

Sesión 09: Índices y *nulles*

Programación 2

Ángel Herranz

2024-2025

Universidad Politécnica de Madrid

En capítulos anteriores

 Tema 1: Clases y Objetos

 Tema 2: Colecciones acotadas de Objetos

En capítulos anteriores

-  Tema 1: Clases y Objetos
-  Tema 2: Colecciones acotadas de Objetos
-  “Programa que lea de la entrada estándar órdenes para insertar y borrar canciones y que imprima la *playlist* resultante final”

En el capítulo de hoy

Índices y *nulles*

Índices y *nulles*

“Programa que lea de la entrada estándar órdenes para insertar y borrar canciones y que imprima la *playlist* resultante final”

En el capítulo de hoy

Índices y *nulles*

“Programa que lea de la entrada estándar órdenes para insertar y borrar canciones y que imprima la *playlist* resultante final”

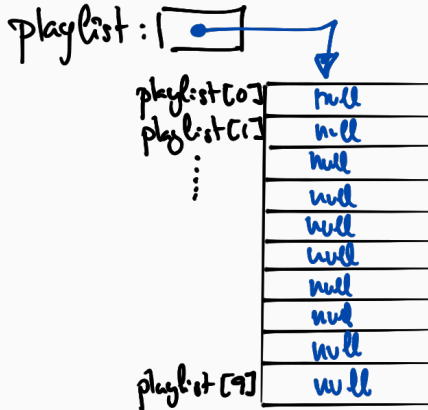
!A dibujar!



Mi *playlist* v2: ordenes.txt

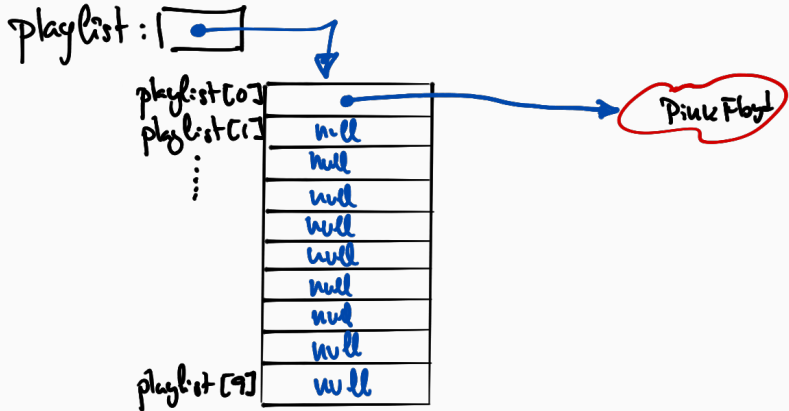
a	a
Wish you where here	SAD!
Pink Floyd	XXXTENTACION
4	3
a	a
Despacito	God's Plan
Luis Fonsi	Drake
2	4
a	r
The logical song	SAD!
Supertramp	a
5	Havana
r	Camila Cabello, Young Thug
Despacito	4

