Computer Program Solutions

QUESTION 6

Ethan Van Rensburg – System Development
Learnership Candidate

18 – 23 June 2024

Contents

a) Purpose Description:

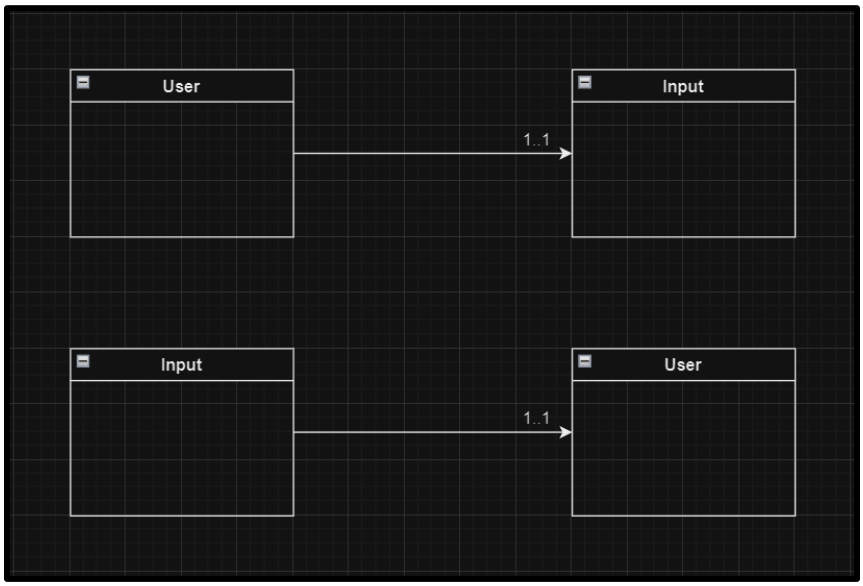
UrbanFurn had requested that the developer create a system for end-user delivery, below is an in-depth description of what was produced:

The program created was a Payroll calculation program that is used for end-user satisfaction and the delivery of results. The program begins its execution by prompting a user to enter their shift at UrbanFurn, then the program will ask for the suitable hours the user had worked during a week. After the hours and shifts are collected, the system then calculates the pay of the user based on the hourly rate per shift. The final results are shown in a display like below:

```
Please select your assigned shift:
1 - Morning Shift
2 - Day Shift
3 - Night Shift
1
How many hours do you work in a week?
40

Payroll:
----------
Hours Worked: 40
Shift: Morning
Hourly Pay Rate: R50.0
Regular Pay: R2000
Overtime Pay: R0.00
Total of Regular and Overtime: R2000
Retirment Deduction: R0.00
Net Pay: R2000
BUILD SUCCESSFUL (total time: 13 seconds)
```

The program designed is based solely on an input output system, which takes in certain information, processes the data given and produces suitable results and outcomes. The development process created a UML Diagram to visualize the relationship between users and outputs:



The program also utilized an interface object to process the information given and store them in a suitable text file for records and management.

```
try {
    String content = ("Hours Worked: " + hours + "\n");
    File fil = new File("C:\\Users\\ETHAN.V\\Documents\\NetBeansProjects\\Ethan_UrbanFurn\\Hours.txt");
    FileWriter fw = new FileWriter((fil.getAbsoluteFile()), true);
    BufferedWriter bw = new BufferedWriter(fw);
    bw.write(content);
    bw.close();
} catch (IOException e) {
    e.printStackTrace();
}
```

```
Hours Worked: 45
Hours Worked: 70
Hours Worked: 40
Hours Worked: 70
Hours Worked: 40
Hours Worked: 70
Hours Worked: 40
Hours Worked: 70
Hours Worked: 50
Hours Worked: 50
Hours Worked: 50
```

b) Code Layout:

Below is the format and structure of the code created and used for UrbanFurn's product:

```
package ethan_urbanfurn;

import java.util.Scanner;
import java.io.*;
import java.text.DecimalFormat;

public class Ethan_UrbanFurn {

    public static void main(String[] args) {
        //Create the needed objects for the methods and functions
        Scanner scan = new Scanner(System.in);
        Ethan_UrbanFurn plan = new Ethan_UrbanFurn();

        //Declare and initialize the variables for the hours and hourly pay
        int hours;
        double hourly1 = 50;
        double hourly2 = 70;
        double hourly3 = 90;

        //Prompt the user to enter their shift number
        System.out.println("""
                           Please select your assigned shift:
                           1 - Morning Shift
                           2 - Day Shift
                           3 - Night Shift""");
        //Create a variable to store their input for the different shifts
        //use scanner
        int shifChoice = Integer.parseInt(scan.nextLine());
```

```java
//Create a switch case for the 3 shifts and their corresponding cases
switch (shifChoice) {
    //Case 1 will link to shift 1 and complete the calculations
    case 1 -> {
        //Prompt the user to enter their number of work hours
        System.out.println("How many hours do you work in a week?");
        //Use the hours variable to store the input
        hours = scan.nextInt();
        //Create a try catch statement for FileWriter and BufferedWriter
        //to read/write the inputs into a text file for records
        try {
            String content = ("Hours Worked: " + hours + "\n");
            File fil = new File("C:\\Users\\ETHAN.V\\Documents\\NetBeansProjects\\Ethan_UrbanFurn\\Hours.txt");
            FileWriter fw = new FileWriter((fil.getAbsoluteFile()), true);
            BufferedWriter bw = new BufferedWriter(fw);
            bw.write(content);
            bw.close();
        } catch (IOException e) {
            e.printStackTrace();
        }
        //Create an if branch for the hours input and their corresponding calculations
        //If hours is less than or equal to 40 and not 0
        //Produce the calculations for regular pay and display the needed results
        if (hours <= 40 && hours != 0) {
            int weekPay = hours * 50;
            System.out.println();
            System.out.println("Payroll:");
            System.out.println("----------");
            System.out.println("Hours Worked: " + hours);
            System.out.println("Shift: Morning");
            System.out.println("Hourly Pay Rate: R" + hourly1);
            System.out.println("Regular Pay: R" + weekPay);
            System.out.println("Overtime Pay: R0.00");
            System.out.println("Total of Regular and Overtime: R" + weekPay);
            System.out.println("Retirment Deduction: R0.00");
            System.out.println("Net Pay: R" + weekPay);
        } //If hours is greater than 40 and not 0
```

```
        //Produce the calculations for overtime pay and display the needed results
        else if (hours > 40 && hours != 0) {
            int overTime = hours - 40;
            int calcOver = overTime * (int) (50 * 1.5);
            float weekOver = calcOver + (40 * 50);
            System.out.println();
            System.out.println("Payroll:");
            System.out.println("----------");
            System.out.println("Hours Worked: " + hours);
            System.out.println("Shift: Morning");
            System.out.println("Hourly Pay Rate: R" + hourly1);
            System.out.println("Regular Pay: R" + (40 * 50));
            System.out.println("Overtime Pay: R" + calcOver);
            System.out.println("Total of Regular and Overtime: R" + weekOver);
            System.out.println("Retirment Deduction: R0.00");
            System.out.println("Net Pay: R" + weekOver);
        } //If hours is invalid, then produce a message for invalid inputs
        else {
            System.out.println("Invalid input");
        }
    }
    //Case 2 will link to shift 2 and complete the calculations
    case 2 -> {
        //Prompt the user to enter their number of work hours
        System.out.println("How many hours do you work in a week?");
        //Use the hours variable to store the input
        hours = scan.nextInt();
        //Create a try catch statement for FileWriter and BufferedWriter
        //to read/write the inputs into a text file for records
        try {
            String content = ("Hours Worked: " + hours + "\n");
            File fil = new File("C:\\Users\\ETHAN.V\\Documents\\NetBeansProjects\\Ethan_UrbanFurn\\Hours.txt");
            FileWriter fw = new FileWriter((fil.getAbsoluteFile()), true);
            BufferedWriter bw = new BufferedWriter(fw);
            bw.write(content);
