Data structures Project

Employee record management system using linked list

Rawan alsufyani 2112024

1- The code:

```
import java.util.Scanner;


public class Project {
  private Node head;
   private  class Node{
    private int ID;
    private String name,day,phone_mumber,address;
    private double hours,salary;
    private Node next;
    public Node(int ID,String name,String day,String phone_mumber,String address,double
hours,double salary){
      this.ID=ID;
      this.name=name;
      this.day=day;
      this.phone_mumber=phone_mumber;
      this.address=address;
      this.hours=hours;
      this.salary=salary;
      this.next=null;
    }
  }


 public void insertEmployee(int ID,String name,String day,String phone_mumber,String
address,double hours,double salary){
    if(!checkRecord(ID)){
      System.out.println("***************");
    Node new_node=new Node(ID,name,day,phone_mumber,address,hours,salary);
    if(head==null){
      head=new_node;
      return;
    }
```

```
      Node temp=head;
      while(temp.next!=null){
         temp=temp.next;
      }
      temp.next=new_node;
     }else{
        System.out.println("Employee with ID "+ID+ " already exist ");
     }
   }
  boolean checkRecord(int ID){
     Node temp=head;
     if(head==null)
        return false;


      while(temp!=null&&head!=null){
         if(temp.ID==ID)
            return true;
         temp=temp.next;
      }
     return false;
  }


  public void search(int ID){
      Node temp=head;
      while (temp!= null) {
          if(temp.ID == ID){
            System.out.println("Employee Info :\n- ID:"+ID+"\n- Name:"+temp.name+"\n- First
day of work: "+
                  temp.day+"\n- Phone number: "+temp.phone_mumber+"\n- Address
:"+temp.address+"\n- Work hours: "+temp.hours
            +"\n- Salary: "+temp.salary);
           return;
          }else{
              temp=temp.next;
          }
      }
      System.out.println("employee with ID ("+ID+") not registered" );
  }
    public  int deleteEmployee(int ID){


       Node temp=head,prev=null;
       if (temp != null && temp.ID == ID) {
          head = temp.next;
          return 1;
      }
       while (temp != null && temp.ID != ID) {
          prev = temp;
          temp = temp.next;
      }
      if(temp==null){
          return 0;
      }
      prev.next=temp.next;
      return 1;
  }
    public void showRecord(){
      Node temp=head;
      while(temp!=null){
            System.out.println("Employee Info :\n- ID:"+temp.ID+"\n- Name:"+temp.name+"\n-
First day of work: "+
                    temp.day+"\n- Phone number: "+temp.phone_mumber+"\n- Address
:"+temp.address+"\n- Work hours: "+temp.hours
              +"\n- Salary: "+temp.salary);
          temp=temp.next;
```

```
   }

}
public void updateSalary(){

    if(head==null){
        return;
    }

    Node temp=head;
   while(temp!=null){
      if(temp.hours>32){
          temp.salary+=(temp.salary*.02);

      }
      temp=temp.next;
   }
    System.out.println("Record updated");
}
public void updateEmployee(int ID){
    if(!checkRecord(ID)){
        System.out.println("Employee not found");
    }else{
        Scanner input=new Scanner(System.in);
        String name,day,phone_mumber,address;
        double hours,salary;
                   System.out.print("Enter new employee Name:");
                   name=input.nextLine();
                   System.out.print("Enter new First day of work:");
                   day=input.nextLine();
                   System.out.print("Enter new Phone number:");
```

