Teacher Initial:
TAB

Daffodil International University

Spring 2021

Department of Computer Science and Engineering

Midterm Open Book Examination Answer Script

Full Marks: 25 Allowed, Time: 2:30 hrs.

Date: Sunday 7, March 2021

General Information (must be filled by the student)

COURSE CODE: CSE132 **SECTION:** PC_I **PROGRAM:** DAY EVEN

STUDENT ID: 201-15-3053 **TIME STARTED:** 9 AM **TIME ENDED:** 11:30 AM

write your detail on the paper using hand)

201-15-3053(PC_I)

**** Plagiarism will be checked while you submit your response. You are advised to be honest during the open book exam.**

Student ID:

**** Plagiarism will be checked while you submit your response. You are advised to be honest during the open book exam.**

Name: Tofayel Ahmed
ID: 201-15-3053 (PC-I)

Ans to the qa no-1

// Generally Java not support multiple inheritance without interface.
By the Interface we can. mutiple inheritance possible creating inteface class and implement by subclass.

Example,

```
class Airplane
  {

                    // Not support

  }
class car
 {

 }
class flyingcar extends Airplane, car
  { }
```

```
interface Airplane
{

}
interface car
{                                    // Supported

}
class flyingcar implements Airplane, car
{

}
```

Ans of the qu no-2

```
import java.util.Scanner;
pubic class vacation
{
Private String name;          // Atributes
Private int day;
Private int nameofvacation;
private string month;

}
.vacation (string n, int d , int nv , String m )

        {name = n;     // Setting value by
         day = d;                  // constructor
         nameofvacation = nv;
         month = m;

        }
```

```
public static void main (String args[ ])
{
Scanner sc = new Scanner (System.in);
String n = sc.nextLine();
int d = sc.nextInt(); //taking inpat
int nv = sc.nextInt();
String m = sc.nextline();
~~vacation VT = new vacation();~~
vacation VT = new vacation(n, d, nv, m);

}

}
```

Ans 2b

```
public bool SameName(vacation obj1, vacation obj2)
{
if(obj1.name == obj2.Name)
    {
     return true;
    }
else
    {
     return false;
    }
}
```

In this method it has two vacationtyp parameter.
calling this method it takes obJ1 and obJ2 object and compare name obJ1 == o of the object. After ending condition it's return true otherwise it returnrse false.