🗨️ Todo con *nulles*

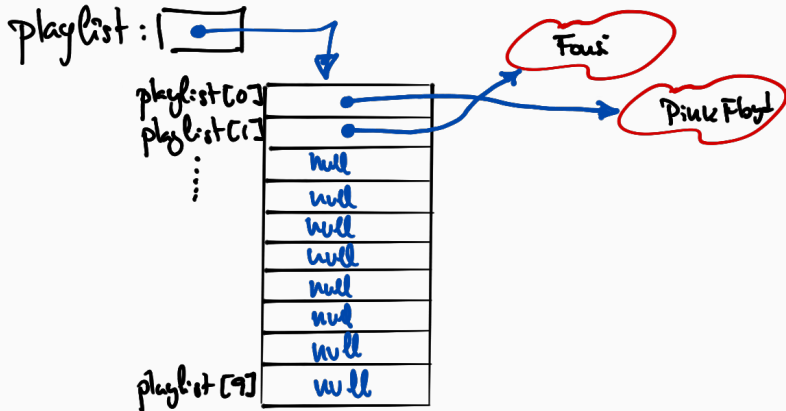




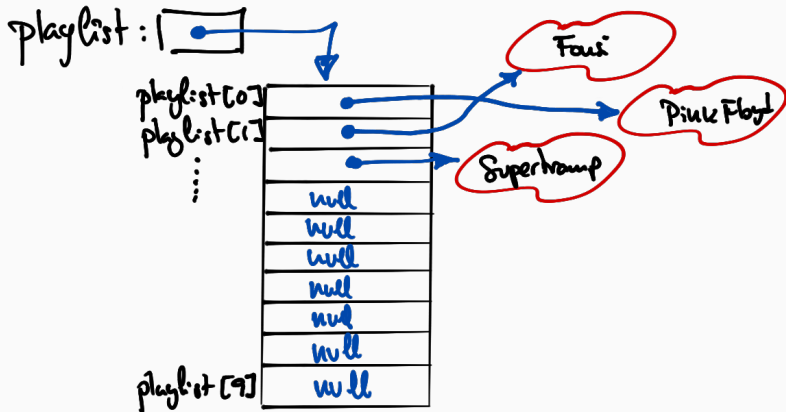
a Wish you were here



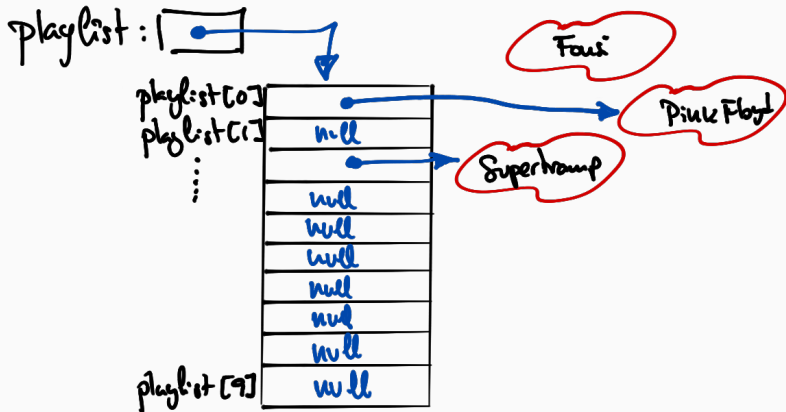
🗨️ a Despacito



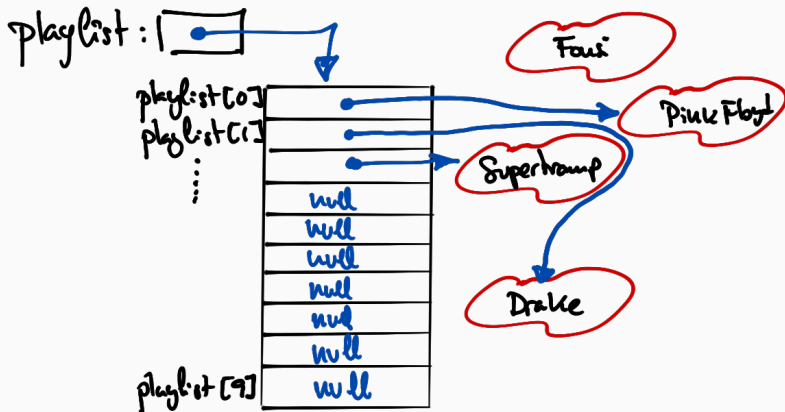
🗨️ a The logical song



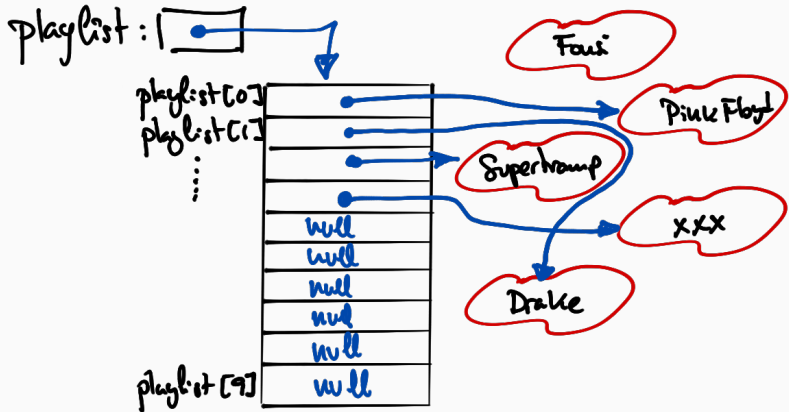
🗨 r Despacito



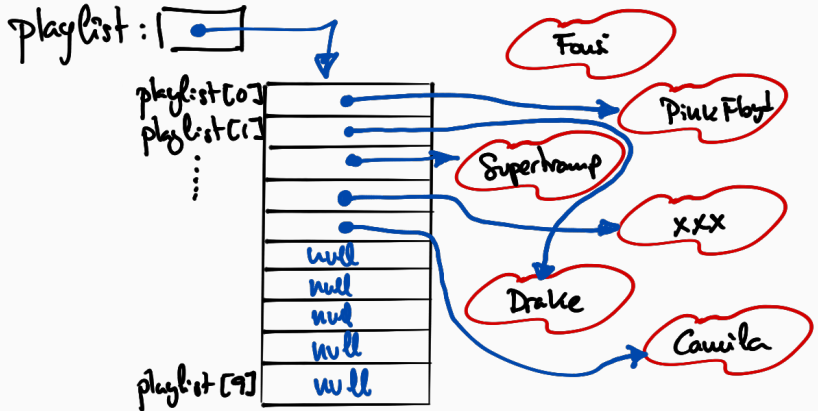
💬 a God's plan

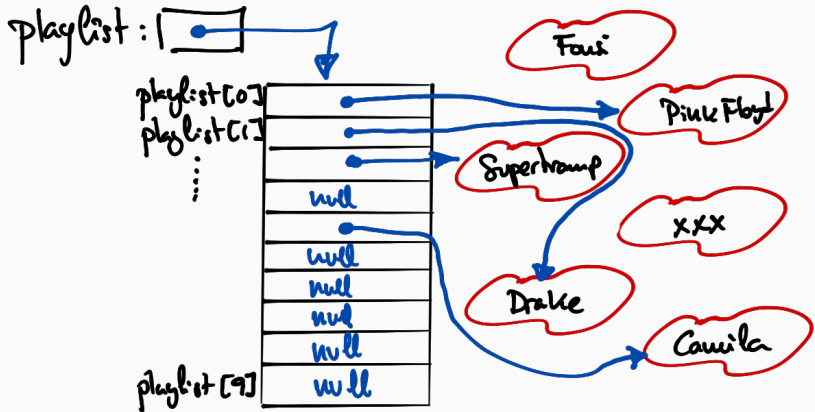


🗨️ a Sad



🗨️ a Havana






```
public class ProcesarOrdenes {  
    public static void main(String[] args) {  
        // Se crea un objeto Scanner para poder  
        // leer de la entrada estándar  
        java.util.Scanner stdin =  
            new java.util.Scanner(System.in);  
  
        // Se declara un array de canciones  
        Cancion[] playlist;  
  
        // Se crea el array con un máximo  
        final int N = 10;  
        playlist = new Cancion[N];  
  
        // Se procesan las órdenes  
        while (stdin.hasNext()) {  
            String orden = stdin.nextLine();
```

¿Problemillas?

- En Cancion, necesitamos un observador para sacar el título de una canción:

```
public String titulo() {  
    return titulo;  
}
```

- Cuidado al hacer `playlist[i].titulo()` cuando `playlist[i]` es **null**
- Cuidado con los espacios al principio y final de los títulos (*leading and trailing spaces*): buscar documentación del método `trim()` de `String`
- Cuidado con el “fin de fichero” (EOF)

¿Problemas con el enfoque nullo?