            bw.close();
        } catch (IOException e) {
            e.printStackTrace();
        }
```

```
        //Create an if branch for the hours input and their corresponding calculations
        //If hours is less than or equal to 40 and not 0
        //Produce the calculations for regular pay and display the needed results
        if (hours <= 40 && hours != 0) {
            int weekPay = hours * 70;
            //Call upon the retire2 method with the plan object
            plan.retire2(weekPay, hours, hourly2);
        } //If hours is greater than 40 and not 0
        //Produce the calculations for overtime pay and display the needed results
        else if (hours > 40 && hours != 0) {
            int overTime = hours - 40;
            double calcOver = overTime * (70 * 1.5);
            double weekOver = calcOver + (40 * 70);
            //Call upon the retireOver2 method with the plan object
            plan.retireOver2(weekOver, hours, calcOver, hourly2);
        } //If hours is invalid, then produce a message for invalid inputs
        else {
            System.out.println("Invalid input");
        }
    }
    //Case 2 will link to shift 2 and complete the calculations
    case 3 -> {
        //Prompt the user to enter their number of work hours
        System.out.println("How many hours do you work in a week?");
        //Use the hours variable to store the input
        hours = scan.nextInt();
        //Create a try catch statement for FileWriter and BufferedWriter
        //to read/write the inputs into a text file for records
        try {
            String content = ("Hours Worked: " + hours + "\n");
            File fil = new File("C:\\Users\\ETHAN.V\\Documents\\NetBeansProjects\\Ethan_UrbanFurn\\Hours.txt");
            FileWriter fw = new FileWriter((fil.getAbsoluteFile()), true);
            BufferedWriter bw = new BufferedWriter(fw);
            bw.write(content);
            bw.close();
        } catch (IOException e) {
            e.printStackTrace();
        }
```

```
            //Create an if branch for the hours input and their corresponding calculations
            //If hours is less than or equal to 40 and not 0
            //Produce the calculations for regular pay and display the needed results
            if (hours <= 40 && hours != 0) {
                int weekPay = hours * 90;
                //Call upon the retire3 method with the plan object
                plan.retire3(weekPay, hours, hourly3);
            } //If hours is greater than 40 and not 0
            //Produce the calculations for overtime pay and display the needed results
            else if (hours > 40 && hours != 0) {
                int overTime = hours - 40;
                double calcOver = overTime * (90 * 1.5);
                double weekOver = calcOver + (40 * 90);
                //Call upon the retireOver3 method with the plan object
                plan.retireOver3(weekOver, hours, calcOver, hourly3);
            } //If hours is invalid, then produce a message for invalid inputs
            else {
                System.out.println("Invalid input");
            }
        }
        //Set the default case of the switch as an Invalid Input message
        default ->
            System.out.println("Invalid input");
    }

}
```

```java
//Create a method to calculate the needed retirment deductions for shift 2
//with regular pay
//Call on the weekpay, hours and hourly2 variables
public void retire2(float weekPay, int hours, double hourly2) {
    //Use scanner for user input
    Scanner scan = new Scanner(System.in);

    //Prompt the user to choose if they want the retirement plan or not
    System.out.println("Do you wish to sign up for the retirement plan? (yes/no)");
    //Store the choice in a variable
    String choice = scan.nextLine();
    //Create an if branch for the choice variable and the calculations based
    //on the outcome
    //If the user chooses yes, the method then calculates the retirement from
    //the regular pay and net pay and displays all final results
    if ("yes".equalsIgnoreCase(choice)) {
        double retire = (weekPay * 0.05);
        double retPay = weekPay - retire;
        System.out.println();
        System.out.println("Payroll:");
        System.out.println("----------");
        System.out.println("Hours Worked: " + hours);
        System.out.println("Shift: Day");
        System.out.println("Hourly Pay Rate: R" + hourly2);
        System.out.println("Regular Pay: R" + weekPay);
        System.out.println("Overtime Pay: R0.00");
        System.out.printf("Total of Regular and Overtime: R%.2f", weekPay);
        System.out.printf("\nRetirment Deduction: R%.2f", retire);
        System.out.printf("\nNet Pay: R%.2f", retPay);
        System.out.println();
    } //If the user chooses no, the method then calculates without retirement from
