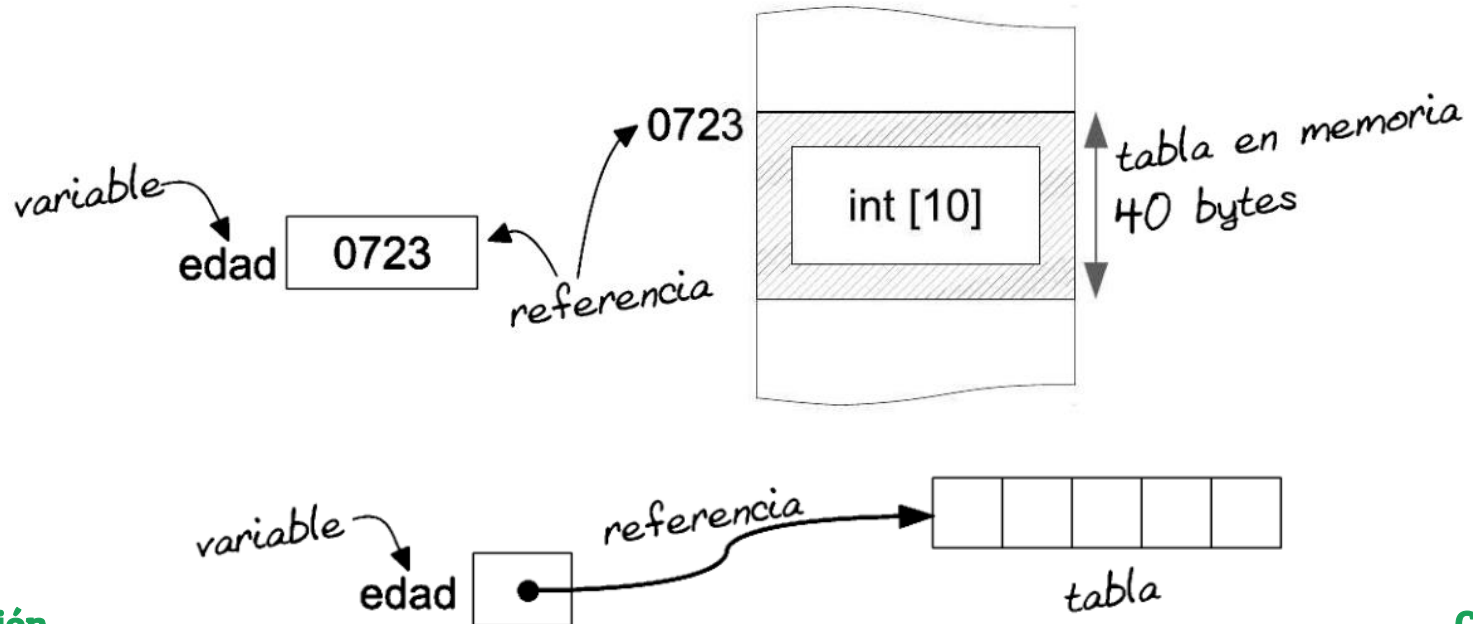


Arrays

Declaración y definición

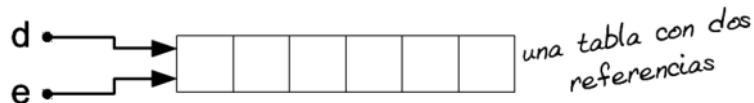
```
int [] edad = new int[5];
```

```
int [] edad;  
edad = new int[5];
```

