**Core Java Assignment 6**

**Assignment on Collection**

**1.**

**import** java.util.Collection;

**import** java.util.Collections;

**import** java.util.Map;

**import** java.util.TreeMap;

**public** **class** TreeMapContactNo {

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

Contact c1 = **new** Contact("Chetana", "chetana@gmail.com");

Contact c2 = **new** Contact("Siddhesh", "siddhesh@gmail.com");

Contact c3 = **new** Contact("Deepali", "deepali@gmail.com");

Contact c4 = **new** Contact("Manisha", "manisha@gmail.com");

Contact c5 = **new** Contact("Prakash", "prakash@gmail.com");

TreeMap <Long, Contact> contactNoMap = **new** TreeMap<Long, Contact>(Collections.*reverseOrder*());

contactNoMap.put(9021520002L, c1);

contactNoMap.put(9423267676L, c2);

contactNoMap.put(9404716076L, c3);

contactNoMap.put(9563259845L, c4);

contactNoMap.put(9589996699L, c5);

**for**(Map.Entry entry : contactNoMap.entrySet()) {

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

}

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

**for**(Map.Entry entry : contactNoMap.entrySet()) {

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

}

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

contactNoMap.forEach((key,value)->{

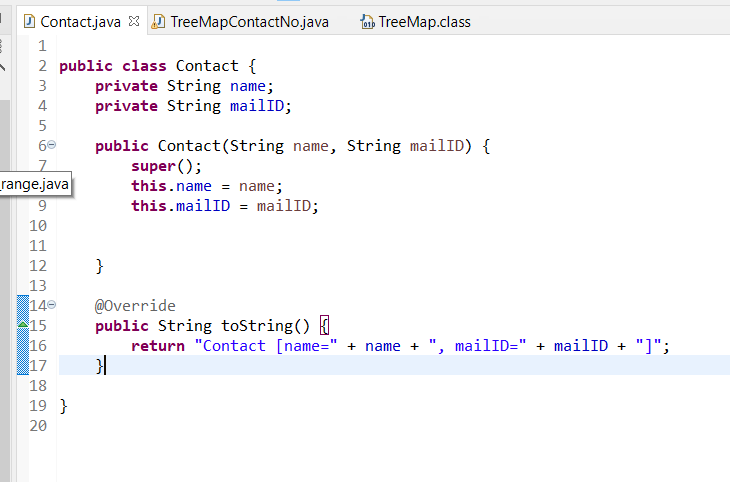
System.***out***.println(key);

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

});

