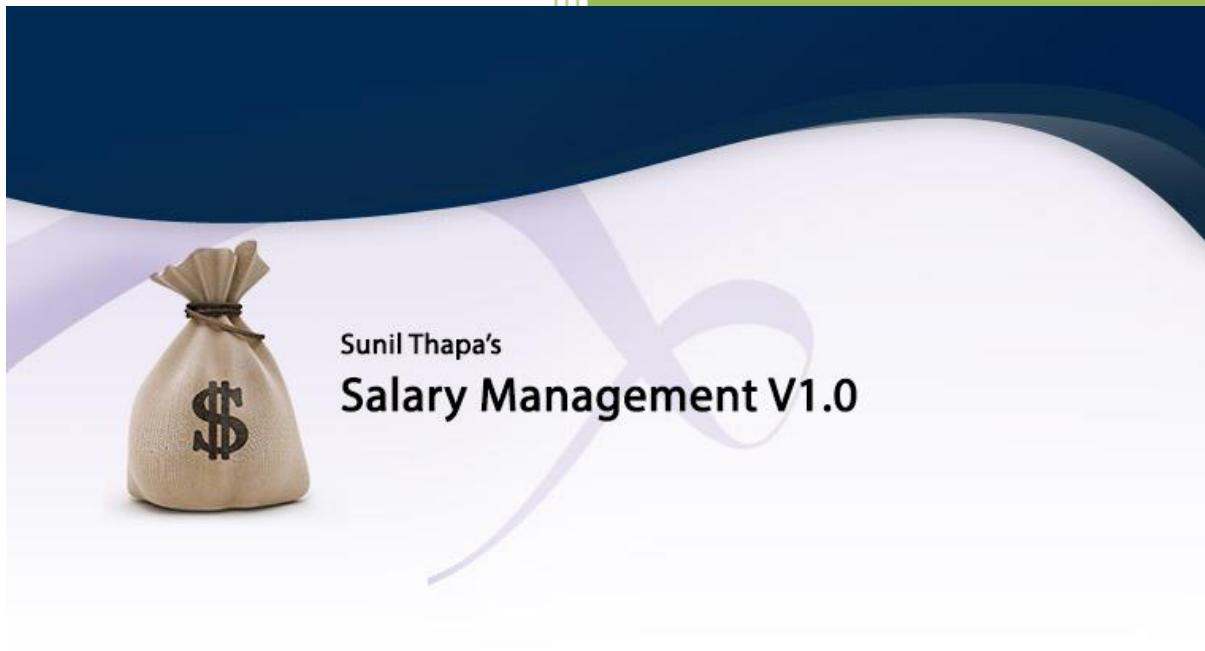


9691/4



# Salary Management V1.0 Documentation

**Sunil Thapa**

**Gems Institute Of Higher**

**Education**

**NP730/3037**

# **PREFACE**

I would like to thank all those who have helped me directly or indirectly in creating this project report for computing. I would like to thank Mr. Badri Thapa for providing me chance to create the software for B.D.Multipurpose. I would like to heartily thank Mr. Rakesh Chaudhary (my Computing teacher) for helping me with the production of the software and for helping me with this documentation. I would like to thank my friends for supporting me.

Once again, I would like to thank everyone for your help.

# Table of Contents

<b>1. Definition, Investigation and Analysis.....</b>	<b>4</b>
<b>Definition .....</b>	<b>4</b>
<b>Methods Currently in Use.....</b>	<b>6</b>
<b><i>Origin of the Data</i> .....</b>	<b>8</b>
<b>Investigation .....</b>	<b>9</b>
<b><i>Alternative Solutions</i> .....</b>	<b>16</b>
<b><i>Advantages and Disadvantages of the Proposed Systems</i>.....</b>	<b>17</b>
<b>Feasibility Study.....</b>	<b>18</b>
<b>2. System Design .....</b>	<b>19</b>
<b><i>Screen Layout</i>.....</b>	<b>19</b>
<b><i>Flowcharts</i> .....</b>	<b>36</b>
<b><i>Table Structure</i> .....</b>	<b>52</b>
<b><i>Intended benefits of the New System</i>.....</b>	<b>56</b>
<b>System Limitations.....</b>	<b>57</b>
<b>Size Estimation of Database .....</b>	<b>58</b>
<b>3. Software Development , Testing and Implementation .....</b>	<b>63</b>
<b><i>Algorithms</i> .....</b>	<b>64</b>
<b><i>Data Layer</i> .....</b>	<b>68</b>
<b><i>Forms with Codes</i> .....</b>	<b>76</b>
<b><i>Testing</i>.....</b>	<b>167</b>
<b><i>Implementation</i> .....</b>	<b>189</b>
<b>4. Documentation .....</b>	<b>190</b>
<b>5. Evaluation .....</b>	<b>192</b>

## **1. Definition, Investigation and Analysis**

### ***Definition***

#### ***About the Client: B.D MULTIPURPOSE***

**B.D multipurpose**, is a furniture currently run by MR. Badri Thapa . It has been providing services for more than a decades. First established in 2000 A.D. It is located in Lokanthali, Bhaktapur, Nepal. Different kinds of products are made and distributed in different showrooms according to their orders. All the products are made from high quality wooden blocks.



**Fig. 1: A Product(window) made in B.D.Multipurpose**

**It has been providing quality service to its customer. The products made by this organization includes:**

- **Windows**
- **Doors**
- **Tables**
- **Chairs**
- **Beds**
- **Wardrobe**
- **And many other wooden products**

## **METHODS CURRENTLY IN USE**

**The current system is working manually, the manager records entry time and exit time of a carpenter in a entry card. There are three types of carpenter that is Helper, Head, and Designer Head. Every individual has their own hourly rate. Total number of hours worked in a month is calculated and multiplied by their personal rate to calculate their earnings. Their expenses of each day is written in last column of that entry card. Entry card is different for every individual every month. All details (no of hours worked, total expenses in a day and in a month) of every month is kept in entry card. Calculations are done manually so there is a greater chance of mistakes.**

क्रमांक	नाम	पत्ता नं.	दिनांक
१०६	हाजिरी कार्ड	३४५७५९	२०८६/१३
१०७	नाम :	राम सिंह	
१०८	मास :	इन्दोड	
१०९	मिति :	२०८६/१५	साल पांच
११०	गते देविति :	३२९	गते सम्म
१११	Address		
११२	Name		
११३	Date		
११४	All		

2086/13 क्रमांक 2086/13  
 20651991

८३ + ९६.५ × ६५ = ८०  
 आवधि ८० × ९०००  
 $(\text{आवधि}) \rightarrow 99.५६$   
 लग्जरी २५२५  
 लग्जरी २८६१  
 लग्जरी १८८१

## **calculations**

**Fig. 2: A Entry Card of B.D.Multipurpose**

## ***ORIGIN OF THE DATA***

The data required for developing the program is obtained from the following sources:

### ***Entry Card of an Employee***

The entry card of an employee acts as a data source for number of hours worked, their expenses, and their names (please refer to Fig. 2).

# ***Investigation***

## **Method of Investigation**

### **Interview**

For better understanding about the problems in current system and to know the expectation from my side, An interview was held with Mr. Badri Thapa which went as follows:

***Q.1. When was B.D.Multipurpose established?***

**Ans.** B.D.Multipurpose was established in 2000 A.D.

***Q.2. Could you describe the whole tasks till making a total income's report ?***

**Ans.** The manager records entry time and exit time of a carpenter in a entry card. Total number of hours worked in a month is calculated and multiplied by their personal rate to calculate their earnings and their expenses is subtracted to calculate total income of an individual.

***Q.3. What if the employee lost the entry card ?***

**Ans.** If an employee lost an employee's entry card then there is no way of regaining the data like numbers of hours worked and expenses.

***Q.4. Are you facing any kind of difficulties in running this system?***

**Ans.** Yes, it is really a slow job to do. If any employee wants to see earnings of his previous months then it takes time to search that file. If anyone lose their entry card than that is a problem. Sometimes we lost the records of employee's previous month's income. There are errors in calculating income as it is a manual system of calculation. We do not have long term storage of employee's details(Income, wages).

***Q.5. Haven't you thought of any alternative solution to this system?***

Ans. Yes, I tried to keep the records in two entry cards one for employee and another for backup purpose but it is too slow and time consuming process. So we came back to single entry card.

***Q.6. How do you expect a desktop application to help you in this matter?***

Ans. It needs to be much faster to use. I need to be able to keep all details of employee. I need to be able to search for the employee's details and records and edit them , much more quickly and efficiently. Everything should be clear. I want it to be simple and easily understandable by me. Interface should be user friendly so that it would be easy for us to use. It should automatically calculate income by multiplying rate and numbers of hours worked and subtracting the expenses. It should allow us to print every individual income with total hours worked and expenses of each day. Only manager should be allowed to use this application (User name and password should be needed to run this application which should be kept by us). A common working hour should be stored and should automatically display in entry and exit time.

## **Method of Investigation**

### **Observation**

- Employee's details are not kept systematically.
- It took time to see 3 months earlier record of an employee.
- There were few mistakes in calculating income of an employee.
- Some of the employee's record were lost
- Manager needs to do all the things manually so at the end of the month, there is large workload for manager.
- Data are not secure.

**Conclusively a computerized system is a must need at this situation to convert the process being too slow to fast, accurate, and efficient.**

## ***Findings from the investigation***

**From the conducted investigation, I came up with a few conclusions. They are:**

- **It is very time consuming to search the previous months details of an employee.**
- **Manager is tired of calculating the income details at the end of the month.**
- **All the details are kept manually and calculated manually.**
- **Because of being a manual system there are chances to commit an error.**
- **Because of being a paper based system there is less security of data.**

## *Analysis*

### ***Limitations in the Current system***

These are few limitations in the current system:

- The most difficult job currently is searching the previous months records
- There is a risk of data being lost of an employee
- If an employee lost an entry card then there is no way of regaining the data
- The manager does not have an accurate calculation
- It is a slower system as it consumes to enter entry and exit time and again calculate the total income at the end of a month.
- Even if employee erased the expenses in a day, then there is no way of knowing it.

## ***Program Requirements***

**From the investigation, I came to a conclusion that my program should include following features:**

- ✓ It should be menu based system with navigation buttons.
- ✓ The new system should provide all the functionality of current system overcoming the problems in this system.
- ✓ A feature of adding, editing and deleting new employee's details.
- ✓ A feature of adding and editing common working hours.
- ✓ The program needs to have simple and user friendly interface which includes all the fields from employee's entry card.
- ✓ A database which stores the values of the fields.
- ✓ A feature of uploading working hours of all employee at once so that it will consume less time.
- ✓ Employee's income should be calculated automatically.
- ✓ A feature of showing employee's previous months details.
- ✓ A feature to print employee's earning details.
- ✓ A feature of providing each employee a different employee ID.

## ***Hardware and Software Requirements***

**Following are the Hardware and Software requirement of the system**

### ***Hardware Requirements***

- A Pentium 3 processor 700 MHZ or better with a compatible motherboard.
- 128 MB or more RAM.
- A hard disk to store and save the files, including the program
- A low resolution monitor to display the output.
- A mouse and a keyboard for entering the data into the system.
- USB flash drive support/CD-writer/Floppy-disk drive, portable hard drives (for backups)
- A inkjet or laser printer to print reports

### ***Software Requirements***

- Operating System (The program runs in all the Windows NT operating system like Windows NT, Windows XP, Windows Vista, etc.)
- .Net framework (3.5 or higher)
- Microsoft Office 2000 or higher with MS Access component installed

## ***ALTERNATIVE SOLUTIONS***

### ***Manual Solution:***

**It is the current method used by the organization. All the records are stored on a employee's entry card. All the calculations are done manually. For more details see the section 1.1.2**

### ***Computerized Solution:***

- ❖ Microsoft Access can be used to store earning details. This method will be much more efficient than storing data on entry card. Microsoft access has various useful features like searching a table in the database, filtering records etc.
- ❖ A Be-Spoke Software (a software created with the purpose to meet the specific demand of a system) can be developed in order to simplify the problem. It would provide features exactly matching what the system needs.

## ***ADVANTAGES AND DISADVANTAGES OF THE PROPOSED SYSTEMS***

### ***Manual Solution***

**Advantage:** The user is flexible to fill any record he wants. There is no limitation of sorting.

**Disadvantage:** It is time consuming. It becomes difficult to handle all records manually. Moreover, the records will not be safe.

### ***Computerized Solutions***

#### ***I. Using MS Access***

**Advantage:** Records will remain in order when it is inserted into different tables based on their fields. It will not be as messy as in notebooks and registers.

**Disadvantage:** The system will not exactly fulfill all the needs of the operator. The data will still need to be entered manually.

#### ***II. Using Be-spoke software***

**Advantage:** Be spoke software is the best possible solution to the organization. As it is tailored, it is totally made to fulfill all system requirements of the organization. It will be fully automatic as every aspect of the existing system will be studied thoroughly before creating a new system. All the records will be handled without error.

**Disadvantage:** The only problem in the existing system is that it will take some time for the members to blend in with the new system and also it will take more time to develop. Moreover, as it is only “one of a kind” software, it will involve more expenses.

## **FEASIBILITY STUDY**

### **Technical Feasibility:**

- The software that is to be designed is a simple be spoke software. Man power for repairs and maintenance is easily available.
- A few Giga bytes of hard disk drive is enough.
- A Pentium 4 or higher processor

### **Economic Feasibility:**

- Due to having constant load shedding in Nepal, a home UPS system would also be needed to supply power for the computer and printer.
- A printer and a computer system are required.
- Printing bills in an A4 paper will be expensive. It was suggested that a dot matrix printer be used but there were difficulties relating drivers

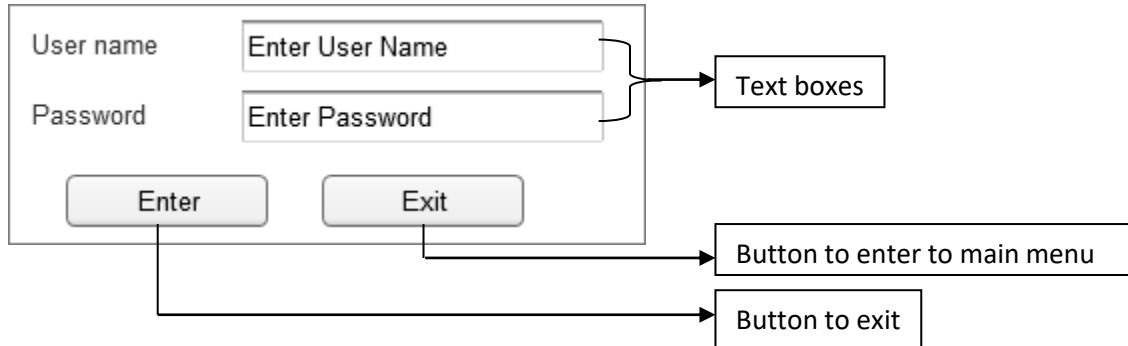
### **Legal Feasibility:**

- The system follows the rule of Data Capture Rights of the country.
- Copyrights of the system must be registered.

## **2. System Design**

### **SCREEN LAYOUT**

#### **Form to login**



This is like a splash screen, it will be shown first. IF the user enters correct username and password then main menu form will be shown and other stuffs like add, edit and so on will be shown. If the user is administrator then he will have right to add other and delete user else he can't delete any users. He will also have option to change password and username once logged in.

## Form to add Employee

Employee ID	<input type="text" value="EnterId"/>
Name	<input type="text" value="Enter Name"/>
Gender	<input type="text" value="Choose Gender"/>
Class	<input type="text" value="Choose Class"/>
Type	<input type="text" value="Choose Type"/>
Rate	<input type="text" value="Enter Rate"/>
Temporary Address	<input type="text" value="Enter Temporary Address"/>
Permanent Address	<input type="text" value="Enter Permanent Address"/>
Mobile No	<input type="text" value="Enter mobile Number"/>
Citizenship No	<input type="text" value="Enter Citizenship Number"/>
Licence No	<input type="text" value="Enter Licence No"/>
Date of Birth	<input type="text" value="Enter Date Of Birth"/>
Date Hired	<input type="text" value="Enter Date Hired"/>



Button to Select Image

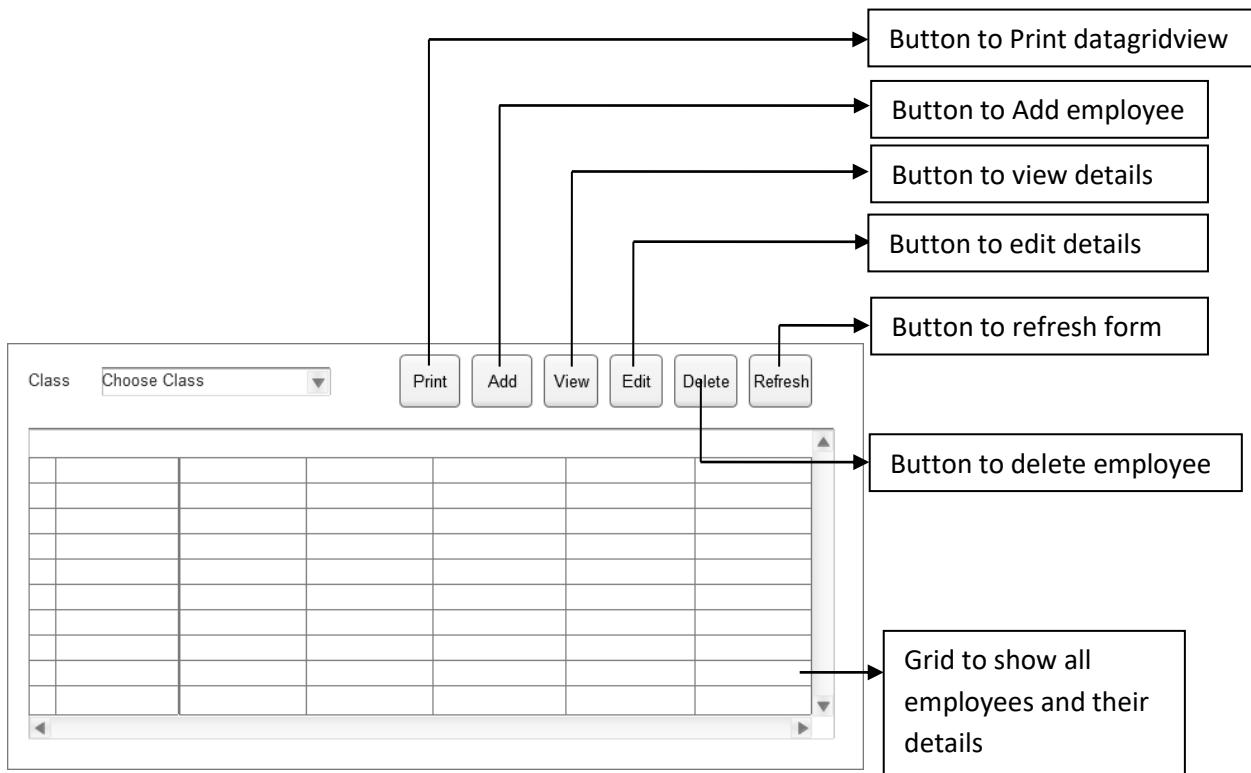
Button to Close

Button to clear all text boxes

Button to Save details

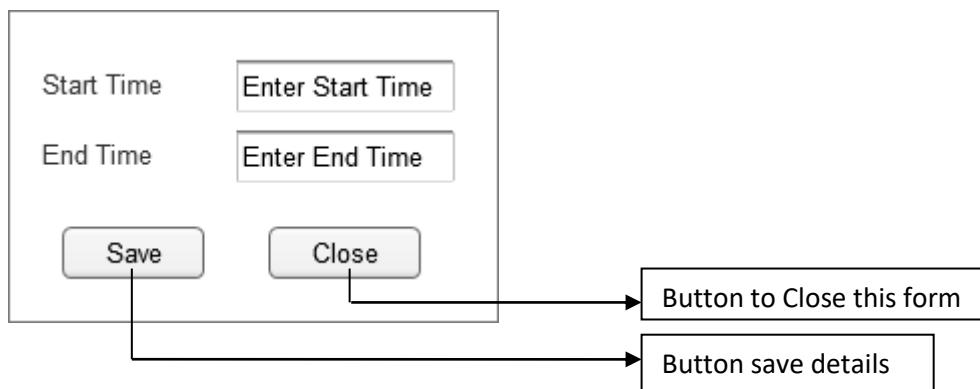
This form allows user to save all details of an employee. Details like id, name, gender, rate, salary, address and so on will be saved. Gender, class and type are selected from combo box. When all details are filled and save button is pressed then all details are saved to database. Clear button clears all details of textboxes and close button closes this form.

### Form to edit employee's details



This form allows user to add employee, edit employee's details, view employee's details, delete employee and print employee details. Datagridview is filtered by choosing class. If carpenter is selected then all carpenter is listed in datagridview else monthly paid employees are listed. Print button show print page and grid can be printed. Add button opens add employee form and new employee can be add through this form. View button opens add employee form but with all text boxes disabled and save and clear button hidden. Edit button opens add employee form with new form savechanges and details can be changed through this form. Delete button deletes the employee which is in selected row. Refresh button refreshes the form.

### Form to change common working hours



This form allows user to save common working hours. When form is loaded previously stored start time and end time is loaded shown in text box. When start time or end time is change and save button is pressed then it replaces old times and store new time from text boxes.

### **Form to edit monthly paid employee's salary**

Year Choose Year ▾

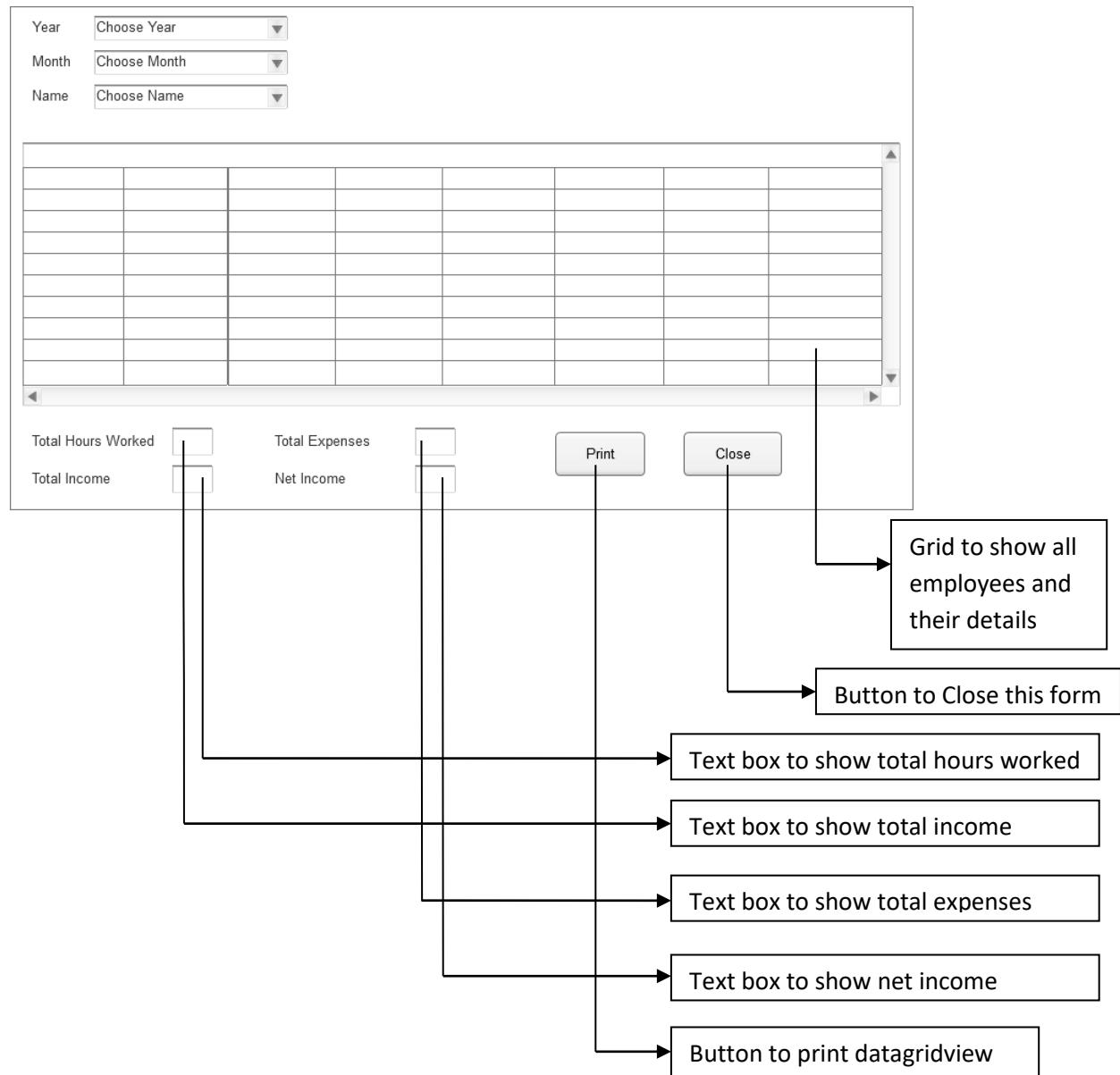
Month Choose Month ▾

Name Choose Name ▾

Grid to show all employees and their details

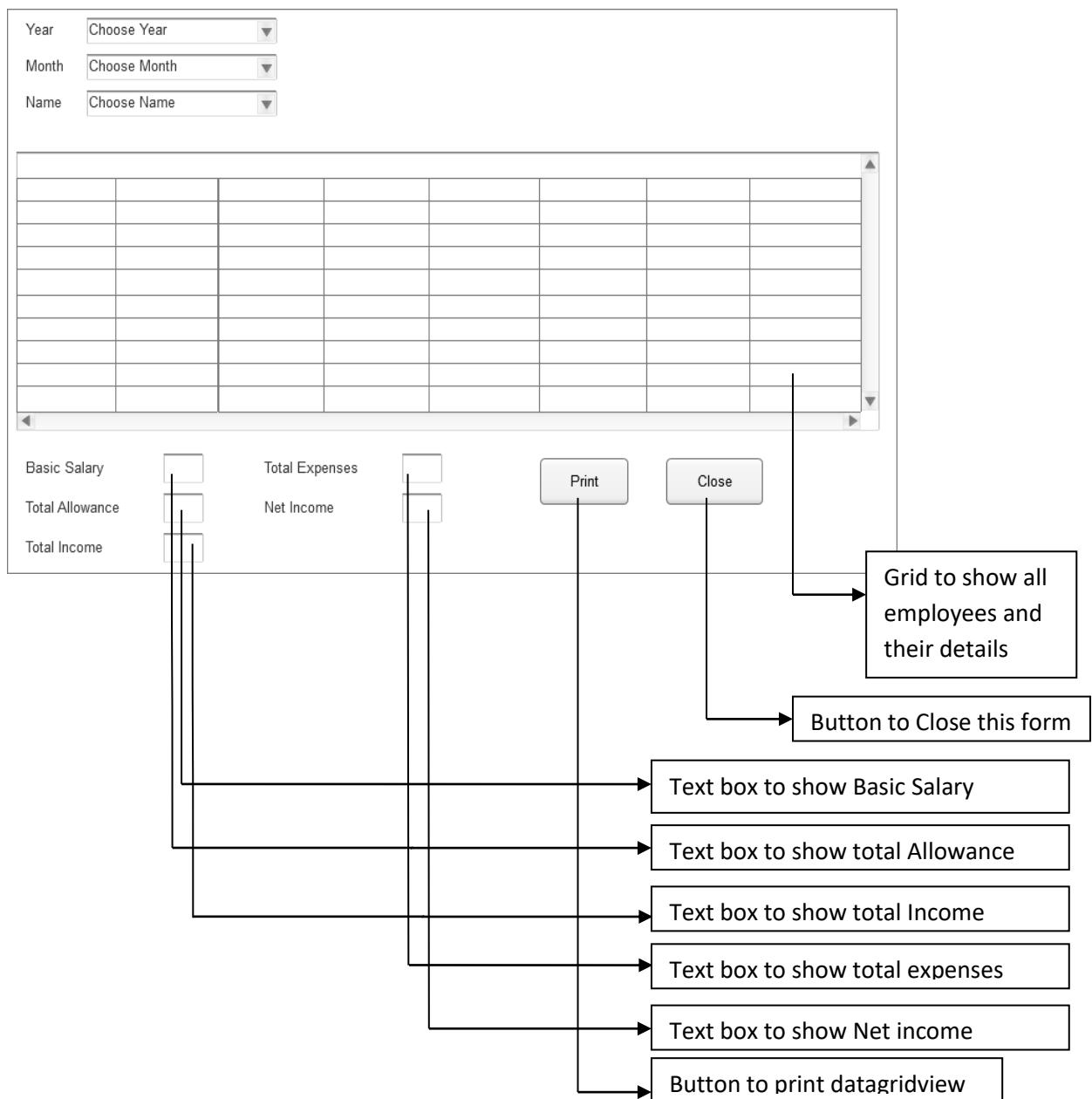
**This form allows user to save salary of a monthly paid employee. Today's year and today's month is selected automatically. All name from database is added to name's combo box. When certain name is selected a row is added to datagridview with previously stored salary and it can be edited by editing the value of salary column.**

## Form to view carpenter's income details



This form allows user to view carpenter's income details. Today's year and today's month is selected automatically. All name from database is added to name's combo box. When certain name is selected rows are added to datagridview with previously stored working hours. Total working hours, total income and total expenses are calculated by adding up their respective columns. Net income is also generated by subtracting total expenses from total income. Print button enables user to print this datagridview and close button closes this form.

## Form to view monthly paid employee's income details

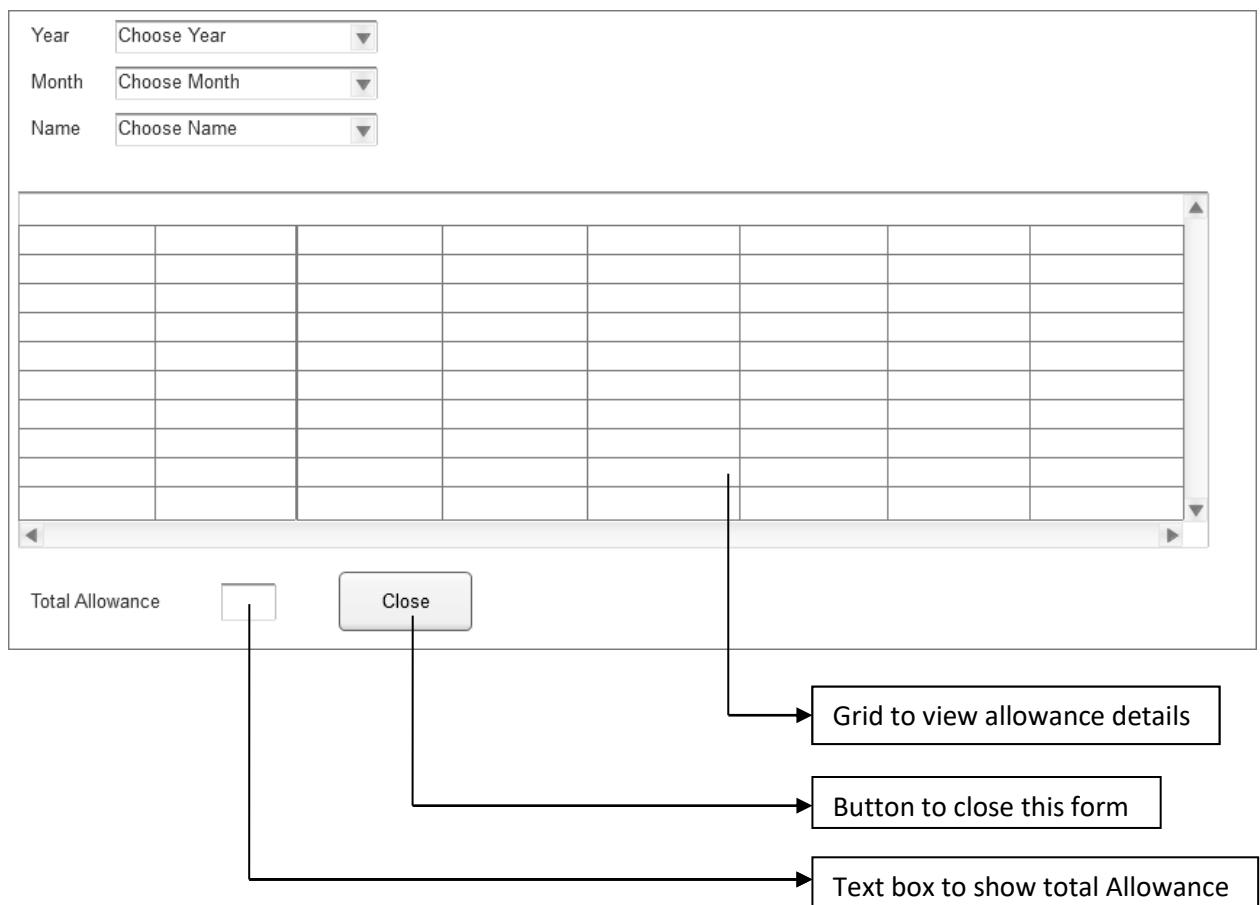


This form allows user to view monthly paid employee's income details. Today's year and today's month is selected automatically. All name from database is added to name's combo box. When certain name is selected rows are added to datagridview with previously stored salary, expenses and allowances. Basic salary, total allowance and total expenses are calculated by adding up their respective columns. Total income is calculated by adding basic salary and allowance. Net income is also generated by subtracting total expenses from total income. Print button enables user to print this datagridview and close button closes this form.

## **Form to view monthly paid employee's salary**

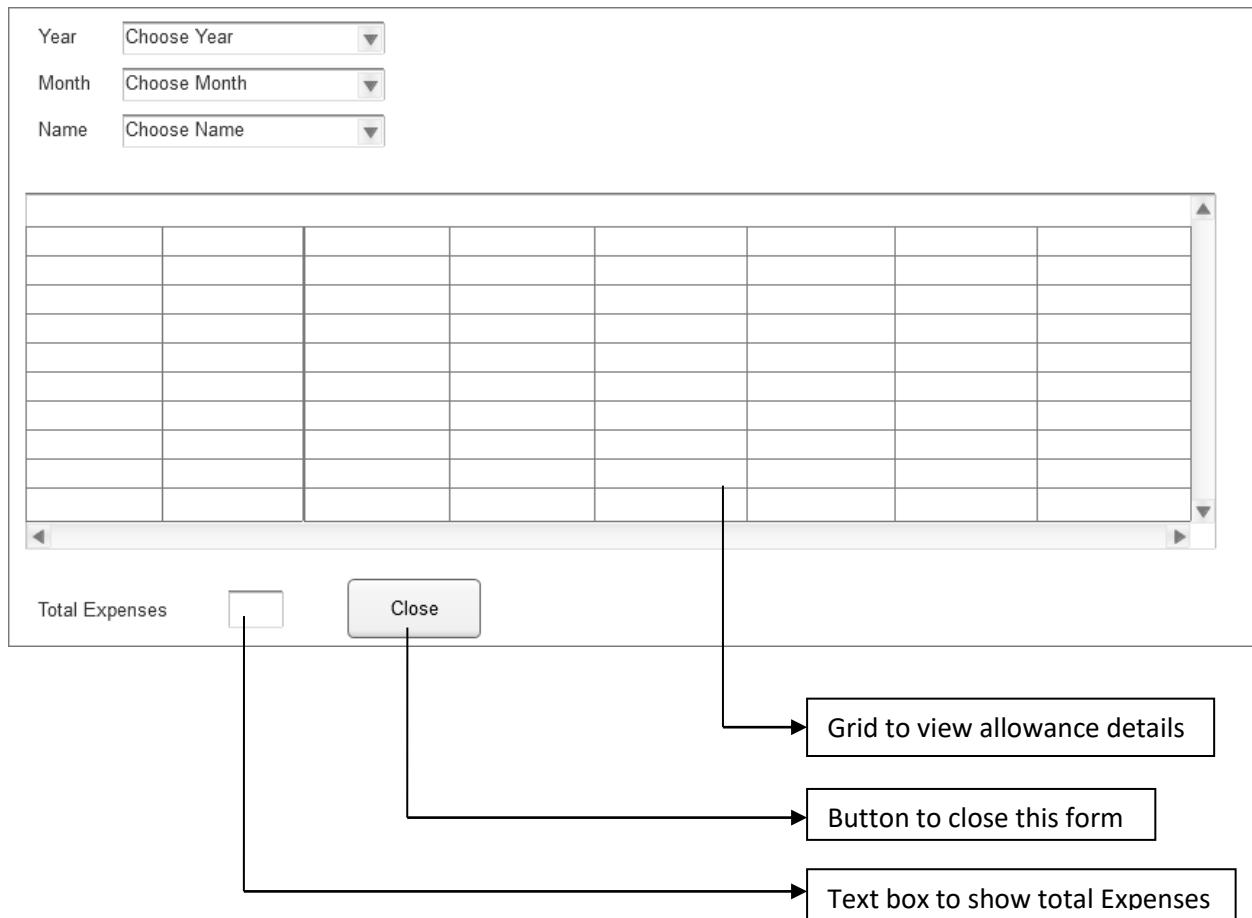
This form allows user to view monthly paid employee's salary. Today's year and today's month is selected automatically. All name from database is added to name's combo box. When certain name is selected rows are added to datagridview with previously stored salary.

### Form to view monthly paid employee's allowances



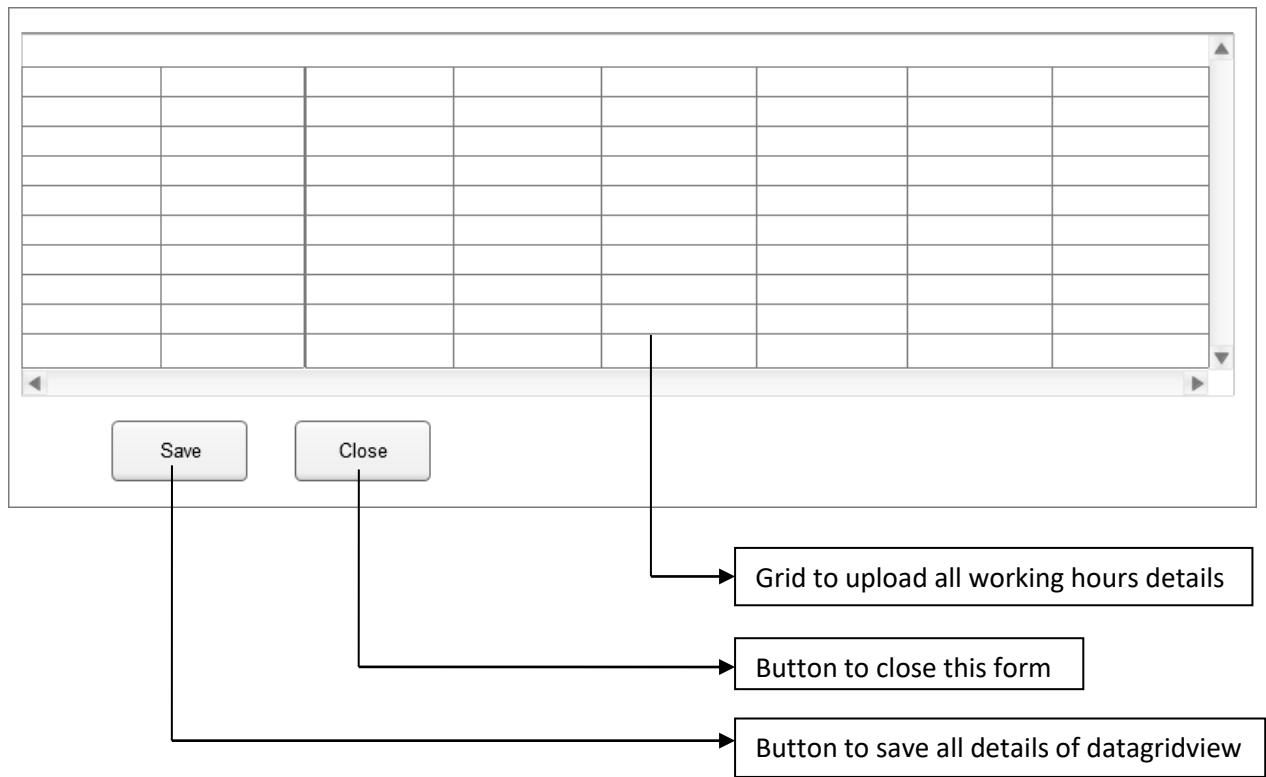
This form allows user to view monthly paid employee's allowance. Today's year and today's month is selected automatically. All name from database is added to name's combo box. When certain name is selected rows are added to datagridview with previously stored allowances. Total allowance is calculated by adding up its columns and shown in text box. Close button closes this form.

### Form to view monthly paid employee's expenses



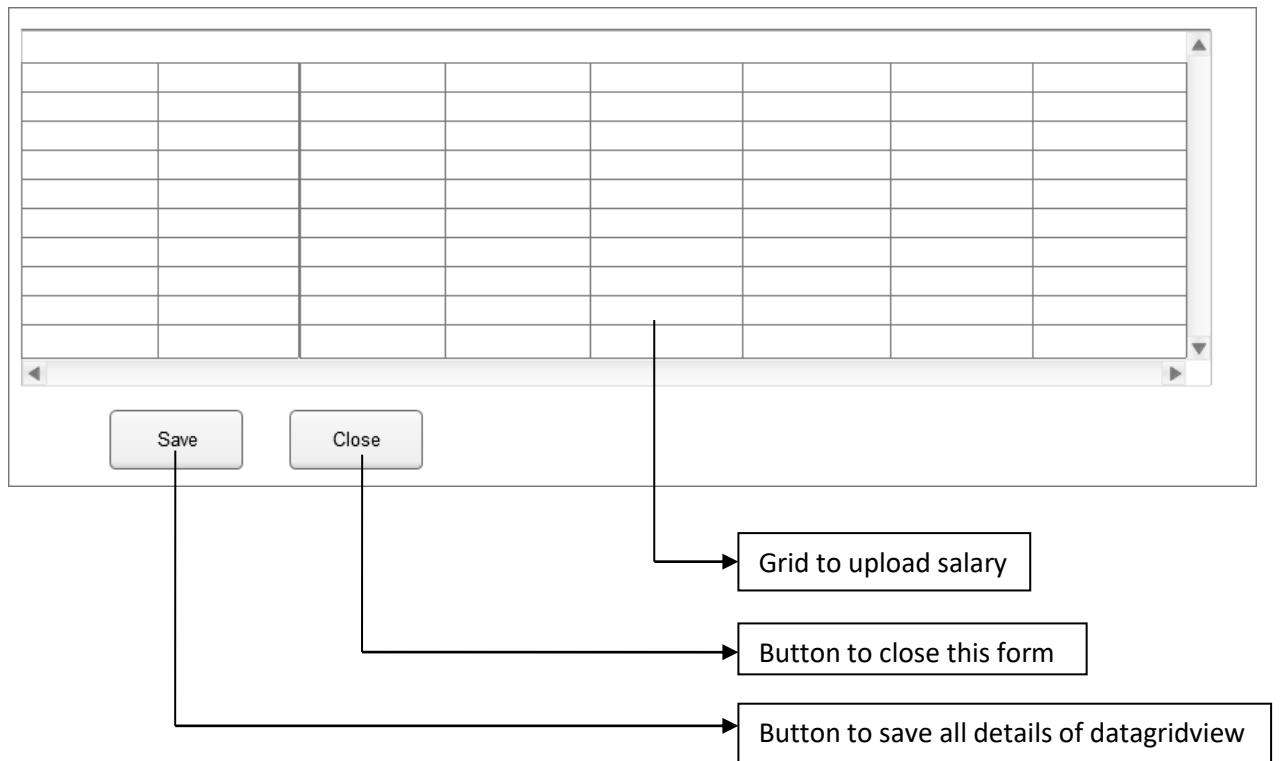
This form allows user to view monthly paid employee's expenses. Today's year and today's month is selected automatically. All name from database is added to name's combo box. When certain name is selected rows are added to datagridview with previously stored expenses. Total expenses is calculated by adding up its columns and shown in text box. Close button closes this form.

### Form to upload carpenter's working hours details



This form allows user to upload carpenters working hours details. When form is loaded rows with all employee's name and common working is added with expenses as zero. Working hours, expenses can be changed by changing its value. Save button saves all data from grid view and close button closes this form.

### Form to upload monthly paid employee's salary



This form allows user to upload monthly paid employee's salary. When form is loaded rows with all employee's name and stored salary is added. Salary can be changed by changing its value. Save button saves all data from grid view and close button closes this form.

