COLLECTION ASSIGNMENT

1)Contact:

```
public class Contact{
    String name;
    String email;
    Enum Gender;

    public Contact(String name,String email,Enum gender) {
        this.name=name;
        this.email=email;
        this.Gender=gender;
    }



    public String getName() {
        return name;
    }

    public void setName(String name) {
        this.name = name;
    }

    public String getEmail() {
        return email;
    }

    public void setEmail(String email) {
        this.email = email;
    }


    public Enum getGender() {
        return Gender;
    }

    public void setGender(Enum gender) {
        Gender = gender;
     }

     public String toString()
     {
         return "Contact [name="+name+",email="+email+",gender="+Gender+"]";
     }
}
```

```
import java.util.Collections;

class Main {

    enum gender{
        Female, Male
    }

    public static void main(String[] args) {

        Main.gender gender = null;
        gender f1=gender.Female;
        gender m1=gender.Male;

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

            Contact Shruti = new Contact("Shruti","shruti170799@gmail.com",f1);
            Contact Sakshi = new Contact("Sakshi","sakshi@gmail.com",f1);
            Contact Rohit = new Contact("Rohit","rohit@gmail.com",m1);
            Contact Amit = new Contact("Amit","amit@gmail.com",m1);

            contact.put((long) 802819281, Shruti);
            contact.put((long) 987291792 , Sakshi);
            contact.put((long) 896273627, Rohit);
            contact.put((long) 902617628, Amit);

            Set set = contact.entrySet();
            Iterator i = set.iterator();

            while(i.hasNext()) {
                Map.Entry me = (Map.Entry)i.next();
                System.out.println("  Phone No. " + me.getKey());
                System.out.println(   me.getValue());
                System.out.println("  Phone No. " + me.getKey() + me.getValue());
            }
```

a)Fetch keys:

```
Problems  Javadoc  Declaration  Consol
<terminated> Main (7) [Java Application] C:\Program
  Phone No. 987291792
  Phone No. 902617628
  Phone No. 896273627
  Phone No. 802819281
```

b)Fetch Values:

```
Problems  Javadoc  Declaration  Console  Error Log
<terminated> Main (7) [Java Application] C:\Program Files\Java\jdk-13.0.2\bin\javaw.exe
Contact [name=Sakshi,email=sakshi@gmail.com,gender=Female]
Contact [name=Amit,email=amit@gmail.com,gender=Male]
Contact [name=Rohit,email=rohit@gmail.com,gender=Male]
Contact [name=Shruti,email=shruti170799@gmail.com,gender=Female]
```

c)Fetch Keys and Values pair:

```
Problems  Javadoc  Declaration  Console  Error Log
<terminated> Main (7) [Java Application] C:\Program Files\Java\jdk-13.0.2\bin\javaw.exe  (09-Aug-2021, 8:17:32 pm – 8:17:35 pm)
{987291792=Contact [name=Sakshi,email=sakshi@gmail.com,gender=Female], 902617628=Contact [name=Amit,email=amit@gmail.com,gender=Male],
```

```
Error Log
s\Java\jdk-13.0.2\bin\javaw.exe  (09-Aug-2021, 8:17:32 pm – 8:17:35 pm)
 896273627=Contact [name=Rohit,email=rohit@gmail.com,gender=Male], 802819281=Contact [name=Shruti,email=shruti170799@gmail.com,gender=Female]}
```

2)Duplicate Items:

```
1  import java.util.ArrayList;
4  public class Duplicate {
5
6      public static void main(String[] args) {
7
8          List<String> foodlist = new ArrayList<String>();
9          foodlist.add("Pizza");
10         foodlist.add("Burger");
11         foodlist.add("Fries");
12         foodlist.add("Samosa");
13         foodlist.add("Vadapav");
14         foodlist.add("Kachori");
15         foodlist.add("Sweet");
16         foodlist.add("Dhokla");
17         foodlist.add("Gulabjamun");
18         foodlist.add("Khandvi");
19         //trying to add Duplicate Item
20         foodlist.add("Fries");
21         //duplicate item will not be printed in Treeset
22         TreeSet<String> treeset=new TreeSet<String>(foodlist);
23         System.out.println(treeset);
24
25     }
26
27 }
```

Output:

```
Problems  Javadoc  Declaration  Console  Error Log
<terminated> Duplicate [Java Application] C:\Program Files\Java\jdk-13.0.2\bin\javaw.exe  (09-Aug-2021, 8:18:59 pm –
[Burger, Dhokla, Fries, Gulabjamun, Kachori, Khandvi, Pizza, Samosa, Sweet, Vadapav]
```

3)Employee Sorting:

a)Employee class:

```java
import java.util.Comparator;
import java.util.Iterator;
import java.util.Objects;
import java.util.Scanner;
import java.util.Set;
import java.util.TreeSet;

class Employee implements Comparable<Employee>{
    int ID;
    String Name;
    String Department;
    int Salary;
    public Employee(int iD, String name, String department, int salary) {
        super();
        ID = iD;
        Name = name;
        Department = department;
        Salary = salary;
    }
    public int getID() {
        return ID;
    }
    public void setID(int iD) {
        ID = iD;
    }
    public String getName() {
        return Name;
    }
    public void setName(String name) {
        Name = name;
    }

    public String getDepartment() {
        return Department;
    }
    public void setDepartment(String department) {
        Department = department;
    }
    public int getSalary() {
        return Salary;
    }
    public void setSalary(int salary) {
        Salary = salary;
    }


    @Override
    public int hashCode() {
        return Objects.hash(Department, ID, Name, Salary);
    }
    @Override
    public boolean equals(Object obj) {
        if (this == obj)
            return true;
        if (obj == null)
            return false;
        if (getClass() != obj.getClass())
            return false;
        Employee other = (Employee) obj;
        return Objects.equals(Department, other.Department) && ID == other.ID && Objects.equals(Name, other.Name)
                && Salary == other.Salary;
    }

    @Override
    public int compareTo(Employee o) {
        // TODO Auto-generated method stub
        return this.getID() - o.getID();
    }
    @Override
    public String toString() {
        return "Employee [ID=" + ID + ", Name=" + Name + ", Department=" + Department + ", Salary=" + Salary + "]";
    }


}
```

