

Clase String - Métodos:

MÉTODO	DESCRIPCIÓN
charAt(n)	Devuelve el carácter que está en la posición n.
compareTo(OtroString)	Devuelve 0 si las dos cadenas son iguales. <0 si la primera es alfabéticamente menor que la segunda ó >0 si la primera es alfabéticamente mayor que la segunda.
compareToIgnoreCase(OtroString)	Igual que compareTo pero sin considerar mayúsculas y minúsculas.
concat(OtroString)	Agrega OtroString al final del String.
contains(OtroString)	Devuelve un booleano comprobando si la cadena contiene o no a la otra.
copyValueOf(char [], n, m)	Devuelve una cadena que representa los caracteres de una matriz de caracteres. n representa el elemento de la matriz en el que inicia y m en el que termina de concatenar.
equals(OtroString)	Compara dos cadenas y devuelve true si son iguales.
equalsIgnoreCase(OtroString)	Igual que equals pero sin considerar mayúsculas y minúsculas.
indexOf('carácter')	Devuelve la posición de la primera aparición de carácter (sino -1).
isEmpty()	Devuelve un booleano comprobando si la cadena está vacía o no.
lastIndexOf('carácter')	Devuelve la posición de la última aparición de carácter (sino -1).
length()	Devuelve la longitud de la cadena.
replace(char, charNuevo)	Cambia el carácter especificado que se encuentre en la cadena por otro carácter nuevo.
substring(n1,n2) o substring(n1)	Devuelve la subcadena comprendida entre las posiciones n1 y n2, o devuelve la subcadena a partir de la posición n1 indicada.
toLowerCase()	Devuelve la cadena convertida a minúsculas.
toUpperCase()	Devuelve la cadena convertida a mayúsculas.
trim()	Devuelve la cadena eliminando los espacios en blanco al comienzo y al final de la cadena original.
valueOf(N)	Método estático. Convierte el valor N a String. N puede ser de cualquier tipo.

HashMap - Métodos:

MÉTODO	DESCRIPCIÓN
clear()	Elimina todos los elementos del mapa.
containsKey(Clave)	Devuelve un boolean indicando si existe un elemento con la clave especificada.
containsValue(Valor)	Devuelve un boolean indicando si existe un elemento con el valor especificado.
get(Clave)	Devuelve el valor del elemento con la clave especificada.
isEmpty()	Devuelve un boolean indicando si el mapa está vacío.
keySet()	Devuelve un conjunto de todas las claves del mapa.
put(Clave, Valor)	Guarda una entrada en el mapa, si ya existe un valor en la clave indicada lo cambiará por el especificado.
remove(Clave) o remove(Clave, Valor)	Elimina la entrada en la clave especificada. Si se pone también un valor solo se eliminará si el valor en la clave coincide con el especificado.
size()	Devuelve el número de entradas que tiene el mapa.
values()	Devuelve una colección de solamente todos los valores que hay en el mapa, sus claves no van incluidas.

ArrayList - LinkedList - Métodos:

MÉTODO	DESCRIPCIÓN
add(Elemento) o add(indice, elemento)	Agregar elementos a la lista o al índice indicado.
clear()	Elimina todos los elementos de la lista.
Collections.sort(Lista)	Ordena los elementos de la lista alfabéticamente o numéricamente. <u>(Si se importa la clase Collections únicamente)</u>
contains(elemento)	Devuelve un boolean indicando si existe el elemento en la lista.
get(indice)	Acceder a un elemento de la lista en el índice especificado.
indexOf(elemento)	Devuelve el índice del elemento especificado, en caso de no estar devuelve -1.
isEmpty()	Devuelve un boolean indicando si la lista está vacía.
remove(indice)	Elimina el elemento en el índice especificado.
set(indice, elemento)	Reemplaza un elemento en la posición indicada.
size()	Devuelve el número de entradas que tiene la lista.

Cola (Queue) - Métodos:

MÉTODO	DESCRIPCIÓN
offer(Elemento)	Agregar elementos al final de la cola.
poll()	Elimina y devuelve el elemento al frente de la cola.
peek()	Devuelve el elemento al frente de la cola sin eliminarlo.
isEmpty()	Comprueba si la cola está vacía.

Pila (Stack) - Métodos:

MÉTODO	DESCRIPCIÓN
push(Elemento)	Agregar elementos a la cima de la pila.
pop()	Elimina y devuelve el elemento de la cima de la pila.
peek()	Devuelve el elemento de la cima de la pila sin eliminarlo.
isEmpty()	Comprueba si la pila está vacía.

Clase Array - Métodos:

MÉTODO	DESCRIPCIÓN
Arrays.deepToString(array)	Este método está diseñado para convertir matrices multidimensionales en cadenas.
Arrays.equals(array1, array2)	Devuelve un booleano comprobando si ambos arrays son iguales.
Arrays.sort(array)	Ordena los elementos de la lista alfabéticamente o numéricamente.
Arrays.toString(array)	Retorna una representación de los contenidos del arreglo especificado. Podemos imprimir un arreglo unidimensional usando este método.

Excepciones :

EXCEPCIÓN	DESCRIPCIÓN
ArithmeticException	Indica un error aritmético, como la división por cero.
ArrayIndexOutOfBoundsException	Se lanza cuando intentas acceder a un índice fuera del rango válido de un arreglo.
FileNotFoundException	Se lanza cuando se intenta acceder a un archivo que no se encuentra en el sistema.
NullPointerException	Ocurre cuando intentas acceder a un objeto que es null.