```

```java
    //the regular pay and net pay and displays all final results
    else if ("no".equalsIgnoreCase(choice)) {
        System.out.println();
        System.out.println("Payroll:");
        System.out.println("----------");
        System.out.println("Hours Worked: " + hours);
        System.out.println("Shift: Day");
        System.out.println("Hourly Pay Rate: R" + hourly2);
        System.out.println("Regular Pay: R" + weekPay);
        System.out.println("Overtime Pay: R0.00");
        System.out.println("Total of Regular and Overtime: R" + weekPay);
        System.out.println("Retirment Deduction: R0.00");
        System.out.println("Net Pay: R" + weekPay);
    } //If the user enters an invalid choice, the program will show an invalid method
    else {
        System.out.println("Invalid Option");
    }
}
```

```java
//Create a method to calculate the needed retirment deductions for shift 2
//with overtime pay
//Call on the weekOver, hours, calcOver and hourly2 variables
public void retireOver2(double weekOver, int hours, double calcOver, double hourly2) {
    //Use scanner for user input
    Scanner scan = new Scanner(System.in);
    //Prompt the user to choose if they want the retirement plan or not
    System.out.println("Do you wish to sign up for the retirement plan? (yes/no)");
    //Store the choice in a variable
    String choice = scan.nextLine();
    //Create an if branch for the choice variable and the calculations based
    //on the outcome
    //If the user chooses yes, the method then calculates the retirement from
    //the overtime pay, regular pay and net pay and displays all final results
    if ("yes".equalsIgnoreCase(choice)) {
        double retire = (0.05 * weekOver);
        double retPay = weekOver - retire;
        System.out.println();
        System.out.println("Payroll:");
        System.out.println("----------");
        System.out.println("Hours Worked: " + hours);
        System.out.println("Shift: Day");
        System.out.println("Hourly Pay Rate: R" + hourly2);
        System.out.println("Regular Pay: R" + (40 * 70));
        System.out.println("Overtime Pay: R" + calcOver);
        System.out.printf("Total of Regular and Overtime: R%.2f", weekOver);
        System.out.printf("\nRetirment Deduction: R%.2f", retire);
        System.out.printf("\nNet Pay: R%.2f", retPay);
        System.out.println();
        //If the user chooses no, the method then calculates without retirement from
        //the overtime pay, regular pay and net pay and displays all final results
    } else if ("no".equalsIgnoreCase(choice)) {
```

```
            //the overtime pay, regular pay and net pay and displays all final results
        } else if ("no".equalsIgnoreCase(choice)) {
            System.out.println();
            System.out.println("Payroll:");
            System.out.println("----------");
            System.out.println("Hours Worked: " + hours);
            System.out.println("Shift: Day");
            System.out.println("Hourly Pay Rate: R" + hourly2);
            System.out.println("Regular Pay: R" + (40 * 70));
            System.out.println("Overtime Pay: R" + calcOver);
            System.out.println("Total of Regular and Overtime: R" + weekOver);
            System.out.println("Retirment Deduction: R0.00");
            System.out.println("Net Pay: R" + weekOver);
        } //If the user enters an invalid choice, the program will show an invalid method
        else {
            System.out.println("Invalid Option");
        }
    }
```

```java
//Create a method to calculate the needed retirment deductions for shift 3
//with regular pay
//Call on the weekPay, hours and hourly3 variables
public void retire3(float weekPay, int hours, double hourly3) {
    //Use scanner for user input
    Scanner scan = new Scanner(System.in);
    //Prompt the user to choose if they want the retirement plan or not
    System.out.println("Do you wish to sign up for the retirement plan? (yes/no)");
    //Store the choice in a variable
    String choice = scan.nextLine();
    //Create an if branch for the choice variable and the calculations based
    //on the outcome
    //If the user chooses yes, the method then calculates the retirement from
    //the regular pay and net pay and displays all final results
    if ("yes".equalsIgnoreCase(choice)) {
        double retire = (0.05 * weekPay);
        double retPay = weekPay - retire;
        System.out.println();
        System.out.println("Payroll:");
        System.out.println("----------");
        System.out.println("Hours Worked: " + hours);
        System.out.println("Shift: Night");
        System.out.println("Hourly Pay Rate: R" + hourly3);
        System.out.println("Regular Pay: R" + weekPay);
        System.out.println("Overtime Pay: R0.00");
