

## Assignment 2

```
import java.util.ArrayList;
import java.util.Collections;
import java.util.Comparator;
import java.util.HashMap;
import java.util.List;
import java.util.Map;

class Employee{
    String name;
    int age;
    Gender gender;
    double salary;//salary per month
    public Employee(String name,int age,Gender gender,double salary){
        this.name = name;
        this.age = age;
        this.gender = gender;
        this.salary = salary;
    }
    public String getName(){
        return name;
    }
    public void setName(String name){
        this.name = name;
    }
    public void raiseSalary(double byPercent){
        salary=salary*byPercent/100.0;
    }
    public String toString(){
        return name + "---" + "salary is =" + salary;
    }
}
enum Gender{
    MALE,
    FEMALE;
}

public class Assignment2 {

    /*Write a method to calculate the social Security Tax of an
employee and print it
    */
    public double socialSecurityTax(Employee employee){
        double tax=0.0;
        if(employee.salary <= 8900){
            tax = 0.062*employee.salary;
        }else{
```

```

        tax = 106800 * 0.062;
    }
    System.out.println(employee.name+"'s social security tax is
"+tax);
    return tax;
}
/*write a method to calculate an employee's contribution for
insurance coverage and print it.
* */
public double insuranceCoverage(Employee employee){
    double rate=0.0;
    if(employee.age <35){
        rate = 0.03 * employee.salary;
    }else if(employee.age >=35 && employee.age <= 50){
        rate = 0.04 * employee.salary;
    }else if(employee.age >50 && employee.age <60){
        rate = 0.05 * employee.salary;
    }else{
        rate = 0.06 * employee.salary;
    }
    System.out.println(employee.name+" 's contribution for
insurance is " + rate);
    return rate;
}
/*Write a method to sort three employees' salary from low to
high,and then print their name
* in order.
* */
private static Comparator<Map.Entry<String,Double>>
bySalaryAssendingOrder =
Comparator.comparing(Map.Entry<String,Double>::getValue);
public void sortSalary(Employee e1,Employee e2, Employee e3){
    HashMap<String,Double> salaryMap = new HashMap<>();
    salaryMap.put(e1.name, e1.salary);
    salaryMap.put(e2.name, e2.salary);
    salaryMap.put(e3.name, e3.salary);
    List<Map.Entry<String,Double>> salarySort=new
ArrayList<>(salaryMap.entrySet());
    Collections.sort(salarySort,bySalaryAssendingOrder);
    List<String> res=new ArrayList<>();
    for(int i=0;i<salarySort.size();i++){
        res.add(salarySort.get(i).getKey());
    }
    System.out.println(res);
}
/*write a method to raise an employee's salary to three times of
his/her original salary.

```

```

    */
    public void tripleSalary(Employee employee){
        System.out.println("before triple");
        System.out.println(employee.toString());
        employee.raiseSalary(300.0);
        System.out.println("after triple");
        System.out.println(employee.toString());
    }
    public static void main(String[] args) {
        Assignment2 assignment=new Assignment2();
        Employee e1=new Employee("alice",21,Gender.FEMALE,9000.0);
        assignment.socialSecurityTax(e1);
        assignment.insuranceCoverage(e1);
        Employee e2=new Employee("peter",26,Gender.MALE,7000.0);
        Employee e3=new Employee("kate",28,Gender.FEMALE,5000.0);
        assignment.sortSalary(e1, e2, e3);
        assignment.tripleSalary(e2);

    }
}

```

//Extra credit

```

/**
 * I have written some code below. What I want is to swap two
Employee objects.
 * One is Jenny and one is John. But after running it, I got the
result below:
 * Before: a=Jenny
 * Before: b=John
 * After: a=Jenny
 * After: b=John
 * There is no change after swap()! Do you know the reason why my
swap failed?
 * Write your understanding of the reason and explain it.
 */
/*
write your understanding here.
*/
public static void main(String[] args) {
    Employee a = new Employee("Jenny", 20, Gender.FEMALE, 2000);
    Employee b = new Employee("John", 30, Gender.MALE, 2500);
    System.out.println("Before: a=" + a.getName());
    System.out.println("Before: b=" + b.getName());
    swap(a, b);
    System.out.println("After: a=" + a.getName());
    System.out.println("After: b=" + b.getName());
}

```

We can not swap an object like what we do in primitive data types, because java doesn't pass method argument by reference; Instead, it passes them by value. In order to swap an object, we can have two solutions, first is to new a temp object and swap each of the attributes. Second is to use wrapper class.

Solution 1:

```
public static void swap(Employee x, Employee y) {
    Employee temp = new
Employee(x.name,x.age,x.gender,x.salary);

    x.name=y.name;
    x.age=y.age;
    x.gender=y.gender;
    x.salary=y.salary;
    y.name=temp.name;
    y.age=temp.age;
    y.gender=temp.gender;
    y.salary=temp.salary;
}
```

Solution 2:

```
class employeeWrapper{
    Employee employee;
    employeeWrapper(Employee employee){
        this.employee = employee;
    }
    public static void swap(employeeWrapper x,employeeWrapper y){
        Employee temp=x.employee;
        x.employee = y.employee;
        y.employee =temp;
    }
}
```

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

```
    Employee a = new Employee("Jenny", 20, Gender.FEMALE, 2000);
    Employee b = new Employee("John", 30, Gender.MALE, 2500);
    System.out.println("Before: a=" + a.getName());
    System.out.println("Before: b=" + b.getName());
    // swap(a, b);
    employeeWrapper ew1=new employeeWrapper(a);
    employeeWrapper ew2=new employeeWrapper(b);
    swap(ew1,ew2);
```

```
    System.out.println("After: a=" + ew1.employee.getName());
    System.out.println("After: b=" + ew2.employee.getName());
```

```
}
```