1. Create a java program to search the key in the tree map using comparative method ?

CODE:

import java.util.TreeMap;

public class TreeMapSearch {

public static void main(String[] args) {

TreeMap<Integer, String>treeMap= new TreeMap<>((key1, key2) -> key1.compareTo(key2));

treeMap.put(1, "Apple");

treeMap.put(2, "Banana");

treeMap.put(3, "Orange");

treeMap.put(4, "Grapes");

int keyToSearch = 3;

if (treeMap.containsKey(keyToSearch)) {

System.out.println("Key " + keyToSearch + " found with value: " + treeMap.get(keyToSearch));

} else {

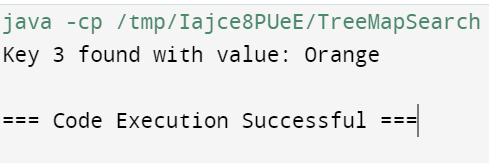
System.out.println("Key " + keyToSearch + " not found in the TreeMap.");

}

}

}

OUTPUT:



1. Create a java program to SORT the key in the tree map using comparative method?

CODE: import java.util.\*;

public class SortTreeMapKeys {

public static void main(String[] args) {

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

treeMap.put("D", 4);

treeMap.put("B", 2);

treeMap.put("C", 3);

treeMap.put("A", 1);

TreeMap<String, Integer> sortedTreeMap = new TreeMap<>(new CustomComparator(treeMap));

sortedTreeMap.putAll(treeMap);

System.out.println("Sorted TreeMap Keys:");

for (String key : sortedTreeMap.keySet()) {

System.out.println(key + " : " + sortedTreeMap.get(key));

}

}

static class CustomComparator implements Comparator<String> {

TreeMap<String, Integer> map;

public CustomComparator(TreeMap<String, Integer> map) {

this.map = map;

}

@Override

public int compare(String a, String b) {

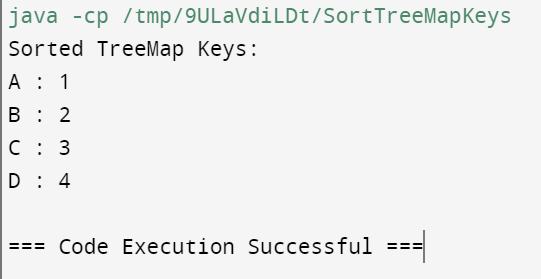
return a.compareTo(b);

}

}

}

OUTPUT:



3.Overload the constructor with different data type, create the class called point with instance variable x and y .one constructor takes interger parameter and the another constructor double parameter and print the value for each constructor(constructor overload)?

CODE:

public class Point {

private int x;

private int y;

public Point(int x, int y) {

this.x = x;

this.y = y;

System.out.println("Integer Constructor - x: " + x + ", y: " + y);

}

public Point(double x, double y) {

this.x = (int) x;

this.y = (int) y;

System.out.println("Double Constructor - x: " + x + ", y: " + y);

}

public static void main(String[] args) {

Point intPoint = new Point(5, 10);

Point doublePoint = new Point(3.5, 7.2);

}

}

OUTPUT:

