



ESCUELA DE
INGENIERÍA EN CIENCIAS Y SISTEMAS
FACULTAD DE INGENIERÍA
UNIVERSIDAD DE SAN CARLOS DE GUATEMALA



Día, Fecha:

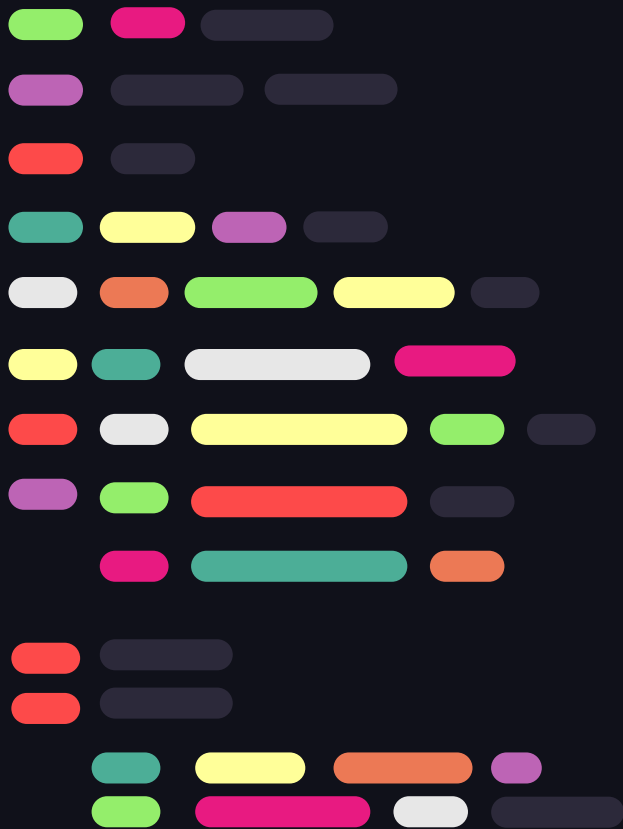
Martes, 04/06/2024

Hora de inicio:

13:00

Introducción a la Programación y Computación 1 [A]

Josué Rodolfo Morales Castillo



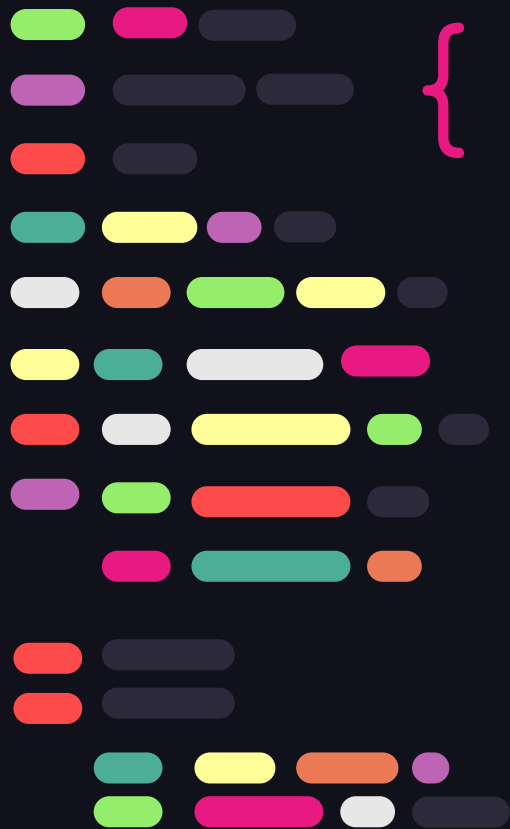
{ ..



Clase 2- Agenda

- Foro No. 1
- Asignación DTT (Formulario)
- Recordatorio corto 1
- Preguntas práctica 1
- Fundamentos de Programación y JAVA

} ..



Fundamentos de Programación y JAVA

Parte 2

Arreglos

Es un objeto que almacena una colección de elementos del mismo tipo. Estos elementos pueden ser variables primitivas o no primitivos. Son estructuras de datos que te permiten almacenar y manipular múltiples valores bajo un solo nombre de variable.

Unidimensionales

Almacena elementos del mismo tipo en una secuencia lineal. Podemos pensar en un arreglo unidimensional como una lista ordenada de elementos, donde cada elemento tiene un índice único que comienza desde cero.