### Form to upload monthly paid employee's allowances

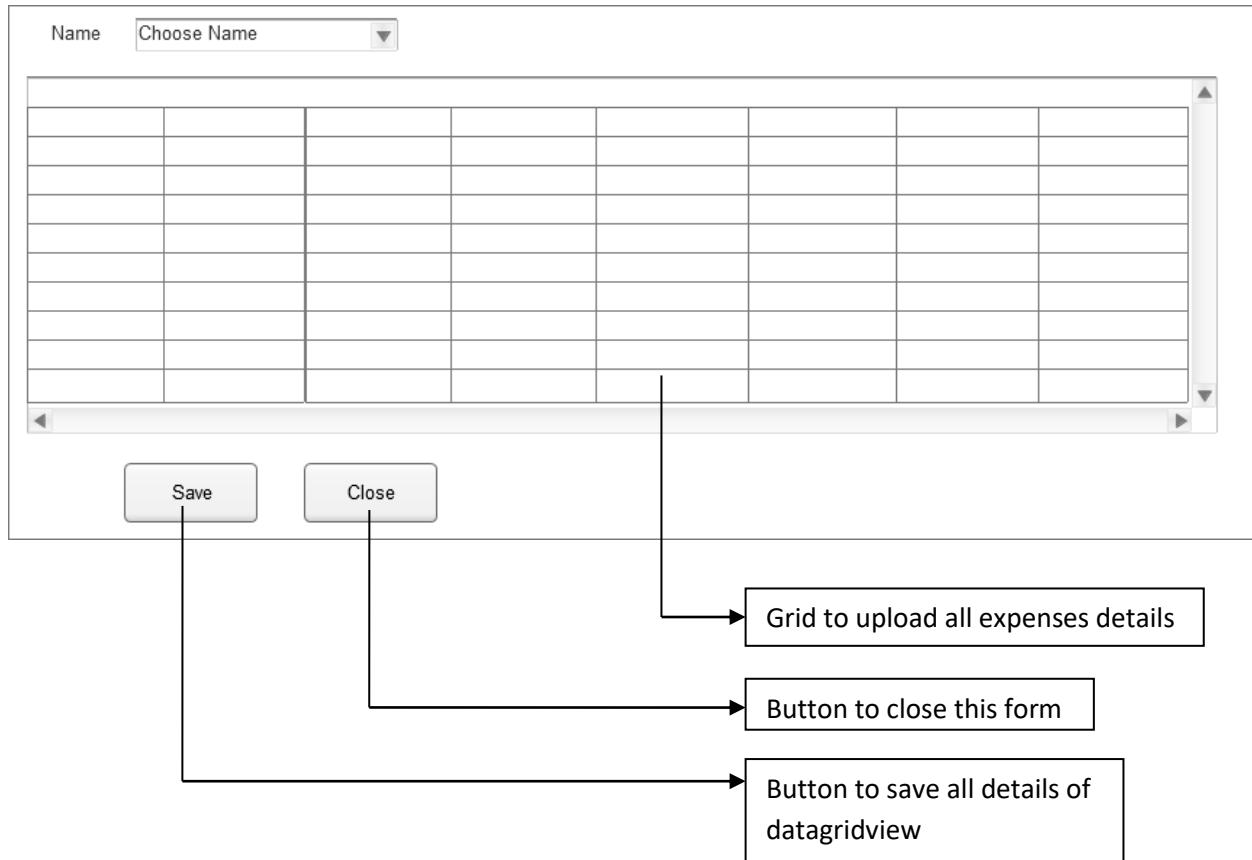
Year	Choose Year
Month	Choose Month
Employee ID	EnterId
Name	Choose Name
Amount	Enter Name
Allowance title	Enter Rate

**Save**    **Close**

→ Button to close this form  
→ Button to save all details of text boxes

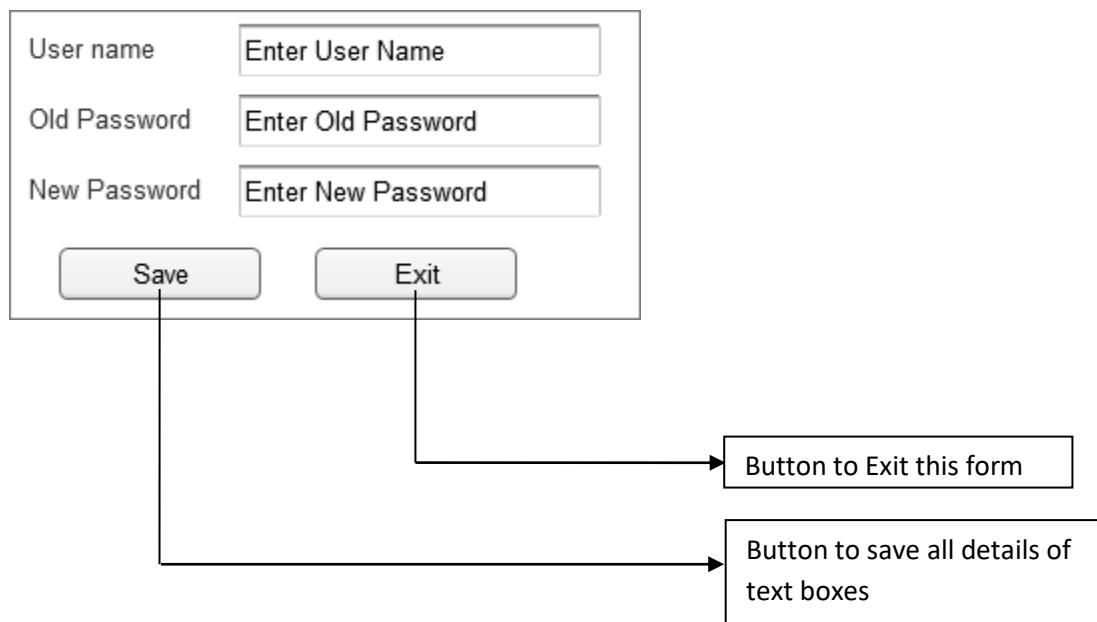
This form allows user to upload monthly paid employee's allowance. Today's year and today's month is selected automatically. All name from database is added to name's combo box. When certain name is selected its id is automatically extracted and shown in id's text box. Amount of allowance and title of allowance is to manually written. Save button saves all data from this form and close button closes this form.

### Form to upload monthly paid employee's expenses



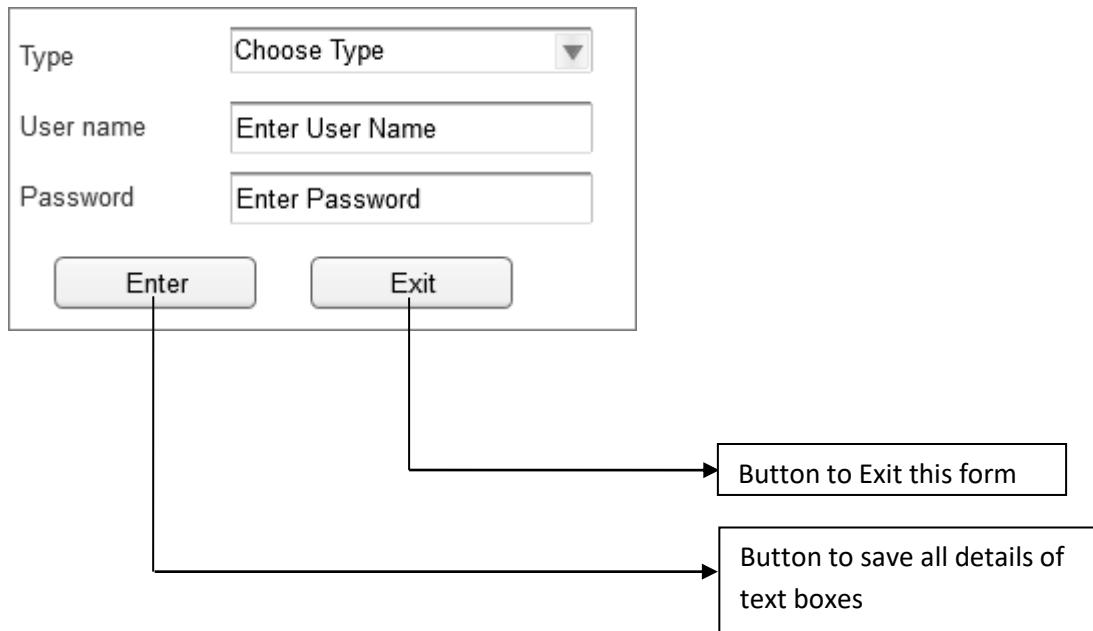
This form allows user to upload monthly paid employee's expenses. All name from database is added to name's combo box .When a name is selected a row is added with date and 0 as an expense. Expenses can be changed by changing its value. Save button saves all data from grid view and close button closes this form.

### Form to change user password



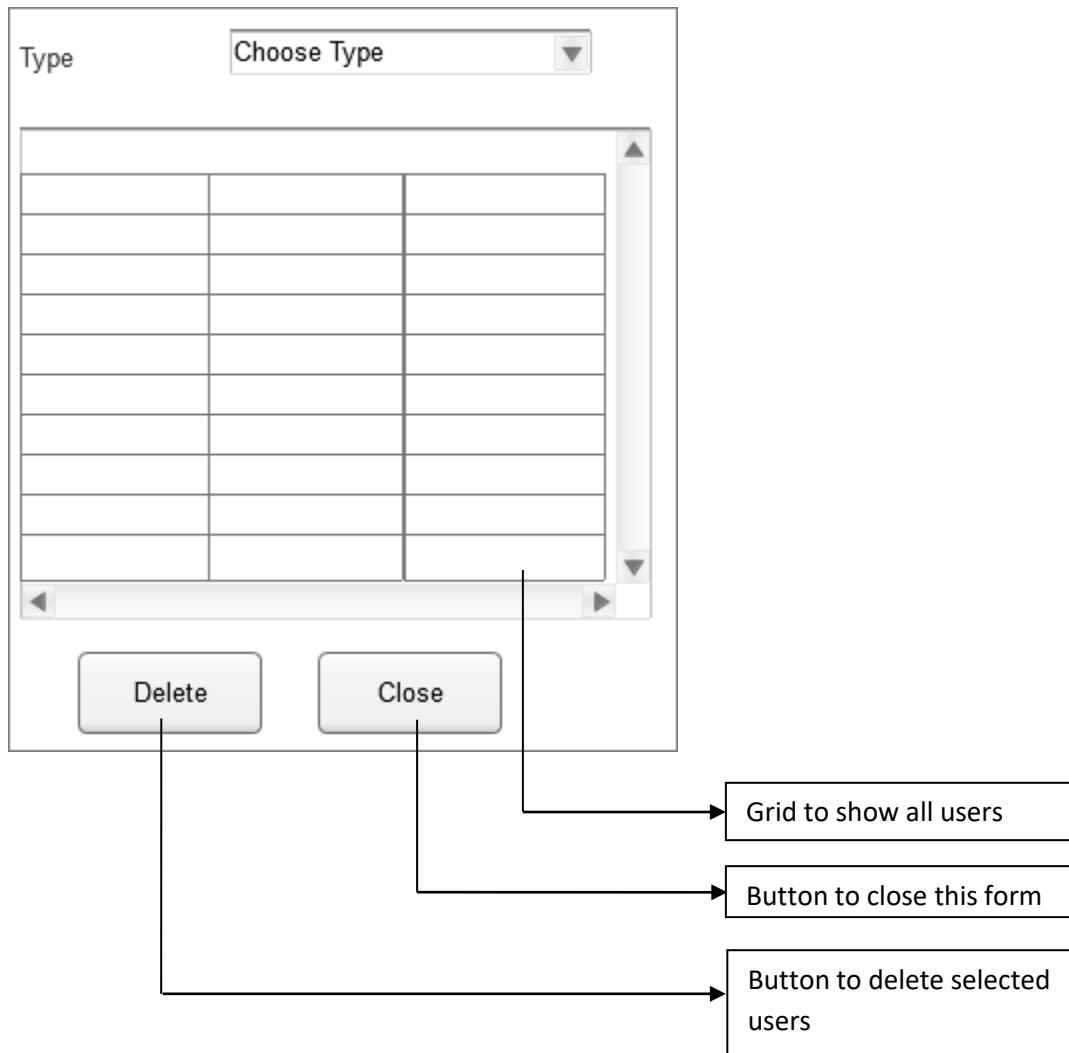
This form allows user to change username or password. User name is automatically shown in username's text box. To change password old password should be same as typed in textbox else it will not change the username or password. After typing old password user should enter new password and click on save to save username and password. Exit button closes this form.

### Form to add other users



**This form allows user to add new username and password. User name is entered in username's text box and password in password's textbox. Type should be selected from combo box. After typing username and password user should click on save to save username and password. Exit button closes this form.**

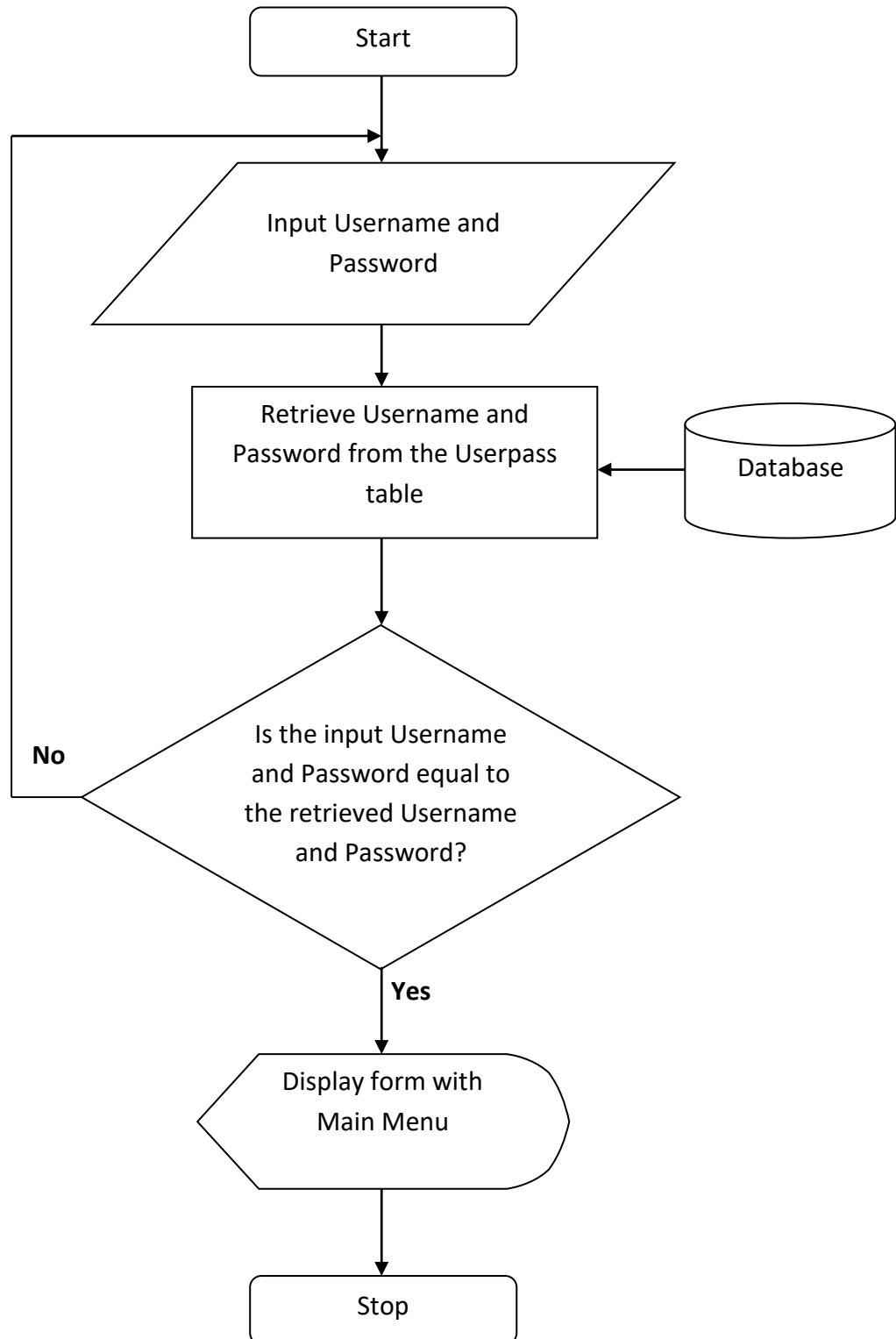
### Form to view and delete users



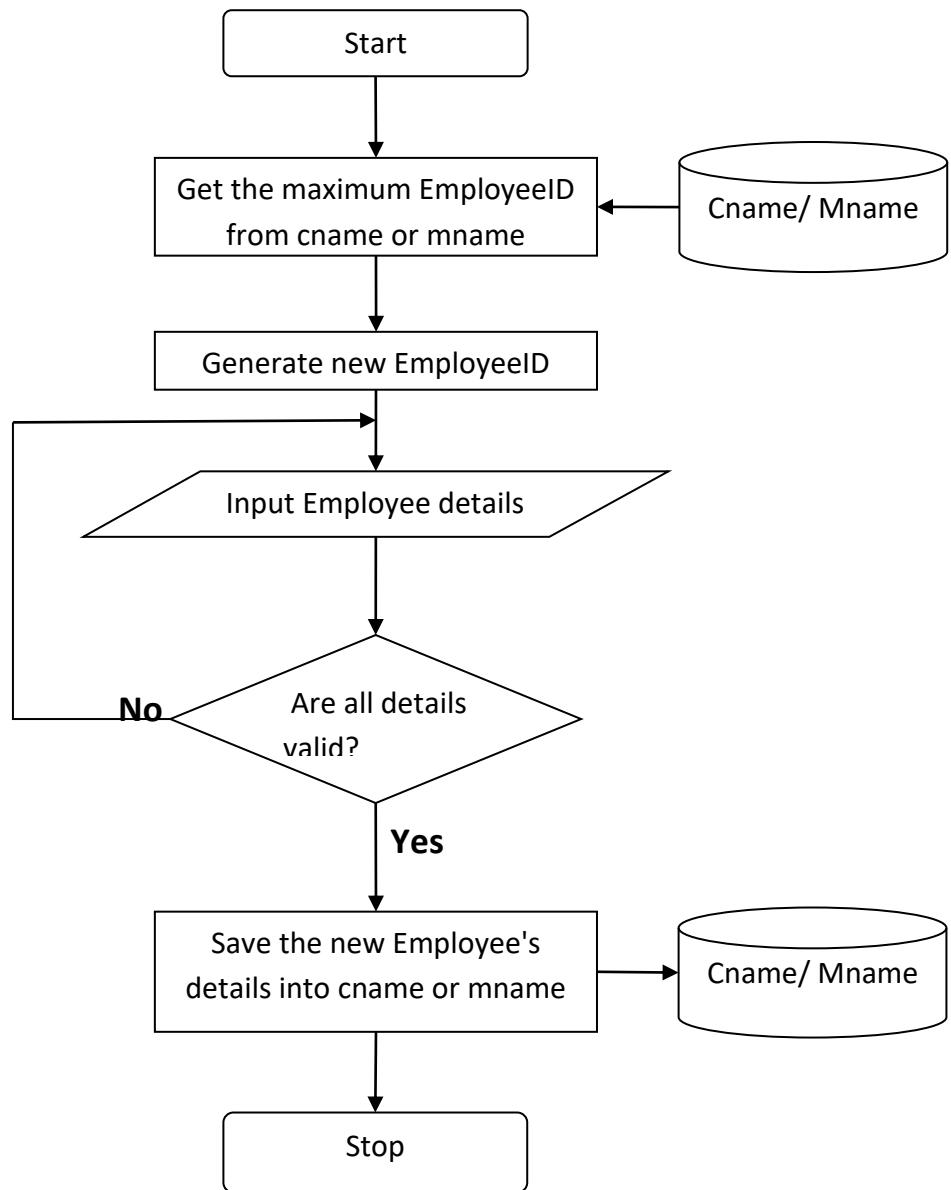
This form allows user to delete user. Type should be selected from combo box. After selecting type all users are listed in grid view. User should select the user which you want to delete. Delete button deletes the user and close button closes this form.

