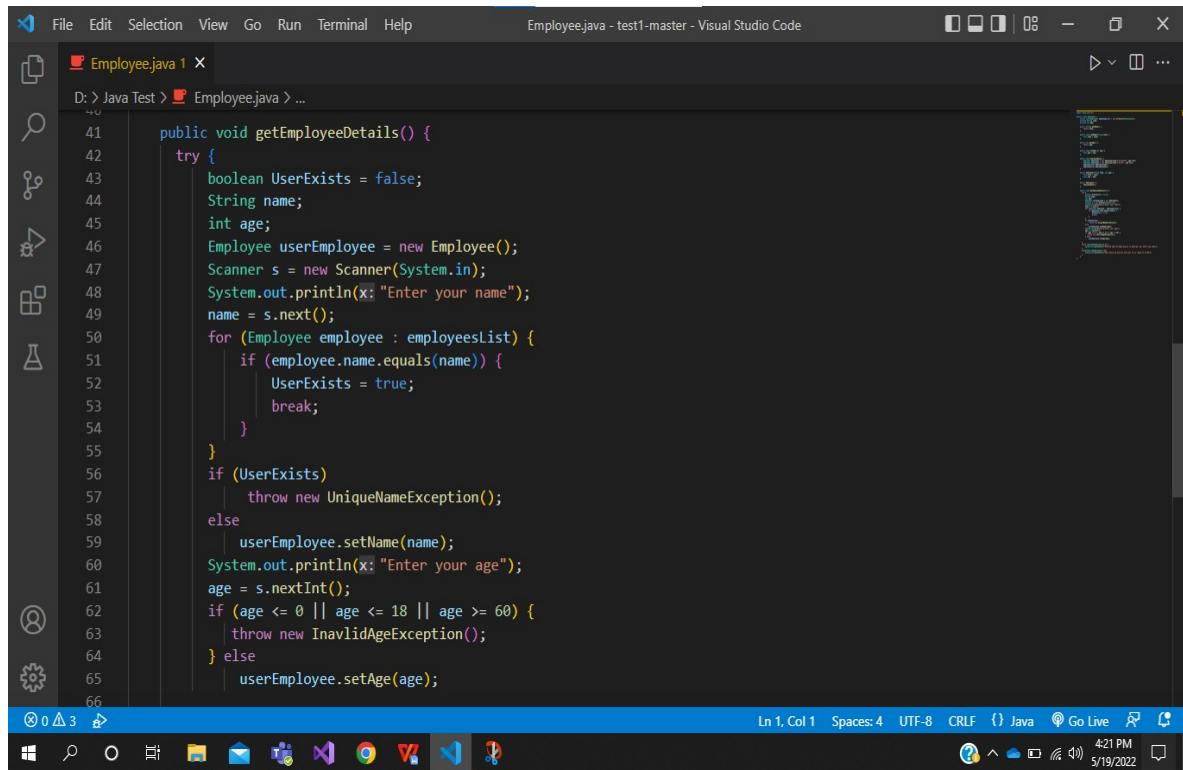


## Question :1

```
1 package com.softura.test1.question1;
2
3 import java.util.Scanner;
4
5 public class InsurancePremium {
6     public void PrintPremiumAmount(){
7         Scanner scanner = new Scanner(System.in);
8         System.out.println(x: "Please enter number of customers");
9         int numberOfCustomers = scanner.nextInt();
10
11         int[] PremiumAmountArray = new int[numberOfCustomers];
12
13         double totalAmount = 0;
14
15         for (int i = 0; i < PremiumAmountArray.length; i++){
16             System.out.println(x: "Enter premium amount type: ");
17             String premiumType = scanner.next();
18             System.out.println(x: "Enter premium amount ");
19             double premiumAmount = scanner.nextDouble();
20             scanner.nextLine();
21
22             System.out.println("Final Amount"+totalAmount);
23
24             switch (premiumType) {
25                 case "Monthly" :{
26                     double monthlytotalAmount = premiumAmount / 12;
```

```
3
Enter premium amount type:
Annual
Enter premium amount
5000
Final Amount0.0
Final Amount5000.0
Enter premium amount type:
BiAnnual
Enter premium amount
6000
Final Amount5000.0
Final Amount8000.0
Final Amount14000.0
Enter premium amount type:
Monthly
Enter premium amount
8000
Final Amount14000.0
Final Amount14666.666666666666
Final Amount16666.666666666664
Final Amount20666.666666666664
Final Amount28666.666666666664
Total premium amount Paid by all Customers are 28666.666666666664
PS D:\Java Test>
```

