FINAL PROJECT

OBJECT ORIENTED PROGRAMMING

COMP 6047001

PROJECT NAME**: BEM // BEST EMPLOYEE OF THE MONTH**

Name**: William Jonathan Mulyadi**

Student ID**: 2502045683**

Class**: L2CC**

Lecturer: **Jude Joseph Lamug Martinez**

“The Best Employee of the Month System”

Name: William Jonathan Mulyadi

NIM: 2502045683

Project Specification

You might be wondering, why I made The Best Employee of the Month system rather than another voting system, for example: for presidential purposes. It has always been my dream to run a café when I’m old in the future. Hence, the purpose behind the making of this project is to decide and give a reward to the best employee in a company. This is to increase the motivation of those employees and increase the competition between one employee and another. The best employee of the month will be decided by the number of visitors that created an account member and supports an employee. Actually, this system can be changed to whatever you want, it doesn't have to be choosing the best employee. This system can be used for a competition too.

In conclusion, the main purpose of this project is to increase the competition between employees which I believe can increase the productivity of each employee.

Background

I first started this project in May, I created the program using NetBeans by using JFrame to create a layout for my full Java project. I used MySQL to create a database and it takes input and sends out output to the database. Then the database will be printed on the JFrame on NetBeans.

But as time went on, I realized that as I used MySQL, it seems that the Java programming language used is not enough to cover all the materials that I learned, therefore with I created another project that uses full Java programming language. The full java project is my main project as the one that uses MySQL is an additional project.

Input

Employee,

First Name

Last Name

Age

Visitor (account member),

First Name

Last Name

Gender

Password

Other,

String adminpass.equals(“password”) (for Admin password)

String themonth (for user to input the month name)

String a and String b (for “Press any key to continue” & for taking inputs between Y/N; this is for making sure if the user wants to log out)

Int admin (letting the user choose whether he/she is an admin or visitor)

Int choose (for options on the admin page)

Int choice (for options on the visitor page)

Int n (for setting up the vote)

Output

Animation

Background of the app making

Result (the result of the voting)

Account member information

Account member information and password

Employee candidates’ details

Solution Design

*Details of how the system work.

First page

The first page prints out hiAnimation String array by using a for-loop from the interface animation that is being implemented in the FinalProject.java. It basically prints out big Hi animation made by “#”.

Second page

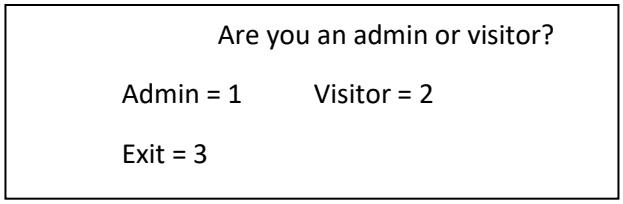
The second page prints out the BACKGROUND String array by using a for-loop from the interface Background that is being implemented in the FinalProject.java. It prints the background details of my name and the reason why I made this system.

Third page

The third page only prints out “Welcome to the voting program!” and asks you to input the month of the election time. The string themonth will get input from here.

Fourth page // The Log in Page

This page let you choose between 3 options, which are Admin, Visitor, or Exit. You can choose it by giving a number input from 1 to 3. The int admin will get input from this page.



If you input 1, the page will navigate you through the admin menu page. Meanwhile, **if you input 2,** the page will navigate you to the visitor menu page, and **if input 3**, it will ask you whether you really one to exit the program or not. If you input Y after the ask statement, it will break the program and go to the logout page.

Fifth page // The Admin page

When you input 1 from the main page, it will redirect you to this page. But first, it will ask for a password before the system lets you into this page. String adminpass will get input from this page and it has to be "password" or else it will redirect you back to the fourth page.

When you input the "password", this page will show you the options for the admin to make.

There are currently: 0 employee candidates
1. View member account password
2. Add the employee information
3. Edit the employee information
4. Display the employee information
5. Log out

If you choose 1, you will get the information about the member account details and password that is created on the visitor menu page. However, if you haven't created an account yet, it will print out "There is no account member-created yet?".

Meanwhile**, if you choose 2,** it lets you fill in the employee information and it updates the There are currently 0 employee candidates to 3. It uses for-loop to take in multiple-input, as you have to fill in 3 candidates.

If you choose 3, It first prints the employee account with its ID (using for-loop) and it lets you choose his/her ID and edit it. If you choose a larger number than the ID, it prints out "Invalid ID". If you choose the correct ID, it will ask you for editing by entering the first name, last name, and age of the employee by using the for loop. However, if you haven't added employee candidates, it will print that you haven't added an account and it will redirect you back to the admin menu page.

If you choose 4, it prints out the data of the employee candidates by using a for-loop. However, if you haven't added an account, it prints "There are no employee candidates registered!".

If you choose 5, it will ask you for input to make sure whether the admin really wants to log out or not. It asks you to select between Y/N. If you input Y, it will break the while-loop. Meanwhile, N will redirect you back to the page. (You can also use small letters (no capital)).

Sixth page // The Visitor page

When you input 2 from the main page, it will redirect you to this page.

```
There are currently: 0 accounts
What would you like to do?
1. Add new member account
2. Edit member account
3. Login and choose the best employee
4. Display account information
5. Employee information
6. Result
7. Logout
[Program: If you don't remember your password,
please ask admin for more information!]
Please select the number:
```

There are 7 options here.

If you choose 2,3,4,6 before adding a new member account, it all prints “There is no account member created.”. You have to add a member first before choosing the other options, except for number 5, if you choose number 5 and the employees are already registered; it prints out the employee’s information.

If you choose 1, it will let you put the first name, last name, gender, and also password.

If you choose 2, it prints out the name and ID of the member account first. Then, it asks you which account you want to edit by asking for the input of the ID number from the user. If you choose a larger number than the ID, it prints out “Invalid ID”. After choosing the ID, the user will be asked to input the password of the account member that’s already been created. If the password is wrong, it redirects you back to the menu of the visitor page. But if it’s right, you will be asked to put first name, last name, and gender again. It prints out “You have account successfully updated” after all of it is done. This basically overwrites the name before as it uses the setter function.

If you choose 3, it first will show the ID and name of the account member. After that, it will ask you to input the ID of the chosen account and ask for its password again. If you choose a larger number than the ID, it prints out “Invalid ID”. If the password is correct, it’ll print out the details of the employee and you will be asked to give a vote by choosing the ID of the employee candidates.

If you choose 4, it prints out the ID, first name, last name, and gender of the account member that has been created by using a for loop.

If you choose 5, it prints out the employee information detail. It has the same function as the display of employee information on the employee page. It uses for-loop to print it out.

If you choose 6, it prints out the result. But if you haven't created an account member, it prints out "There are no account members created.", and if you haven't registered the employee candidates, it prints "There are no candidates registered." So first, you have to create an account for members and register the candidate. Besides that, if not all account has voted, it prints "Not all account has voted!". So, all accounts have to vote first. After that, it prints out the result of the voting. It first prints out the details of who votes for who. The next thing is, it will calculate and prints out the winner with its percentage.

If you choose 7, it breaks the while-loop of the visitor menu page and it redirects you back to the log-in page.