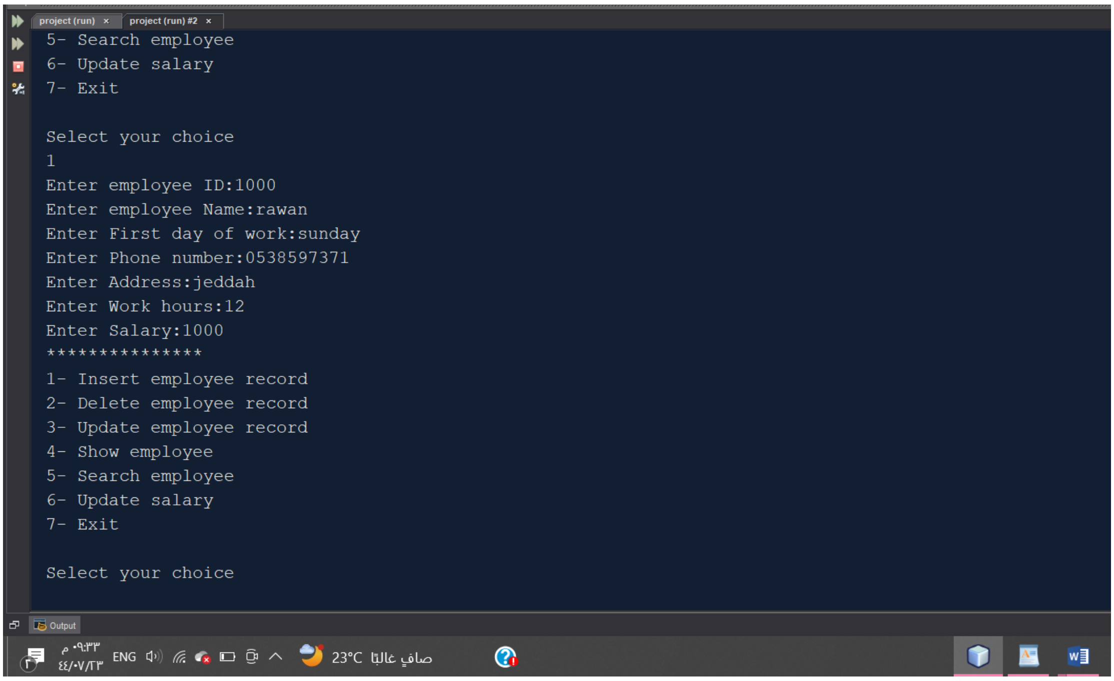
```
         phone_mumber=input.nextLine();
         System.out.print("Enter new Address:");
         address=input.nextLine();
         System.out.print("Enter new Work hours:");
         hours=input.nextDouble();
         System.out.print("Enter new Salary:");
         salary=input.nextDouble();
    Node temp=head;
     while(temp!=null){
       if(temp.ID==ID){
          temp.name=name;
          temp.salary=salary;
          temp.address=address;
          temp.day=day;
          temp.hours=hours;
          temp.phone_mumber=phone_mumber;
       }
       temp=temp.next;
     }
  }

}




public static void main(String[] args) {
   Scanner input=new Scanner(System.in);
   int choose,ID;
   String name,day,phone_mumber,address;
   double hours,salary;
   boolean repeat=true;
```

```
         Project employees=new Project();


        while(repeat){
            System.out.println("1- Insert employee record\n" +
                        "2- Delete employee record\n" +
                        "3- Update employee record\n" +
                        "4- Show employee\n" +
                        "5- Search employee\n" +
                        "6- Update salary\n"+
                        "7- Exit\n");
            System.out.println("Select your choice ");
            choose=input.nextInt();
            switch(choose){
                case 1:
                    System.out.print("Enter employee ID:");
                    ID=input.nextInt();
                    input.nextLine();
                    System.out.print("Enter employee Name:");
                    name=input.nextLine();
                    System.out.print("Enter First day of work:");
                    day=input.nextLine();
                    System.out.print("Enter Phone number:");
                    phone_mumber=input.nextLine();
                    System.out.print("Enter Address:");
                    address=input.nextLine();
                    System.out.print("Enter Work hours:");
                    hours=input.nextDouble();
                    System.out.print("Enter Salary:");
                    salary=input.nextDouble();
                    employees.insertEmployee(ID, name, day, phone_mumber, address, hours,
salary);
```

```
  break;
case 2:
   System.out.println("Enter employee ID to delete :");
   ID=input.nextInt();
   if(employees.deleteEmployee(ID)==1){
      System.out.println("Emplyee has been deleted ");

   }else{
      System.out.println("Employee with "+ID+" doesn't Exist");
   }
   break;
case 3:
    System.out.println("Enter employee ID to Update :");
    ID=input.nextInt();
    employees.updateEmployee(ID);
   break;
case 4:
   employees.showRecord();
   break;
case 5:
   System.out.println("Enter employee ID to Search :");
   ID=input.nextInt();
   employees.search(ID);
   break;
case 6:
   employees.updateSalary();
   break;
case 7:
   repeat=false;
   break;
default:
```