## **FLOWCHARTS**

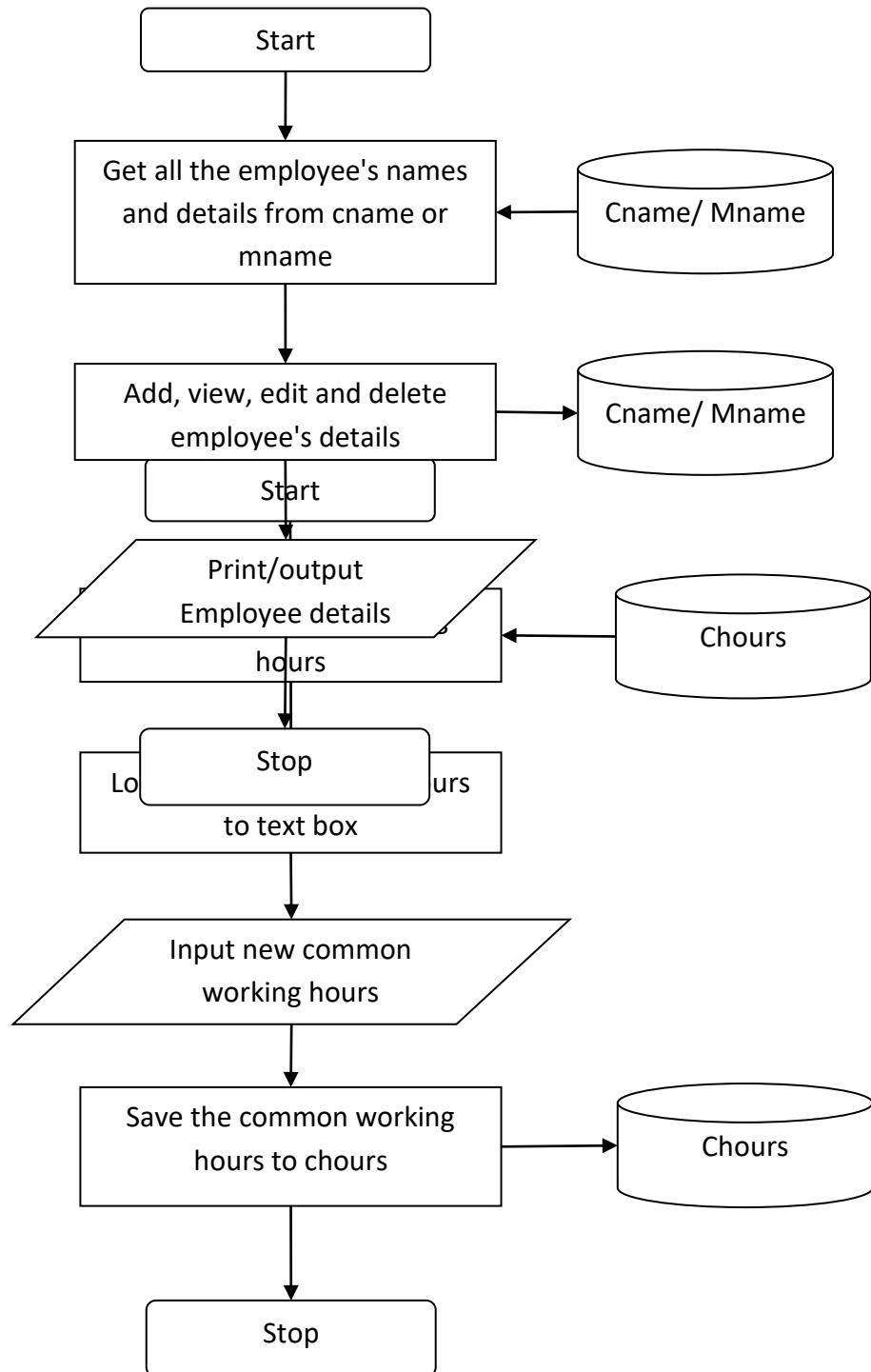
### **Log In**



## Adding Employee

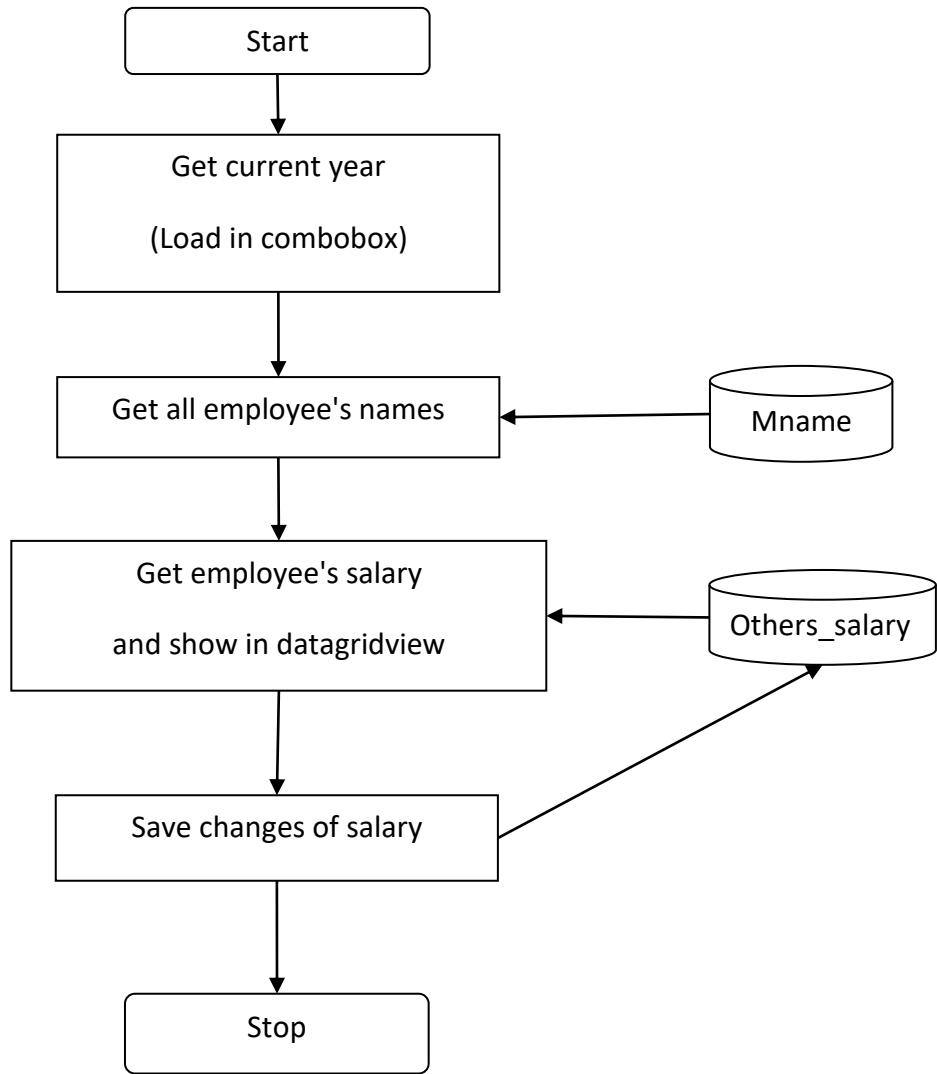


## Editing Employee

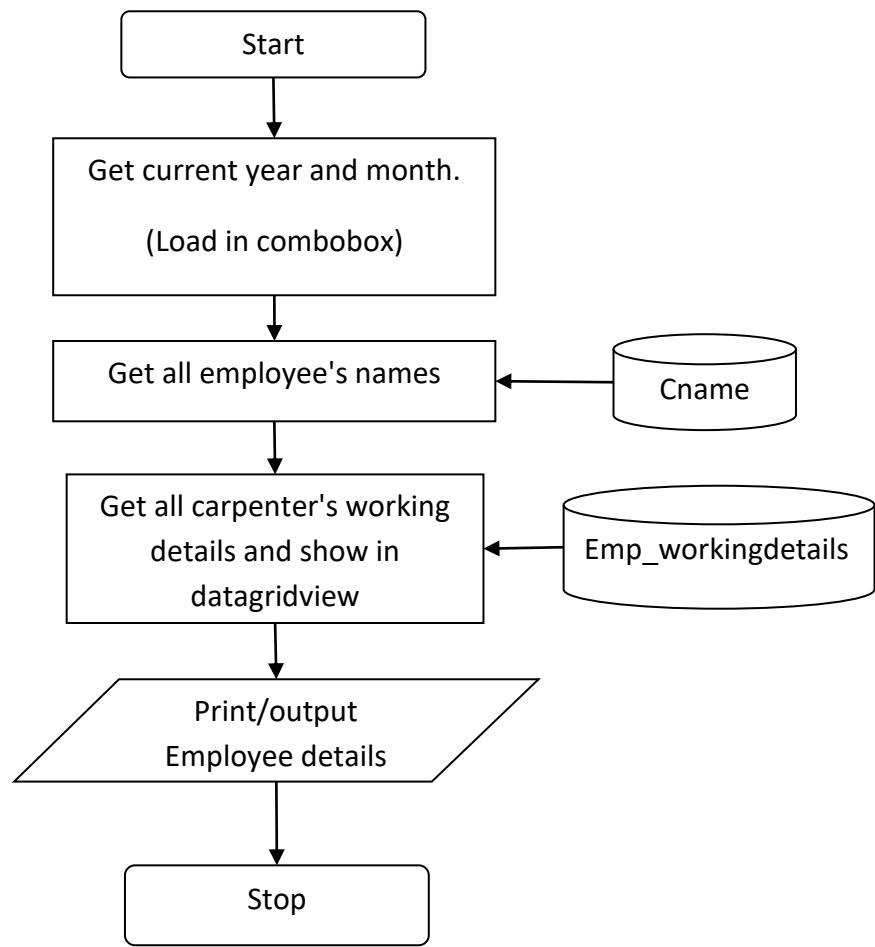


## Add and Edit Common working Hours

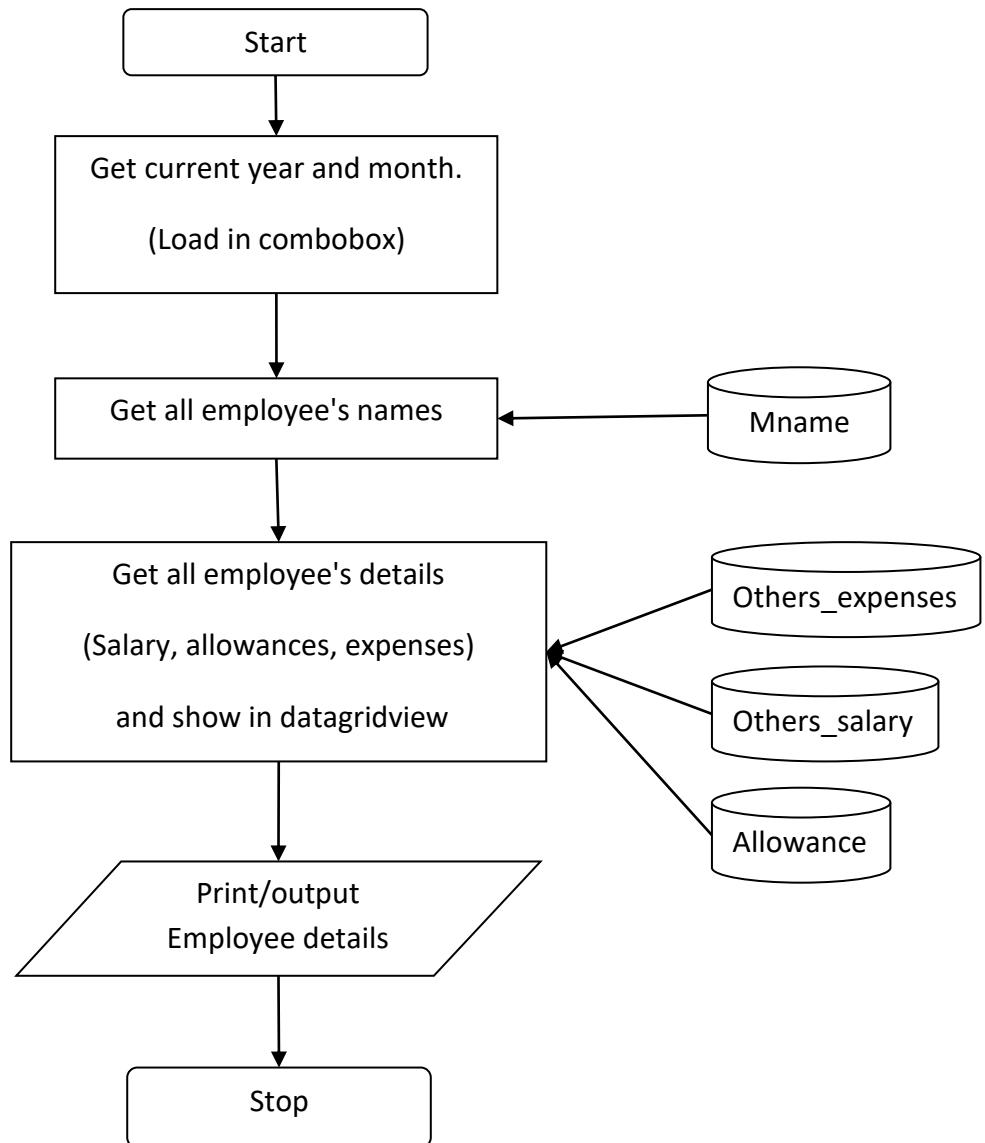
## Edit Monthly Paid Employee's Salary



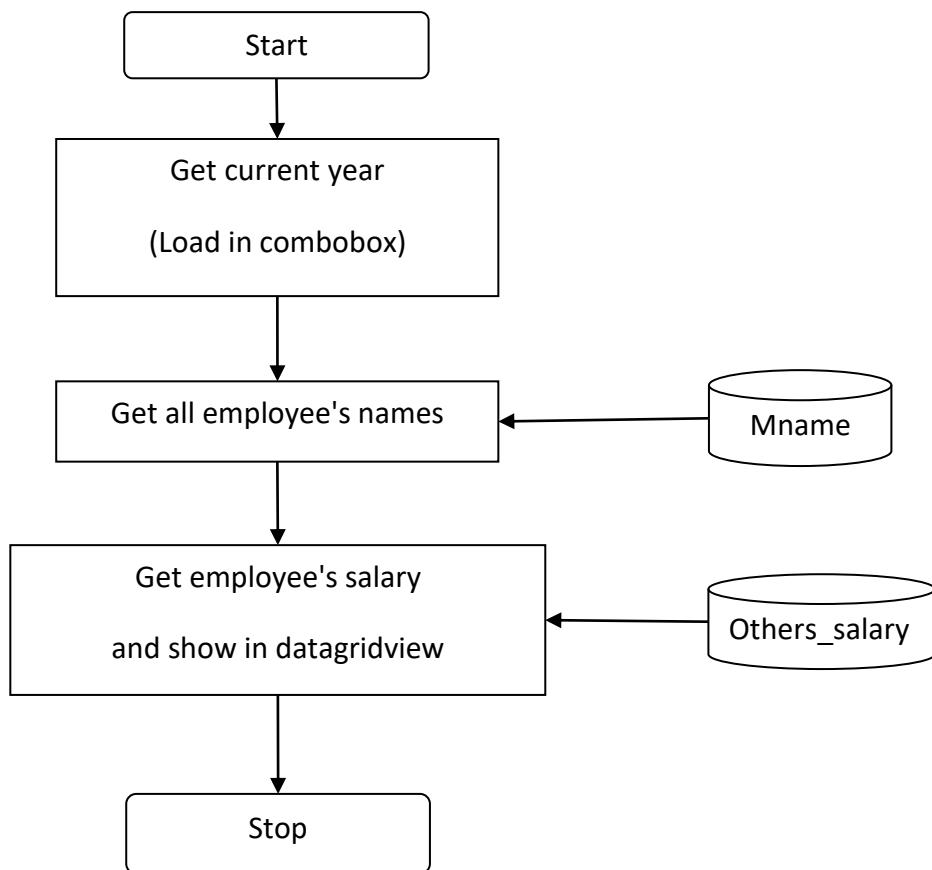
## **View carpenters income details**



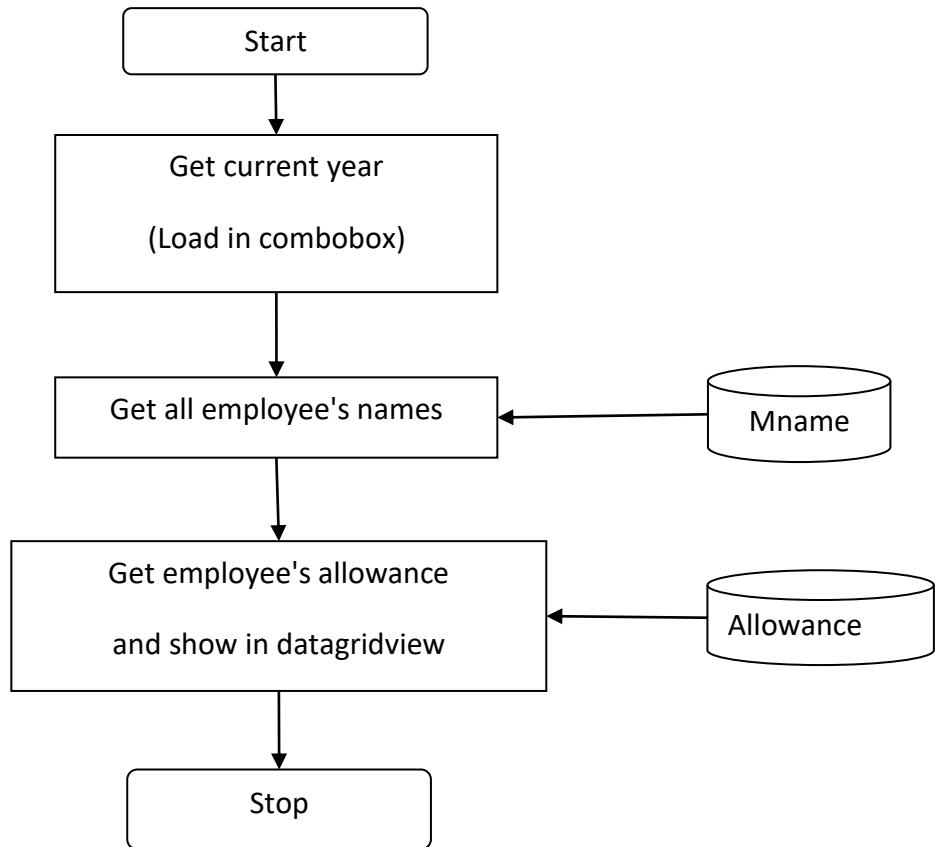
## **View Monthly Paid employee's income details**



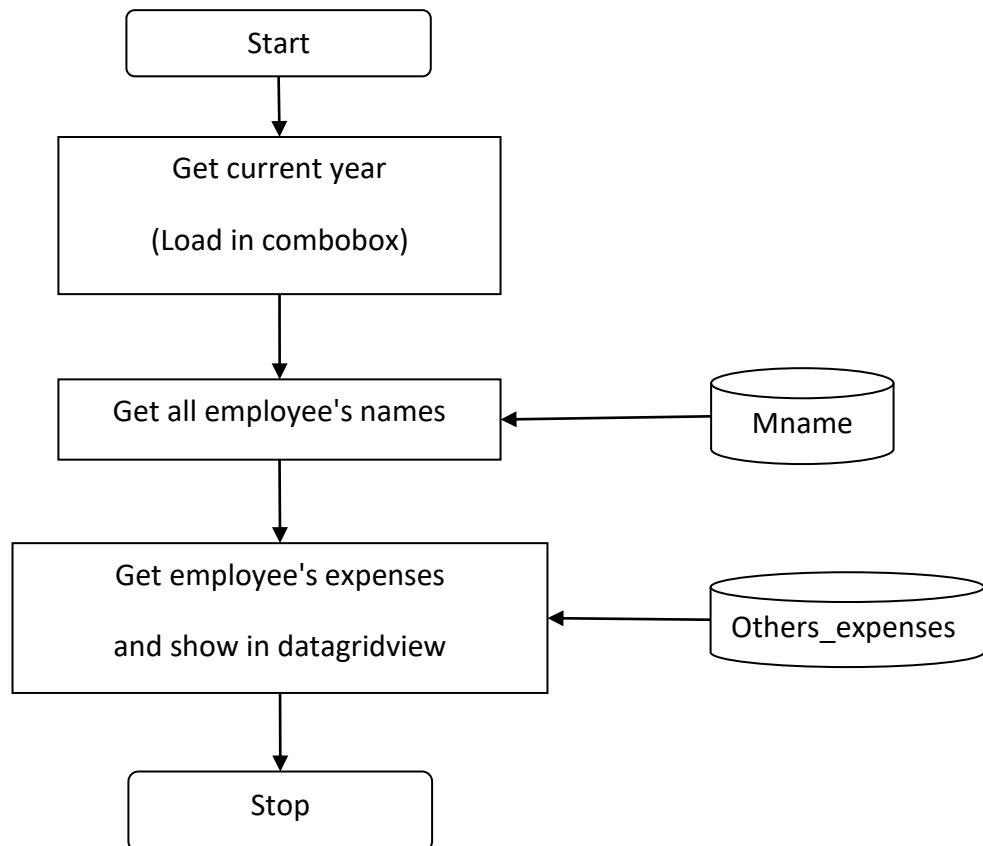
## **View Monthly Paid employee's salary details**



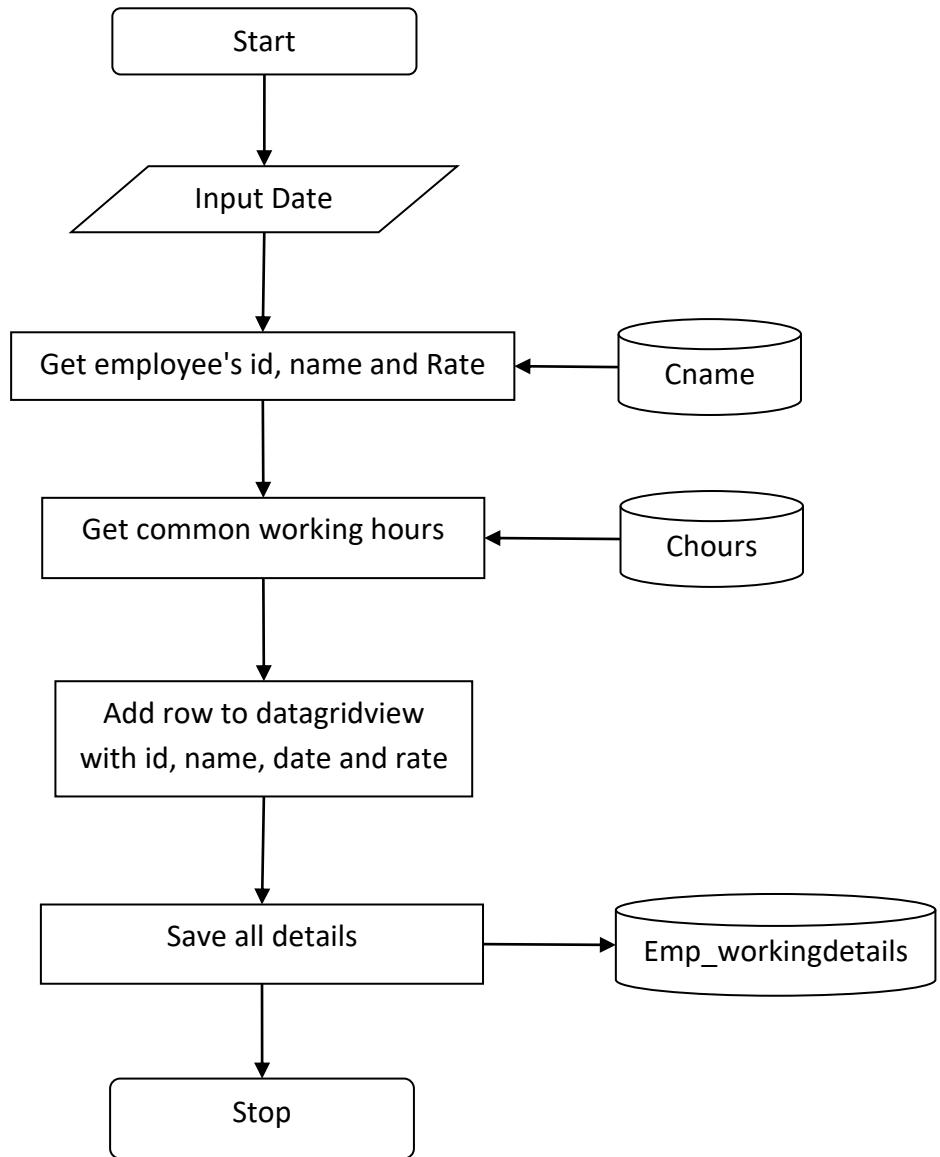
## **View Monthly Paid employee's allowance details**



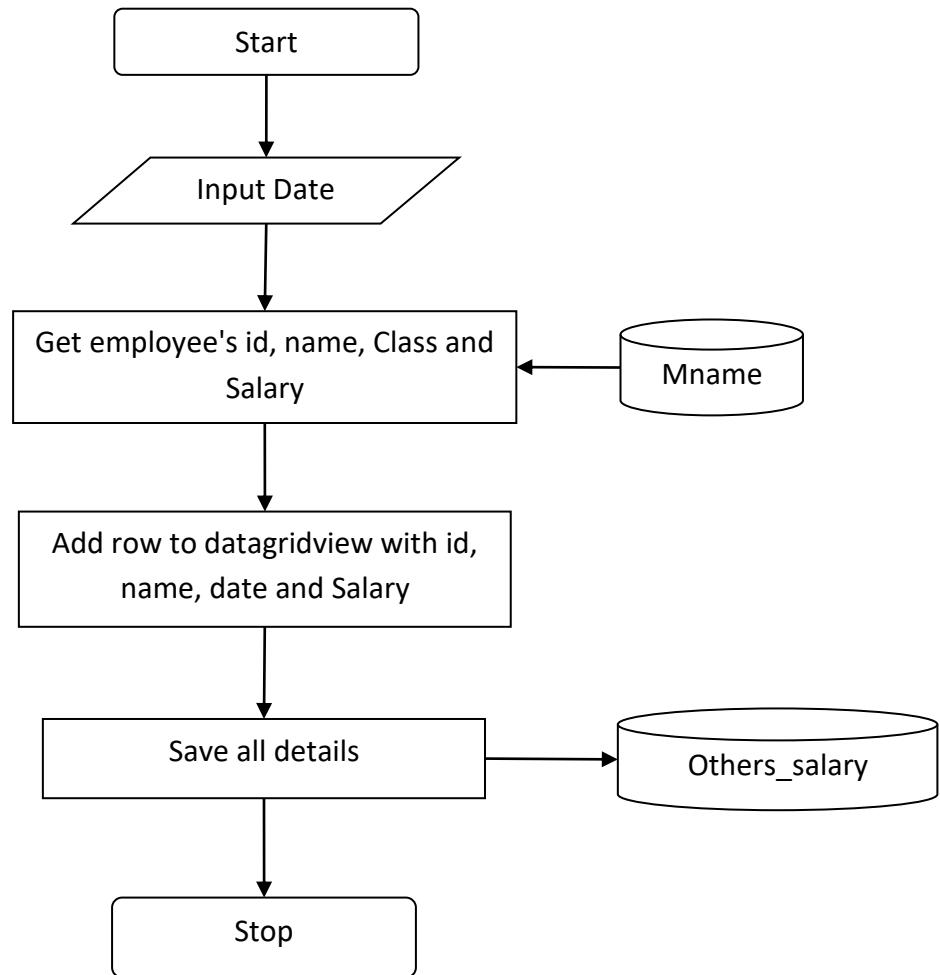
## **View Monthly Paid employee's Expenses details**



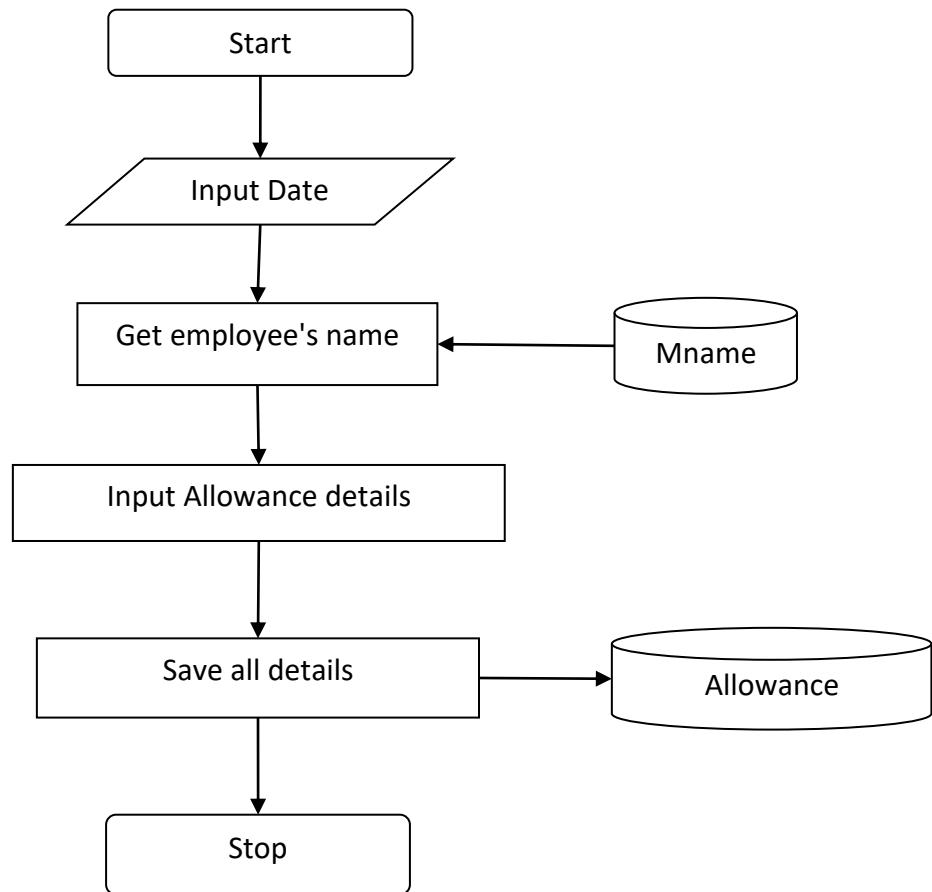
## Upload carpenters working hours



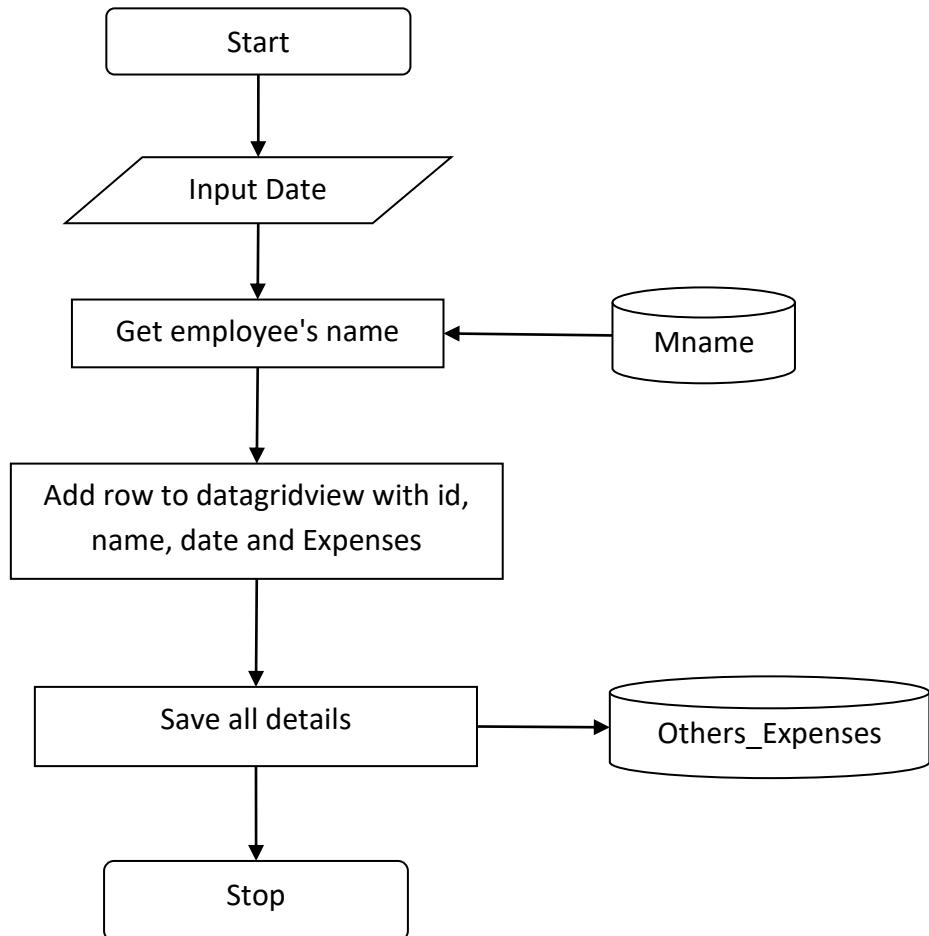
## Upload Monthly paid employee's salary