Bidimensionales

Almacena elementos en filas y columnas, formando una especie de cuadrícula o matriz. Podemos pensar en un arreglo bidimensional como una tabla en la que cada elemento tiene dos índices: uno para la fila y otro para la columna.

Unidimensionales

Tipo de dato
de los elementos del
vector

Nombre del vector

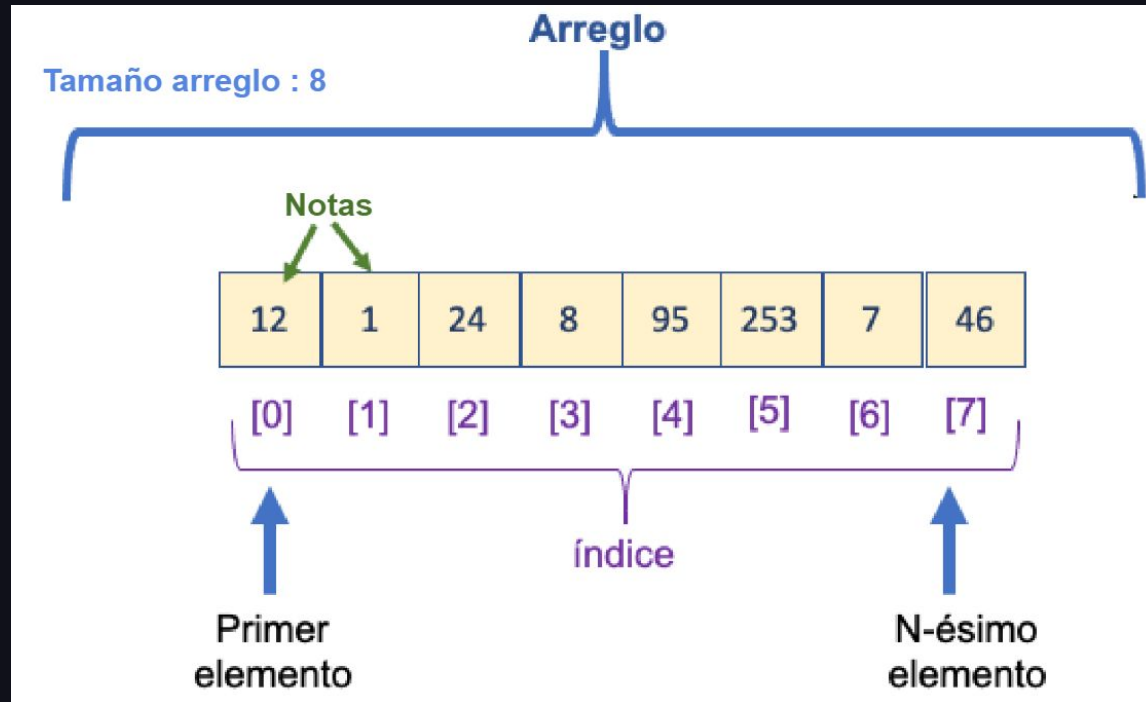
```
int[] notas = new int[8];
```

Número de
elementos del vector

```
notas[0] = 12 ;
```

Asignación de valores

Unidimensionales



Bidimensionales

The diagram illustrates the syntax for creating and accessing a 2D array in Java. It features two lines of code with arrows pointing to specific parts and descriptive labels.

Line 1: `int[][] notas = new int[filas][columnas];`

- An arrow points from the label "Tipo de dato de los elementos de la matriz" to the `int` type.
- An arrow points from the label "Nombre de la matriz" to the `notas` variable.
- An arrow points from the label "Filas y columnas de la matriz" to the `filas` and `columnas` dimensions.

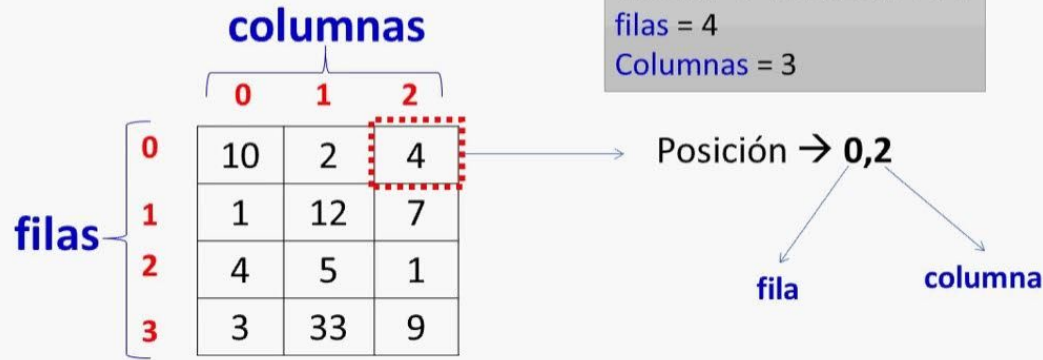
Line 2: `notas[0][0] = 14;`

- An arrow points from the label "Asignación de valores" to the `[0][0]` indices.