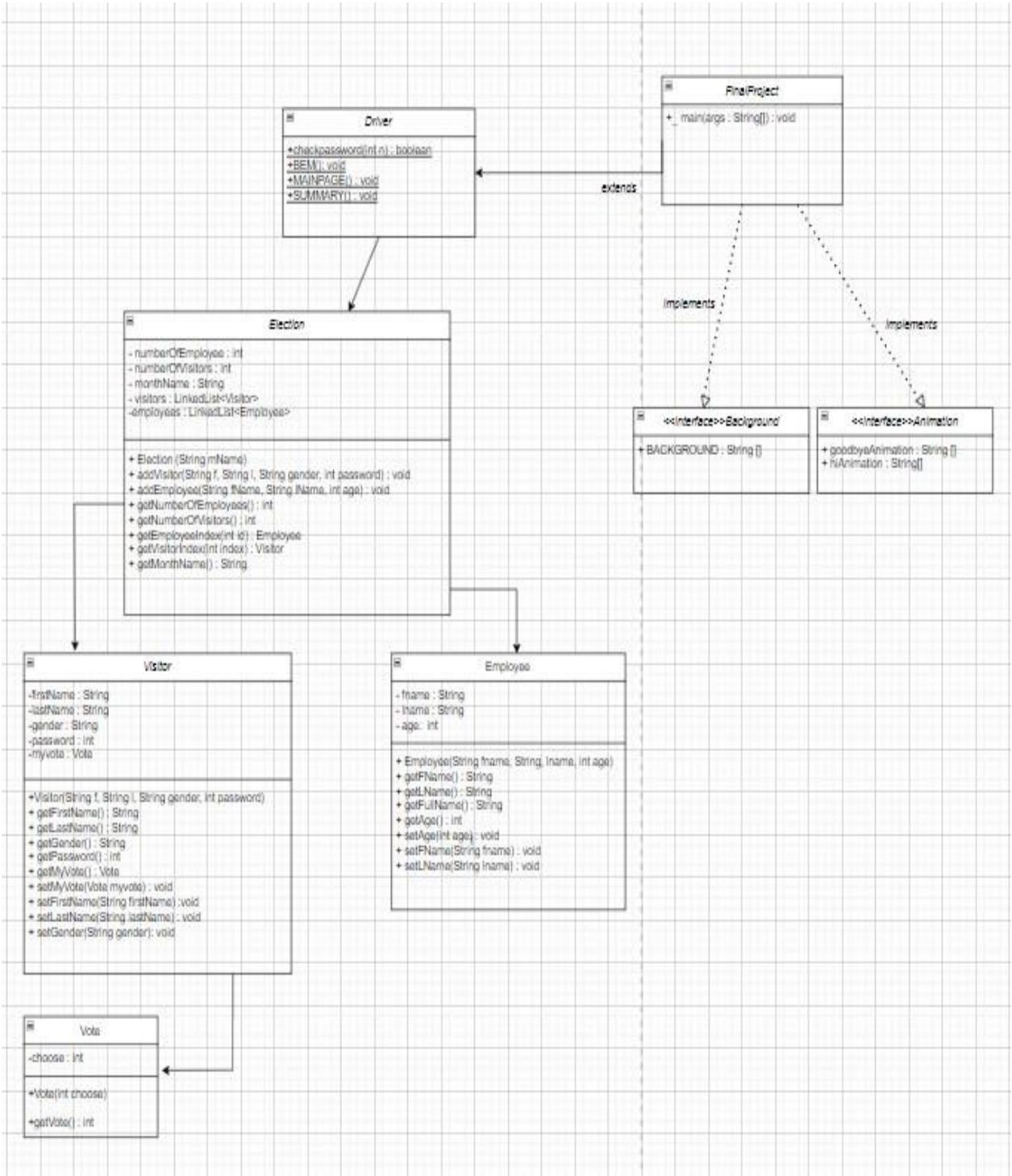
Seventh page // The Exit page

If you choose 3 from the Log in the page, you will be asked to input between Y/N to make sure if you are sure to Exit the program. If you input N, it will redirect you back to the program, meanwhile, if you input Y, it will redirect you to the goodbye page of the program.

Eight page// The end of the program

After breaking the while loop, it will print out a summary of the program, it will give details and information about the winner and who votes for who, the month's name, etc. Then it will print out the goodbyeAnimation from the animation interface which prints the text of " thank you for using our app.", and a smile animation created by "#".

UML DIAGRAM


FinalProject
+ main(args : String[]) : void
Driver
+checkpassword(int n) : boolean
+BEM(): void
+MAINPAGE() : void
+SUMMARY() : void
extends
Election
- numberOfEmployee : int
- numberOfVisitors : int
- monthName : String
- visitors : LinkedList<Visitor>
-employees : LinkedList<Employee>
+ Election (String mName)
+ addVisitor(String f, String l, String gender, int password) : void
+ addEmployee(String fName, String lName, int age) : void
+ getNumberOfEmployees() : int
+ getNumberOfVisitors() : int
+ getEmployeeIndex(int id) : Employee
+ getVisitorIndex(int index) : Visitor
+ getMonthName() : String
Implements
Implements
<<Interface>>Background
+ BACKGROUND : String []
<<Interface>>Animation
+ goodbyeAnimation : String []
+ hiAnimation : String[]
Visitor
-firstName : String
-lastName : String
-gender : String
-password : int
-myvote : Vote
+Visitor(String f, String l, String gender, int password)
+ getFirstName() : String
+ getLastName() : String
+ getGender() : String
+ getPassword() : int
+ getMyVote() : Vote
+ setMyVote(Vote myvote) : void
+ setFirstName(String firstName) :void
+ setLastName(String lastName) : void
+ setGender(String gender): void
Employee
- fname : String
- lname : String
- age: int
+ Employee(String fname, String, lname, int age)
+ getFName() : String
+ getLName() : String
+ getFullName() : String
+ getAge() : int
+ setAge(int age) : void
+ setFName(String fname) : void
+ setLName(String lname) : void
Vote
-choose : int
+Vote(int choose)
+getVote() : int

CODE EXPLANATION

For this program, I used 6 classes and 2 interfaces and I will be explaining the classes and interfaces one by one.

Here is the list of classes:

1. Vote
2. Visitor
3. Employee
4. Election
5. Driver
6. FinalProject

Here is the list of the interfaces:

1. Background
2. animation

Vote class

```
1  public class Vote {
2      //the vote class to get the vote
3      private int choose;
4      //constructor
5      public Vote(int choose){
6          this.choose = choose;
7      }
8
9      //getter function to get the vote
10     public int getVote(){
11         return this.choose;
12     }
13
14
15 }
```

The vote class is used for getting the vote number later and to set up the vote.

In this class I declared private int choose; this is for setting up the vote later on the constructor.

Visitor class

```java
public class Visitor {
    //private variables ; can't be accessed directly by user
    private String firstName;
    private String lastName;
    private String gender;
    private int password;
    private Vote myvote;
    //the public method
    //constructor
    public Visitor(String f, String l,String gender,int password){
        this.firstName = f;
        this.lastName = l;
        this.gender = gender;
        this.password = password;
    }
    //the getter function
    //getting first name
    public String getFirstName(){
        return this.firstName;
    }
    //getting last name
    public String getLastName(){
        return this.lastName;
    }
    //getting the gender
    public String getGender(){
        return this.gender;
    }
    //getting the password
    public int getPassword(){
        return this.password;
    }
    //method that is used to set Vote
    public void setMyVote(Vote myvote){
        this.myvote = myvote;
    }
```

```java
    //method that is used to set Vote
    public void setMyVote(Vote myvote){
        this.myvote = myvote;
    }
    //getting the vote
    public Vote getMyVote(){
        return myvote;
    }
    //setter function
    public void setFirstName(String firstName) {
        this.firstName = firstName;
    }
    public void setLastName(String lastName) {
        this.lastName = lastName;
    }
    public void setGender(String gender){
        this.gender = gender;
    }

}
```

The visitor class contains private variables which are as shown above. I also made the constructor, which will be used to create the Visitor object. The visitor constructor contains 3 strings and 1 integer. The 3 strings are the first name, last name, and gender; meanwhile, the integer is the password.

I have the getter function which I will use to get the information that is needed. Besides that, I have the setter function to set the vote from Vote class; edit the information of the Visitor later on in the Driver class.

Employee class

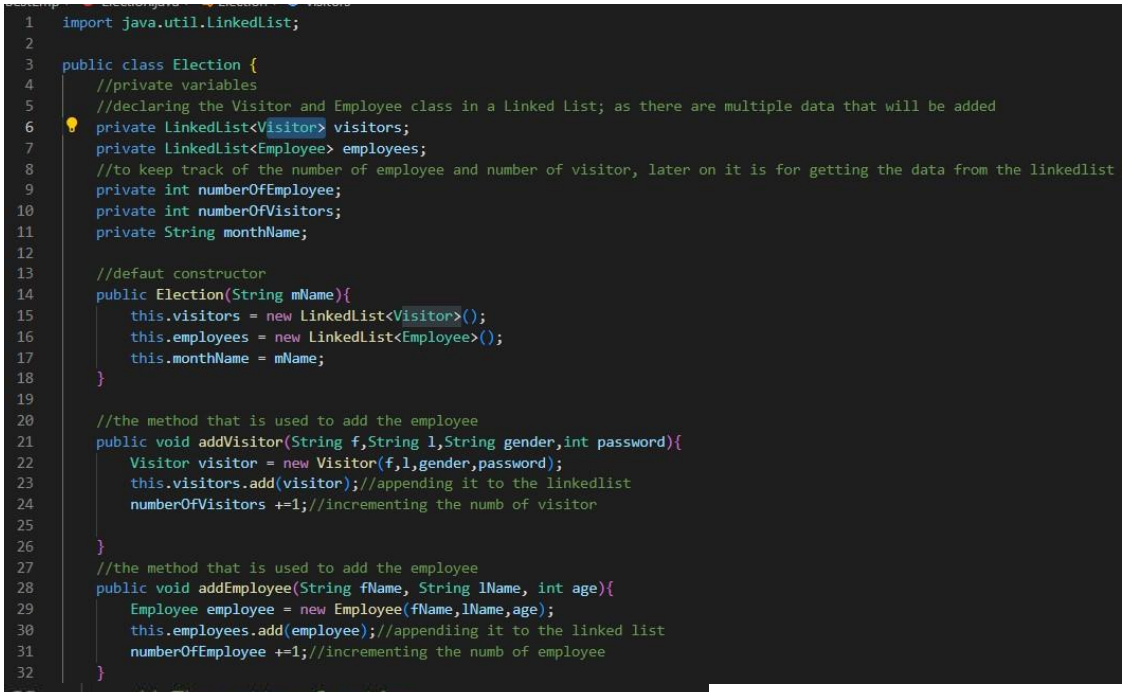
```java
public class Employee {
    //the publice method
    private String fname;
    private String lname;
    private int age;

    //constructor
    public Employee(String fname,String lname, int age) {
        this.fname = fname;
        this.lname = lname;
        this.age = age;
    }
    //the getter function
    public String getFullName(){
        return this.fname+" "+this.lname;
    }
    //getting the first name
    public String getFName() {
        return this.fname;

    }
    //getting the last name
    public String getLName() {
        return this.lname;
    }
    //getting the age
    public int getAge() {
        return this.age;
    }
    //setter function
    public void setAge(int age) {
        this.age = age;
    }
    public void setFName(String fname) {
        this.fname = fname;
    }
    public void setLName(String lname) {
        this.lname = lname;
    }

}

```

I introduced the private variables for 2 strings and 1 integer. It is for the employee's first name, last name, and age.

I made a constructor that contains the information of the private variables I declared before.

The getter function is made to get these private variables. Meanwhile, the setter function is used to set these private variables. The setter will be used later on for the Edit option.