        System.out.printf("Total of Regular and Overtime: R%.2f", weekPay);
        System.out.printf("\nRetirment Deduction: R%.2f", retire);
        System.out.printf("\nNet Pay: R%.2f", retPay);
        System.out.println();
    } //If the user chooses no, the method then calculates without retirement from
    //the regular pay and net pay and displays all final results
```

```
        else if ("no".equalsIgnoreCase(choice)) {
            System.out.println();
            System.out.println("Payroll:");
            System.out.println("----------");
            System.out.println("Hours Worked: " + hours);
            System.out.println("Shift: Night");
            System.out.println("Hourly Pay Rate: R" + hourly3);
            System.out.println("Regular Pay: R" + weekPay);
            System.out.println("Overtime Pay: R0.00");
            System.out.println("Total of Regular and Overtime: R" + weekPay);
            System.out.println("Retirment Deduction: R0.00");
            System.out.println("Net Pay: R" + weekPay);
        } //If the user enters an invalid choice, the program will show an invalid method
        else {
            System.out.println("Invalid Option");
        }
    }
```

```
    //Create a method to calculate the needed retirment deductions for shift 3
    //with overtime pay
    //Call on the weekOver, hours, calcOver and hourly3 variables
    public void retireOver3(double weekOver, int hours, double calcOver, double hourly3) {
        //Use scanner for user input
        Scanner scan = new Scanner(System.in);
        //Prompt the user to choose if they want the retirement plan or not
        System.out.println("Do you wish to sign up for the retirement plan? (yes/no)");
        //Store the choice in a variable
        String choice = scan.nextLine();
        //Create an if branch for the choice variable and the calculations based
        //on the outcome
        //If the user chooses yes, the method then calculates the retirement from
        //the overtime pay, regular pay and net pay and displays all final results
        if ("yes".equalsIgnoreCase(choice)) {
            double retire = (0.05 * weekOver);
            double retPay = weekOver - retire;
            System.out.println();
            System.out.println("Payroll:");
            System.out.println("----------");
            System.out.println("Hours Worked: " + hours);
            System.out.println("Shift: Night");
            System.out.println("Hourly Pay Rate: R" + hourly3);
            System.out.println("Regular Pay: R" + (40 * 90));
            System.out.println("Overtime Pay: R" + calcOver);
            System.out.printf("Total of Regular and Overtime: R%.2f", weekOver);
            System.out.printf("\nRetirment Deduction: R%.2f", retire);
            System.out.printf("\nNet Pay: R%.2f", retPay);
            System.out.println();
        } //If the user chooses no, the method then calculates without retirement from
```

```
        } //If the user chooses no, the method then calculates without retirement from
        //the overtime pay, regular pay and net pay and displays all final results
        else if ("no".equalsIgnoreCase(choice)) {
            System.out.println();
            System.out.println("Payroll:");
            System.out.println("----------");
            System.out.println("Hours Worked: " + hours);
            System.out.println("Shift: Night");
            System.out.println("Hourly Pay Rate: R" + hourly3);
            System.out.println("Regular Pay: R" + (40 * 90));
            System.out.println("Overtime Pay: R" + calcOver);
            System.out.println("Total of Regular and Overtime: R" + weekOver);
            System.out.println("Retirment Deduction: R0.00");
            System.out.println("Net Pay: R" + weekOver);
        } //If the user enters an invalid choice, the program will show an invalid method
        else {
            System.out.println("Invalid Option");
        }
    }

}
```

d) Change Log:

EthanGitHubbie Add files via upload	37de3eb · now	6 Commits
EthanQ1.docx	Add files via upload	5 minutes ago
EthanQ1.pdf	Add files via upload	5 minutes ago
EthanQ2.docx	Add files via upload	5 minutes ago
EthanQ2.pdf	Add files via upload	5 minutes ago
EthanQ4.docx	Add files via upload	5 minutes ago
EthanQ4.pdf	Add files via upload	5 minutes ago
EthanQ5a.docx	Add files via upload	5 minutes ago
EthanQ5a.pdf	Add files via upload	5 minutes ago
EthanQ5b.docx	Add files via upload	5 minutes ago
EthanQ5b.pdf	Add files via upload	5 minutes ago
EthanQ5c.docx	Add files via upload	5 minutes ago
EthanQ5c.pdf	Add files via upload	5 minutes ago
EthanQ6.docx	Add files via upload	2 minutes ago
PAYROLL_README	Update and rename README.md to PAYROLL_README	7 minutes ago
Solution1.zip	Add files via upload	now
Solution2.zip	Add files via upload	now