Bidimensionales

Matriz

- Arreglo bidimensional que se compone de **filas** y **columnas**
- En cada posición se almacena un elemento de un tipo
- Todos los elementos de la matriz son del mismo tipo



Listas Dinámicas

Las listas dinámicas son estructuras de datos que pueden cambiar de tamaño durante la ejecución del programa, lo que las hace útiles cuando no conocemos de antemano la cantidad exacta de elementos que contendrá la lista. Se implementan generalmente mediante la interfaz `List` del paquete **java.util**.

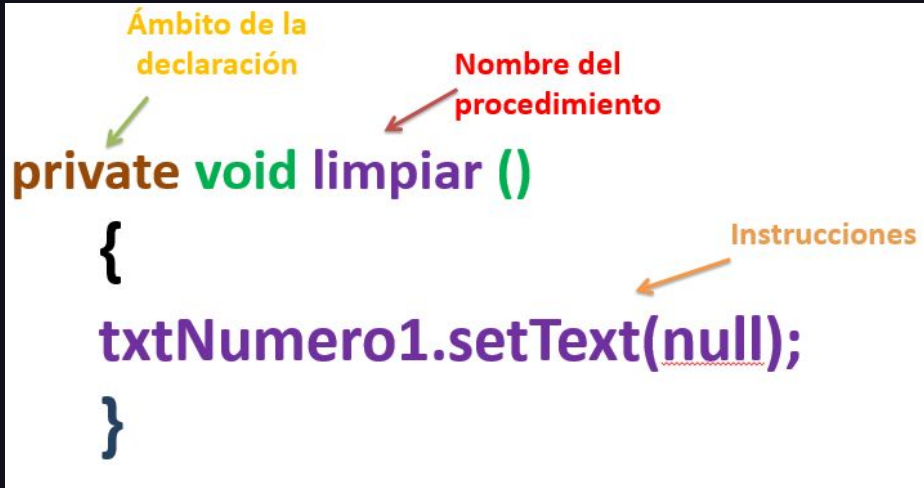
```
import java.util.ArrayList;
public class ArrayListExample {
    public static void main(String[] args) {
        // Crear un ArrayList de tipo String
        ArrayList<String> listaDinamica = new ArrayList<>();
        // Agregar elementos
        listaDinamica.add("Juan");

        // Acceder a elementos
        System.out.println("Elemento en la posición 1: " + listaDinamica.get(0));
        //Resultado: Elemento en la posición 1: Juan
    }
}
```

Procedimientos en Java

Es un bloque de código que realiza una tarea específica y se ejecuta cuando es llamado. Pueden recibir (no siempre) datos como entrada (parámetros), procesar estos datos y realizar acciones, pero no generan un resultado que se pueda utilizar directamente en la llamada al procedimiento, es decir no retorna nada.