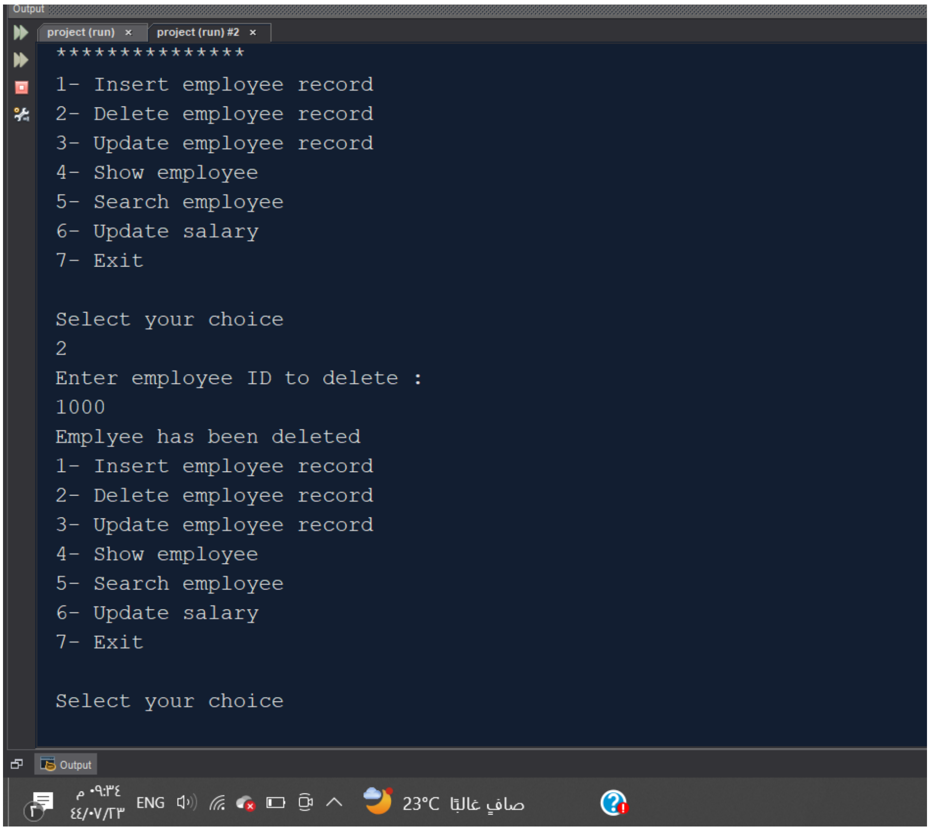
```
                    System.out.println("Wrong choice");



                }

            }




        }


    }
```

Insert output :