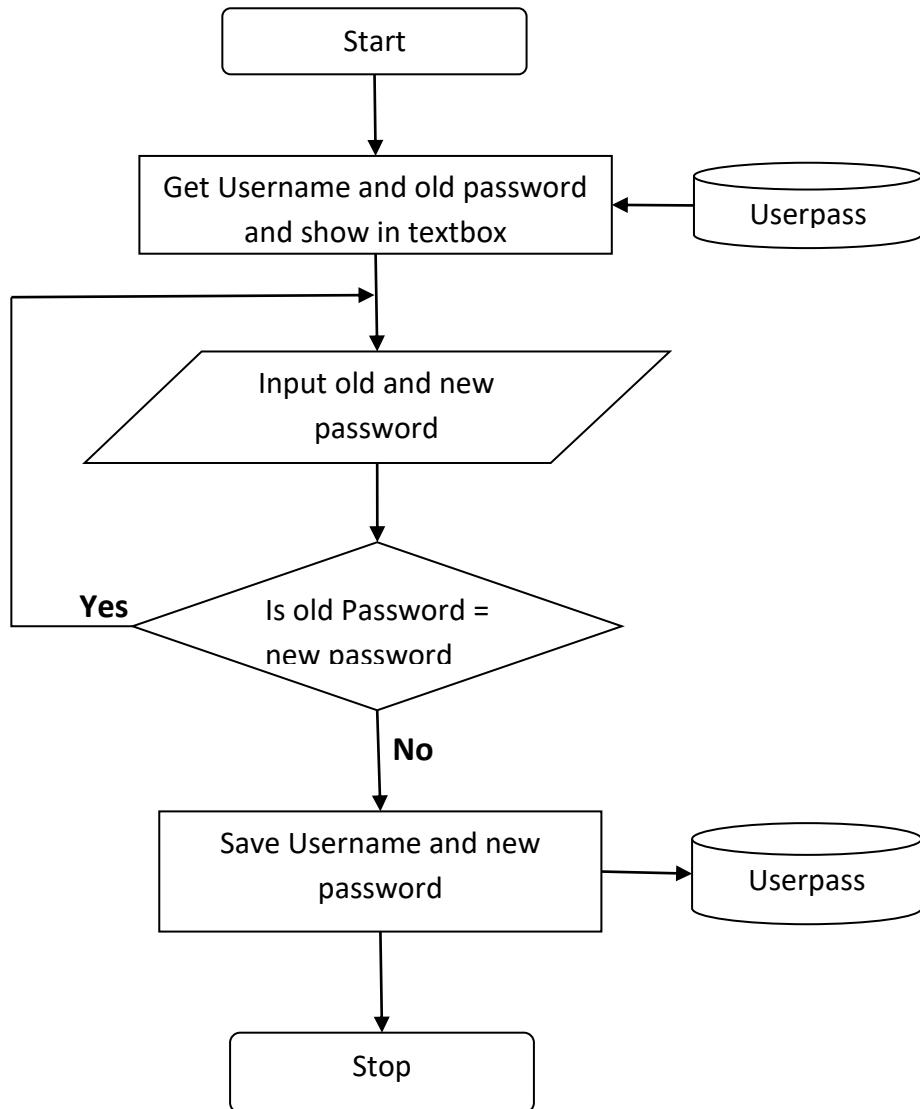
## Upload Monthly paid employee's Allowance



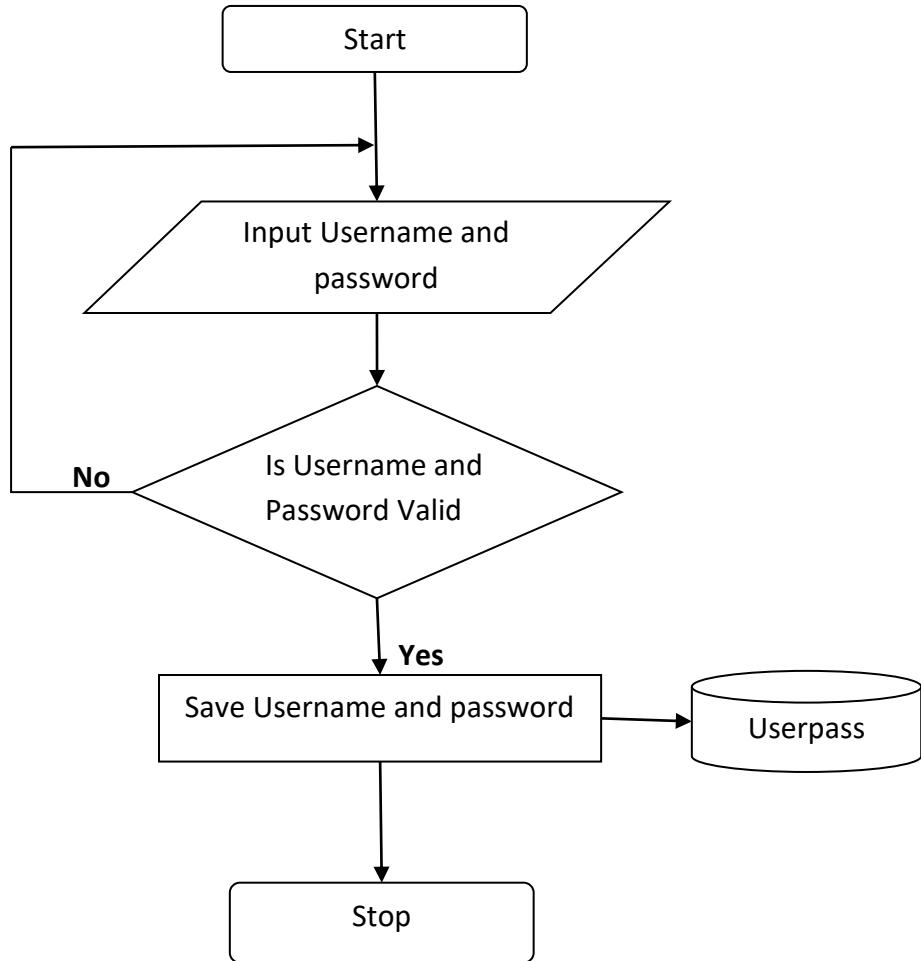
## Upload Monthly paid employee's Expenses



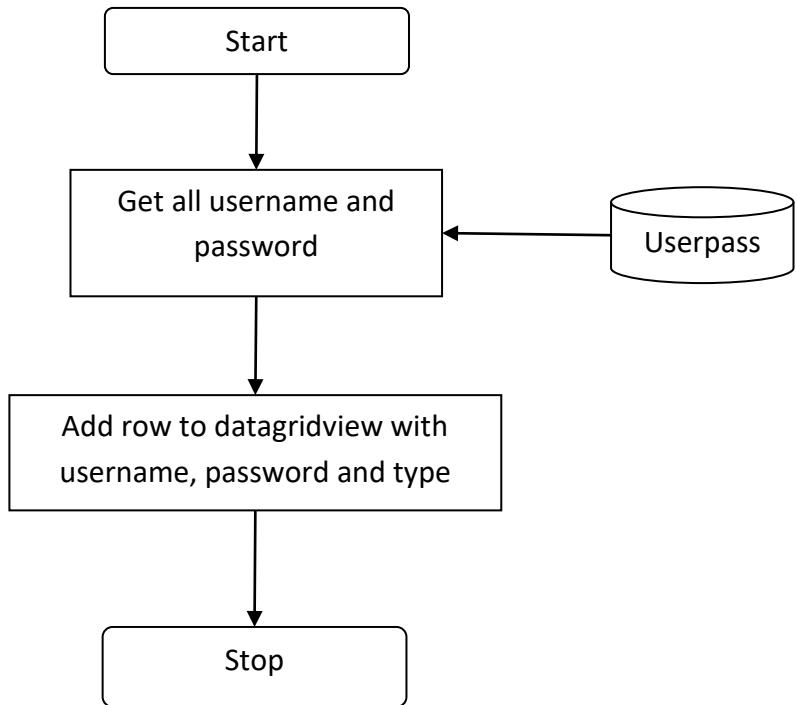
## Change Password



## Add Other Users



## Show all Users



## TABLE STRUCTURE

Table: Allowances

Field Name	Data Type	Description
Fyear	Number	Allowances of which year
FMonth	Number	Allowances of which month
Emp_ID	Number	Employee's employee id is stored here
Emp_Name	Text	Employee's employee name is stored here
Allowance Title	Text	Title of allowance is stored here
Amount	Number	Amount of allowance is stored here

Table: Cname

Field Name	Data Type	Description
Emp_ID	Number	Identification number for each carpenter
Carpenter_Name	Text	this field is for carpenter's name
Gender	Text	gender of carpenter is stored in this field
EType	Text	sub type of carpenter is stored in this field
ERate	Number	this is to store per hour rate of carpenter
EClass	Text	this is to store class of an employee
Temporary_Address	Text	this is to store temporary address of carpenter
Permanent_Address	Text	this is to store Permanent address of carpenter
Mobile_No	Text	this field is to store mobile number of carpenter
Citizenship_No	Text	this is to store citizenship no of carpenter
License_No	Text	this is to store license no of carpenter
Date_of_Birth	Date/Time	This field is for entering the Date of Birth of carpenter
Date_Hired	Date/Time	This field is for entering the hired Date of carpenter
Imagepic	Text	this is to store location of picture of carpenter

Table: Mname

Field Name	Data Type	Description
Emp_ID	Number	Identification number for each employee
Emp_Name	Text	this field is for manager's name
Gender	Text	gender of employee is stored in this field
Salary	Number	this field is for storing salary of manager
EClass	Text	this is class of an employee
Temporary_Address	Text	this is to store temporary address of manager
Permanent_Address	Text	this is to store Permanent address of manager
Mobile_No	Text	this field is to store mobile number of manager
Citizenship_No	Text	this is to store citizenship no of manager
License_No	Text	this is to store license no of carpenter
Date_of_Birth	Date/Time	This field is for entering the Date of Birth of manager
Date_Hired	Date/Time	This field is for entering the hired Date of manager
Imagepic	Text	this is to store location of picture of manager

Table: Ctime

Field Name	Data Type	Description
AM	Number	common starting AM time is store in this field
PM	Number	common ending PM time is store in this field

Table: Emp Workingdetails

Field Name	Data Type	Description
HMonth	Number	month(number) is stored in this field
HDay	Number	Day(number) is stored in this field
HYear	Number	Year(number) is stored in this field
Emp_ID	Number	Employee' id (number) is stored in this field
Emp_Name	Text	employee's name is stores in this field
Start Time	Number	starting time of employee's work of that day is stored here
End Time	Number	End time of employee's work of that day is stored here
Expenses	Number	Expenses of employee of that day is stored here
Rate	Number	Rate of employee in that day is stored here

Table: Others Expenses

Field Name	Data Type	Description
HYear	Number	Year(number) is stored in this field
HMonth	Number	month(number) is stored in this field
HDay	Number	Day(number) is stored in this field
Emp_ID	Number	Employee' id (number) is stored in this field
Emp_Name	Text	employee's name is stores in this field
EClass	Number	Employee's class is stored here
Expenses	Number	Expenses of employee of that day is stored here

**Table: Others\_Salary**

Field Name	Data Type	Description
HYear	Number	Year(number) is stored in this field
HMonth	Number	month(number) is stored in this field
Emp_ID	Number	Employee' id (number) is stored in this field
Emp_Name	Text	employee's name is stores in this field
EClass	Number	Employee's class is stored here
Salary	Number	Salary of employee of that month is stored here

**Table: Userpass**

Field Name	Data Type	Description
UserName	Text	User name of users are stored here
Password	Text	password for the users are stored in this field
Type	Text	this is to store type of users

### ***INTENDED BENEFITS OF THE NEW SYSTEM***

- All data are managed in a systematic way and it can be easily accessed with less time.
- Employees details are also stored.
- There is less chances of errors in calculation than by doing it manually
- Work load is reduced to managers as all calculation is done automatically.
- Data are more secure than paper based system.

## **SYSTEM LIMITATIONS**

**Some of the limitations of proposed system are : -**

- ❖ Only 499 employees can be added in carpenter because other's employee id start from 500 onwards.
- ❖ Employee Id number is discontinuous if employee having employee id less than maximum employee id is deleted.(Say we deleted employees having employee id 20 to 30 where there are employees up to id 100,then new employees will never placed in those empty places(20 to 30)).
- ❖ The new system doesn't use the network for data sharing.
- ❖ It can be run only on a single/stand-alone computer.
- ❖ There is no easier way to change the print format and layout, though it is possible by modifying source codes.
- ❖ The current system though runs on all major platforms with Wine Libraries, the program is not a native application for Linux, Mac OS X and BSDs. The program is not natively cross-platform.

## SIZE ESTIMATION OF DATABASE

Table: Allowances

Field Name	Data Type	Size (in bytes)
Fyear	Integer	4
FMonth	Byte	2
Emp_ID	Integer	4
Emp_Name	String	40
Allowance Title	String	20
Amount	Integer	10

*Total Size (per record) = 100 bytes*

*Total Records (approximately) = 1000*

Table: Cname

Field Name	Data Type	Size (in bytes)
Emp_ID	Integer	4
Carpenter_Name	String	40
Gender	String	6
EType	String	10
ERate	Integer	4
EClass	String	15
Temporary_Address	String	30
Permanent_Address	String	30
Mobile_No	String	20
Citizenship_No	String	20
License_No	String	20
Date_of_Birth	Date/Time	8
Date_Hired	Date/Time	8
Imagepic	String	100

*Total Size (per record) = 315 bytes*

*Total Records (approximately) = 1000*

Table: Mname

Field Name	Data Type	Size (in bytes)
Emp_ID	Integer	4
Emp_Name	String	40
Gender	String	6
Salary	Integer	10
EClass	String	10
Temporary_Address	String	30
Permanent_Address	String	30
Mobile_No	String	20
Citizenship_No	String	20
License_No	String	20
Date_of_Birth	String	8
Date_Hired	Date/Time	8
Imagepic	Date/Time	100

**Total Size (per record) = 306 bytes**

**Total Records (approximately) = 1000**

Table: Ctime

Field Name	Data Type	Size (in bytes)
AM	Byte	2
PM	Byte	2

**Total Size (per record) = 4 bytes**

**Total Records (approximately) = 1**

Table: Emp Workingdetails

Field Name	Data Type	Size (in bytes)
HMonth	Byte	2
HDay	Byte	2
HYear	Integer	4
Emp_ID	Integer	4
Emp_Name	String	40
Start Time	Byte	2
End Time	Byte	2
Expenses	Integer	6
Rate	Integer	4

*Total Size (per record) = 66 bytes*

*Total Records (approximately) = 1000*

Table: Others Expenses

Field Name	Data Type	Size (in bytes)
HYear	Integer	4
HMonth	Byte	2
HDay	Byte	2
Emp_ID	Integer	4
Emp_Name	String	40
EClass	String	10
Expenses	Integer	6

*Total Size (per record) = 68 bytes*

*Total Records (approximately) = 1000*

Table: Others\_Salary

Field Name	Data Type	Size (in bytes)
HYear	Integer	4
HMonth	Byte	2
Emp_ID	Byte	4
Emp_Name	String	40
EClass	String	10
Salary	Integer	6

**Total Size (per record) = 66 bytes**

**Total Records (approximately) = 1000**

Table: Userpass

Field Name	Data Type	Size (in bytes)
UserName	String	10
Password	String	10
Type	String	15

**Total Size (per record) = 35 bytes**

**Total Records (approximately) = 5**

**Total Size (of all records) =  $(100 * 1000 + 315 * 1000 + 306 * 1000 + 4 * 1 + 66 * 1000$**

**+  $68 * 1000 + 66 * 1000 + 35 * 5)$**

**= 921179 bytes**

**= 900 kilobytes (approximately)**

### **3. SOFTWARE DEVELOPMENT , TESTING AND IMPLEMENTATION**

#### ***Program and Database Specification***

The program is a desktop application developed using Microsoft visual studio 2008 using general development Environment using VB.NET programming language.

Microsoft Access database has been used to store data of the application. Tables have been used to store the data of the application and queries have been used to create reports. MS Access also provides database password protection feature.

The Visual Studio 2008 GDE has been used during the design, development, debugging and testing of the program.

## ***ALGORITHMS***

### **Adding new users**

1. Begin
2. Accept new users details
3. Add in the userpass table
4. End

### **Editing users**

1. Begin
2. Show all users
3. Select user to edit
4. Accept modified details of user
5. Update in the userpass table
6. End

### **Adding new employee**

1. Begin
2. Accept new employees details
3. Add in the cname/mname table
4. End

### **Editing Employee**

1. Begin
2. Show all employees
3. Select employee to edit
4. Accept modified details of an employee
5. Update in the cname/mname table
6. End

### **Viewing common working hours**

1. Begin

2. Show previously stored working hours
3. Are details modified?
  - Yes: Go to 4
  - No: Go to 5
4. Update in the userpass table
5. End

#### **Editing Monthly paid employee salary**

1. Begin
2. Show previously stored salary
3. Are details modified?
  - Yes: Go to 4
  - No: Go to 5
4. Update in the others\_salary table
5. End

#### **Viewing carpenter's working details**

1. Begin
2. Select employee's name and date
3. Is data present?
  - Yes: Go to 4
  - No: Display "Data is not present"
4. Show all employees working hours and income details
5. End

#### **Viewing Monthly paid employee income**

1. Begin
2. Select employee's name and date
3. Is data present?
  - Yes: Go to 4
  - No: Display "Data is not present"
4. Show all employees income details

5. End

**Viewing Monthly paid employee salary**

1. Begin
2. Select employee's name and date
3. Is data present?  
    Yes: Go to 4  
    No: Display "Data is not present"
4. Show all employees salary details
5. End

**Viewing Monthly paid employee expenses**

1. Begin
2. Select employee's name and date
3. Is data present?  
    Yes: Go to 4  
    No: Display "Data is not present"
4. Show all employees expenses details
5. End

**Viewing Monthly paid employee allowance**

1. Begin
2. Select employee's name and date
3. Is data present?  
    Yes: Go to 4  
    No: Display "Data is not present"
4. Show all employees allowance details
5. End

**Uploading carpenter's working details**

1. Begin

2. Select employee's name and date
3. Is data already present of that day?  
    Yes: Go to 4  
    No: add a row with names and date with zero expenses
4. Show Previously stored details
5. End

#### **Uploading Monthly paid employee salary**

1. Begin
2. Select employee's name and date
3. Is data already present of that Month?  
    Yes: Go to 4  
    No: add a row with names and date with salary from mname
4. Show Previously stored details
5. End

#### **Uploading Monthly paid employee expenses**

1. Begin
2. Select employee's name and date
3. Is data already present of that day?  
    Yes: Go to 4  
    No: add a row with names and date with zero expenses
4. Show Previously stored details
5. End

#### **Uploading Monthly paid employee allowance**

1. Begin
2. Accept allowance details
3. Add in the others\_allowance table
4. End

## **DATA LAYER**

### **Tables**

#### **Allowance**

**Snapshot of the table in Microsoft Office Access:**

Allowances		
	Field Name	Data Type
	FYear	Number
	FMonth	Number
	Emp_ID	Number
	Emp_Name	Text
	Allowance Title	Text
	Amount	Number

#### **Example:**

**Fyear = 2014**

**Fmonth = 3**

**Emp\_ID = 12**

**Emp\_name = Alexis Conron**

**Allowance title = Petrol**

**Amount = 1500**

### Cname

Snapshot of the table in Microsoft Office Access:

Field Name	Data Type	Description
Emp_ID	Number	Identification number for each carpenter
Carpenter_Name	Text	this field is for carpenter's name
Gender	Text	gender of carpenter is stored in this field
EType	Text	sub type of carpenter is stored in this field
ERate	Number	this is to store per hour rate of carpenter
EClass	Text	this is to store class of an employee
Temporary_Address	Text	this is to store temporary address of carpenter
Permanent_Address	Text	this is to store Permanent address of carpenter
Mobile_no	Text	this field is to store mobile number of carpenter
Citizenship_No	Text	this is to store citizenship no of carpenter
licence_No	Text	this is to store licence no of carpenter
Date_of_Birth	Date/Time	This field is for entering the Date of Birth of carpenter
Date_Hired	Date/Time	This field is for entering the hired Date of carpenter
Imagepic	Text	this is to store location of picture of carpenter

Example:

**Emp\_ID = 45**

**Carpenter\_name = Paul Walker**

**Gender = Male**

**Etype = Head**

**Erata = 100**

**Temporary\_Address = Thimi,nepal**

**Permanent\_Address = Sindhupalchowk,nepal**

**Mobile\_no = +977 -9841859784**

**Citizenship\_no = 00-4327-00897**

**Licence\_No = 88-77419824**

**Date\_of\_Birth = 1982/4/22**

**Date\_Hired = 2014/2/2**

**Imagepic = C:\Users\Public\Pictures\Sample Pictures\paul.jpg**

ctime

Snapshot of the table in Microsoft Office Access:

Field Name	Data Type	Description
AM	Number	common starting AM time is store in this field
PM	Number	common ending PM time is store in this field

Example:

Am = 7

Pm = 7

### Emp\_workingdetails

Snapshot of the table in Microsoft Office Access:

Field Name	Data Type	Description
HMonth	Number	month(number) is stored in this field
HDay	Number	Day(number) is stored in this field
HYear	Number	Year(number) is stored in this field
Emp_ID	Number	Employee' id (number) is stored in this field
Emp_Name	Text	employee's name is stores in this field
Start Time	Number	starting time of employee's work of that day is stored here
End Time	Number	End time of employee's work of that day is stored here
Expenses	Number	Expenses of employee of that day is stored here
Rate	Number	Rate of employee in that day is stored here

Example:

Hmonth = 3

Hday = 23

Hyear = 2014

Emp\_ID = 12

Emp\_name = Alexis Conron

Start Time = 7

End Time = 7

Expenses = 4500

Rate = 75

### mname

Snapshot of the table in Microsoft Office Access:

Field Name	Data Type	Description
Emp_ID	Number	Identification number for each employee
Emp_Name	Text	this field is for manager's name
Gender	Text	gender of employee is stored in this field
Salary	Number	this field is for storing salary of manager
Class	Text	this is class of an employee
Temporary_Address	Text	this is to store temporary address of manager
Permanent_Address	Text	this is to store Permanent address of manager
Mobile_no	Text	this field is to store mobile number of manager
Citizenship_No	Text	this is to store citizenship no of manager
licence_No	Text	this is to store licence no of manager
Date_of_Birth	Date/Time	This field is for entering the Date of Birth of manager
Date_Hired	Date/Time	This field is for entering the hired Date of manager
Imagepic	Text	this is to store location of picture of manager

Example:

**Emp\_ID = 45**

**Emp\_name = James Bond**

**Gender = Male**

**Salary = 25000**

**Class = Driver**

**Temporary\_Address = Thimi,nepal**

**Permanent\_Address = Sindhupalchowk,nepal**

**Mobile\_no = +977 9844569784**

**Citizenship\_no = 00-4456-32897**

**Licence\_No = 88-45619824**

**Date\_of\_Birth = 1972/2/24**

**Date\_Hired = 2014/3/1**

**Imagepic = C:\Users\Public\Pictures\Sample Pictures\james.jpg**

### others expenses

Snapshot of the table in Microsoft Office Access:

Others_Expenses			
	Field Name	Data Type	Description
	HYear	Number	Year(number) is stored in this field
	HMonth	Number	month(number) is stored in this field
	HDay	Number	Day(number) is stored in this field
	Emp_ID	Number	Employee' id (number) is stored in this field
	Emp_Name	Text	employee's name is stores in this field
	EClass	Text	class of employee is stored here
	Expenses	Number	Expenses of employee of that day is stored here

Example:

**Hyear = 2014**

**Hmonth = 3**

**Hday = 23**

**Emp\_ID = 12**

**Emp\_name = Alexis Conron**

**Eclass = Driver**

**Expenses = 4500**

### **Others\_salary**

**Snapshot of the table in Microsoft Office Access:**

Field Name	Data Type	Description
HYear	Number	Year(number) is stored in this field
HMonth	Number	month(number) is stored in this field
Emp_ID	Number	Employee' id (number) is stored in this field
Emp_Name	Text	employee's name is stores in this field
Class	Text	class of employee is stored here
Salary	Number	Salary of manager of that months is stored here

**Example:**

**Hyear = 2014**

**Hmonth = 3**

**Emp\_ID = 12**

**Emp\_name = Alexis Conron**

**Eclass = Driver**

**Expenses = 15000**

### Userpass

Snapshot of the table in Microsoft Office Access:

Field Name	Data Type	Description
UserName	Text	User name of users are stored here
Password	Text	password for the users are stored in this field
Type	Text	this is to store type of users

Example:

**Username = admin**

**Password = admin**

**Type = administrator**

## FORMS WITH CODES

### Form: Login



```
Public Class Log_In
    'declaring variable
    Public username As String
    Public password As String
    Public passwordfromdb As String
    Public usernamefromdb As String
    Public typefromdb As String
    Private Sub Button2_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Btnexit.Click
        Me.Close()
    End Sub
    Private Sub Log_In_Load(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles MyBase.Load
        username = txtusername.Text
        password = txtpassword.Text
        'getting usernames and password from database
        getusernameandpassword()
        'focus on username
        txtusername.Focus()
    End Sub
    Private Sub getusernameandpassword()
        'declaring connection
        Dim databaseconnection As New OleDb.OleDbConnection
```

```

        Dim database_provider As String =
"Provider=Microsoft.ace.oledb.12.0;" 
        'declaring data source
        Dim Datasource As String = "Data Source =" & CurDir() &
"\database.accdb"
        databaseconnection.ConnectionString = database_provider &
Datasource
        'opening connection to database
        databaseconnection.Open()
        Dim databasecommand As New OleDb.OleDbCommand
        Dim dbreader As OleDb.OleDbDataReader
        'sendind command to database
        databasecommand.CommandText = "SELECT * FROM Userpass WHERE
UserName = " & "****" & txtusername.Text & "****"
        databasecommand.Connection = databaseconnection
        dbreader = databasecommand.ExecuteReader
        'reading datas until end
        While dbreader.Read()
            usernamefromdb = dbreader.Item(0)
            passwordfromdb = dbreader.Item(1)
            typefromdb = dbreader.Item(2)
        End While
        'Closing connection to database
        databaseconnection.Close()
    End Sub
    'checking password that is correct or not from database
    Private Sub checkpassword()
        getusernameandpassword()
        'clearing errorprovider
        ErrorProvider1.Clear()
        If txtpassword.Text = passwordfromdb And txtusername.Text =
usernamefromdb Then
            Main.Show()
            If typefromdb = "Administrator" Then
                'showing tools only for administrator
                Main.AddOtherUsersToolStripMenuItem.Visible = True
                Main.ShowAllUsersToolStripMenuItem.Visible = True
            End If
            Me.Hide()
        ElseIf txtusername.Text <> usernamefromdb Then
            'show error
            ErrorProvider1.SetError(txtusername, "Sorry Username is
incorrect")
        ElseIf txtpassword.Text <> passwordfromdb Then
            'show error
            ErrorProvider1.SetError(txtpassword, "Sorry Password is
incorrect")
        End If
    End Sub

    Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnenter.Click
        'clearing errorprovider
        ErrorProvider1.Clear()
        If txtusername.Text = "" Then
            'show error
            ErrorProvider1.SetError(txtusername, "Please Enter User Name")
            'focus txtusername
            txtusername.Focus()
        ElseIf txtpassword.Text = "" Then

```

```

        'show error
        ErrorProvider1.SetError(txtpassword, "Please Enter Password")
        'focus txtpassword
        txtpassword.Focus()
    Else
        'clearing errorprovider
        ErrorProvider1.Clear()
        checkpassword()
    End If
End Sub

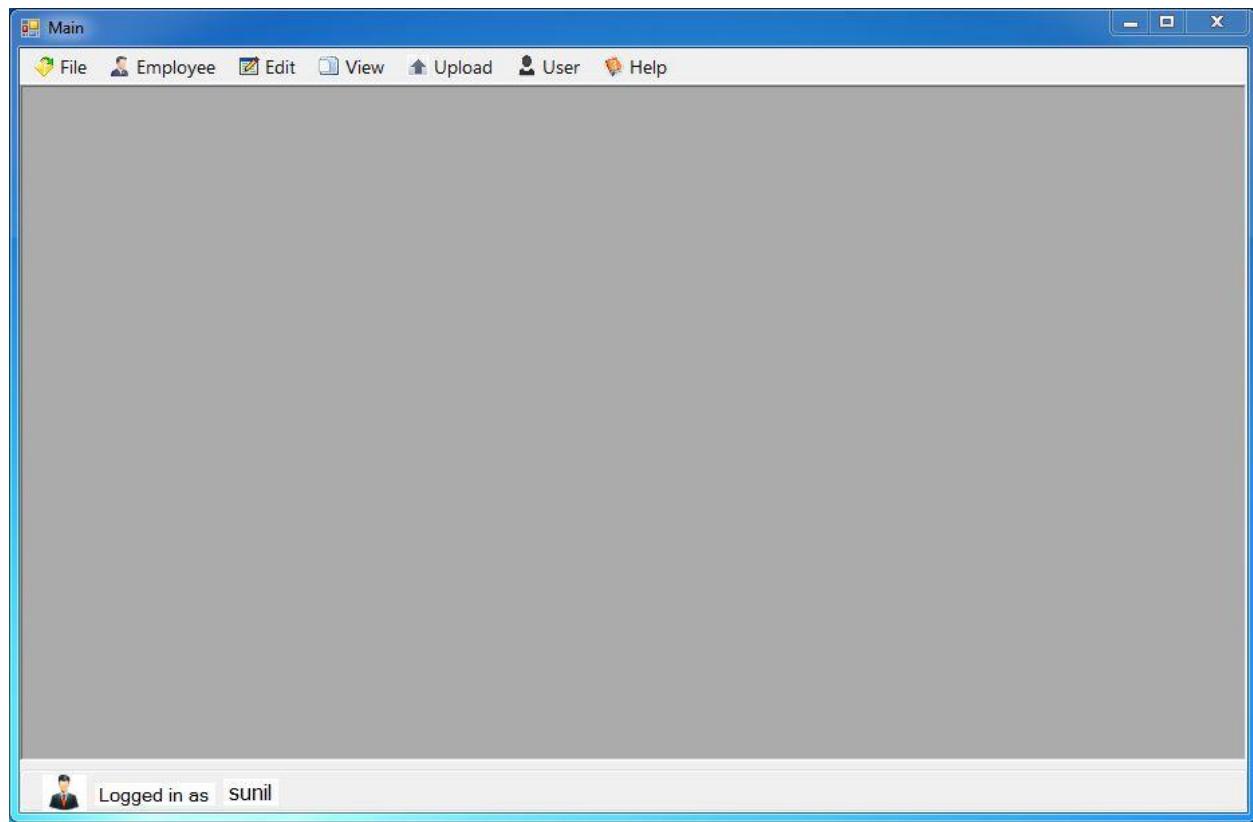
Private Sub txtusername_KeyPress(ByVal sender As Object, ByVal e As
System.Windows.Forms.KeyPressEventArgs) Handles txtusername.KeyPress
    Select Case Asc(e.KeyChar)
        Case 8, 32, 65 To 90, 97 To 122
            e.Handled = False
        Case Else
            e.Handled = True
    End Select
End Sub

Private Sub txtusername_TextChanged(ByVal sender As System.Object,
ByVal e As System.EventArgs) Handles txtusername.TextChanged
    txtusername.MaxLength = 10
End Sub

Private Sub txtpassword_TextChanged(ByVal sender As System.Object,
ByVal e As System.EventArgs) Handles txtpassword.TextChanged
    txtpassword.MaxLength = 10
End Sub
End Class

```

### **Form: Main**



```
Public Class Main
    Public stype As String
    Public dte As String
    Private Sub ExitToolStripMenuItem_Click(ByVal sender As System.Object,
    ByVal e As System.EventArgs) Handles ExitToolStripMenuItem.Click
        'closing forms
        Me.Close()
        Log_In.Close()
    End Sub
    Private Sub AddEmployeeToolStripMenuItem_Click(ByVal sender As
System.Object, ByVal e As System.EventArgs)
        'showing another form
        frmadd.ShowDialog()
    End Sub
    Private Sub CommonWorkingHoursToolStripMenuItem_Click(ByVal sender As
System.Object, ByVal e As System.EventArgs)
        'showing another form
        Common_working_hour.Show()
    End Sub
    Private Sub EmployeeToolStripMenuItem1_Click(ByVal sender As
System.Object, ByVal e As System.EventArgs)
        'showing another form
        Frmemp_Edit_Delete.Show()
    End Sub
    Private Sub Main_FormClosing(ByVal sender As Object, ByVal e As
System.Windows.Forms.FormClosingEventArgs) Handles Me.FormClosing
        'closing form
        Log_In.Close()
    End Sub
```

```

    Private Sub Main_Load(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles MyBase.Load
        'loading this form in its maximum size
        Me.WindowState = FormWindowState.Maximized
        'hiding menus
        AddOtherUsersToolStripMenuItem.Visible = False
        ShowAllUsersToolStripMenuItem.Visible = False
        If Log_In.txtusername.Text = "" Then
            PictureBox1.Hide()
            Label1.Hide()
            lblname.Hide()
        Else
            lblname.Text = Log_In.txtusername.Text
        End If
    End Sub
    Private Sub WorkingHoursToolStripMenuItem_Click(ByVal sender As
System.Object, ByVal e As System.EventArgs)
        If frmWorking_detail.Visible = True Then
            'bringing that form to front
            frmWorking_detail.BringToFront()
        Else
            frmWorking_detail.MdiParent = Me
            'showing another form
            frmWorking_detail.Show()
        End If
    End Sub
    Private Sub AddToolStripMenuItem_Click(ByVal sender As System.Object,
ByVal e As System.EventArgs) Handles AddToolStripMenuItem.Click
        'showing another form
        frmadd.Show()
    End Sub
    Private Sub DeleteToolStripMenuItem_Click(ByVal sender As
System.Object, ByVal e As System.EventArgs)
        'showing another form
        Frmemp_Edit_Delete.Show()
    End Sub
    Private Sub EditToolStripMenuItem1_Click(ByVal sender As System.Object,
ByVal e As System.EventArgs) Handles EditToolStripMenuItem1.Click
        If Frmemp_Edit_Delete.Visible = True Then
            'bringing that form to front
            Frmemp_Edit_Delete.BringToFront()
        Else
            Frmemp_Edit_Delete.MdiParent = Me
            'showing another form
            Frmemp_Edit_Delete.Show()
        End If
    End Sub
    Private Sub EmployeesIncomeToolStripMenuItem_Click(ByVal sender As
System.Object, ByVal e As System.EventArgs)
    End Sub
    Private Sub DatabaseToolStripMenuItem1_Click(ByVal sender As
System.Object, ByVal e As System.EventArgs)
        If frmShow_working_details.Visible = True Then
            'bringing that form to front
            frmShow_working_details.BringToFront()
        Else
            frmShow_working_details.MdiParent = Me
            'showing another form
            frmShow_working_details.Show()
        End If
    End Sub
End Sub

```

```

    Private Sub WorkingHoursToolStripMenuItem1_Click_1(ByVal sender As
System.Object, ByVal e As System.EventArgs)
        If frmShow_working_details.Visible = True Then
            'bringing that form to front
            frmShow_working_details.BringToFront()
        Else
            frmShow_working_details.MdiParent = Me
            'showing another form
            frmShow_working_details.Show()
        End If
    End Sub
    Private Sub ChangePasswordToolStripMenuItem_Click(ByVal sender As
System.Object, ByVal e As System.EventArgs)
        'showing another form
        Change_User_Password.Show()
    End Sub
    Private Sub AddOtherUsersToolStripMenuItem_Click(ByVal sender As
System.Object, ByVal e As System.EventArgs) Handles
AddOtherUsersToolStripMenuItem.Click
        'showing another form
        Add_Users.Show()
    End Sub
    Private Sub ChangePasswordToolStripMenuItem1_Click(ByVal sender As
System.Object, ByVal e As System.EventArgs) Handles
ChangePasswordToolStripMenuItem.Click
        'showing another form
        Change_User_Password.Show()
    End Sub
    Private Sub ShowAllUsersToolStripMenuItem_Click(ByVal sender As
System.Object, ByVal e As System.EventArgs) Handles
ShowAllUsersToolStripMenuItem.Click
        'showing another form
        Form_Edit_Users.Show()
    End Sub
    Private Sub EmployeesIncomeDetailsToolStripMenuItem_Click(ByVal sender
As System.Object, ByVal e As System.EventArgs) Handles
EmployeesIncomeDetailsToolStripMenuItem.Click
        If frmShow_working_details.Visible = True Then
            'bringing that form to front
            frmShow_working_details.BringToFront()
        Else
            frmShow_working_details.MdiParent = Me
            'showing another form
            frmShow_working_details.Show()
        End If
    End Sub
    Private Sub WorkingHoursToolStripMenuItem_Click_2(ByVal sender As
System.Object, ByVal e As System.EventArgs) Handles
WorkingHoursToolStripMenuItem.Click
        If frmWorking_detail.Visible = True Then
            'bringing that form to front
            frmWorking_detail.BringToFront()
        Else
            'showing inputbox
            dte = InputBox("Enter Date of that day you want to edit", ,
Today)
            frmWorking_detail.MdiParent = Me
            If dte <> "" Then
                'showing another form
                frmWorking_detail.Show()
                frmWorking_detail.WindowState = FormWindowState.Maximized

```

```

        End If
    End If
End Sub
Private Sub ExpensesToolStripMenuItem_Click(ByVal sender As
System.Object, ByVal e As System.EventArgs) Handles
ExpensesToolStripMenuItem.Click
    If Frm_Manager_expenses_upload.Visible = True Then
        'bringing that form to front
        Frm_Manager_expenses_upload.BringToFront()
    Else
        Frm_Manager_expenses_upload.MdiParent = Me
        'showing inputbox
        dte = InputBox("Enter Date of that day you want to Keep
Expenses of monthly paid employee", , Today)
        If dte = "" Then
        Else
            'showing another form
            Frm_Manager_expenses_upload.Show()
        End If
    End If
End Sub
Private Sub IncomeToolStripMenuItem_Click_1(ByVal sender As
System.Object, ByVal e As System.EventArgs) Handles
IncomeToolStripMenuItem.Click
    If Manager_s_Salary_details.Visible = True Then
        'bringing that form to front
        Manager_s_Salary_details.BringToFront()
    Else
        Manager_s_Salary_details.MdiParent = Me
        'showing another form
        Manager_s_Salary_details.Show()
    End If
End Sub
Private Sub AllowancesToolStripMenuItem_Click(ByVal sender As
System.Object, ByVal e As System.EventArgs) Handles
AllowancesToolStripMenuItem.Click
    'showing another form
    Frm_Emp_Allowances_upload.Show()
End Sub
Private Sub CommonWorkingHoursToolStripMenuItem_Click_1(ByVal sender As
System.Object, ByVal e As System.EventArgs) Handles
CommonWorkingHoursToolStripMenuItem.Click
    'showing another form
    Common_working_hour.Show()
End Sub
Private Sub AllowancesToolStripMenuItem1_Click(ByVal sender As
System.Object, ByVal e As System.EventArgs) Handles
AllowancesToolStripMenuItem1.Click
    'showing another form
    Frm_View_allowances.Show()
End Sub
Private Sub SalaryToolStripMenuItem1_Click(ByVal sender As
System.Object, ByVal e As System.EventArgs) Handles
SalaryToolStripMenuItem1.Click
    Frm_View_others_salary.Btnsavechanges.Visible = False
    Frm_View_others_salary.dghours.ReadOnly = True
    'showing another form
    Frm_View_others_salary.Show()
End Sub

```

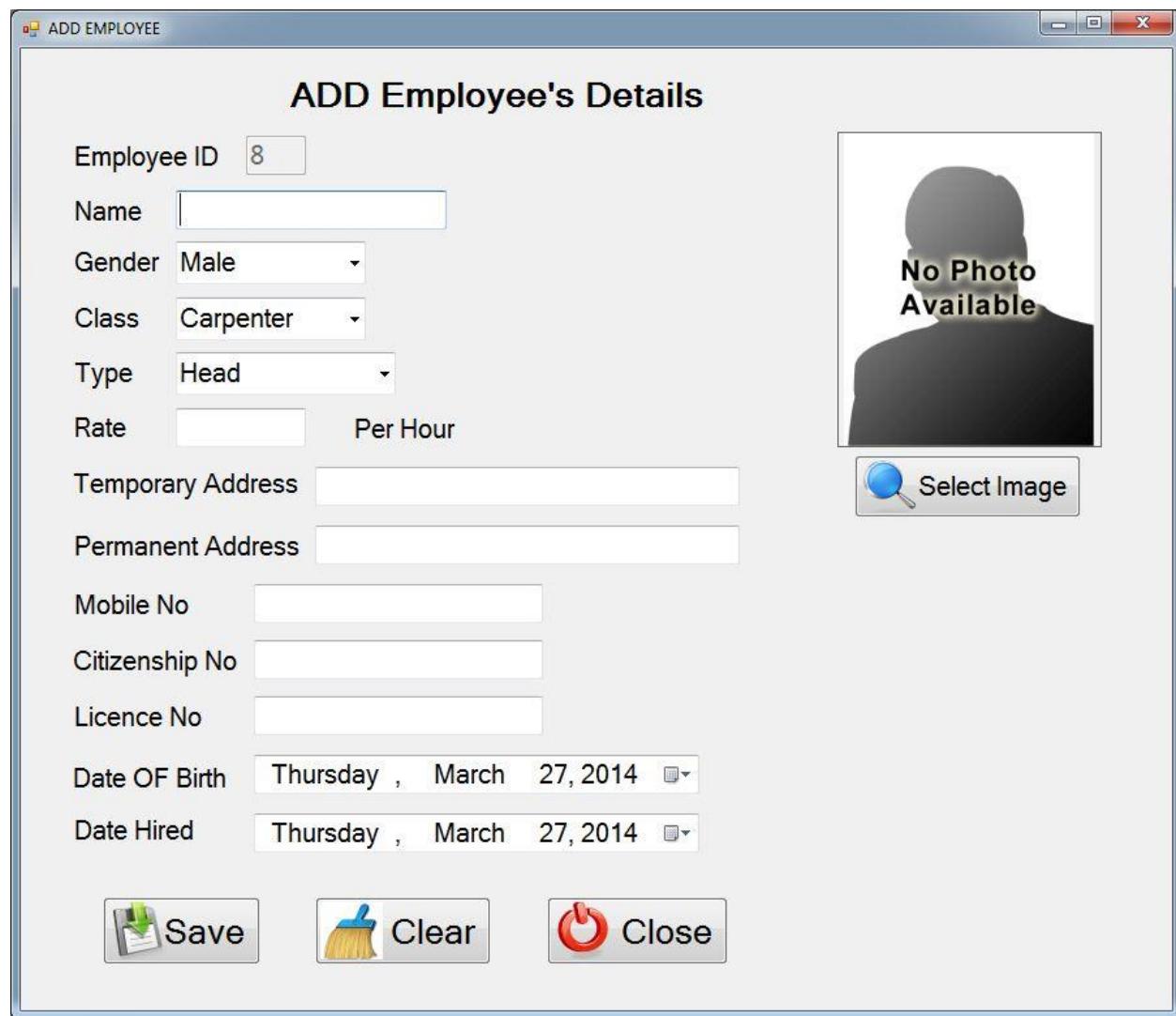
```

    Private Sub SalaryToolStripMenuItem2_Click(ByVal sender As
System.Object, ByVal e As System.EventArgs) Handles
SalaryToolStripMenuItem2.Click
    'showing another form
    Frm_View_others_salary.Show()
End Sub
Private Sub SalaryToolStripMenuItem_Click(ByVal sender As
System.Object, ByVal e As System.EventArgs) Handles
SalaryToolStripMenuItem.Click
    If Frm_others_salary_Upload.Visible = True Then
        'bringing that form to front
        Frm_others_salary_Upload.BringToFront()
    Else
        Frm_others_salary_Upload.MdiParent = Me
        'showing inputbox
        dte = InputBox("Enter Month and year of which you want to
upload salary of monthly paid employee. (Month/Year) (MM/YYYY)", ,
Today.Month & "/" & Today.Year)
        If dte <> "" Then
            'showing another form
            Frm_others_salary_Upload.Show()
        End If
    End If
End Sub
Private Sub ExpensesToolStripMenuItem1_Click(ByVal sender As
System.Object, ByVal e As System.EventArgs) Handles
ExpensesToolStripMenuItem1.Click
    'showing another form
    Show_Manager_expenses.Show()
End Sub
Private Sub PrintToolStripMenuItem_Click(ByVal sender As System.Object,
ByVal e As System.EventArgs)
    PrintForm1.PrintAction = Printing.PrintAction.PrintToPreview
    PrintForm1.Print()
End Sub

Private Sub HowDoIToolStripMenuItem_Click(ByVal sender As
System.Object, ByVal e As System.EventArgs) Handles
HowDoIToolStripMenuItem.Click
    HelpProvider1.HelpNamespace = CurDir() & "\Help.chm"
    Help.ShowHelp(Me, HelpProvider1.HelpNamespace)
End Sub
End Class