}

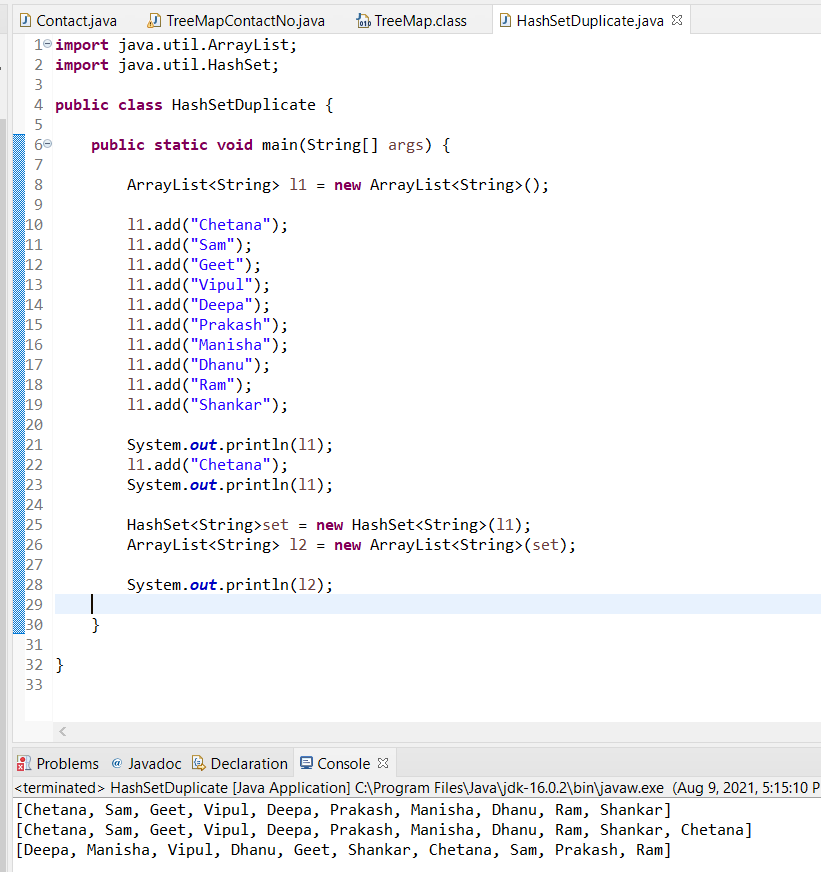
}



**Output:**



**2.**



**3.**

**import** java.util.Objects;

**public** **class** Employee11 {

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

**private** String name;

**private** **final** String Department;

**private** **double** Salary;

**public** Employee11(**int** id, String name, String Department, **double** Salary) {

**this**.id = id;

**this**.name = name;

**this**.Department = Department;

**this**.Salary = Salary;

}

**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 getDepartment() {

**return** Department;

}

**public** **double** getSalary() {

**return** Salary;

}

**public** **void** setSalary(**double** Salary) {

**this**.Salary = Salary;

}

@Override

**public** **boolean** equals(Object o) {

**if** (**this** == o) **return** **true**;

**if** (o == **null** || getClass() != o.getClass()) **return** **false**;

Employee employee = (Employee) o;

**return** id == employee.id;

}

@Override

**public** **int** hashCode() {

**return** Objects.*hash*(id);

}

**public** **int** Employee11(Employee11 employee) {

**return** **this**.getId() - employee.getId();

}

@Override

**public** String toString() {

**return** "Employee{" +

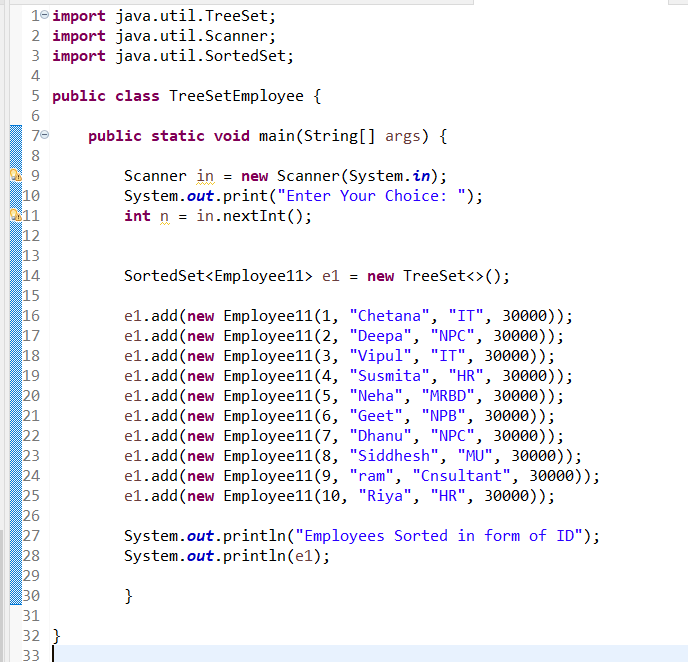
"id=" + id +

", name=" + name + ", Department= "+Department+" Salary= "+Salary +

'}';

}

}



**4.**

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

**class** Birth{

**int** day;

**int** month;

**int** year;

**public** Birth(**int** day, **int** month, **int** year) {

**super**();

**this**.day = day;

**this**.month = month;

**this**.year = year;

}

**public** **int** getDay() {

**return** day;

}

**public** **int** getMonth() {

**return** month;

}

**public** **int** getYear() {

**return** year;

}

**public** **void** LeapYear() {

**boolean** leap = **false**;

**if**(year%4 == 0) {

**if**(year%100 == 0) {

**if**(year%400 == 0)

leap = **true**;

**else**

leap=**false**;

}

**else**

leap=**true**;

}

**else**

leap = **false**;

**if**(leap)

System.***out***.println("It is a leap year");

**else**

System.***out***.println("It is not a leap year");

}

}

**public** **class** Linked\_Birth\_Leap\_Year {

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

LinkedList<Birth> l1 = **new** LinkedList<Birth>();

Birth d1 = **new** Birth(23,2,2000);

Birth d2 = **new** Birth(3,9,2017);

l1.add(d1);

l1.add(d2);

System.***out***.println("leap year");

**for**(Birth element : l1) {

System.***out***.println("date of Birth: "+element.getDay()+"/"+element.getMonth()+"/"+element.getYear());

element.LeapYear();

}

}

}

