *CantidadDeCasillas*-1.
* Intentar acceder a una casilla inexistente dará error de aplicación.
* Los datos de un Arreglo son homogéneos. Esto quiere decir que son todos del mismo tipo (int, float, String, etc).
* La cantidad de casillas es fija.
* Un arreglo puede tener en sus casillas otro arreglo.  
  (Arreglo multidimensional)

#### 7. Ejemplos

1. //declaración de un arreglo:
2. int[] miArreglo = new int[5];
3. //asignación de un valor en una casilla:
4. miArreglo[2] = 125;
5. miArreglo[3] = 250;
6. //imprime en consola:
7. println("todo el arreglo:");
8. println(miArreglo);
9. println("TERCER casilla: " + miArreglo[2]);
10. println("El máximo es: " + max(miArreglo));

“El Juego del Camaleón”

Ver más ejemplos adjuntos en el post.