b)sorting class:

```java
import java.util.Iterator;
import java.util.Scanner;
import java.util.Set;
import java.util.TreeSet;
import java.util.Comparator;
import java.util.Objects;

public class Sorting {

    public static void main(String[] args) {

        Scanner scan=new Scanner(System.in);
        String ch;
        System.out.println("Run Application: \n a)ID \n b)Name \n c)Department \n d)Salary");
        System.out.println("Select one option to Run Application : ");
        ch=scan.next();


        Set<Employee> set=new TreeSet<>();
        set.add(new Employee(1,"Shruti","JFSA",200000));
        set.add(new Employee(3,"Rohit","CS",220000));
        set.add(new Employee(2,"Ritesh","IT",310000));
        set.add(new Employee(5,"Akash","JFS",230000));
        set.add(new Employee(4,"Yogesh","MECH",250000));
        set.add(new Employee(6,"Fheba","ML",340000));
        set.add(new Employee(7,"Ankita","AI",270000));
        set.add(new Employee(9,"Pooja","SCIENCE",260000));
        set.add(new Employee(8,"Nikita","COMPS",290000));
        set.add(new Employee(10,"Ruchira","IOT",280000));

        if(ch.equals("a")) {
            //System.out.println(set);
            Iterator<Employee> it=set.iterator();
            while(it.hasNext()) {
                System.out.println(it.next());
            }
        }
        else if(ch.equals("b")) {

            set = new TreeSet<>(Comparator.comparing(Employee::getName));
            set.add(new Employee(1,"Shruti","JFSA",200000));
            set.add(new Employee(3,"Rohit","CS",220000));
            set.add(new Employee(2,"Ritesh","IT",310000));
            set.add(new Employee(5,"Akash","JFS",230000));
            set.add(new Employee(4,"Yogesh","MECH",250000));
            set.add(new Employee(6,"Fheba","ML",340000));
            set.add(new Employee(7,"Ankita","AI",270000));
            set.add(new Employee(9,"Pooja","SCIENCE",260000));
            set.add(new Employee(8,"Nikita","COMPS",290000));
            set.add(new Employee(10,"Ruchira","IOT",280000));


            //System.out.println(set);
            Iterator<Employee> it=set.iterator();
            while(it.hasNext()) {
                System.out.println(it.next());
            }

        }
        else if(ch.equals("c")) {
```

```
            set = new TreeSet<>(Comparator.comparing(Employee::getDepartment));
            set.add(new Employee(1,"Shruti","JFSA",200000));
            set.add(new Employee(3,"Rohit","CS",220000));
            set.add(new Employee(2,"Ritesh","IT",310000));
            set.add(new Employee(5,"Akash","JFS",230000));
            set.add(new Employee(4,"Yogesh","MECH",250000));
            set.add(new Employee(6,"Fheba","ML",340000));
            set.add(new Employee(7,"Ankita","AI",270000));
            set.add(new Employee(9,"Pooja","SCIENCE",260000));
            set.add(new Employee(8,"Nikita","COMPS",290000));
            set.add(new Employee(10,"Ruchira","IOT",280000));

            //System.out.println(set);
            Iterator<Employee> it=set.iterator();
            while(it.hasNext()) {
                System.out.println(it.next());
            }

        }
        else if(ch.equals("d")) {

            set = new TreeSet<>(Comparator.comparing(Employee::getSalary));
            set.add(new Employee(1,"Shruti","JFSA",200000));
            set.add(new Employee(3,"Rohit","CS",220000));
            set.add(new Employee(2,"Ritesh","IT",310000));
            set.add(new Employee(5,"Akash","JFS",230000));
            set.add(new Employee(4,"Yogesh","MECH",250000));
            set.add(new Employee(6,"Fheba","ML",340000));
            set.add(new Employee(7,"Ankita","AI",270000));
            set.add(new Employee(9,"Pooja","SCIENCE",260000));
            set.add(new Employee(8,"Nikita","COMPS",290000));
            set.add(new Employee(10,"Ruchira","IOT",280000));

            //System.out.println(set);
            Iterator<Employee> it=set.iterator();
            while(it.hasNext()) {
                System.out.println(it.next());
            }
            }

    }

}
```

Output:

```
Problems  Javadoc  Declaration  Console  Error Log
<terminated> Sorting [Java Application] C:\Program Files\Java\jdk-13.0.2\bin\javaw.exe  (09-Aug-20
Run Application:
 a)ID
 b)Name
 c)Department
 d)Salary
Select one option to Run Application :
a
Employee [ID=1, Name=Shruti, Department=JFSA, Salary=200000]
Employee [ID=2, Name=Ritesh, Department=IT, Salary=310000]
Employee [ID=3, Name=Rohit, Department=CS, Salary=220000]
Employee [ID=4, Name=Yogesh, Department=MECH, Salary=250000]
Employee [ID=5, Name=Akash, Department=JFS, Salary=230000]
Employee [ID=6, Name=Fheba, Department=ML, Salary=340000]
Employee [ID=7, Name=Ankita, Department=AI, Salary=270000]
Employee [ID=8, Name=Nikita, Department=COMPS, Salary=290000]
Employee [ID=9, Name=Pooja, Department=SCIENCE, Salary=260000]
Employee [ID=10, Name=Ruchira, Department=IOT, Salary=280000]
```

Problems Javadoc Declaration Console Error Log

<terminated> Sorting [Java Application] C:\Program Files\Java\jdk-13.0.2\bin\javaw.exe (09-

```
Run Application:
 a)ID
 b)Name
 c)Department
 d)Salary
Select one option to Run Application :
b
Employee [ID=5, Name=Akash, Department=JFS, Salary=230000]
Employee [ID=7, Name=Ankita, Department=AI, Salary=270000]
Employee [ID=6, Name=Fheba, Department=ML, Salary=340000]
Employee [ID=8, Name=Nikita, Department=COMPS, Salary=290000]
Employee [ID=9, Name=Pooja, Department=SCIENCE, Salary=260000]
Employee [ID=2, Name=Ritesh, Department=IT, Salary=310000]
Employee [ID=3, Name=Rohit, Department=CS, Salary=220000]
Employee [ID=10, Name=Ruchira, Department=IOT, Salary=280000]
Employee [ID=1, Name=Shruti, Department=JFSA, Salary=200000]
Employee [ID=4, Name=Yogesh, Department=MECH, Salary=250000]
```

