

Introducción a Java

//Parte 1

IT BOARDING

BOOTCAMP



Índice



01 Colecciones y listas

03 Mapas y diccionarios

02 Arrays, ArrayList

04 Iteradores

IT BOARDING

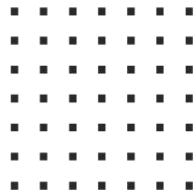
BOOTCAMP

// Colección o lista de objetos

Aunque, en Java, **Collection** y **List** son tipos de dato (*interfaces*), genéricamente llamamos “Colección” a cualquier conjunto de objetos.

```
// un array de 10 int (de 0 a 9)
int a[] = new int[10];
for(int i=0; i<10; i++ )
{
    a[i] = i;
}
```

```
// un arrayList con 10 elementos int
ArrayList<Integer> al = new ArrayList<>();
for(int i=0; i<10; i++ )
{
    al.add(i);
}
```





Iterar una colección

Podemos utilizar un **for** “tradicional”, o un **for each**.

```
for( int i=0; i<a.length; i++ )  
{  
    System.out.println( a[i] );  
}
```

```
for( int i=0; i<a1.size(); i++ )  
{  
    System.out.println( a1.get(i) );  
}
```

```
for(int x: a)  
{  
    System.out.println(x);  
}
```

```
for(int x:a1)  
{  
    System.out.println(x);  
}
```



Clases “wrapper”, variables y **autoboxing**

Cada tipo de dato tienen una clase “**wrapper**”, que lo complementa con **funciones utilitarias** y otras facilidades.

```
// obtenemos el valor decimal de un número expresado en base 2
int b = Integer.parseInt("100110011110011",2);

// obtenemos el valor decimal de un número expresado en base 16
int h = Integer.parseInt("9AE4F",16);
```

```
// averiguamos si una letra está escrita en minúscula
boolean x = Character.isUpperCase('a');

// convertimos una letra a mayúscula
char m = Character.toUpperCase('a');
```

Mapas o diccionarios

En este tipo de colecciones, cada elemento es un **par**: {key, value}

Map<Integer,String>

key

value

1	"Uno"
2	"Dos"
3	"Tres"
4	"Cuatro"

Al agregar un valor, quedará vinculado a una *key*. Dicha *key*, posteriormente, nos permitirá recuperarlo, modificarlo o eliminarlo.



Agregar, acceder y remover entradas en el Map

Los métodos más utilizados son **put**, **get** y **remove**.

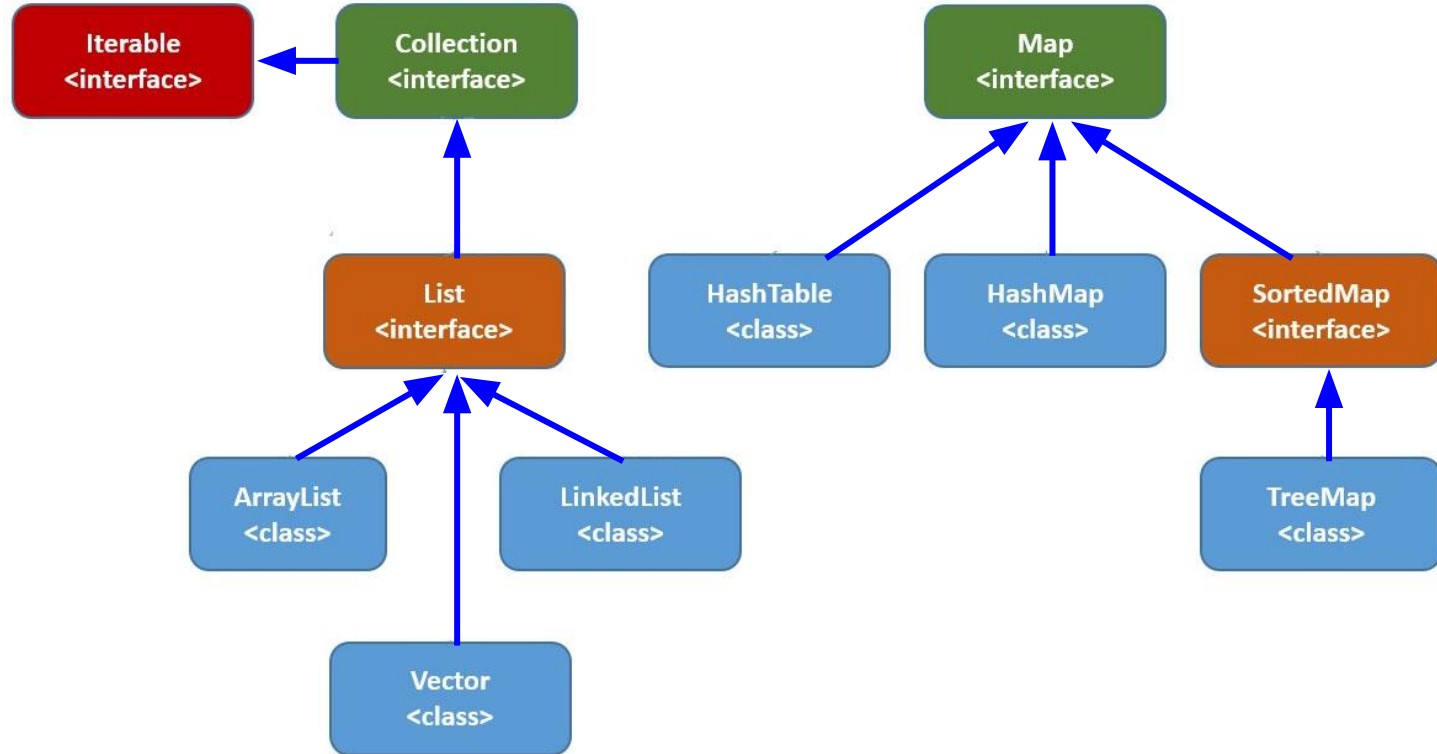
```
HashMap<Integer,String> hm = new HashMap<>();  
hm.put(1,"Uno");  
hm.put(2,"Dos");  
hm.put(3,"Tres");  
hm.put(4,"Cuatro");  
  
String x = hm.get(2);  
System.out.println(x); // Salida: Dos  
hm.remove(2); // Chau entrada
```



Iterar un map

```
HashMap<Integer,String> hm = new HashMap<>();  
hm.put(1,"Uno");  
hm.put(2,"Dos");  
hm.put(3,"Tres");  
hm.put(4,"Cuatro");  
  
for(Map.Entry<Integer,String> entry:hm.entrySet())  
{  
    Integer key = entry.getKey();  
    String value = entry.getValue();  
    System.out.println("Key="+key+", value="+value);  
}
```


Java Collection Framework (JCF)



¿Dudas? ¿Preguntas?



Gracias.

IT BOARDING

BOOTCAMP