```

**Form: Frmadd**



```
Public Class frmadd
    Public cmax As Integer
    Public mmax As Integer
    Public empid As Integer
    Public Sub disabletxtbox()
        'disable all text boxes so that it cannot be edited
        txtename.Enabled = False
        txttempaddress.Enabled = False
        txtperaddress.Enabled = False
        txtcitizen.Enabled = False
        txtlno.Enabled = False
        txtmobileno.Enabled = False
        txtrate.Enabled = False
        txtsalary.Enabled = False
        DateTimePicker1.Enabled = False
        DateTimePicker2.Enabled = False
        cboclass.Enabled = False
        cbotype.Enabled = False
        cbogender.Enabled = False
        PictureBox1.Enabled = False
    End Sub
```

```

Private Sub clearall()
    'clearing all boxes and selecting default items
    txtename.Clear()
    txttempaddress.Clear()
    txtperaddress.Clear()
    txtcitizen.Clear()
    txtlno.Clear()
    txtmobileno.Clear()
    txtrate.Clear()
    txtsalary.Clear()
    DateTimePicker1.Value = Today
    DateTimePicker2.Value = Today
    cboclass.SelectedItem = "Carpenter"
    cbotype.SelectedIndex = 1
    cbogender.SelectedIndex = 0
    PictureBox1.Image = My.Resources._error
    PictureBox1.Image.Tag = My.Resources._error.Tag

End Sub
'return carpenter's maximum id so that next id would be greater than
maximum id by 1
Private Sub getcarpenter_id()
    'declaring connection
    Dim databaseconnection As New OleDb.OleDbConnection
    Dim database_provider As String =
"Provider=Microsoft.ace.oledb.12.0;"
    'declaring data source
    Dim Datasource As String = "Data Source =" & CurDir() &
"\database.accdb"
    databaseconnection.ConnectionString = database_provider &
Datasource
    'opening connection to database
    databaseconnection.Open()
    'declaring command to send to database
    Dim databasecommand As New OleDb.OleDbCommand
    Dim dbreader As OleDb.OleDbDataReader
    databasecommand.CommandText = "SELECT * FROM Cname"
    databasecommand.Connection = databaseconnection
    dbreader = databasecommand.ExecuteReader
    'initialising cmax as 0
    cmax = 0
    'reading datas until end
    While dbreader.Read
        If dbreader.Item(0) > cmax Then
            cmax = dbreader.Item(0)
        End If
    End While
    'increasing cmax by value 1
    cmax = cmax + 1
    'Closing connection to database
    databaseconnection.Close()
End Sub
'return managers's maximum id so that next id would be greater than
maximum id by 1
Private Sub getmanager_id()
    'declaring connection
    Dim databaseconnection As New OleDb.OleDbConnection
    Dim database_provider As String =
"Provider=Microsoft.ace.oledb.12.0;"
    'declaring data source

```

```

        Dim Datasource As String = "Data Source =" & CurDir() &
"\database.accdb"
        databaseconnection.ConnectionString = database_provider &
Datasource
        'opening connection to database
        databaseconnection.Open()
        Dim databasecommand As New OleDb.OleDbCommand
        Dim dbreader As OleDb.OleDbDataReader
        databasecommand.CommandText = "SELECT * FROM mname"
        databasecommand.Connection = databaseconnection
        dbreader = databasecommand.ExecuteReader
        'initialising cmax as 500
        mmax = 500
        'reading datas until end
        While dbreader.Read
            If dbreader.Item(0) > mmax Then
                mmax = dbreader.Item(0)
            End If
        End While
        'increasing mmax by value 1
        mmax = mmax + 1
        'Closing connection to database
        databaseconnection.Close()
    End Sub
    Private Sub frmadd_Load(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles MyBase.Load
        'focus is provided to Employee name
        txtename.Focus()
        'textbox of id is not enabled so that it can't be changed
        txtid.Enabled = False
        'Carpenter is selected by default
        cboclass.SelectedItem = "Carpenter"
        'Head is selected by default
        cbotype.SelectedIndex = 1
        'Male is selected by default
        cbogender.SelectedIndex = 0
        PictureBox1.Image = My.Resources._error
        PictureBox1.Tag = ""
        'datetimepicker1 and 2 has default value of today's date
        DateTimePicker1.Value = Today
        DateTimePicker2.Value = Today
    End Sub

    Private Sub btnclose_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnclose.Click
        'this form is closed
        Me.Close()
    End Sub
    'saving details of employees
    Public Sub saveemployee()
        If cboclass.SelectedItem = "Carpenter" Then
            'declaring connection
            Dim databaseconnection As New OleDb.OleDbConnection
            Dim database_provider As String =
"Provider=Microsoft.ace.oledb.12.0;"
            'declaring data source
            Dim Datasource As String = "Data Source =" & CurDir() &
"\database.accdb"
            databaseconnection.ConnectionString = database_provider &
Datasource
            'opening connection to database

```

```

        databaseconnection.Open()
        Dim databasecommand As New OleDb.OleDbCommand
        databasecommand.CommandText = "INSERT INTO cname
VALUES (?,?,?,?,?,?,?,?,?,?,?,?,?,?)"
        With databasecommand.Parameters
            .AddWithValue("pempid", txtid.Text)
            .AddWithValue("pname", txtename.Text)
            .AddWithValue("pgender", cbogender.SelectedItem)
            .AddWithValue("ptype", cbotype.SelectedItem)
            .AddWithValue("prate", txtrate.Text)
            .AddWithValue("pclass", "Carpenter")
            .AddWithValue("ptempaddress", txttempaddress.Text)
            .AddWithValue("pperaddress", txtperaddress.Text)
            .AddWithValue("pmobile_no", txtmobileno.Text)
            .AddWithValue("pcitizenship", txtcitizen.Text)
            .AddWithValue("plicence", txtlno.Text)
            .AddWithValue("pdob", DateTimePicker1)
            .AddWithValue("pdatehired", DateTimePicker2)
            .AddWithValue("pimage", PictureBox1.Tag)
        End With
        databasecommand.Connection = databaseconnection
        databasecommand.ExecuteNonQuery()
        databaseconnection.Close()
        'showing messagebox
        MsgBox(txtename.Text & " is added to carpenter")
        Frmemp_Edit_Delete.Refresh()
        'clearing all boxes and selecting default items
        clearall()
        'getting new carpenter id
        getcarpenter_id()
        txtid.Text = cmax
        'clearing error provider
        ErrorProvider1.Clear()
    Else
        Dim msgname As String
        'declaring connection
        Dim databaseconnection As New OleDb.OleDbConnection
        Dim database_provider As String =
"Provider=Microsoft.ace.oledb.12.0;"
        'declaring data source
        Dim Datasource As String = "Data Source =" & CurDir() &
"\database.accdb"
        databaseconnection.ConnectionString = database_provider &
Datasource
        'opening connection to database
        databaseconnection.Open()
        Dim databasecommand As New OleDb.OleDbCommand
        databasecommand.CommandText = "INSERT INTO mname
VALUES (?,?,?,?,?,?,?,?,?,?)"
        'adding values to commandtext
        With databasecommand.Parameters
            .AddWithValue("pempid", txtid.Text)
            .AddWithValue("pname", txtename.Text)
            .AddWithValue("pgender", cbogender.SelectedItem)
            .AddWithValue("psalary", txtsalary.Text)
            If cboclass.SelectedItem = "Manager" Then
                .AddWithValue("pclass", "Manager")
                msgname = "Manager"
            Else
                .AddWithValue("pclass", txtspclass.Text)
            End If
        End With
    End If
End Sub

```

```

        msgname = txtspclass.Text
    End If
    .AddWithValue("ptempaddress", txttempaddress.Text)
    .AddWithValue("pperaddress", txtperaddress.Text)
    .AddWithValue("pmobile_no", txtmobileno.Text)
    .AddWithValue("pcitizenship", txtcitizen.Text)
    .AddWithValue("plicence", txtlno.Text)
    .AddWithValue("pdob", DateTimePicker1)
    .AddWithValue("pdatehired", DateTimePicker2)
    .AddWithValue("pimage", PictureBox1.Tag)
End With
databasecommand.Connection = databaseconnection
databasecommand.ExecuteNonQuery()
databaseconnection.Close()
MsgBox(txtename.Text & " is added to " & msgname)
'get new id for manager
getmanager_id()
txtid.Text = mmax
'clearing all boxes and selecting default items
clearall()
'clearing error provider
ErrorProvider1.Clear()
End If
End Sub
Private Sub btnclick_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnclick.Click
    If txtename.Text = "" Then
        'show error in txtname
        ErrorProvider1.Clear()
        ErrorProvider1.SetError(txtename, "Name field cannot be empty")
        'set focus to txtname
        txtename.Focus()
    ElseIf txtrate.Visible = True And txtrate.Text = "" Then
        'show error in rate
        ErrorProvider1.Clear()
        ErrorProvider1.SetError(txtrate, "Rate field cannot be empty")
        'set focus to txtrate
        txtrate.Focus()
    ElseIf txtsalary.Visible = True And txtsalary.Text = "" Then
        'show error in txtsalary
        ErrorProvider1.Clear()
        ErrorProvider1.SetError(txtsalary, "Salary field cannot be empty")
        'set focus to txtsalary
        txtsalary.Focus()
    ElseIf DateTimePicker1.Value = Today Then
        'show error in datetimetype
        ErrorProvider1.Clear()
        ErrorProvider1.SetError(DateTimePicker1, "Date of birth cannot be Today")
    Else
        'save all details of employees
        saveemployee()
        'set focus to txtname
        txtename.Focus()
    End If
End Sub

```

```

    Private Sub cboclass_SelectedIndexChanged(ByVal sender As
System.Object, ByVal e As System.EventArgs) Handles
cboclass.SelectedIndexChanged
        If cboclass.SelectedItem = "Carpenter" Then
            'get carpenter's new id
            getcarpenter_id()
            txtid.Text = cmax
            lblsalary.Visible = False
            txtsalary.Visible = False
            lblpermonth.Visible = False
            lblrate.Visible = True
            lblperhour.Visible = True
            txtrate.Visible = True
            cbotype.Visible = True
            lbltype.Visible = True
            lblclass.Visible = False
            txtspclass.Visible = False
            txtsalary.Clear()
            txtspclass.Clear()
        ElseIf cboclass.SelectedItem = "Manager" Then
            'get managers's new id
            getmanager_id()
            txtid.Text = mmax
            'making labels, textboxes and combobox is made invisible
            lblrate.Visible = False
            lblperhour.Visible = False
            txtrate.Visible = False
            cbotype.Visible = False
            lbltype.Visible = False
            lblsalary.Visible = True
            txtsalary.Visible = True
            lblpermonth.Visible = True
            lblclass.Visible = False
            txtspclass.Visible = False
            txtrate.Clear()
            txtspclass.Clear()
        Else
            'get managers's new id
            getmanager_id()
            txtid.Text = mmax
            'making labels, textboxes and combobox is made invisible
            lblrate.Visible = False
            lblperhour.Visible = False
            txtrate.Visible = False
            cbotype.Visible = False
            lbltype.Visible = False
            lblsalary.Visible = True
            txtsalary.Visible = True
            lblpermonth.Visible = True
            lblclass.Visible = True
            txtspclass.Visible = True
            txtrate.Clear()
        End If
    End Sub

    Private Sub btnselectimage_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnselectimage.Click
        'filtering files with limited extensions
        OpenFileDialog1.Filter = "image file (*.jpg, *.bmp, *.png) | *.jpg;
*.bmp; *.png| all files (*.*) | *.*"
    End Sub

```

```

        If OpenFileDialog1.ShowDialog <> Windows.Forms.DialogResult.Cancel
    Then
        'seting image location
        PictureBox1.ImageLocation = OpenFileDialog1.FileName
        'providing tag to picturebox
        PictureBox1.Tag = OpenFileDialog1.FileName
    End If
End Sub

Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnclear.Click
    'clearing all boxes and selecting default items
    clearall()
    'clearing errorprovider
    ErrorProvider1.Clear()
    'getting focus to ename
    txtename.Focus()
End Sub

Private Sub txtename_KeyPress(ByVal sender As Object, ByVal e As
System.Windows.Forms.KeyPressEventArgs) Handles txtename.KeyPress
    Select Case Asc(e.KeyChar)
        Case 8, 32, 65 To 90, 97 To 122
            e.Handled = False
        Case Else
            e.Handled = True
    End Select
End Sub

Private Sub txtename_TextChanged(ByVal sender As System.Object, ByVal e
As System.EventArgs) Handles txtename.TextChanged
    txtename.MaxLength = 40
End Sub

Private Sub txtspclass_KeyPress(ByVal sender As Object, ByVal e As
System.Windows.Forms.KeyPressEventArgs) Handles txtspclass.KeyPress
    Select Case Asc(e.KeyChar)
        Case 8, 32, 65 To 90, 97 To 122
            e.Handled = False
        Case Else
            e.Handled = True
    End Select
End Sub

Private Sub txtspclass_TextChanged(ByVal sender As System.Object, ByVal
e As System.EventArgs) Handles txtspclass.TextChanged
    txtspclass.MaxLength = 10
End Sub

Private Sub txtsalary_KeyPress(ByVal sender As Object, ByVal e As
System.Windows.Forms.KeyPressEventArgs) Handles txtsalary.KeyPress
    If Asc(e.KeyChar) <> 8 Then
        If Asc(e.KeyChar) < 48 Or Asc(e.KeyChar) > 57 Then
            e.Handled = True
        End If
    End If
End Sub

Private Sub txtsalary_TextChanged(ByVal sender As System.Object, ByVal
e As System.EventArgs) Handles txtsalary.TextChanged
    txtsalary.MaxLength = 10

```

```

End Sub

Private Sub txtrate_KeyPress(ByVal sender As Object, ByVal e As
System.Windows.Forms.KeyPressEventArgs) Handles txtrate.KeyPress
    If Asc(e.KeyChar) <> 8 Then
        If Asc(e.KeyChar) < 48 Or Asc(e.KeyChar) > 57 Then
            e.Handled = True
        End If
    End If
End Sub

Private Sub txtrate_TextChanged(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles txtrate.TextChanged
    txtrate.MaxLength = 4
End Sub

Private Sub txttempaddress_KeyPress(ByVal sender As Object, ByVal e As
System.Windows.Forms.KeyPressEventArgs) Handles txttempaddress.KeyPress
    Select Case Asc(e.KeyChar)
        Case 8, 32, 44, 65 To 90, 97 To 122
            e.Handled = False
        Case Else
            e.Handled = True
    End Select
End Sub

Private Sub txttempaddress_TextChanged(ByVal sender As System.Object,
ByVal e As System.EventArgs) Handles txttempaddress.TextChanged
    txttempaddress.MaxLength = 30
End Sub

Private Sub txtperaddress_KeyPress(ByVal sender As Object, ByVal e As
System.Windows.Forms.KeyPressEventArgs) Handles txtperaddress.KeyPress
    Select Case Asc(e.KeyChar)
        Case 8, 32, 44, 65 To 90, 97 To 122
            e.Handled = False
        Case Else
            e.Handled = True
    End Select
End Sub

Private Sub txtperaddress_TextChanged(ByVal sender As System.Object,
ByVal e As System.EventArgs) Handles txtperaddress.TextChanged
    txtperaddress.MaxLength = 30
End Sub

Private Sub txtmobileno_KeyPress(ByVal sender As Object, ByVal e As
System.Windows.Forms.KeyPressEventArgs) Handles txtmobileno.KeyPress
    If Asc(e.KeyChar) <> 8 Then
        If Asc(e.KeyChar) < 48 Or Asc(e.KeyChar) > 57 Then
            e.Handled = True
        End If
    End If
End Sub

Private Sub txtmobileno_TextChanged(ByVal sender As System.Object,
ByVal e As System.EventArgs) Handles txtmobileno.TextChanged
    txtmobileno.MaxLength = 20
End Sub

```

```

    Private Sub txtcitizen_KeyPress(ByVal sender As Object, ByVal e As
System.Windows.Forms.KeyPressEventArgs) Handles txtcitizen.KeyPress
        If Asc(e.KeyChar) <> 8 Then
            If Asc(e.KeyChar) < 48 Or Asc(e.KeyChar) > 57 Then
                e.Handled = True
            End If
        End If
    End Sub

    Private Sub txtcitizen_TextChanged(ByVal sender As System.Object, ByVal
e As System.EventArgs) Handles txtcitizen.TextChanged
        txtcitizen.MaxLength = 20
    End Sub

    Private Sub txtlno_KeyPress(ByVal sender As Object, ByVal e As
System.Windows.Forms.KeyPressEventArgs) Handles txtlno.KeyPress
        If Asc(e.KeyChar) <> 8 Then
            If Asc(e.KeyChar) < 48 Or Asc(e.KeyChar) > 57 Then
                e.Handled = True
            End If
        End If
    End Sub

    Private Sub txtlno_TextChanged(ByVal sender As System.Object, ByVal e
As System.EventArgs) Handles txtlno.TextChanged
        txtlno.MaxLength = 20
    End Sub
End Class

    Private Sub cboclass_SelectedIndexChanged(ByVal sender As
System.Object, ByVal e As System.EventArgs) Handles cboclass.SelectedIndexChanged
        If cboclass.SelectedItem = "Carpenter" Then
            'get carpenter's new id
            getcarpenter_id()
            txtid.Text = cmax
            lblsalary.Visible = False
            txtsalary.Visible = False
            lblpermonth.Visible = False
            lblrate.Visible = True
            lblperhour.Visible = True
            txtrate.Visible = True
            cbotype.Visible = True
            lbltype.Visible = True
            lblclass.Visible = False
            txtspclass.Visible = False
            txtsalary.Clear()
            txtspclass.Clear()
        ElseIf cboclass.SelectedItem = "Manager" Then
            'get managers's new id
            getmanager_id()
            txtid.Text = mmax
            'making labels, textboxes and combobox is made invisible
            lblrate.Visible = False
            lblperhour.Visible = False
            txtrate.Visible = False
            cbotype.Visible = False
            lbltype.Visible = False
            lblsalary.Visible = True
            txtsalary.Visible = True
            lblpermonth.Visible = True
            lblclass.Visible = False
        End If
    End Sub

```

```

        txtspclass.Visible = False
        txtrate.Clear()
        txtspclass.Clear()
    Else
        'get managers's new id
        getmanager_id()
        txtid.Text = mmax
        'making labels, textboxes and combobox is made invisible
        lblrate.Visible = False
        lblperhour.Visible = False
        txtrate.Visible = False
        cbotype.Visible = False
        lbltype.Visible = False
        lblsalary.Visible = True
        txtsalary.Visible = True
        lblpermonth.Visible = True
        lblclass.Visible = True
        txtspclass.Visible = True
        txtrate.Clear()
    End If
End Sub
Private Sub btnselectimage_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnselectimage.Click
    'filtering files with limited extensions
    OpenFileDialog1.Filter = "image file (*.jpg, *.bmp, *.png) | *.jpg; *.bmp; *.png| all files (*.*) | *.*"
    If OpenFileDialog1.ShowDialog <> Windows.Forms.DialogResult.Cancel Then
        'seting image location
        PictureBox1.ImageLocation = OpenFileDialog1.FileName
        'providing tag to picturebox
        PictureBox1.Tag = OpenFileDialog1.FileName
    End If
End Sub

Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnclear.Click
    'clearing all boxes and selecting default items
    clearall()
    'clearing errorprovider
    ErrorProvider1.Clear()
    'getting focus to ename
    txtename.Focus()
End Sub

Private Sub txtename_TextChanged(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles txtename.TextChanged
    txtename.MaxLength = 40
End Sub

Private Sub txtspclass_TextChanged(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles txtspclass.TextChanged
    txtspclass.MaxLength = 10
End Sub

Private Sub txtsalary_KeyPress(ByVal sender As Object, ByVal e As System.Windows.Forms.KeyPressEventArgs) Handles txtsalary.KeyPress
    If Asc(e.KeyChar) <> 8 Then
        If Asc(e.KeyChar) < 48 Or Asc(e.KeyChar) > 57 Then
            e.Handled = True
        End If
    End If

```

```

        End If
    End Sub

    Private Sub txtsalary_TextChanged(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles txtsalary.TextChanged
        txtsalary.MaxLength = 10
    End Sub

    Private Sub txtrate_KeyPress(ByVal sender As Object, ByVal e As System.Windows.Forms.KeyPressEventArgs) Handles txtrate.KeyPress
        If Asc(e.KeyChar) <> 8 Then
            If Asc(e.KeyChar) < 48 Or Asc(e.KeyChar) > 57 Then
                e.Handled = True
            End If
        End If
    End Sub

    Private Sub txtrate_TextChanged(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles txtrate.TextChanged
        txtrate.MaxLength = 4
    End Sub

    Private Sub txttempaddress_TextChanged(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles txttempaddress.TextChanged
        txttempaddress.MaxLength = 30
    End Sub

    Private Sub txtperaddress_TextChanged(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles txtperaddress.TextChanged
        txtperaddress.MaxLength = 30
    End Sub

    Private Sub txtmobileno_KeyPress(ByVal sender As Object, ByVal e As System.Windows.Forms.KeyPressEventArgs) Handles txtmobileno.KeyPress
        If Asc(e.KeyChar) <> 8 Then
            If Asc(e.KeyChar) < 48 Or Asc(e.KeyChar) > 57 Then
                e.Handled = True
            End If
        End If
    End Sub

    Private Sub txtmobileno_TextChanged(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles txtmobileno.TextChanged
        txtmobileno.MaxLength = 20
    End Sub

    Private Sub txtcitizen_KeyPress(ByVal sender As Object, ByVal e As System.Windows.Forms.KeyPressEventArgs) Handles txtcitizen.KeyPress
        If Asc(e.KeyChar) <> 8 Then
            If Asc(e.KeyChar) < 48 Or Asc(e.KeyChar) > 57 Then
                e.Handled = True
            End If
        End If
    End Sub

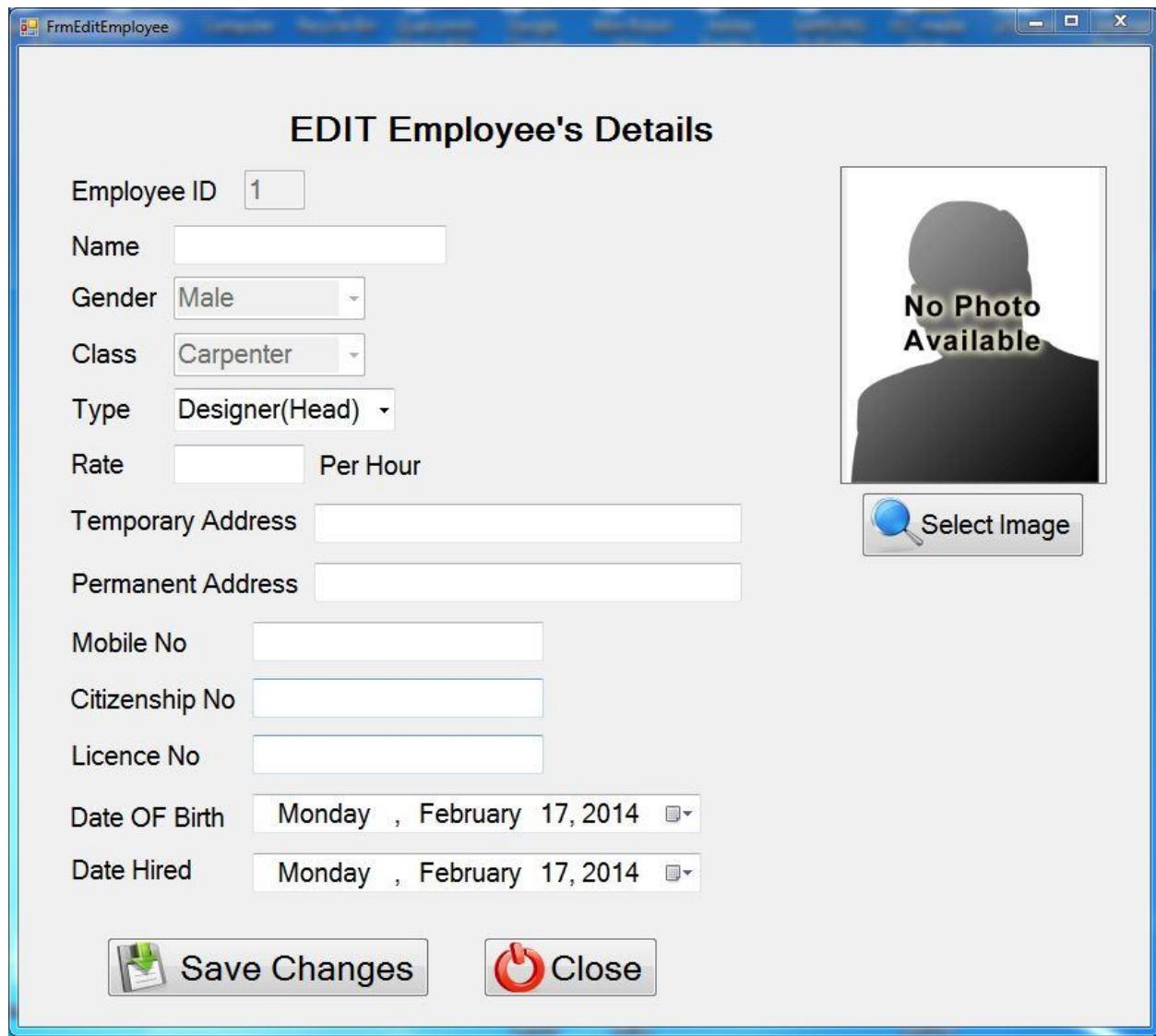
    Private Sub txtcitizen_TextChanged(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles txtcitizen.TextChanged
        txtcitizen.MaxLength = 20
    End Sub