¿Problemas con el enfoque nullo?

Se pierde el orden de inserción

¿Problemas con el enfoque nullo?

Se pierde el orden de inserción
y
añadir una nueva canción implica
recorrer la *playlist* hasta encontrar un
null

¿Solución?

¿Solución?

Además del *array*, se introduce un *índice* entero que diga cuál es el *siguiente* hueco en el que añadir (ie. contador de canciones) y todas las canciones están en la parte de *arriba* del array

```
public class ProcesarOrdenes {  
    public static void main(String[] args) {  
        // Se crea un objeto Scanner para poder  
        // leer de la entrada estándar  
        java.util.Scanner stdin =  
            new java.util.Scanner(System.in);  
  
        // Se declara un array de canciones  
        Cancion[] playlist;  
  
        // Se crea el array con un máximo  
        final int N = 10;  
        playlist = new Cancion[N];  
        int siguiente = 0;  
  
        // Se procesan las órdenes  
        while (stdin.hasNext()) {
```


¿Problemas con el enfoque *índices*?

¿Problemas con el enfoque *índices*?

Cuando se borra un elemento es
necesario desplazar todos...

¿Problemas con el enfoque *índices*?

Cuando se borra un elemento es
necesario desplazar todos...
¿O no?

(colocar el último en el lugar del borrado)

¿Problemas con el enfoque *índices*?

Cuando se borra un elemento es
necesario desplazar todos...

¿O no?

(colocar el último en el lugar del borrado)

¿O sí?

(¡me importa el orden!)

class Playlist