Election class



```java
import java.util.LinkedList;

public class Election {
    //private variables
    //declaring the Visitor and Employee class in a Linked List; as there are multiple data that will be added
    private LinkedList<Visitor> visitors;
    private LinkedList<Employee> employees;
    //to keep track of the number of employee and number of visitor, later on it is for getting the data from the linkedlist
    private int numberOfEmployee;
    private int numberOfVisitors;
    private String monthName;

    //defaut constructor
    public Election(String mName){
        this.visitors = new LinkedList<Visitor>();
        this.employees = new LinkedList<Employee>();
        this.monthName = mName;
    }

    //the method that is used to add the employee
    public void addVisitor(String f,String l,String gender,int password){
        Visitor visitor = new Visitor(f,l,gender,password);
        this.visitors.add(visitor);//appending it to the linkedlist
        numberOfVisitors +=1;//incrementing the numb of visitor

    }
    //the method that is used to add the employee
    public void addEmployee(String fName, String lName, int age){
        Employee employee = new Employee(fName,lName,age);
        this.employees.add(employee);//appendiing it to the linked list
        numberOfEmployee +=1;//incrementing the numb of employee
    }
```

```java
    // The getter function
    public int getNumberOfEmployees(){
        return numberOfEmployee;
    }
    public int getNumberOfVisitors(){
        return numberOfVisitors;
    }
    //to get the index of the employee
    public Employee getEmployeeIndex(int id){
        return this.employees.get(id);
    }
    //to get the index of the visitor
    public Visitor getVisitorIndex(int index){
        return this.visitors.get(index);
    }
    //to get the mont name
    public String getMonthName(){
        return this.monthName;
    }
```

In the election class, I imported a LinkedList so that I could use it in my program. Then I created LinkedList of Visitor class and Employee class. I also made private variables to take in the number of employees and visitor accounts made, and also the month name. s

I made the constructor that sets up the month name and in it, there's also a LinkedList created.

I made the addVisitor and addEmployee function where I can add multiple information of Visitor and Employee to the LinkedList.

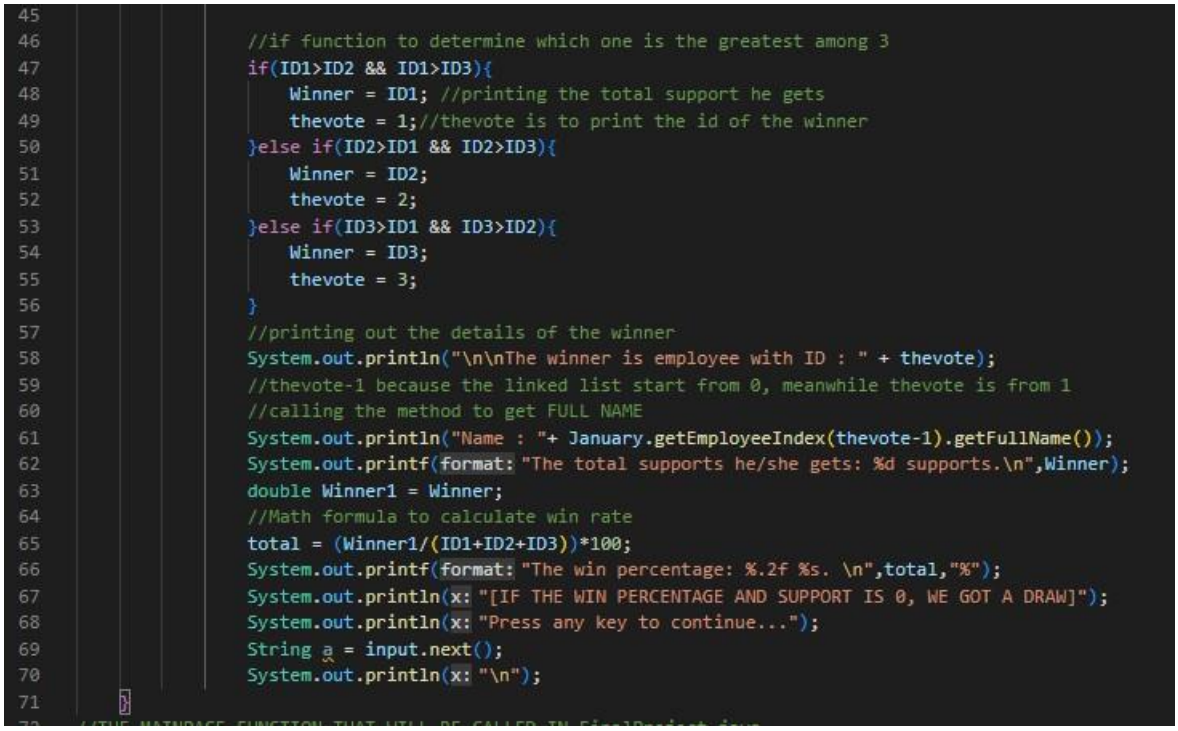
Both getNumberOfEmployees and getNumberOfVisitors is incremented by 1 every time the user use add method. The getIndex is to get the index of the employee and visitor later on. While getMonthName is to get the month.

Driver class

```java
//import that is used to take in input
import java.util.Scanner;
public class Driver {
    // creating the election object
    static Election January = new Election(mName: "January");
    static Scanner input = new Scanner(System.in);

    //method to checkpassword; whether the password is correct or not
    //this is for the voting purpose
    public static boolean checkpassword(int n){
        System.out.printf(format: "Enter password %s %s: ", January.getVisitorIndex(n).getFirstName(),
         January.getVisitorIndex(n).getLastName());//the index or n will be achieved by choosing from for loops
        int password = input.nextInt();//Taking password as interget input
        return password == January.getVisitorIndex(n).getPassword();
    }
```

I created a new object Election. Besides that, I created a method to check the password: so it first prints the first name and last name of the visitor. Then it asks for the password, the password should be equal to the getPassword of the visitor.

```java
    public static void BEM(){
        //This function is to decide the best employee of the month
                System.out.println(x: "\nThe Winner of the month! \n");
                System.out.println(x: "Details !\n");
                //for calculating the winner variables
                //assigning and initializing the variables

                int ID1=0;
                int ID2=0;
                int ID3=0;
                int thevote=0;//variables that is assined as ID number of the employees
                double total;//to calculate the percentage
                int Winner =0;//to know how many support the winner gets

                //for loop for calculating vote purposes
                for(int i = 0; i < January.getNumberOfVisitors(); i++){
                    // it will print the first name + votes for + numb of votes of that person.
                    System.out.printf(format: "%d. %s votes for %s \n", i+1,  January.getVisitorIndex(i).getFirstName(),January.getVisitorIndex(i).getMyVote().getVote());
                    //i assigned thevote to get the vote number of i
                    thevote = January.getVisitorIndex(i).getMyVote().getVote();
                    //if statement
                    if(thevote==1){
                        ID1 ++;//if thevote == 1; ID1 will increase 1; starts from 0
                    }else if(thevote==2){
                        ID2 ++;//if thevote == 2; ID2 will increase 1; starts from 0
                    }else if(thevote==3){
                        ID3 ++;//if thevote == 3; ID3 will increase 1; starts from 0
                    }
                }
```

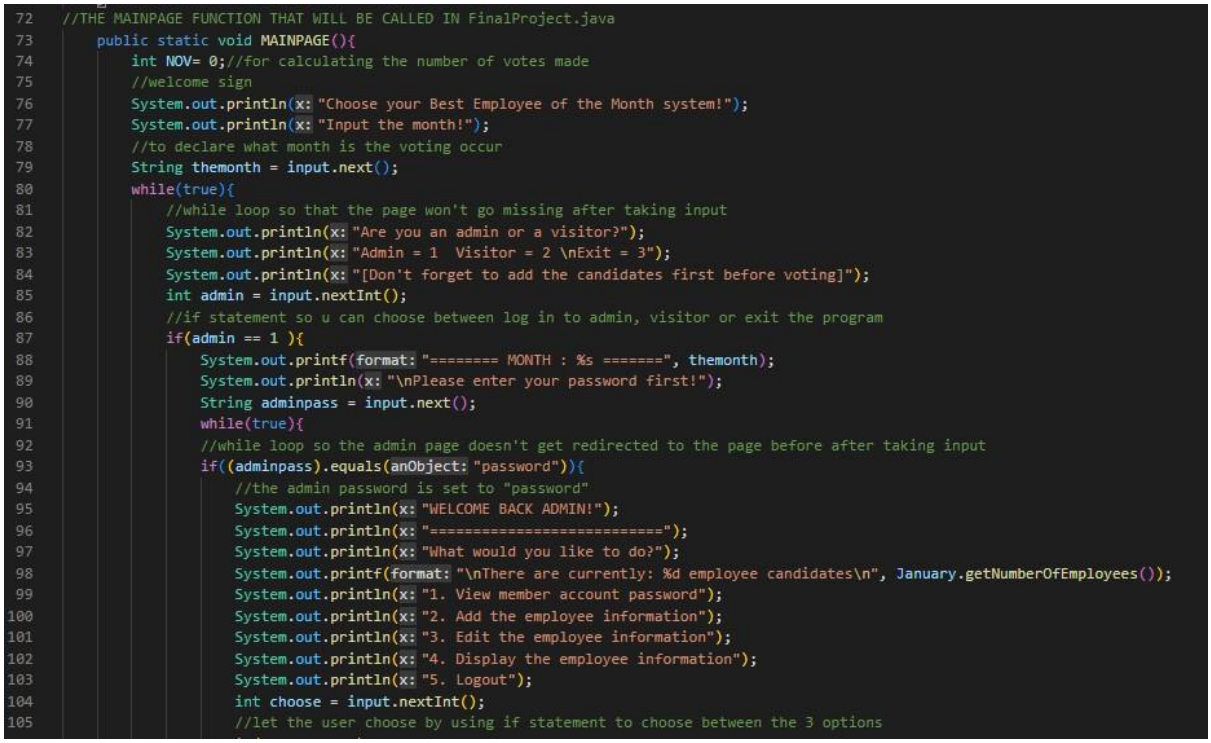


```
45
46              //if function to determine which one is the greatest among 3
47              if(ID1>ID2 && ID1>ID3){
48                  Winner = ID1; //printing the total support he gets
49                  thevote = 1;//thevote is to print the id of the winner
50              }else if(ID2>ID1 && ID2>ID3){
51                  Winner = ID2;
52                  thevote = 2;
53              }else if(ID3>ID1 && ID3>ID2){
54                  Winner = ID3;
55                  thevote = 3;
56              }
57              //printing out the details of the winner
58              System.out.println("\n\nThe winner is employee with ID : " + thevote);
59              //thevote-1 because the linked list start from 0, meanwhile thevote is from 1
60              //calling the method to get FULL NAME
61              System.out.println("Name : "+ January.getEmployeeIndex(thevote-1).getFullName());
62              System.out.printf(format: "The total supports he/she gets: %d supports.\n",Winner);
63              double Winner1 = Winner;
64              //Math formula to calculate win rate
65              total = (Winner1/(ID1+ID2+ID3))*100;
66              System.out.printf(format: "The win percentage: %.2f %s. \n",total,"%");
67              System.out.println(x: "[IF THE WIN PERCENTAGE AND SUPPORT IS 0, WE GOT A DRAW]");
68              System.out.println(x: "Press any key to continue...");
69              String a = input.next();
70              System.out.println(x: "\n");
71          }
```

