CS1073
FR03B
Assignment #2

Daniyal Faheem Khan
3765942

Question I:

```
/**
 This class represents a salesperson.
 @author Daniyal Khan 3765942
*/
public class Salesperson {

     /**
     The name of the salesperson.
     */
     private String name;



     /**
     The hourly wage of the salesperson.
     */
     private double hourlyWage;



     /**
     The rate of commission of the salesperson (as a decimal).
     */
     private double commissionRate;


     /**
     Hours the salesperson has worked since their last pay.
     */
     private int numHours;


     /**
     Total value of sales the salesperson has made since their
last pay.
     */
     private int salesValue;




     /**
     This method constructs a salesperson with a specific hourly
wage and rate of commission.
     The hours worked and sales made are initially zero.
```

```
     @param nameIn The name of the salesperson.
     @param wageIn The hourly wage of the salesperson.
     @param commissionIn The commission of the salesperson (as a
decimal).
     */
     public Salesperson(String nameIn, double wageIn, double
commissionIn) {
          name = nameIn;
          hourlyWage = wageIn;
          commissionRate = commissionIn;
          numHours = 0;
          salesValue = 0;
     }



     /**
     This method retrieves the name of the salesperson.
     @return The salesperson's name.
     */
     public String getName() {
          return name;
     }



     /**
     This method retrieves the hourly wage of the salesperson.
     @return The salesperson hourly wage.
     */
     public double getWage() {
          return hourlyWage;
     }


     /**
     This method retrieves the commission rate of the
salesperson.
     @return The salesperson commission rate.
     */
     public double getCommission() {
          return commissionRate;
     }
```

```
/**
This method retrieves the hours the salesperson has worked.
@return The salesperson's hours since their last pay.
*/
public int getHours() {
     return numHours;
}


/**
This method retrieves the sales the salesperson has made.
@return The salesperson's sales since their last pay.
*/
public int getSales() {
     return salesValue;
}




/**
This method calculates the pay of the salesperson.
The sales and hours are reset to zero.
@return The salesperson's pay.
*/
public double calcPay() {
     double bonus = commissionRate * salesValue;
     double pay = hourlyWage * numHours;
     salesValue = 0;
     numHours = 0;
     return pay + bonus;
}




/**
This method increases the salesperson's hours.
@param hoursWorked Amount of hours worked.
*/
public void incHours(int hoursWorked) {
     numHours = numHours + hoursWorked;
}
```

```
     /**
     This method increases the salesperson's sales.
     @param salesMade Amount of sales made.
     */
     public void incSales(int salesMade) {
          salesValue = salesValue + salesMade;
     }

} //end Salesperson
```

Driver Class:

```
/**
 @author Daniyal Khan 3765942
*/

public class SalespersonDriver {
     public static void main(String[] args) {
          Salesperson person1 = new Salesperson("Luffy", 19.80,
2.4);
          Salesperson person2 = new Salesperson("Zoro", 16.50,
1.6);

          // adding the number of hours each person worked
          person1.incHours(40);
          person2.incHours(35);

          // the number of sales each person made
          person1.incSales(3);
          person2.incSales(5);

          // hours worked by each person
          System.out.println("Hours Worked by " +
person1.getName() + ": " + person1.getHours());
          System.out.println("Hours Worked by " +
person2.getName() + ": " + person2.getHours());

          // sales made by each person
          System.out.println("Sales made by " +
person1.getName() + ": " + person1.getSales());
          System.out.println("Sales made by " +
person2.getName() + ": " + person2.getSales());

          // pay of each person
```

```
          System.out.println("Pay of " + person1.getName() + ":
" + person1.calcPay());
          System.out.println("Pay of " + person2.getName() + ":
" + person2.calcPay());

          System.out.println();

          // test to see if the values have been reset to zero
          System.out.println("Hours Worked by " +
person1.getName() + ": " + person1.getHours());
          System.out.println("Hours Worked by " +
person2.getName() + ": " + person2.getHours());

          System.out.println("Sales made by " +
person1.getName() + ": " + person1.getSales());
          System.out.println("Sales made by " +
person2.getName() + ": " + person2.getSales());
     }
}
```

Output:

```
Hours Worked by Luffy: 40
Hours Worked by Zoro: 35
Sales made by Luffy: 3
Sales made by Zoro: 5
Pay of Luffy: 799.2
Pay of Zoro: 585.5

Hours Worked by Luffy: 0
Hours Worked by Zoro: 0
Sales made by Luffy: 0
Sales made by Zoro: 0
```