Ans of the c

c/

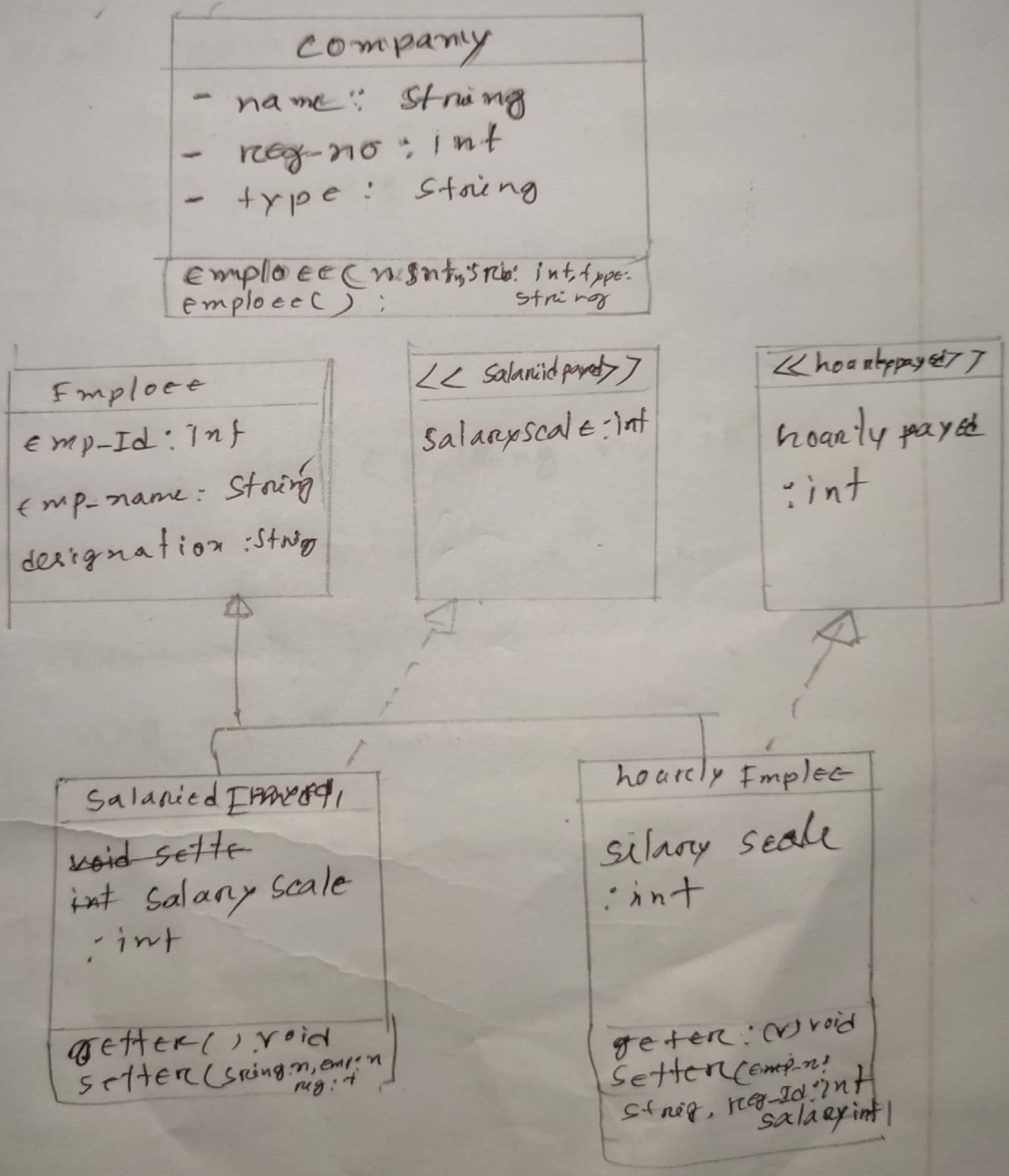
```
import Java.utill.scanner;
Public voidint calculate Vacdtion ( )
{
  Scanner sc = new Scanner(System.in)
  int numofvacation = sc. nextInt( );
  int percentage = numofvacation / 8
```

Ans to the qu no 2c

```
import java.util.Scanner;
public void calculatevacation()
{
Scanner sc = new Scanner(System.in);
numofvacation = sc.nextInt();
int percantage = num of vacation * 8/100;

}
```

Ans to the qu no - 3a



Ans to the qu no- 3b

```
class company
{
Private String name;

private int reg_no;

Private inf string type;

Private

}
public setter (String n, int r, String t)
{
        name = n;
        reg_no = r;
        type = t;

}
```

```
public ~~void~~ getter
public void getter ( )
{
System.out.println (name+" "+ " " + reg-no "
                    + " " + type);
}
class Employee
{
private int emp_id;
private String emp_name;
private String designation;
}
```

```
Employee Public setter (int e, String n, String d)
    {
Private emp_Id = e;
private name = n;
private designation = d;
    }
Public void getter
employee{
Employee System.out.println (emp_ID + " " + name + " "
                + " " + d);
    }
    System.out.println (em-ID + " " + name
                + " " + d);
}
interface Salaryscale
  {
```

~~class Hourly~~

class SalariedEmployee extends

Employee

implements Salary

{

int SalaryScale

void setSalary(int Salary)

{

salaryscale = salary;

}

}

interface { hourly_payment; }

class hrainly employe extend,

employee implements hourly payment;

{

void setter(ints){

salary = s;

}

```
void getter
{
System.out.println(salary);

}
```