Then I created the BEM method. BEM method is basically Best Employee of the Month, this method calculates the result of the votes and prints out the details. I basically created this first in the visitor page, result option. But then I decided to make it a method so it can be called in other parts of the code not only the MAINPAGE ().

By using for-loop I print out the details of who votes for who. Basically, prints out the name and the vote number by using getVote. Then if the getVote is equal to 1, it will add 1 to ID1, if it's equal to 2, it will add 1 to ID2, and the same with 3.

After that, I make another if statement to know which ID has the greatest number (using ID1>ID2, etc). The ones with the greatest number will make the vote assigned that 1/2/3 (this prints out the ID winner). As for the Winner integer, it keeps the total number of the winner. I created total int to calculate the win rate of the ID. It uses the formula of an ID/by total votes times a hundred.



```
72  //THE MAINPAGE FUNCTION THAT WILL BE CALLED IN FinalProject.java
73      public static void MAINPAGE(){
74          int NOV= 0;//for calculating the number of votes made
75          //welcome sign
76          System.out.println(x: "Choose your Best Employee of the Month system!");
77          System.out.println(x: "Input the month!");
78          //to declare what month is the voting occur
79          String themonth = input.next();
80          while(true){
81              //while loop so that the page won't go missing after taking input
82              System.out.println(x: "Are you an admin or a visitor?");
83              System.out.println(x: "Admin = 1  Visitor = 2 \nExit = 3");
84              System.out.println(x: "[Don't forget to add the candidates first before voting]");
85              int admin = input.nextInt();
86              //if statement so u can choose between log in to admin, visitor or exit the program
87              if(admin == 1 ){
88                  System.out.printf(format: "======== MONTH : %s =======", themonth);
89                  System.out.println(x: "\nPlease enter your password first!");
90                  String adminpass = input.next();
91                  while(true){
92                  //while loop so the admin page doesn't get redirected to the page before after taking input
93                  if((adminpass).equals(anObject: "password")){
94                      //the admin password is set to "password"
95                      System.out.println(x: "WELCOME BACK ADMIN!");
96                      System.out.println(x: "=========================");
97                      System.out.println(x: "What would you like to do?");
98                      System.out.printf(format: "\nThere are currently: %d employee candidates\n", January.getNumberOfEmployees());
99                      System.out.println(x: "1. View member account password");
100                     System.out.println(x: "2. Add the employee information");
101                     System.out.println(x: "3. Edit the employee information");
102                     System.out.println(x: "4. Display the employee information");
103                     System.out.println(x: "5. Logout");
104                     int choose = input.nextInt();
105                     //let the user choose by using if statement to choose between the 3 options
```

The MAINPAGE () method will be used in the FinalProject class. First, it prints out the welcome sign and asks you to input the month's name.

After that, I used a while loop to keep printing the menu for the log in page. Then it asks you to choose your role. The role is divided into 2, which are the visitor and admin. By inputting numbers between 1-3, you can choose the options.

Admin == 1; ----------

The int admin is responsible for choosing the role option. If you put 1, it redirects you to the page of the admin. But before that, you have to input the admin password first. The adminpass is set to "password", so unless you input password, you can't get into the page. If you get into the page, it prints out the menu option and asks you again to input an integer between those numbers shown.

I used another while loop to make sure that the page of admin is shown non-stop unless it is "break".

```
104                 int choose = input.nextInt();
105                 //let the user choose by using if statement to choose between the 3 options
106                 if(choose == 1){
107                     // To view the password of the member account made
108                     if(January.getNumberOfVisitors()==0){
109                         //to check whether an account has been made or not
110                         System.out.println(x: "There is no account yet!");
111                         System.out.println(x: "Press any key to continue...");
112                         String a = input.next();
113                     }else{
114                     System.out.println(x: "These are the passwords of the accounts:");
115                     System.out.println(x: "==========================");
116                     //for loops to get all of the account member data
117                     for(int a = 0; a < January.getNumberOfVisitors(); a++){
118                         //for the ID
119                         System.out.printf(format: "ID : %d \n",a+1);
120                         //for the First Name
121                         System.out.printf(format: "First name : %s \n", January.getVisitorIndex(a).getFirstName());
122                         //for the Last Name
123                         System.out.printf(format: "Last name : %s \n", January.getVisitorIndex(a).getLastName());
124                         //for getting the password
125                         System.out.printf(format: "Password : %s \n", January.getVisitorIndex(a).getPassword());
126                     }
127                     System.out.println(x: "Press any key to continue...");
128                     String a = input.next();
129                         }
130                     }
131                 else if(choose == 2){
```

If you input 1; it firsts checks the condition whether the information is enough to execute the command or not.

Choose == 1;

View member account password

First, it checks if there's an account or not. If there's not, it means that the number of visitors is 0; it will print out information stating that there's no account yet. Meanwhile, if there has been an account made, it'll execute in printing the details of the visitor.

I used for-loops to get all the account details; first, we print out the ID; i+1, and then first name, last name, and the password. I used the getter that I created on the visitor class and getIndex to get the index of the visitor in the employee class.

To end choose ==1; you will have to input a key. (design-wise; not essential)

```
131                 else if(choose == 2){
132                     if(January.getNumberOfEmployees()<4){
133                         if(January.getNumberOfEmployees()==3){
134                         System.out.println(x: "\nThe employees candidate are fully registered!");
135                         System.out.println(x: "Press any key to continue...");
136                         String a = input.next();
137                         System.out.println(x: "\n");
138                         }else{
139                     System.out.println(x: "Please enter 3 employees candidate!");
140                     for(int i = 0; i <3;i++){
141                     //using the for loops to take multiple inputs. Its actually 3 inputs
142                     System.out.printf(format: "ID  %d. Employee first name: ",i+1 );
143                     String front = input.next();
144                     System.out.printf(format: "ID  %d. Employee last name: ",i+1 );
145                     String back = input.next();
146                     System.out.printf(format: "ID  %d. Employee age: ",i+1 );
147                     int age = input.nextInt();
148                     //adding the employee
149                     January.addEmployee(front,back,age);
150                     System.out.println(x: "==========================");
151                     }
152                     }
153                     }else{
154                         System.out.println(x: "Sorry, the employees are already fully registered!");
155                     }
```

Choose== 2;

Add the employee information

It will ask you to input the first name, last name, and age of the employee. However, if you've registered them before, it prints out that the candidate is fully registered 3. There's an if statement to set this up. (If January.NumberOfEmployee()<4) this set up the max number of employee candidates is less than 4. (if January.NumberOfEmployee()==3) it prints out that the employees are fully registered.

