1. **import** java.util.\*;

**public** **class** QTree

{

**public** **static** **void** main(String[] args)

{

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

treeMap.put(989328444, "Priyanshi");

treeMap.put(989328442, "Sakshi");

treeMap.put(989328443, "Ishika");

treeMap.put(989328445, "Anamika");

treeMap.put(989328446, "Aditya");

Set keys = treeMap.keySet();

Iterator i = keys.iterator();

**while** (i.hasNext())

{

System.***out***.println(i.next());

}

System.***out***.println();

Collection getValues = treeMap.values();

i = getValues.iterator();

**while** (i.hasNext())

{

System.***out***.println(i.next());

}

System.***out***.println();

System.***out***.println(treeMap);

}

}

1. **import** java.util.TreeSet;

**public** **class** second

{

**public** **static** **void** main(String[] args)

{

TreeSet<Function> func = **new** TreeSet<>();

func.add(**new** Function("Cheese",1));

func.add(**new** Function("Pasta",2));

func.add(**new** Function("Chips",3));

func.add(**new** Function("Coke",4));

func.add(**new** Function("Chocolates",2));

func.add(**new** Function("Maggi",4));

**for**(Function f : func)

{

System.***out***.println(f);

}

}

}

**public** **class** Function **implements** Comparable<Function>

{

**private** String name;

**private** **int** id;

Function(String product\_name, **int** product\_id)

{

**this**.id = product\_id;

**this**.name = product\_name;

}

**private** String getName()

{

**return** name;

}

**public** **int** getId()

{

**return** id;

}

**public** **int** compareTo(Function f)

{

**if**(id == f.getId())

{

**return** 0;

}

**else** **if**(name.compareTo(f.getName()) < 0)

{

**return** -1;

}

**else**

{

**return** -1;

}

}

**public** String toString()

{

**return** name + " - " + id;

}

}

1. **import** java.util.\*;

**class** Employee **implements** Comparable<Employee>

{

**int** id;

String Name;

String Department;

**int** Salary;

**public** Employee(**int** id,String name,String department,**int** salary)

{

**this**.id=id;

**this**.Name=name;

**this**.Department=department;

**this**.Salary=salary;

}

**public** **int** compareTo(Employee e)

{

**if**(id>e.id)

{

**return** 1;

}

**else** **if**(id<e.id)

{

**return** -1;

}

**else**

{

**return** 0;

}

}

}

**public** **class** third {

**public** **static** **void** main(String[] args) {

Set <Employee>set=**new** TreeSet<Employee>();

Employee emp1=**new** Employee(1,"Priyanshi","Developer",30000);

Employee emp2=**new** Employee(2,"Sakshi","Java Developer",30000);

Employee emp3=**new** Employee(3,"Ishika","Analyst",20000);

Employee emp4=**new** Employee(4,"Aditya","Business Analyst",45000);

set.add(emp1);

set.add(emp2);

set.add(emp3);

set.add(emp4);

**for**(Employee e:set) {

System.***out***.println(e.Name);

}

}

}

1. **import** java.time.LocalDate;

**import** java.time.format.DateTimeFormatter;

**import** java.util.LinkedList;

**public** **class** Fourth {

**public** **static** **void** main(String[] args) {

LocalDate date1 = LocalDate.*of*(2000, 12, 23);

LocalDate date2 = LocalDate.*of*(1998, 2, 27);

LocalDate date3 = LocalDate.*of*(2001, 12, 23);

LinkedList<LocalDate> list = **new** LinkedList<LocalDate>();

list.add(date1);

list.add(date2);

list.add(date3);

**for**(LocalDate l : list)

{

String printDate = l.format(DateTimeFormatter.*ofPattern*("dd-MM-YYYY"));

**if**(l.isLeapYear())

{

System.***out***.println("Your Date of Birth is " + printDate + " and it was a Leap Year");

}

**else**

{

System.***out***.println("Your Date of Birth is " + printDate + " and it was not a Leap Year");

}

}

}

}