Permite la reutilización y organización eficiente del código



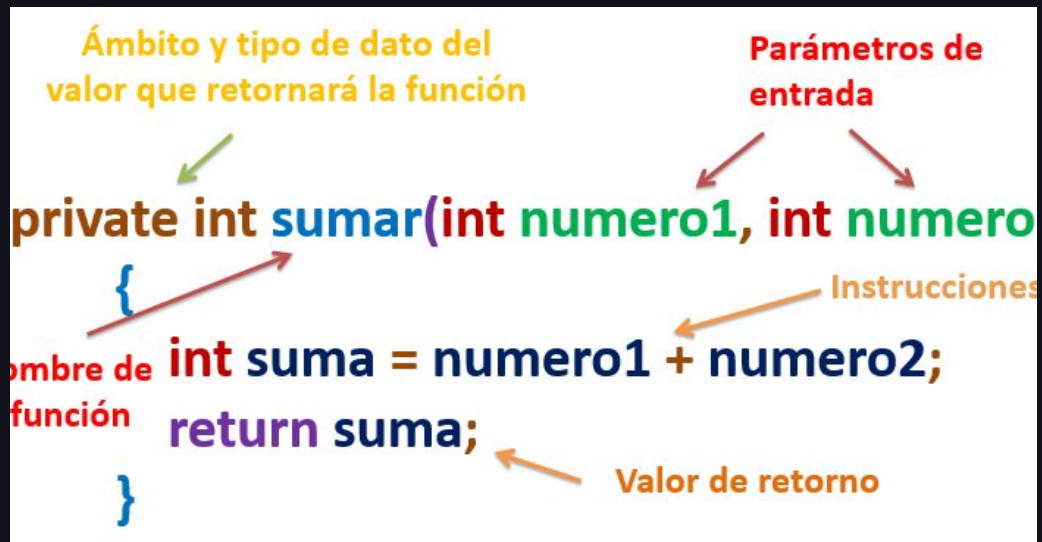
```
private void limpiar ()  
{  
    txtNumero1.setText(null);  
}
```

The diagram illustrates the components of a Java procedure declaration. It shows the code: `private void limpiar ()`, `{`, `txtNumero1.setText(null);`, and `}`. Annotations with arrows point to specific parts: 'Ámbito de la declaración' (Scope of the declaration) points to 'private'; 'Nombre del procedimiento' (Procedure name) points to 'limpiar'; and 'Instrucciones' (Instructions) points to the body of the procedure, specifically the line `txtNumero1.setText(null);`.

Funciones en Java

Es un bloque de código que realiza una tarea específica y se ejecuta cuando es llamado. Las funciones pueden (no siempre) recibir datos como entrada (parámetros), procesar estos datos, la diferencia con los procedimientos es que las funciones devuelven un resultado (valor de retorno).

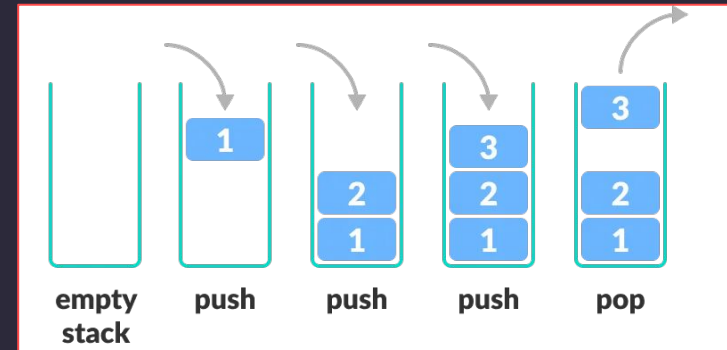
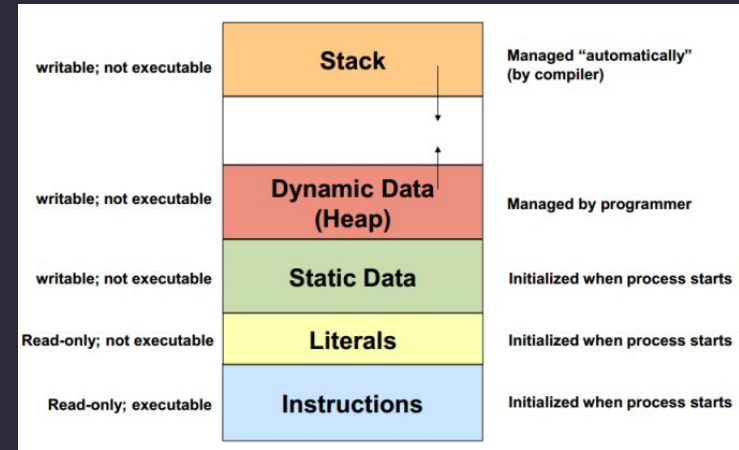
En java, las funciones deben definirse con un tipo de Dato, y el dato que retorna es el mismo tipo con el que se definió.