If you've put the inputs, it will call the method addEmployee and add the employee information to the LinkedList. To get multiple inputs, I used a for-loop again.

```
156                 }else if(choose == 3){
157                     if(January.getNumberOfEmployees()==0){
158                         System.out.println(x: "There is no account yet!");
159                         System.out.println(x: "Press any key to continue...");
160                         String a = input.next();
161                         System.out.println(x: "\n");
162                     }else{
163                         System.out.println(x: "Edit employee profile.\n");
164                         //To print the member index and name
165                         for(int i = 0; i < January.getNumberOfEmployees(); i++){
166                             System.out.printf(format: "%d. %s %s \n", i+1,  January.getEmployeeIndex(i).getFName(),
167                             January.getEmployeeIndex(i).getLName());
168                         }
169                         System.out.println(x: "Please enter the index of your account: ");
170                         int choice =input.nextInt();
171                         if(choice > January.getNumberOfEmployees()){
172                         //Just in case the ID chosen is greater than the number of account member
173                             System.out.println(x: "Invalid index");
174                         }else{
175                             System.out.println(x: "Edit your first name: ");
176                             String fname = input.next();
177                         //calling the setter function which basically overwrite the information
178                             January.getEmployeeIndex(choice-1).setFName(fname);
179                             System.out.println(x: "Edit your last name: ");
180                             String lname = input.next();
181                             January.getEmployeeIndex(choice-1).setLName(lname);
182                             System.out.println(x: "Edit your age: ");
183                             int age = input.nextInt();
184                             January.getEmployeeIndex(choice-1).setAge(age);
185                             System.out.println(x: "Account successfully updated!!");
186                             System.out.printf(format: "ID %d %s %s",choice,January.getEmployeeIndex(choice-1).getFName(),
187                             January.getEmployeeIndex(choice-1).getLName());
188                             System.out.println(x: "\nPress any key to continue...");
189                             String a = input.next();
190                             System.out.println(x: "\n");
191                         }
192                     }
```

Choose ==3;

Edit the employee information

First if statement is to know whether the candidates are registered or not. If it's not, it will print out that there is no account yet.

If the candidates are already registered, it will first print out the list of the employees with their IDs using a for-loop. Then it will ask you to put the chosen ID, then asks you for editing the employee's information using the setter function. It prints out "Account successfully updated!!" if the option is executed.

```
193                 }else if(choose ==4){
194                     //Printing employee information
195                     if(January.getNumberOfEmployees()==0){
196                         //to indicate if no employee candidates are registered
197                         System.out.println(x: "No employee candidate is registered yet!");
198                         System.out.println(x: "Press any key to continue...\n");
199                         String a = input.next();
200                         System.out.println(x: "\n");
201                     }else{
202                     //Printing all the employees candidate details
203                     System.out.println(x: "\nThese are the employee candidates. \n");
204                     //for loop that execute the printing method of choose==4
205                     for(int a = 0; a < January.getNumberOfEmployees(); a++){
206                         //for the ID
207                         System.out.printf(format: "ID : %d \n",a+1);
208                         //for the First Name
209                         System.out.printf(format: "First name : %s \n", January.getEmployeeIndex(a).getFName());
210                         //for the Last Name
211                         System.out.printf(format: "Last name : %s \n", January.getEmployeeIndex(a).getLName());
212                         //for getting the Age
213                         System.out.printf(format: "Age : %s \n", January.getEmployeeIndex(a).getAge());
214 
215                         }
216                     System.out.println(x: "Press any key to continue...\n");
217                     String a = input.next();
218                     System.out.println(x: "\n");
219                     }
```

Choose ==4;

Display employee's information.

First it checks whether there are employees registered or not. If the employees are registered, it uses for loop to print their ID, first name, last name, and age. It uses the getter function to print it.

```
220                 }else if(choose == 5){
221                     //Exit choose
222                     System.out.println(x: "Are you sure you want to log out?(Y/N)");
223                     //Asking the user if he/she is sure to log out
224                     String b = input.next();
225                     //if statement to check input to decide if the user is sure to log out
226                     if(b.equals(anObject: "Y") || b.equals(anObject: "y")){
227                         System.out.println(x: "Goodbye Admin! :)");
228                         System.out.println(x: "See you next time!");
229                         System.out.println(x: "================================== ");
230                         break;
231                     }else if(b.equals(anObject: "N") || b.equals(anObject: "n")){
232                         System.out.println(x: "==================================");
233                         System.out.println(x: "Welcome back! :)");
234                     }else{
235                         System.out.println(x: "Please select between Y or N!");
236                     }
237                 }
```

Choose ==5;

Exit

In this option, it breaks the while-loop if it fulfils all the conditions. So first, you will be asked to select y/n to make sure if the user really wants to log out. If you choose y, it will break the loop; meanwhile, if you choose n, it will continue the while loop. (Any other key other than y will continue the while-loop).

```
237                 }
238             }else{
239                     //if the user doesnt enter "password"
240                     System.out.println(x: "Wrong password!");
241                     System.out.println(x: "Press any key to continue...");
242                     String a = input.next();
243                     System.out.println(x: "\n\n\n\n\n");
244                     break;
245 
246                 }
247             }//while of admin ==1
248         }
249 //-----------------------------------------------------------------------------------------------------------------------
```

Else

This statement is if the password the user enter is wrong. If the password is not equal to “password” then the else statement will be executed. It’ll print wrong password.

Admin ==2 ----------

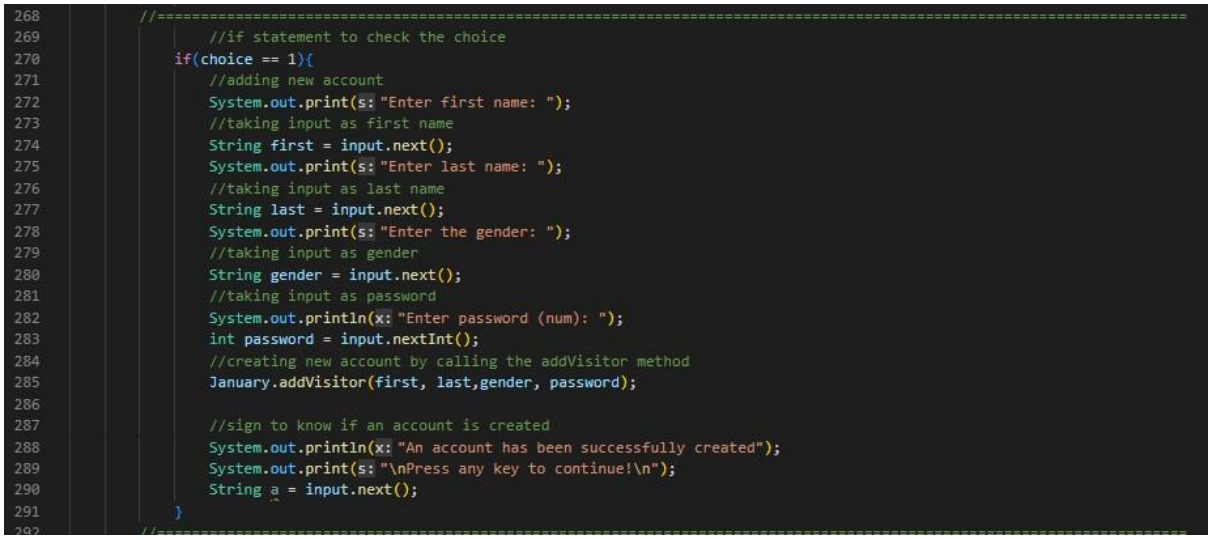
```
249 //-----------------------------------------------------------------------------------------------------------------------
250         else if(admin==2){
251             while(true){
252                 //main menu for the user to choose                    String themonth - Driver.MAINPAGE()
253                 System.out.printf(format: "======== MONTH : %s =======", themonth);
254                 //to indicates if an account is created or not; or how many account has been created
255                 System.out.printf(format: "\nThere are currently: %d accounts\n", January.getNumberOfVisitors());
256                 System.out.println(x: "What would you like to do?\n ");
257                 System.out.println(x: "1. Add new member account");
258                 System.out.println(x: "2. Edit member account");
259                 System.out.println(x: "3. Login and choose the best employee");
260                 System.out.println(x: "4. Display account information");
261                 System.out.println(x: "5. Employee information");
262                 System.out.println(x: "6. Result");
263                 System.out.println(x: "7. Logout");
264                 System.out.println(x: "[Program: If you don't remember your password, please ask admin for more information!]");
265                 System.out.print(s: "\nPlease select the number: ");
266                 //taking input as choice from the user
267                 int choice = input.nextInt();
```

If admin ==2, it immediately redirects you to the visitor page.

I created another while loop; so the admin menu page will stay after executing an option unless it is break.

It prints out the menu and takes in integer input so the user can choose between the menu listed.

It prints out “There are currently: %d accounts”; this is to let you know how many visitor accounts have been created.



```
268     //=============================================================================================================
269             //if statement to check the choice
270         if(choice == 1){
271             //adding new account
272             System.out.print(s: "Enter first name: ");
273             //taking input as first name
274             String first = input.next();
275             System.out.print(s: "Enter last name: ");
276             //taking input as last name
277             String last = input.next();
278             System.out.print(s: "Enter the gender: ");
279             //taking input as gender
280             String gender = input.next();
281             //taking input as password
282             System.out.println(x: "Enter password (num): ");
283             int password = input.nextInt();
284             //creating new account by calling the addVisitor method
285             January.addVisitor(first, last,gender, password);
286 
287             //sign to know if an account is created
288             System.out.println(x: "An account has been successfully created");
289             System.out.print(s: "\nPress any key to continue!\n");
290             String a = input.next();
291         }
292     //=============================================================================================================
```

Choice ==1;