## Question :2



```
41 public void getEmployeeDetails() {
42     try {
43         boolean UserExists = false;
44         String name;
45         int age;
46         Employee userEmployee = new Employee();
47         Scanner s = new Scanner(System.in);
48         System.out.println(x: "Enter your name");
49         name = s.next();
50         for (Employee employee : employeeslist) {
51             if (employee.name.equals(name)) {
52                 UserExists = true;
53                 break;
54             }
55         }
56         if (UserExists)
57             throw new UniqueNameException();
58         else
59             userEmployee.setName(name);
60         System.out.println(x: "Enter your age");
61         age = s.nextInt();
62         if (age <= 0 || age <= 18 || age >= 60) {
63             throw new InavlidAgeException();
64         } else
65             userEmployee.setAge(age);
66     }
```

```
Enter your name
Lakshmi
Enter your age
-9
Age should be postive and must be in range of 18-60
PS D:\Java Test> |
```

```
Enter your name
Srikanth
Entered name Already Exists in Records! pls enter new name
PS D:\Java Test> |
```

### Question :3

```
SalariedWorker.java X
src > main > java > com > softura > test1 > question3 > model > SalariedWorker.java > SalariedWorker > pay(double)
1 package com.softura.test1.question3.model;
2
3 import lombok.EqualsAndHashCode;
4
5 @EqualsAndHashCode(callSuper = false)
6 public class SalariedWorker extends Worker{
7     private static int noOfWorkingHoursInWeek = 40;
8     private double workerGetsAmount;
9
10
11
12     @Override
13     public double pay(double pay) {
14
15         return noOfWorkingHoursInWeek * pay;
16     }
17
18     public static int getNoOfWorkingHoursInWeek() {
19         return noOfWorkingHoursInWeek;
20     }
21
22     public static void setNoOfWorkingHoursInWeek(int noOfWorkingHoursInWeek) {
23         SalariedWorker.noOfWorkingHoursInWeek = noOfWorkingHoursInWeek;
24     }
25
26     public double getWorkerGetsAmount() {
```

```
src > main > java > com > softura > test1 > question3 > utils > App.java > App > main(String[])
12
Run | Debug
13 public static void main(String[] args){
14     Scanner scanner = new Scanner(System.in);
15     System.out.println(x: "Enter type of employee");
16     Worker worker = new DailyWorker();
17     worker.setTypeOfEmployee(scanner.next());
18     System.out.println(x: "Enter number of employee");
19     int numberOfEmp = scanner.nextInt();
20     DailyWorker[] dailyWorkers = new DailyWorker[numberOfEmp];
21     SalariedWorker[] salariedWorkers = new SalariedWorker[numberOfEmp];
22     DailyWorker dailyWorker = null;
23     SalariedWorker salariedWorker = null;
24
25     if(Objects.equals(worker.getTypeOfEmployee(), b: "DailyWorker")){
26         for(int i = 0; i<dailyWorkers.length; i++){
27             dailyWorker = new DailyWorker();
28             System.out.println(x: "Enter Employee name");
29             dailyWorker.setName(scanner.next());
30             scanner.nextLine();
31             System.out.println(x: "Enter salary rate");
32             dailyWorker.setSalaryRate(scanner.nextDouble());
33             scanner.nextLine();
34             System.out.println(x: "Enter start appointment yyyy-mm-dd");
35             dailyWorker.setStartDate(LocalDate.parse(scanner.nextLine()));
36             System.out.println(x: "Enter end appointment yyyy-mm-dd");
37             dailyWorker.setEndDate(LocalDate.parse(scanner.nextLine()));
```

```
Enter type of employee
SalariedWorker
Enter number of employee
2
Enter Employee name
Srikanth
Enter salary rate
200
Enter Employee name
Kiran
Enter salary rate
500
```

## Question :4

```
@EqualsAndHashCode(callSuper = false)
public class Courses {
    public Courses(String courseName, double coursePrice, CourseType courseType) {
        this.courseName = courseName;
        this.coursePrice = coursePrice;
        this.courseType = courseType;
    }

    private String courseName;
    private double coursePrice;
    private CourseType courseType;

    public Courses() {
    }

    public String getCourseName() {
        return courseName;
    }

    public void setCourseName(String courseName) {
        this.courseName = courseName;
    }

    public double getCoursePrice() {
        return coursePrice;
    }
}
```

```
System.out.println(x: "Enter Employee id");
employee.setEmployeeId(scanner.nextInt());

System.out.println(x: "Enter Employee name");
employee.setEmployeeName(scanner.next());

System.out.println(x: "Enter course name");
employee.setCourseName(scanner.next());
if(Objects.equals(employee.getCourseName(), b: "java")){
    employee.setCoursePrice(coursePrice: 2000);
    employee.setCourseType(CourseType.valueOf("Online"));
}else if(Objects.equals(employee.getCourseName(), b: "python")){
    employee.setCoursePrice(coursePrice: 4000);
    employee.setCourseType(CourseType.valueOf("ClassRoomBased"));
}else{
    System.out.println(x: "Please provide a valid course");
}
employees[i] = employee;
}

for(Employee employeeObj : employees)
    return employeeObj;

return null;
}
```