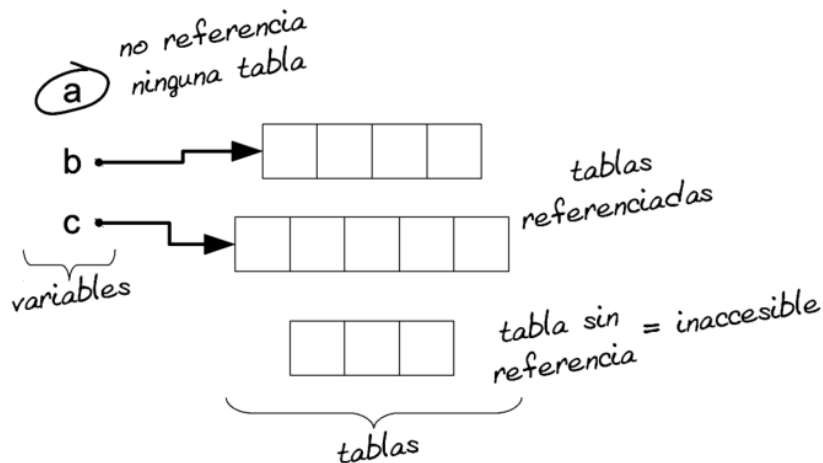


Multirreferencias y null

```
int [] d, e;  
d = new int[6];  
e = d;
```



```
int [] a, b, c;  
b = new int[5];  
c = new int[5];
```



null y GC

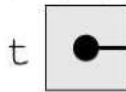
```
int [] t1, t2;  
t1 = new int[100];  
t2 = t1;  
  
t1 = null;  
t2 = null;
```

Arrays y funciones

```
int t[] = new int[5];
```

```
...
```

```
ejemploFuncion(t);
```



4	0	-1	3	7
---	---	----	---	---

```
void ejemploFuncion(int x[]) {  
    //cuerpo de la función  
}
```

Clase Arrays

Imprimir

```
int [] t = {8, 41, 37, 22, 19};  
System.out.println(Arrays.toString(t));  
...  
[8, 41, 37, 22, 19]
```

Inicializar

```
Arrays.fill(t, 123);  
Arrays.fill(t, 3, 7, 123);
```

Comparar

```
int [] t1 = {8, 41, 37};  
int [] t2 = {8, 41, 37};  
System.out.println(t1 == t2);  
  
System.out.println(Arrays.equals(t1, t2));
```

```
import java.util.Arrays;
```

Ordenar

```
int [] t = {8, 41, 37};  
  
Arrays.sort(t);  
  
// Ahora t = {8, 37, 41}
```

Copiar

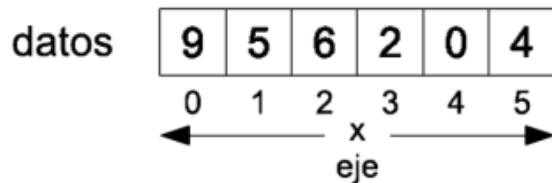
```
int [] t = {1, 2, 1, 6, 23};  
int [] a, b;  
a = Arrays.copyOf(t, 3); // a = [1, 2, 1]  
b = Arrays.copyOfRange(t, 1, 4); // b = [2, 1, 6]
```

Buscar

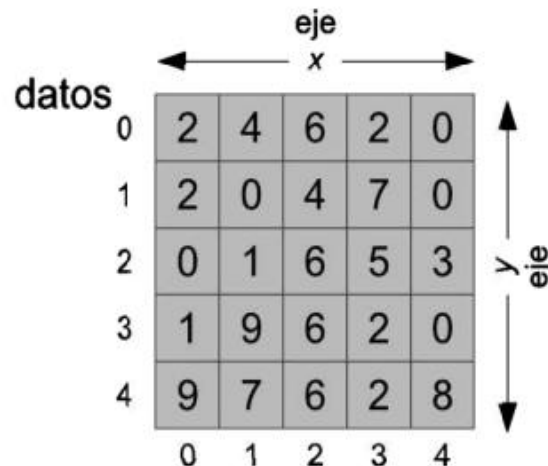
```
int pos = Arrays.binarySearch(datos, 19);  
if (pos >= 0) {  
    System.out.println("Encontrado en el índice:  
" + pos);  
} else {  
    System.out.println("No se ha encontrado");  
}
```

Arrays multidimensionales

```
int [] datos = new int[5];
```



```
int [][] datos = new int[5][5];
```



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

END



jgardur081@g.educaand.es