Add a new member account

It asks you to enter your first name, last name, gender, and password in the integer. Then it will use the addVisitor method which will add visitor information to the LinkedList. If all is done, it prints out “An account has been successfully created!”.

```
293             else if(choice ==3){
294                 //The login and choose employee candidates
295                 if(January.getNumberOfVisitors()==0){
296                     //To check account memmber has been made or not
297                     System.out.println(x: "There is no account member yet!");
298                     System.out.println(x: "Press any key to continue...");
299                     String a = input.next();
300                 }else if(January.getNumberOfEmployees()==0){
301                     //To check employee candidats account has been made or not
302                     System.out.println(x: "There is no employee candidate resgistered!");
303                     System.out.println(x: "Press any key to continue...");
304                     String a = input.next();
305 
306                 }else{
307                 //THE FIRST VOTE WILL OVERWRITE THE SECOND VOTE
308                 System.out.println(x: "[REMINDER: IF YOU VOTE TWICE, THE FIRST VOTE WILL BE LOST]");
309                 //for loop that prints index; the index is added by 1 as it start from 0 and the first name
310                 System.out.print(s: "First log in to your account!\n");
311                 //Printing the information of the account created
312                 for(int i = 0; i < January.getNumberOfVisitors(); i++){
313                     System.out.printf(format: "%d. %s %s \n", i+1,  January.getVisitorIndex(i).getFirstName(),
314                      January.getVisitorIndex(i).getLastName());
315                 }
316                 //getting the account by using index
317                 System.out.println(x: "Please enter the index of your account: ");
318                 choice =input.nextInt();
319                 if(choice>January.getNumberOfVisitors()){
320                     //if statment; just in case the user input a super big number
321                     System.out.println(x: "Invalid ID of the employee!");
322                     System.out.println(x: "Press any key to continue...");
323                     String a = input.next();
324 
325                 }else{
326                 //calling the checkpassword method
327                 //if the password is wrong, it will not take any input for setting the visitor vote
328                     if(checkpassword(choice-1)){
329                         System.out.print(s: "\nChoose your employee ID!\n\n");
330                         for(int i = 0; i < January.getNumberOfEmployees(); i++){
331                             //for loops to print the information of the employee
332                             System.out.printf(format: "%d. %s  \n", i+1,  January.getEmployeeIndex(i).getFullName());
333                         }
334                     int n = input.nextInt();
335                     //asking input to set up vote
336                     if(n>January.getNumberOfEmployees()){
337                         System.out.println(x: "Invalid ID!");
338                     }else{
339                         Vote vot = new Vote(n);
340                         NOV++;
341                         //choice-1 bcs it starts from 0 ; it is -1 from the ID shown
342                         January.getVisitorIndex(choice-1).setMyVote(vot);
343                         System.out.println(x: "You have successfully voted!");
344                         System.out.println(x: "\n");
345                         }
346                         }
347                     }
348                 }
```

Choice ==3;

Log in and choose the best employee

It first checks whether employees and visitor accounts have been created or not. If it hasn't, it prints out the statement that indicates it hasn't been created.

If both have been created, it prints out the information of the account member ID, first name, and last name. It asks you to input the ID of your account.

If the ID is greater than the number of accounts created, it prints out "Invalid ID". If not, it calls the checkpassword method, to ask for password input and check the password inputted. Then by using for-loop, it prints out the ID and name of the employees.

It asks you to input the ID of your chosen employees, and the ID inputted will be set as a vote. After a vote is added, NOV of short for (Number of Vote) is added by 1. If you've successfully added the vote, it prints out you have successfully voted.

```
350         //=============================================================================================================
351             else if(choice ==4){
352                 //To check whether the account has been created or not
353                 if(January.getNumberOfVisitors()==0){
354                     System.out.println(x: "There is no account member yet!");
355                     System.out.println(x: "Press any key to continue...");
356                     String a = input.next();
357                     System.out.println(x: "\n");
358                 }else{
359                 //it shows the information account member that has been created by the user
360                 System.out.println(x: "\n These are the list of account: \n");
361                 for(int a = 0; a < January.getNumberOfVisitors(); a++){
362                     //by using for loops, it can print all of the visitor li
363                     System.out.printf(format: "ID : %d \n",a+1);
364                     System.out.printf(format: "First name : %s \n", January.getVisitorIndex(a).getFirstName());
365                     System.out.printf(format: "Last name : %s \n", January.getVisitorIndex(a).getLastName());
366                     System.out.printf(format: "Age : %s \n", January.getVisitorIndex(a).getGender());
367 
368                 }
369                 System.out.println(x: "Press any key to continue !\n");
370                 String a = input.next();
371                 System.out.println(x: "\n");
372                 }
373             }
```

Visitor Election.getVisitorIndex(int index)

Choice == 4;

Display account information

First if statement to check whether an account has been created or not. If an account has been created, it prints out all the member account information by using the for-loops. It prints out all the details except for the password.

```
374         //=============================================================================================================
375             else if(choice ==6){
376                 //The result page
377                 if(January.getNumberOfVisitors()==0){
378                     //condition if no account member is made
379                     System.out.println(x: "There is no account member created");
380                     System.out.println(x: "Press any key to continue...");
381                     String a = input.next();
382                     System.out.println(x: "\n");
383                 }else if(January.getNumberOfEmployees()==0){
384                     //condition if there's no candidate registered
385                         System.out.println(x: "There is no employee candidate is registered");
386                         System.out.println(x: "Press any key to continue...");
387                         String a = input.next();
388                         System.out.println(x: "\n");
389 
390                 }else if(NOV<January.getNumberOfVisitors()){
391                     System.out.println("Vote :"+NOV);
392                     System.out.println("visitor" + January.getNumberOfVisitors());
393                     //condition if not all of the account has voted
394                     System.out.println(x: "Not all account has voted! Please vote first!");
395                     System.out.println(x: "Press any key to continue...");
396                     String a = input.next();
397                     System.out.println(x: "\n");
398                 }else {
399                 //Calling the BEM function to decide the winner
400                 BEM();
401             }
402         }
403         //=============================================================================================================
```

Choice == 6;

Result

The first if statement is to check whether an account member has been created or not. The else if, is to check whether the employees have been registered or not, the next else if is to check whether all account has voted or not. Next else, execute the result of the vote by calling the BEM method.

As stated before the BEM method is a method that is used to decide the winner of the election.

```
403         //=============================================================================================================
404             else if(choice == 5){
405                 //Printing employee information
406                 if(January.getNumberOfEmployees()==0){
407                     //to indicate if no employee candidates are registered
408                     System.out.println(x: "No employee candidate is registered yet!");
409                     System.out.println(x: "Press any key to continue...\n");
410                     String a = input.next();
411                     System.out.println(x: "\n");
412                 }else{
413                 //Printing all the employees candidate details
414                 System.out.println(x: "\nThese are the employee candidates. \n");
415                 //for loop that execute the printing method of choice==5
416                 for(int a = 0; a < January.getNumberOfEmployees(); a++){
417                     //for the ID
418                     System.out.printf(format: "ID : %d \n",a+1);
419                     //for the First Name
420                     System.out.printf(format: "First name : %s \n", January.getEmployeeIndex(a).getFName());
421                     //for the Last Name
422                     System.out.printf(format: "Last name : %s \n", January.getEmployeeIndex(a).getLName());
423                     //for getting the Age
424                     System.out.printf(format: "Age : %s \n", January.getEmployeeIndex(a).getAge());
425 
426                 }
427                 System.out.println(x: "Press any key to continue...\n");
428                 String a = input.next();
429                 System.out.println(x: "\n");
430             }
431         }
432         //=============================================================================================================
```

Choice == 5;

Display the employee information

The first if statement is to check whether the employee's candidate has been registered or not. If it has, it'll print out the employee details by using for-loop. It prints out the ID, first name, last name, and age. It'll get the value by using the getter method.

```
432         //=============================================================================================================
433             else if(choice==7){
434                 //choice==7 is the logout , it prints successfully logged out text and breaks while loop
435                 System.out.println(x: "Sucessfully logged out! :) ");
436                 System.out.println(x: "\n");
437                 break;
438         }
```

Choice == 7;

Log out

This basically breaks the while loop in the visitor menu page.

```
439         //=============================================================================================================
440             else if(choice==2){
441                 //Edit account information
442                 if(January.getNumberOfVisitors()==0){
443                     //If there is no member account has been made
444                     System.out.println(x: "There is no account yet!");
445                     System.out.println(x: "Press any key to continue...");
446                     String a = input.next();
447                     System.out.println(x: "\n");
448                 }else{
449                     //executing the choice==2 program
450                     System.out.println(x: "Edit account profile.\n");
451                     //To print the member index and name
452                     for(int i = 0; i < January.getNumberOfVisitors(); i++){
453                         System.out.printf(format: "%d. %s %s \n", i+1,  January.getVisitorIndex(i).getFirstName(),
454                          January.getVisitorIndex(i).getLastName());
455                     }
456                     System.out.println(x: "Please enter the index of your account: ");
457                     choice =input.nextInt();
458                     if(choice > January.getNumberOfVisitors()){
459                         //Just in case the ID chosen is greater than the number of account member
460                         System.out.println(x: "Invalid index");
461                     }
462                     else if(checkpassword(choice-1)){
463                         //calling check password to ensure the edit page is only accessible by the account holder
464                         System.out.println(x: "Edit your first name: ");
465                         String fname = input.next();
466                         //calling the setter function which basically overwrite the information
467                         January.getVisitorIndex(choice-1).setFirstName(fname);
468                         System.out.println(x: "Edit your last name: ");
469                         String lname = input.next();
470                         January.getVisitorIndex(choice-1).setLastName(lname);
471                         System.out.println(x: "Edit your gender: ");
472                         String gender = input.next();
473                         January.getVisitorIndex(choice-1).setGender(gender);
474                         System.out.println(x: "Account successfully updated!!");
475                         System.out.printf(format: "ID %d %s %s",choice,January.getVisitorIndex(choice-1).getFirstName(),
476                         January.getVisitorIndex(choice-1).getLastName());
477                         }
478                     else{
479                         System.out.println(x: "Wrong password!");
480                         System.out.println(x: "\n");
481                         }
482                     }
483                 }
```

Choice ==2;

Edit the account member information

It first checks whether a member account has been created or not. If it has been created it will execute a for-loop printing the ID, first name, and last name of the visitors.