Question II:

```
/**
```

```
This class is for a resort that offers several activities to
their guests, each for a fee
@author Daniyal Khan 3765942
*/

public class ActivityTab {

     /**
     Name of the guest.
     */
     private String guest;

     /**
     Room Number of the guest.
     */
     private int roomNum;

     /**
     Tracks the owing amount of the guest.
     */
     private double owedAmount;

     /**
     Constructs a ActivityTab object given the guest's name and
the room number.
     @param guestIn name of the guest
     @param roomNumIn room number of the guest
     */
     public ActivityTab(String guestIn, int roomNumIn) {
          guest = guestIn;
          roomNum = roomNumIn;
          owedAmount = 0.0;
     }

     /**
     Returns the guest name.
     @return name of the guest
     */
     public String getName() {
          return guest;
     }

     /**
     Returns the guest's room number.
     @return room number of the guest
     */
     public int getRoomNumber() {
```

```
          return roomNum;
     }

     /**
     Returns the amount which the guest owes.
     @return amount owed by the guest
     */
     public double getOwedAmount() {
          return owedAmount;
     }

     /**
     Updates the cost of activity done by the guest as their
owed amount.
     @param activityCost cost of activity done by the guest
     */
     public void addActivityPrice(double activityCost) {
          owedAmount += activityCost;
     }

     /**
     Returns the tip amount after taking in tip percentage from
the guest.
     @param tipPercent tip percentage from the guest
     @return amount of tip
     */
     public double getTipAmount(int tipPercent) {
          double tip = tipPercent * 0.01;
          return owedAmount * tip;
     }
}
```

Driver Class:

```
/**
This is a driver class for ActivityTab
@author Daniyal Khan 3765942
*/

public class ComputerScienceRetreat{
     public static void main(String[] args) {
          ActivityTab anniesTab = new ActivityTab("Annie
Easley", 73);
          anniesTab.addActivityPrice(4.50);
```

```
          ActivityTab alansTab = new ActivityTab("Alans Turing",
342);
          alansTab.addActivityPrice(9.75);

          ActivityTab clarencesTab = new ActivityTab("Clarence
Ellis", 214);
          clarencesTab.addActivityPrice(6.00);
          clarencesTab.addActivityPrice(8.75);

          ActivityTab gracesTab = new ActivityTab("Grace
Hopper", 742);
          gracesTab.addActivityPrice(9.75);

          gracesTab.addActivityPrice(11.25);
          clarencesTab.addActivityPrice(11.25);

          alansTab.addActivityPrice(12.75);
          anniesTab.addActivityPrice(7.80);

          System.out.println("Guest's Name: " +
anniesTab.getName() + "\nRoom Number: " +
anniesTab.getRoomNumber()+ "\nAmount Owed: " +
anniesTab.getOwedAmount());

          System.out.println();
          System.out.println("Guest's Name: " +
alansTab.getName() + "\nRoom Number: " +
alansTab.getRoomNumber()+ "\nAmount Owed: " +
alansTab.getOwedAmount());

          System.out.println();
          System.out.println("Guest's Name: " +
clarencesTab.getName() + "\nRoom Number: " +
clarencesTab.getRoomNumber()+ "\nAmount Owed: " +
clarencesTab.getOwedAmount());

          System.out.println();
          System.out.println("Guest's Name: " +
gracesTab.getName() + "\nRoom Number: " +
gracesTab.getRoomNumber()+ "\nAmount Owed: " +
gracesTab.getOwedAmount());

          System.out.println();

          System.out.println(anniesTab.getName() + " Tip: " +
anniesTab.getTipAmount(20));
```

```
          System.out.println(alansTab.getName() + " Tip: " +
alansTab.getTipAmount(20));
          System.out.println(clarencesTab.getName() + " Tip: " +
clarencesTab.getTipAmount(18));
          System.out.println(gracesTab.getName() + " Tip: " +
gracesTab.getTipAmount(15));
     }
}
```

Output:

```
Guest's Name: Annie Easley
Room Number: 73
Amount Owed: 12.3

Guest's Name: Alans Turing
Room Number: 342
Amount Owed: 22.5

Guest's Name: Clarence Ellis
Room Number: 214
Amount Owed: 26.0

Guest's Name: Grace Hopper
Room Number: 742
Amount Owed: 21.0

Annie Easley Tip: 2.4600000000000004
Alans Turing Tip: 4.5
Clarence Ellis Tip: 4.68
Grace Hopper Tip: 3.15
```