Problems Javadoc Declaration Console Error Log

<terminated> Sorting [Java Application] C:\Program Files\Java\jdk-13.0.2\bin\javaw.exe

```
Run Application:
 a)ID
 b)Name
 c)Department
 d)Salary
Select one option to Run Application :
c
Employee [ID=7, Name=Ankita, Department=AI, Salary=270000]
Employee [ID=8, Name=Nikita, Department=COMPS, Salary=290000]
Employee [ID=3, Name=Rohit, Department=CS, Salary=220000]
Employee [ID=10, Name=Ruchira, Department=IOT, Salary=280000]
Employee [ID=2, Name=Ritesh, Department=IT, Salary=310000]
Employee [ID=5, Name=Akash, Department=JFS, Salary=230000]
Employee [ID=1, Name=Shruti, Department=JFSA, Salary=200000]
Employee [ID=4, Name=Yogesh, Department=MECH, Salary=250000]
Employee [ID=6, Name=Fheba, Department=ML, Salary=340000]
Employee [ID=9, Name=Pooja, Department=SCIENCE, Salary=260000]
```

```
Run Application:
 a)ID
 b)Name
 c)Department
 d)Salary
Select one option to Run Application :
d
Employee [ID=1, Name=Shruti, Department=JFSA, Salary=200000]
Employee [ID=3, Name=Rohit, Department=CS, Salary=220000]
Employee [ID=5, Name=Akash, Department=JFS, Salary=230000]
Employee [ID=4, Name=Yogesh, Department=MECH, Salary=250000]
Employee [ID=9, Name=Pooja, Department=SCIENCE, Salary=260000]
Employee [ID=7, Name=Ankita, Department=AI, Salary=270000]
Employee [ID=10, Name=Ruchira, Department=IOT, Salary=280000]
Employee [ID=8, Name=Nikita, Department=COMPS, Salary=290000]
Employee [ID=2, Name=Ritesh, Department=IT, Salary=310000]
Employee [ID=6, Name=Fheba, Department=ML, Salary=340000]
```

4)Date Of Birth (leap Year):

```
1  import java.util.ArrayList;
8
9  public class Date {
10
11     public static void main(String[] args) {
12         {
13
14         LocalDate date1 = LocalDate.of(2000, 12, 23);
15         LocalDate date2 = LocalDate.of(2001, 12, 23);
16
17         Collection<Object> obj=new LinkedList<>();
18         obj.add(date1);
19         obj.add(date2);
20         for(Object i: obj) {
21
22             int a,c;
23             int y=date1.getYear();
24             int y1=date2.getYear();
25             if(y!=0)
26             {
27             a=(y%400==0)?(c=1):(y%100==0)?(c=0):((y%4==0)?(c=1):(c=0));
28             if(a==1)
29                 System.out.println("Your Date of Birth is " +date1+" and it was a leap year");
30             else
31                 System.out.println("Your Date of Birth is " +date1+" and it was not a leap year");
32             if(y1!=0)
33             {
34             a=(y1%400==0)?(c=1):(y1%100==0)?(c=0):((y1%4==0)?(c=1):(c=0));
35             if(a==1)
36                 System.out.println("Your Date of Birth is " +date2+" and it was a leap year");
37             else
38                 System.out.println("Your Date of Birth is " +date2+" and it was not a leap year");
39
40
41             Iterator<Object> itr=obj.iterator();
42             while(itr.hasNext()) {
43                             }
44
45             }
46         }
47         }
48     }
49 }
50 }
51
```

Output:

```
Problems  Javadoc  Declaration  Console  Error Log
Date [Java Application] C:\Program Files\Java\jdk-13.0.2\bin\javaw.exe  (09-Aug-2021,
Your Date of Birth is 2000-12-23 and it was a leap year
Your Date of Birth is 2001-12-23 and it was not a leap year
```