It will ask you to input the ID, and it will ask you for your password and check it by using checkpassword method. Then, it will ask you to input first name, last name, and gender. By using the setter method, it will edit the details and it'll print out "Account successfully updated!". Then it'll print the updated account name and ID. However, the password is wrong, it will print out wrong password.

If the account has not been created, it prints out "There is no account created yet.".

```
484         //=============================================================================================================
485             else{
486                 //If the user input is not 1,2,3,4,5,6,7
487                 System.out.println(x: "Please choose the right number!");
488                 System.out.println(x: "\n");
489             }
490         }
491 //-----------------------------------------------------------------------------------------------------------------
```

Else

It prints out, please choose the right number and continue the while-loop. So, it's back to the visitor menu page.

Admin ==3 ----------

```
            }else if(admin==3 ){
                //EXIT
                System.out.println(x: "Are you sure that you want to exit? (Y/N)");
                //To ensure the user is sure that he wants to exit
                String b = input.next();
                if(b.equals(anObject: "Y") || b.equals(anObject: "y")){
                    System.out.println(x: "\n\n\n\n\n\n\n\n\n\n\n\n\n\n\tGoodbye! Thank you for using our program! :)");
                    System.out.println(x: "\tSee you next time!");
                    System.out.println(x: "================================================================");
                    SUMMARY();//It calls of the summary method in code below
                    break;//break the first while loop
                }else if(b.equals(anObject: "N") || b.equals(anObject: "n")){
                    System.out.println(x: "==================================");
                    System.out.println(x: "Welcome back! :)");
                }else{
                    System.out.println(x: "Please select between Y or N!");
                }
            }else{
                //If he/she not choose between 1-3
                System.out.println(x: "Please select between 1,2,3!");
                System.out.println(x: "\n");
            }
        }//first while loop statement
    }
//-----------------------------------------------------------------------------------------------------------------
```

It first asks you to input Y/N. If the user inputs N, it will not break the while loop; meaning it will not end the program. Meanwhile, if the user inputs Y, it will break the while loop and end the program. It will print “Thank you for using our program and See you next time!” and SUMMARY method. If its input is other than N and Y, it will still continue the while-loop but it first prints “Please select between Y or N!”.

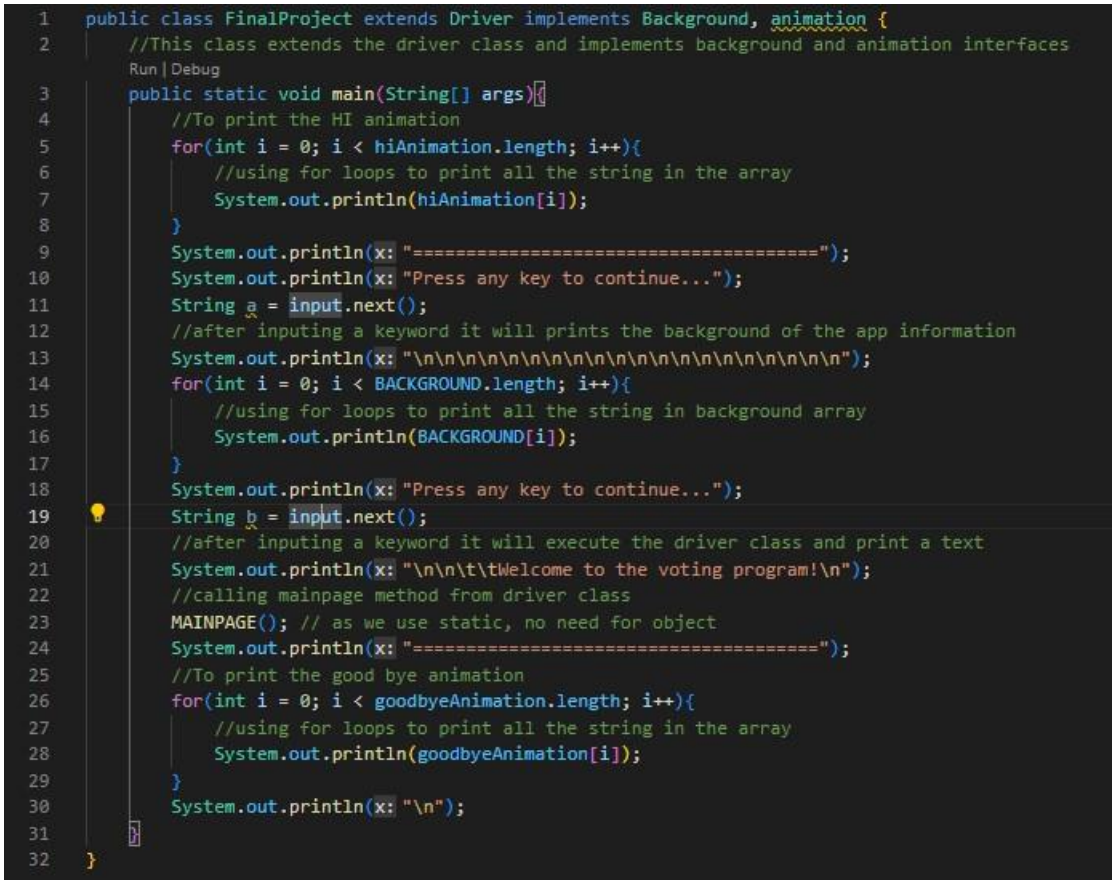
```
//-----------------------------------------------------------------------------------------------------------------
    //Summary method to print out the details of the system
    public static void SUMMARY(){
        int z = 0;//for if condition later on
        System.out.printf(format: "======================THE SUMMARY======================\n");
        System.out.println(x: "");
        //Get month name
        System.out.println("THE MONTH : " + January.getMonthName());
        // if no account member has been made, no employees have been registered or -
        //no vote has been made or not all vote has been made.
        for (int i = 0; i< January.getNumberOfVisitors(); i++){
            if(January.getVisitorIndex(i).getMyVote()==null){
                z = -1; //indicating if there's an account that hasn't vote
            }
        }
        if(January.getNumberOfVisitors()==0){
            System.out.println(x: "Not enough information to decide the winner!");
        }else if(January.getNumberOfEmployees()==0){
            System.out.println(x: "Not enough information to decide the winner!");
        }else if(z==-1){
            System.out.println(x: "Not enough information to decide the winner!");
        }else{
            //It prints the winner and text THE BEST EMPLOYE ... .
            System.out.println(x: "THE BEST EMPLOYEE OF THE MONTH DETAILS!");
            System.out.println(x: "");
            BEM();//Calling the best employee of the month method
        }
    }
}
//THE END OF THIS PAGE CODE
//-----------------------------------------------------------------------------------------------------------------
```

SUMMARY()

It prints the month name and if the information is complete; it prints out the BEM method again which decide the winner; however if it’s not complete, it prints out not enough information.

The z = 0 and if there's an account that hasn't voted; z will become -1. This is to know whether all account has voted or not.

FinalProject class



```java
public class FinalProject extends Driver implements Background, animation {
    //This class extends the driver class and implements background and animation interfaces
    Run | Debug
    public static void main(String[] args){
        //To print the HI animation
        for(int i = 0; i < hiAnimation.length; i++){
            //using for loops to print all the string in the array
            System.out.println(hiAnimation[i]);
        }
        System.out.println(x: "====================================");
        System.out.println(x: "Press any key to continue...");
        String a = input.next();
        //after inputing a keyword it will prints the background of the app information
        System.out.println(x: "\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n");
        for(int i = 0; i < BACKGROUND.length; i++){
            //using for loops to print all the string in background array
            System.out.println(BACKGROUND[i]);
        }
        System.out.println(x: "Press any key to continue...");
        String b = input.next();
        //after inputing a keyword it will execute the driver class and print a text
        System.out.println(x: "\n\n\t\tWelcome to the voting program!\n");
        //calling mainpage method from driver class
        MAINPAGE(); // as we use static, no need for object
        System.out.println(x: "====================================");
        //To print the good bye animation
        for(int i = 0; i < goodbyeAnimation.length; i++){
            //using for loops to print all the string in the array
            System.out.println(goodbyeAnimation[i]);
        }
        System.out.println(x: "\n");
    }
}
```

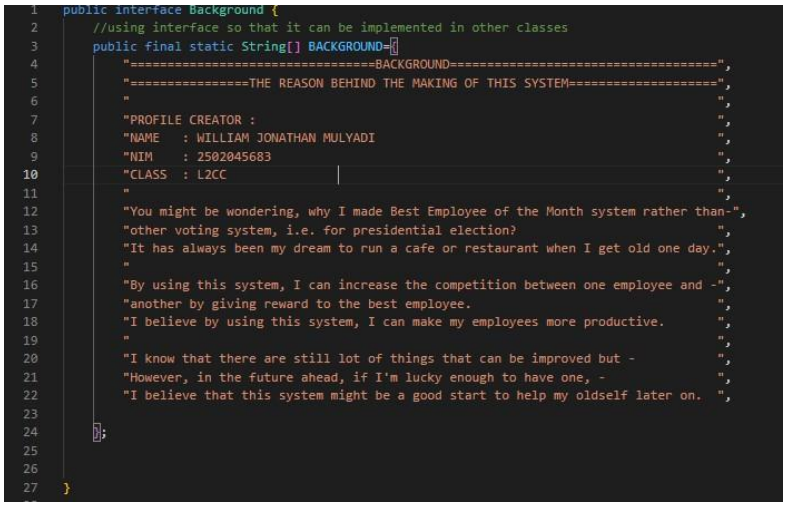
This class extends the Driver class and implements 2 interfaces.

