**Colecciones en JAVA**

**TREESET**

public class EjemploTreeSet {

public static void main(String args[]) {

//Crear y agregar elementos

//con string

TreeSet<String> al = new TreeSet<String>();

al.add("Ramiro");

al.add("Victoria");

al.add("Rafael");

al.add("Ana");

//Traversing elements

Iterator<String> itr = al.iterator();

while (itr.hasNext()) {

System.out.println(itr.next());

}

//con numeros

TreeSet<Integer> set = new TreeSet<Integer>();

set.add(24);

set.add(66);

set.add(12);

set.add(15);

for (Integer val : set) {

System.out.println(val);

}

}

}

**Crear un paquete modelo con la siguiente clase:**

public class Book implements Comparable<Book> {

private int id;

private String name, author, publisher;

private int quantity;

public int getId() {

return id;

}

public void setId(int id) {

this.id = id;

}

public String getName() {

return name;

}

public void setName(String name) {

this.name = name;

}

public String getAuthor() {

return author;

}

public void setAuthor(String author) {

this.author = author;

}

public String getPublisher() {

return publisher;

}

public void setPublisher(String publisher) {

this.publisher = publisher;

}

public int getQuantity() {

return quantity;

}

public void setQuantity(int quantity) {

this.quantity = quantity;

}

public Book(int id, String name, String author, String publisher, int quantity) {

this.id = id;

this.name = name;

this.author = author;

this.publisher = publisher;

this.quantity = quantity;

}

@Override

public String toString() {

return "Book{" + "id=" + id + ", name=" + name + ", author=" + author + '}';

}

public int compareTo(Book b) {

if (id > b.getId()) {

return 1;

} else if (id < b.getId()) {

return -1;

} else {

return 0;

}

}

}

public class EjemploTreeSetClase {

public static void main(String[] args) {

Set<Book> set=new TreeSet<Book>();

//Creating Books

Book b1=new Book(121,"Let us C","Yashwant Kanetkar","BPB",8);

Book b2=new Book(233,"Operating System","Galvin","Wiley",6);

Book b3=new Book(101,"Data Communications & Networking","Forouzan","Mc Graw Hill",4);

//Adding Books to TreeSet

set.add(b1);

set.add(b2);

set.add(b3);

//Traversing TreeSet

for(Book b:set){

System.out.println(b);

}

}

}

**HASHSET**

Aplicar los mismos ejercicios pero con HashSet

**HASHMAP**

public class EjemploHashMap {

static HashMap<String, Float> listaProductos;

public static void main(String[] args) {

listaProductos = new HashMap<String, Float>();

Scanner sc = new Scanner(System.in);

int opcionElegida = 0;

float precio;

String codigo;

while (opcionElegida != 5) {

System.out.println("Introduce el numero de la opción que quieras:");

System.out.println("1.- Introducir producto");

System.out.println("2.- Modificar precio");

System.out.println("3.- Mostrar todos los productos");

System.out.println("4.- Eliminar producto");

System.out.println("5.- Salir");

opcionElegida = sc.nextInt();

switch (opcionElegida) {

case 1:

System.out.println("Introduce el códido del producto:");

codigo = sc.next();

System.out.println("Introduce el precio del producto:");

precio = sc.nextFloat();

guardarProducto(codigo, precio);

break;

case 2:

System.out.println("Introduce el códido del producto del que quieres cambiar el precio:");

codigo = sc.next();

modificaPrecio(codigo);

break;

case 3:

mostrarProductos();

break;

case 4:

System.out.println("Introduce el códido del producto que quieres eliminar:");

codigo = sc.next();

eliminaProducto(codigo);

break;

case 5:

break; // Si la opcion es 5 no se hace nada

default:

System.out.println("Tienes que introducir una opción valida");

}

}

}

}

**TREEMAP**

public class EjemploTreeMap {

public static void main(String[] args) {

TreeMap<Integer, String> mapHttpStatus = new TreeMap<>();

mapHttpStatus.put(100, "Continue");

mapHttpStatus.put(200, "OK");

mapHttpStatus.put(300, "Multiple Choices");

mapHttpStatus.put(400, "Bad Request");

mapHttpStatus.put(401, "Unauthorized");

mapHttpStatus.put(402, "Payment Required");

mapHttpStatus.put(403, "Forbidden");

mapHttpStatus.put(404, "Not Found");

mapHttpStatus.put(500, "Internal Server Error");

mapHttpStatus.put(501, "Not Implemented");

mapHttpStatus.put(502, "Bad Gateway");

System.out.println("Todos los pares clave-valor ");

for (Integer code : mapHttpStatus.keySet()) {

System.out.println(code + " -> " + mapHttpStatus.get(code));

}

System.out.println();

Integer firstKey = mapHttpStatus.firstKey();

String firstValue = mapHttpStatus.get(firstKey);

System.out.println("Primer estado: " + firstKey + " -> " + firstValue);

System.out.println();

Integer lastKey = mapHttpStatus.lastKey();

String lastValue = mapHttpStatus.get(lastKey);

System.out.println("Ultimo estado: " + lastKey + " -> " + lastValue);

System.out.println();

}

}