Manejo de memoria

Memoria Stack

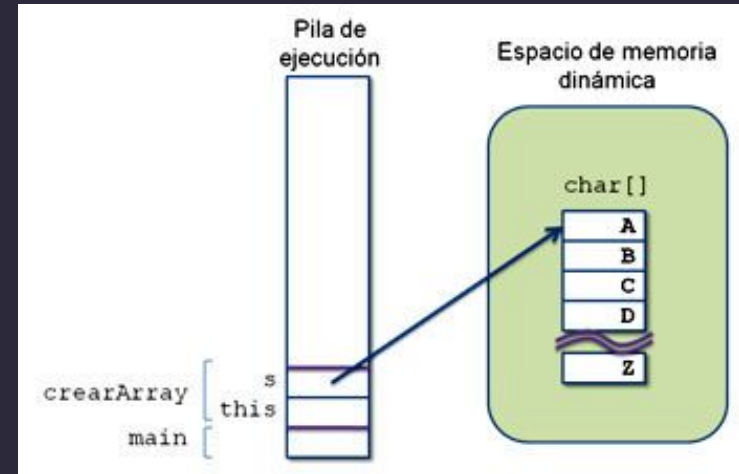
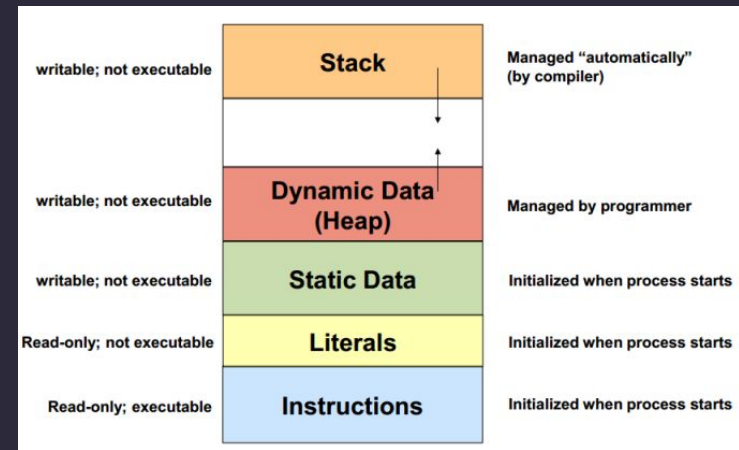
En programación, la memoria de la pila (stack) es una región de la memoria que se utiliza para almacenar variables locales y gestionar llamadas a funciones. Es una estructura de datos tipo pila (stack), lo que significa que las operaciones de inserción y eliminación de datos se realizan en el extremo superior de la pila.



Manejo de memoria

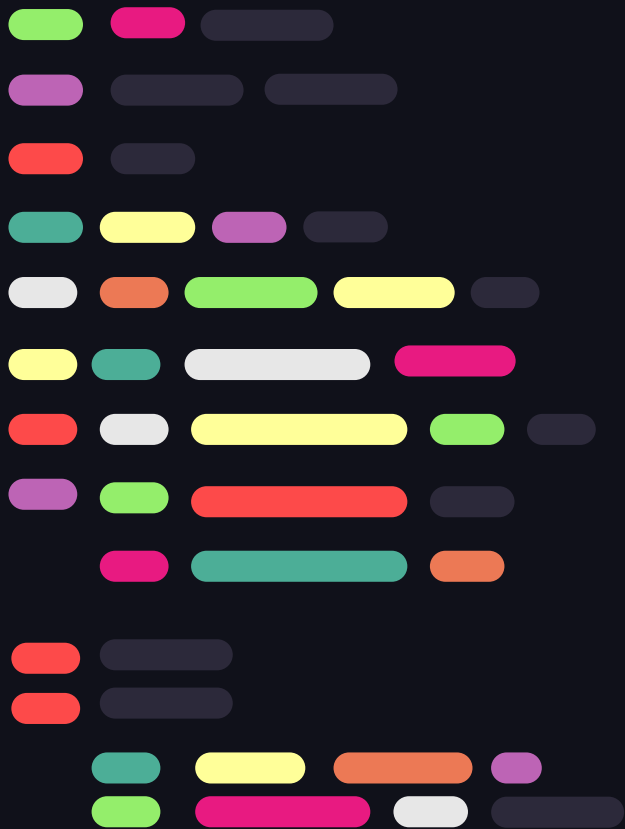
Memoria Heap

La memoria heap es una región de memoria dinámica que se utiliza para la asignación de memoria durante la ejecución de un programa. Es un área de memoria que se reserva para almacenar objetos y estructuras de datos que se crean y eliminan durante la ejecución del programa.



¿Dudas?





{ ..



Ejemplo

} ..