The main function is here; to run the program.

It uses for loop to print the array of hiAnimation in the animation interface, then it prints out the BACKGROUND array from the Background interface using the for-loop again. After that, it calls of the method MAINPAGE() . The MAINPAGE() is a method from the driver class, then it will print out the goodbyeAnimation array from the animation interface by using for loop.

INTERFACES

Background interface



```
public interface Background {
    //using interface so that it can be implemented in other classes
    public final static String[] BACKGROUND={
        "================================BACKGROUND=================================",
        "===============THE REASON BEHIND THE MAKING OF THIS SYSTEM==================",
        "                                                                           ",
        "PROFILE CREATOR :                                                          ",
        "NAME   : WILLIAM JONATHAN MULYADI                                          ",
        "NIM    : 2502045683                                                        ",
        "CLASS  : L2CC                                                              ",
        "                                                                           ",
        "You might be wondering, why I made Best Employee of the Month system rather than-",
        "other voting system, i.e. for presidential election?                       ",
        "It has always been my dream to run a cafe or restaurant when I get old one day.",
        "                                                                           ",
        "By using this system, I can increase the competition between one employee and -",
        "another by giving reward to the best employee.                             ",
        "I believe by using this system, I can make my employees more productive.   ",
        "                                                                           ",
        "I know that there are still lot of things that can be improved but -       ",
        "However, in the future ahead, if I'm lucky enough to have one, -           ",
        "I believe that this system might be a good start to help my oldself later on.  ",

    };


}
```

It only contains a BACKGROUND array which basically consists of the background why I made this system.

animation interface

```
public interface animation {
    //using interface so that it can be implemented in other classes
    final static String[] hiAnimation={
        "                                    "
        ,"      ######      #######   ###"
        ,"      ######      #######   ###"
        ,"      ######      #######      "
        ,"      ######      #######   ###"
        ,"      ##################   ###"
        ,"      ##################   ###"
        ,"      ##################   ###"
        ,"      ######      #######   ###"
        ,"      ######      #######   ###"
        ,"      ######      #######   ###"
        ,"      ######      #######   ###"
    };
    final static String[] goodbyeAnimation={
        "                                                  "
        ,"                                                  "
        ,"           ######             ######              "
        ,"         ########           ########             "
        ,"         ########           ########             "
        ,"          ######             ######              "
        ,"   #####                                 ######    "
        ,"     ######                            ######     "
        ,"       ######                        ######       "
        ,"         ############################             "
        ,"            ########################              "
        ,"              ####################                "
        ,"                                                  "
    };
```

Interfaces have to be final!

It only contains hiAnimation array which contain of # making a HI and goodbyeAnimation which contain of # making a :).

REFLECTION

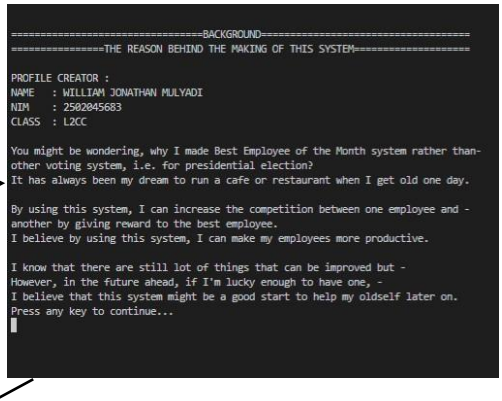
As I mentioned before in the background, I created 2 projects. The one with MySQL and the one that uses full java language. For the one with MySQL, I use many references from StackOverflow and YouTube. It takes less time to create the one with MySQL even though I have to remember MySQL and search for more references. For the full java one, I did it without references and I tried to implement all the knowledge that I got from Mr. Jude's class before. It was quite hard as there seem to be many errors occurring. Debugging the program takes a very long time, after I seem to finish it, there always seems to be another problem. It takes a while to think about how to solve those problems. Not to mention I still found problems when I started making video demonstrations. Hence, I tried to solve it again and again.

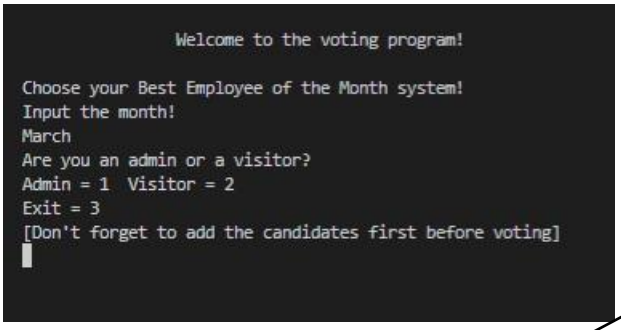
What I learned from making this project is that we have to be really patient to code. Besides, it takes time to think. Don't forget to relax, as you can't really force your idea to come up under stressful conditions.

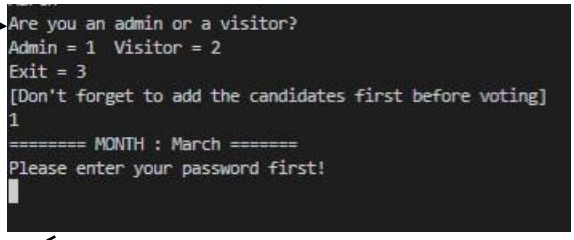
REFERENCES:

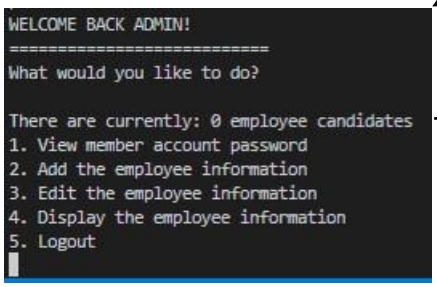
- Mr. Jude Joseph Lamug Martinez task, class, and forums.

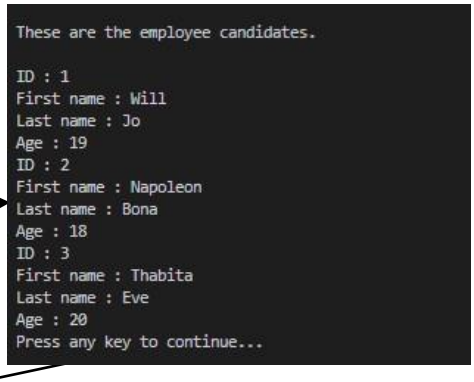
WORKING PROGRAM PROOF

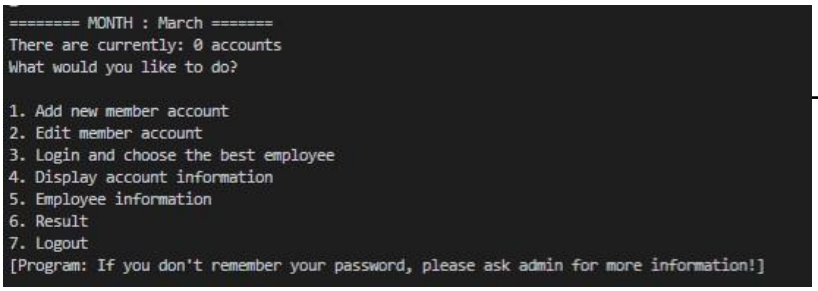


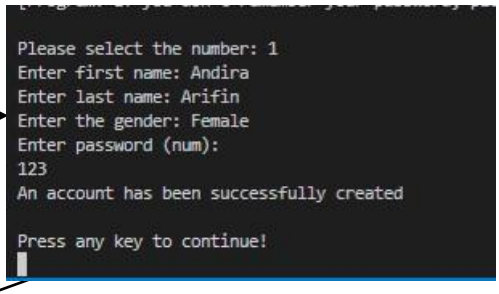


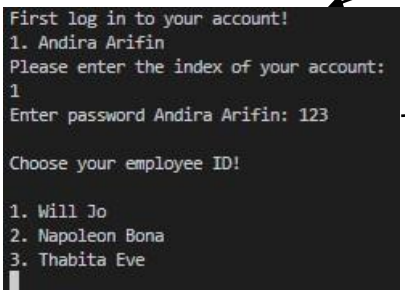


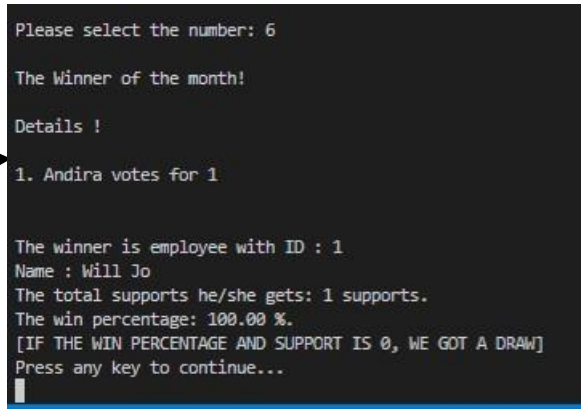












```
Are you an admin or a visitor?
Admin = 1  Visitor = 2
Exit = 3
[Don't forget to add the candidates first before voting]
3
Are you sure that you want to exit? (Y/N)
y
```