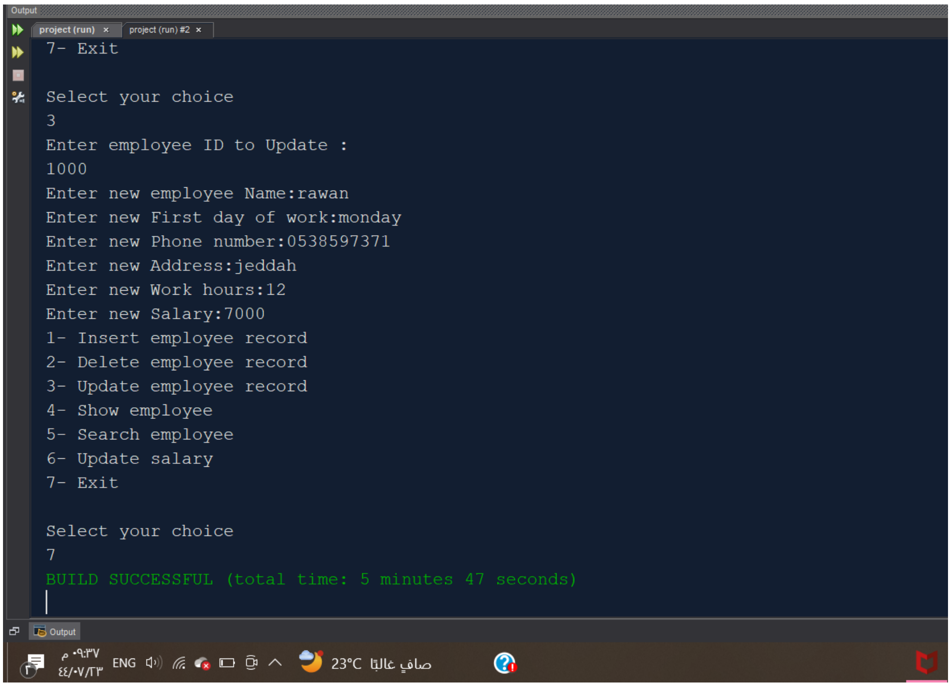
```
5- Search employee
6- Update salary
7- Exit

Select your choice
1
Enter employee ID:1000
Enter employee Name:rawan
Enter First day of work:sunday
Enter Phone number:0538597371
Enter Address:jeddah
Enter Work hours:12
Enter Salary:1000
***************
1- Insert employee record
2- Delete employee record
3- Update employee record
4- Show employee
5- Search employee
6- Update salary
7- Exit

Select your choice
```

Delete output:



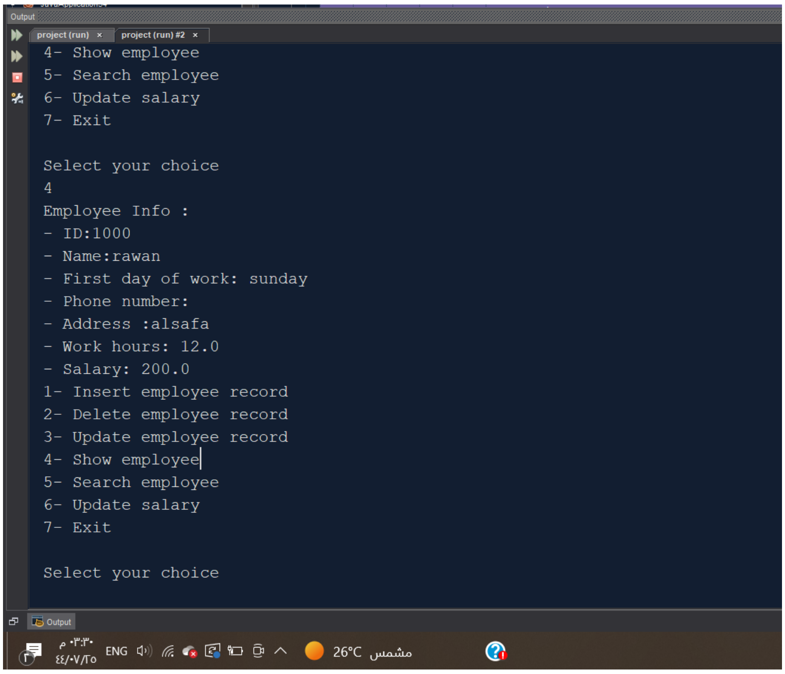
```
***************
1- Insert employee record
2- Delete employee record
3- Update employee record
4- Show employee
5- Search employee
6- Update salary
7- Exit

Select your choice
2
Enter employee ID to delete :
1000
Emplyee has been deleted
1- Insert employee record
2- Delete employee record
3- Update employee record
4- Show employee
5- Search employee
6- Update salary
7- Exit

Select your choice
```

Update output:



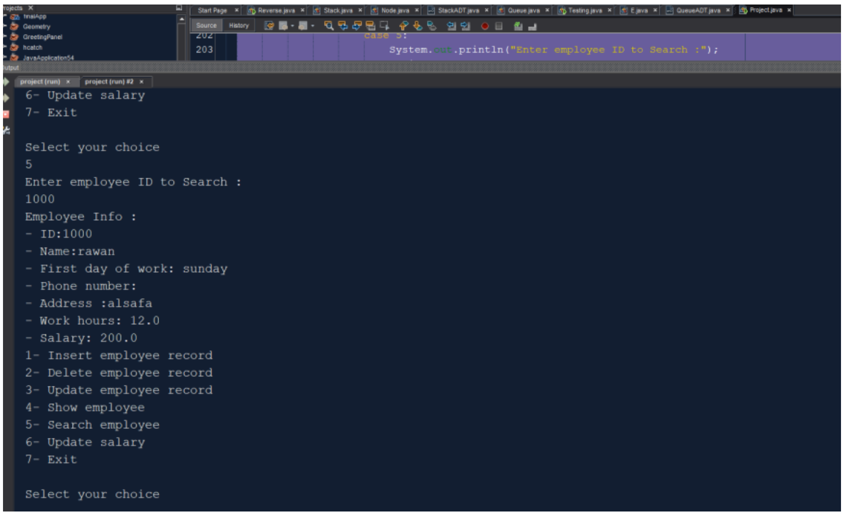
```
7- Exit

Select your choice
3
Enter employee ID to Update :
1000
Enter new employee Name:rawan
Enter new First day of work:monday
Enter new Phone number:0538597371
Enter new Address:jeddah
Enter new Work hours:12
Enter new Salary:7000
1- Insert employee record
2- Delete employee record
3- Update employee record
4- Show employee
5- Search employee
6- Update salary
7- Exit

Select your choice
7
BUILD SUCCESSFUL (total time: 5 minutes 47 seconds)
```

Show:



```
4- Show employee
5- Search employee
6- Update salary
7- Exit

Select your choice
4
Employee Info :
- ID:1000
- Name:rawan
- First day of work: sunday
- Phone number:
- Address :alsafa
- Work hours: 12.0
- Salary: 200.0
1- Insert employee record
2- Delete employee record
3- Update employee record
4- Show employee
5- Search employee
6- Update salary
7- Exit

Select your choice
```

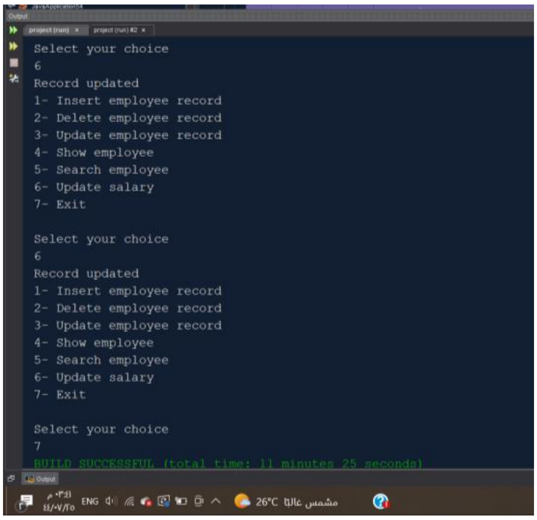
Search:



```
                case 5:
                    System.out.println("Enter employee ID to Search :");
```

```
6- Update salary
7- Exit

Select your choice
5
Enter employee ID to Search :
1000
Employee Info :
- ID:1000
- Name:rawan
- First day of work: sunday
- Phone number:
- Address :alsafa
- Work hours: 12.0
- Salary: 200.0
1- Insert employee record
2- Delete employee record
3- Update employee record
4- Show employee
5- Search employee
6- Update salary
7- Exit

Select your choice
```

Update + exist:



```
Select your choice
6
Record updated
1- Insert employee record
2- Delete employee record
3- Update employee record
4- Show employee
5- Search employee
6- Update salary
7- Exit

Select your choice
6
Record updated
1- Insert employee record
2- Delete employee record
3- Update employee record
4- Show employee
5- Search employee
6- Update salary
7- Exit

Select your choice
7
BUILD SUCCESSFUL (total time: 11 minutes 25 seconds)
```