```

```
Private Sub txtlno_KeyPress(ByVal sender As Object, ByVal e As System.Windows.Forms.KeyPressEventArgs) Handles txtlno.KeyPress
    If Asc(e.KeyChar) <> 8 Then
        If Asc(e.KeyChar) < 48 Or Asc(e.KeyChar) > 57 Then
            e.Handled = True
        End If
    End If
End Sub

Private Sub txtlno_TextChanged(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles txtlno.TextChanged
    txtlno.MaxLength = 20
End Sub
End Class
```

Form:FrmEditEmployee



```
Public Class FrmEditEmployee
    Private Sub btnsave_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnsavechanges.Click
        If txtename.Text = "" Then
            'show error in txtname
            ErrorProvider1.SetError(txtename, "Name field cannot be empty")
            txtename.Focus()
        ElseIf txtrate.Visible = True And txtrate.Text = "" Then
            'show error in rate
            ErrorProvider1.SetError(txtrate, "Rate field cannot be empty")
            txtrate.Focus()
        ElseIf txtsalary.Visible = True And txtsalary.Text = "" Then
            'show error in txtsalary
            ErrorProvider1.SetError(txtsalary, "Salary field cannot be
empty")
            txtsalary.Focus()
        Else
            'saving all changes
            save_employee_changes()
            txtename.Focus()
        End If
    End Sub
```

```

        End If
    End Sub
    Private Sub FrmEditEmployee_Load(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles MyBase.Load
        txtid.Enabled = False
        cbogender.Enabled = False
        cboclass.Enabled = False
        getlabelandtext()
        PictureBox1.Image = My.Resources._error
        PictureBox1.Tag = ""
    End Sub
    Private Sub btnclose_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnclose.Click
        Me.Close()
    End Sub
    'get visibility of textboxes and lalels
    Private Sub getlabelandtext()
        If Frmemp_Edit_Delete.cboclass.SelectedItem = "Carpenter" Then
            lblsalary.Visible = False
            txtsalary.Visible = False
            lblpermonth.Visible = False
            lblrate.Visible = True
            lblperhour.Visible = True
            txtrate.Visible = True
            cbotype.Visible = True
            lbltype.Visible = True
        Else
            lblrate.Visible = False
            lblperhour.Visible = False
            txtrate.Visible = False
            cbotype.Visible = False
            lbltype.Visible = False
            lblsalary.Visible = True
            txtsalary.Visible = True
            lblpermonth.Visible = True
        End If
    End Sub
    Private Sub save_employee_changes()
        If cboclass.SelectedItem = "Carpenter" Then
            'declaring connection
            Dim databaseconnection As New OleDb.OleDbConnection
            Dim database_provider As String =
"Provider=Microsoft.ace.oledb.12.0;" 
            'declaring data source
            Dim Datasource As String = "Data Source =" & CurDir() &
"\database.accdb"
            databaseconnection.ConnectionString = database_provider &
Datasource
            'opening connection to database
            databaseconnection.Open()
            Dim databasecommand As New OleDb.OleDbCommand
            databasecommand.CommandText = "UPDATE Cname SET" -
                & " Carpenter_Name=?,EType=?,ERate=?," -
                & "Temporary_Address=?,Permanent_Address=?,Mobile_no=?," -
                & "Citizenship_No=?,licence_No=?,Date_of_Birth?," -
                & "Date_Hired=?,Imagepic=? WHERE Emp_ID=?"
            With databasecommand.Parameters
                .AddWithValue("pname", txtename.Text)
                .AddWithValue("ptype", cbotype.Text)
                .AddWithValue("prate", txtrate.Text)
                .AddWithValue("ptempaddress", txttempaddress.Text)
            End With
        End If
    End Sub

```

```

        .AddWithValue("pperaddress", txtperaddress.Text)
        .AddWithValue("pmobile_no", txtmobileno.Text)
        .AddWithValue("pcitizenship", txtcitizen.Text)
        .AddWithValue("plicence", txtlno.Text)
        .AddWithValue("pdob", DateTimePicker1.Value)
        .AddWithValue("pdatehired", DateTimePicker2.Value)
        .AddWithValue("pimage", PictureBox1.Tag)
        .AddWithValue("pEmpid", txtid.Text)
    End With
    databasecommand.Connection = databaseconnection
    databasecommand.ExecuteNonQuery()
    'Closing connection to database
    databaseconnection.Close()
    MsgBox(txtename.Text & "'s details is changed")
Else
    'declaring connection
    Dim databaseconnection As New OleDb.OleDbConnection
    Dim database_provider As String =
"Provider=Microsoft.ace.oledb.12.0;" 
    'declaring data source
    Dim Datasource As String = "Data Source =" & CurDir() &
"\database.accdb"
    databaseconnection.ConnectionString = database_provider &
Datasource
    'opening connection to database
    databaseconnection.Open()
    Dim databasecommand As New OleDb.OleDbCommand
    databasecommand.CommandText = "UPDATE mname" _
        & " SET Manager_Name=? ,Salary=? ,Temporary_Address=? , " _
        & " Permanent_Address=? ,Mobile_no=? ,Citizenship_No=? , " _
        & " licence_No=? ,Date_of_Birth=? ,Date_Hired=? ,Imagepic=? "
    WHERE Emp_ID=?"
    With databasecommand.Parameters
        .AddWithValue("pname", txtename.Text)
        .AddWithValue("psalary", txtsalary.Text)
        .AddWithValue("ptempaddress", txttempaddress.Text)
        .AddWithValue("pperaddress", txtperaddress.Text)
        .AddWithValue("pmobile_no", txtmobileno.Text)
        .AddWithValue("pcitizenship", txtcitizen.Text)
        .AddWithValue("plicence", txtlno.Text)
        .AddWithValue("pdob", DateTimePicker1)
        .AddWithValue("pdatehired", DateTimePicker2)
        .AddWithValue("pimage", PictureBox1.Tag)
        .AddWithValue("pEmpid", txtid.Text)
    End With
    databasecommand.Connection = databaseconnection
    databasecommand.ExecuteNonQuery()
    'Closing connection to database
    databaseconnection.Close()
    MsgBox(txtename.Text & "'s details is changed")
End If
End Sub
Private Sub btnselectimage_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnselectimage.Click
    'filtering files with limited extensions
    OpenFileDialog1.Filter = "image file (*.jpg, *.bmp, *.png) | *.jpg; *.bmp; *.png| all files (*.*) | *.*"
    If OpenFileDialog1.ShowDialog <> Windows.Forms.DialogResult.Cancel Then
        'seting image location
        PictureBox1.ImageLocation = OpenFileDialog1.FileName
    End If
End Sub

```

```

        'providing tag to picturebox
        PictureBox1.Tag = OpenFileDialog1.FileName
    End If
End Sub
Private Sub txtename_KeyPress(ByVal sender As Object, ByVal e As System.Windows.Forms.KeyPressEventArgs) Handles txtename.KeyPress
    Select Case Asc(e.KeyChar)
        Case 8, 32, 65 To 90, 97 To 122
            e.Handled = False
        Case Else
            e.Handled = True
    End Select
End Sub

Private Sub txtename_TextChanged(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles txtename.TextChanged
    txtename.MaxLength = 40
End Sub
Private Sub txtsalary_KeyPress(ByVal sender As Object, ByVal e As System.Windows.Forms.KeyPressEventArgs) Handles txtsalary.KeyPress
    If Asc(e.KeyChar) <> 8 Then
        If Asc(e.KeyChar) < 48 Or Asc(e.KeyChar) > 57 Then
            e.Handled = True
        End If
    End If
End Sub
Private Sub txtsalary_TextChanged(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles txtsalary.TextChanged
    txtsalary.MaxLength = 10
End Sub
Private Sub txtrate_KeyPress(ByVal sender As Object, ByVal e As System.Windows.Forms.KeyPressEventArgs) Handles txtrate.KeyPress
    If Asc(e.KeyChar) <> 8 Then
        If Asc(e.KeyChar) < 48 Or Asc(e.KeyChar) > 57 Then
            e.Handled = True
        End If
    End If
End Sub
Private Sub txtrate_TextChanged(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles txtrate.TextChanged
    txtrate.MaxLength = 4
End Sub
Private Sub txttempaddress_KeyPress(ByVal sender As Object, ByVal e As System.Windows.Forms.KeyPressEventArgs) Handles txttempaddress.KeyPress
    Select Case Asc(e.KeyChar)
        Case 8, 32, 44, 65 To 90, 97 To 122
            e.Handled = False
        Case Else
            e.Handled = True
    End Select
End Sub
Private Sub txttempaddress_TextChanged(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles txttempaddress.TextChanged
    txttempaddress.MaxLength = 30
End Sub
Private Sub txtperaddress_KeyPress(ByVal sender As Object, ByVal e As System.Windows.Forms.KeyPressEventArgs) Handles txtperaddress.KeyPress
    Select Case Asc(e.KeyChar)
        Case 8, 32, 44, 65 To 90, 97 To 122
            e.Handled = False
        Case Else

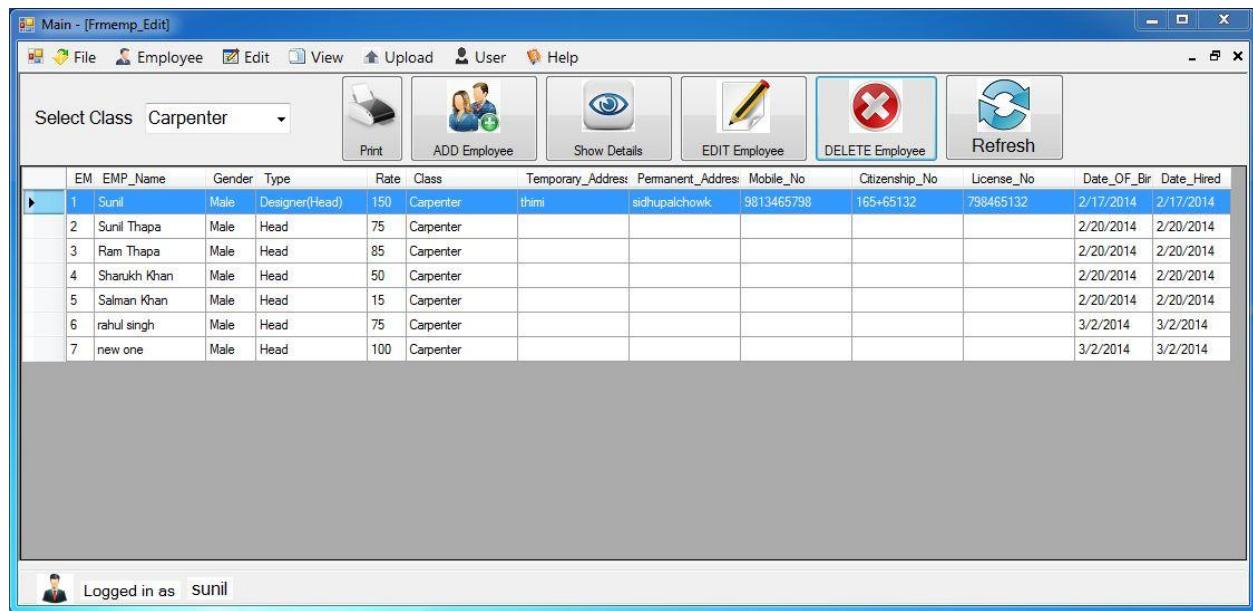
```

```

        e.Handled = True
    End Select
End Sub
Private Sub txtperaddress_TextChanged(ByVal sender As System.Object,
 ByVal e As System.EventArgs) Handles txtperaddress.TextChanged
    txtperaddress.MaxLength = 30
End Sub
Private Sub txtmobileno_KeyPress(ByVal sender As Object, ByVal e As
System.Windows.Forms.KeyPressEventArgs) Handles txtmobileno.KeyPress
    If Asc(e.KeyChar) <> 8 Then
        If Asc(e.KeyChar) < 48 Or Asc(e.KeyChar) > 57 Then
            e.Handled = True
        End If
    End If
End Sub
Private Sub txtmobileno_TextChanged(ByVal sender As System.Object,
 ByVal e As System.EventArgs) Handles txtmobileno.TextChanged
    txtmobileno.MaxLength = 20
End Sub
Private Sub txtcitizen_KeyPress(ByVal sender As Object, ByVal e As
System.Windows.Forms.KeyPressEventArgs) Handles txtcitizen.KeyPress
    If Asc(e.KeyChar) <> 8 Then
        If Asc(e.KeyChar) < 48 Or Asc(e.KeyChar) > 57 Then
            e.Handled = True
        End If
    End If
End Sub
Private Sub txtcitizen_TextChanged(ByVal sender As System.Object, ByVal
e As System.EventArgs) Handles txtcitizen.TextChanged
    txtcitizen.MaxLength = 20
End Sub
Private Sub txtlno_KeyPress(ByVal sender As Object, ByVal e As
System.Windows.Forms.KeyPressEventArgs) Handles txtlno.KeyPress
    If Asc(e.KeyChar) <> 8 Then
        If Asc(e.KeyChar) < 48 Or Asc(e.KeyChar) > 57 Then
            e.Handled = True
        End If
    End If
End Sub
Private Sub txtlno_TextChanged(ByVal sender As System.Object, ByVal e
As System.EventArgs) Handles txtlno.TextChanged
    txtlno.MaxLength = 20
End Sub
End Class

```

### Form: Frmemp\_Edit\_Delete



```

Imports DataGridViewPrinter.DataGridViewPrinter
Public Class Frmemp_Edit_Delete
    Public cname As String
    Public btnclick As String
    Dim MyDataGridViewPrinter As DataGridViewPrinter.DataGridViewPrinter

    Private Sub Frmemp_delete_Load(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles MyBase.Load
        cboclass.SelectedItem = "Carpenter"
        'showing this form in maximum size
        Me.WindowState = FormWindowState.Maximized
    End Sub
    Public databaseconnection As New OleDb.OleDbConnection
    'clearing rows and coloumns
    Public Sub clearall()
        DataGridView1.ClearSelection()
        DataGridView1.Columns.Clear()
        DataGridView1.Rows.Clear()
    End Sub
    'getting carpenter names from database
    Private Sub get_carpenter_names()
        'getting coloumns for carpenter
        getcarcoloumns()
        'declaring connection
        Dim databaseconnection As New OleDb.OleDbConnection
        Dim database_provider As String =
"Provider=Microsoft.ace.oledb.12.0;"
        'declaring data source
        Dim Datasource As String = "Data Source =" & CurDir() &
"\database.accdb"
        databaseconnection.ConnectionString = database_provider &
Datasource
        'opening connection to database
        databaseconnection.Open()
        Dim databasecommand As New OleDb.OleDbCommand
        Dim dbreader As OleDb.OleDbDataReader
        databasecommand.CommandText = "SELECT * FROM Cname"
    End Sub

```

```

databasecommand.Connection = databaseconnection
dbreader = databasecommand.ExecuteReader
'read database until end of file
While dbreader.Read
    Dim rowdata() As String = {dbreader.Item(0), dbreader.Item(1),
dbreader.Item(2), dbreader.Item(3), dbreader.Item(4), dbreader.Item(5),
dbreader.Item(6), dbreader.Item(7), dbreader.Item(8), dbreader.Item(9),
dbreader.Item(10), dbreader.Item(11), dbreader.Item(12)}
    'adding rows to datagridview
    DataGridView1.Rows.Add(rowdata)
End While
databaseconnection.Close()
End Sub
'getting managers names from database
Private Sub getmanagernames()
    'getting coloumns for manager
getmancoloumns()
    'declaring connection
    Dim databaseconnection As New OleDb.OleDbConnection
    Dim database_provider As String =
"Provider=Microsoft.ace.oledb.12.0;"
    'declaring data source
    Dim Datasource As String = "Data Source =" & CurDir() &
"\database.accdb"
    databaseconnection.ConnectionString = database_provider &
Datasource
    'opening connection to database
    databaseconnection.Open()
    Dim databasecommand As New OleDb.OleDbCommand
    Dim dbreader As OleDb.OleDbDataReader
    databasecommand.CommandText = "SELECT * FROM Mname"
    databasecommand.Connection = databaseconnection
    dbreader = databasecommand.ExecuteReader
    'read database until end of file
    While dbreader.Read
        Dim rowdata() As String = {dbreader.Item(0), dbreader.Item(1),
dbreader.Item(2), dbreader.Item(3), dbreader.Item(4), dbreader.Item(5),
dbreader.Item(6), dbreader.Item(7), dbreader.Item(8), dbreader.Item(9),
dbreader.Item(10), dbreader.Item(11)}
        'adding rows to datagridview
        DataGridView1.Rows.Add(rowdata)
    End While
    databaseconnection.Close()
End Sub
Private Sub cboclass_SelectedIndexChanged(ByVal sender As
System.Object, ByVal e As System.EventArgs) Handles
cboclass.SelectedIndexChanged
    If cboclass.SelectedItem = "Carpenter" Then
        'open carpenter's names'
        DataGridView1.Columns.Clear()
        get_carpenter_names()
        aligndatagridviewfor_car()
    Else
        'open managers's names'
        DataGridView1.Columns.Clear()
        getmanagernames()
        aligndatagridviewfor_man()
    End If
End Sub

```

```

    Private Sub DataGridView1_CellClick(ByVal sender As Object, ByVal e As
System.Windows.Forms.DataGridViewCellEventArgs) Handles
DataGridView1.CellClick
    DataGridView1.CurrentRow.Selected = True
    'getting id of that row which is clicked
    clname = DataGridView1.Item(0,
DataGridView1.CurrentRow.Index).Value.ToString
End Sub
'getting carpenter's columns to datagridview
Private Sub getcarcoloumns()
    DataGridView1.Columns.Add("Eid", "EMP_ID")
    DataGridView1.Columns.Add("pname", "EMP_Name")
    DataGridView1.Columns.Add("pgender", "Gender")
    DataGridView1.Columns.Add("ptype", "Type")
    DataGridView1.Columns.Add("prate", "Rate")
    DataGridView1.Columns.Add("pclass", "Class")
    DataGridView1.Columns.Add("ptempaddress", "Temporary_Address")
    DataGridView1.Columns.Add("pperaddress", "Permanent_Address")
    DataGridView1.Columns.Add("pmobile_no", "Mobile_No")
    DataGridView1.Columns.Add("pcitizenship", "Citizenship_No")
    DataGridView1.Columns.Add("plicence", "License_No")
    DataGridView1.Columns.Add("pdob", "Date_OF_Birth")
    DataGridView1.Columns.Add("pdatehired", "Date_Hired")
End Sub
'getting managers's columns to datagridview
Private Sub getmancoloumns()
    DataGridView1.Columns.Add("Eid", "EMP_ID")
    DataGridView1.Columns.Add("pname", "EMP_Name")
    DataGridView1.Columns.Add("pgender", "Gender")
    DataGridView1.Columns.Add("psalary", "Salary")
    DataGridView1.Columns.Add("pclass", "Class")
    DataGridView1.Columns.Add("ptempaddress", "Temporary_Address")
    DataGridView1.Columns.Add("pperaddress", "Permanent_Address")
    DataGridView1.Columns.Add("pmobile_no", "Mobile_No")
    DataGridView1.Columns.Add("pcitizenship", "Citizenship_No")
    DataGridView1.Columns.Add("plicence", "License_No")
    DataGridView1.Columns.Add("pdob", "Date_OF_Birth")
    DataGridView1.Columns.Add("pdatehired", "Date_Hired")
End Sub

Private Sub btndelete_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btndelete.Click
    If DataGridView1.Rows.Count = 0 Then Exit Sub
    If cboclass.SelectedItem = "Carpenter" Then
        'delete carpenter's names'
        If MessageBox.Show("Are you sure you want to delete this
employee", "WARNING", MessageBoxButtons.YesNo, MessageBoxIcon.Warning) =
Windows.Forms.DialogResult.Yes Then
            'getting id of that row which is clicked
            clname = DataGridView1.Item(0,
DataGridView1.CurrentRow.Index).Value.ToString
            'delete names'
            deletecarpenter()
        Else
            'Dont delete
        End If
    ElseIf cboclass.SelectedItem = "Others" Then
        'delete managers's names'

```

```

        If MessageBox.Show("Are you sure you want to delete this
employee", "WARNING", MessageBoxButtons.YesNo, MessageBoxIcon.Warning) =
Windows.Forms.DialogResult.Yes Then
    cname = DataGridView1.Item(0,
DataGridView1.CurrentRow.Index).Value.ToString
    'delete(names)
    deleteManager()
Else
    'Dont delete
End If
End If
End Sub
Private Sub deletecarpenter()
    'declaring connection
    Dim databaseconnection As New OleDb.OleDbConnection
    Dim database_provider As String =
"Provider=Microsoft.ace.oledb.12.0;"
    'declaring data source
    Dim Datasource As String = "Data Source =" & CurDir() &
"\database.accdb"
    databaseconnection.ConnectionString = database_provider &
Datasource
    'opening connection to database
    databaseconnection.Open()
    'Send delete command
    Dim databasecommand As New OleDb.OleDbCommand
    databasecommand.CommandText = "DELETE FROM cname WHERE Emp_ID = " &
cname
    databasecommand.Connection = databaseconnection
    databasecommand.ExecuteNonQuery()
    databaseconnection.Close()
    MsgBox("Employee having employee's id " & cname & " is deleted
from Carpenter")
    DataGridView1.Columns.Clear()
    get_carpenter_names()
End Sub
Private Sub deleteManager()
    'declaring connection
    Dim databaseconnection As New OleDb.OleDbConnection
    Dim database_provider As String =
"Provider=Microsoft.ace.oledb.12.0;"
    'declaring data source
    Dim Datasource As String = "Data Source =" & CurDir() &
"\database.accdb"
    databaseconnection.ConnectionString = database_provider &
Datasource
    'opening connection to database
    databaseconnection.Open()
    Dim databasecommand As New OleDb.OleDbCommand
    databasecommand.CommandText = "DELETE FROM mname WHERE Emp_ID = " &
cname
    databasecommand.Connection = databaseconnection
    databasecommand.ExecuteNonQuery()
    databaseconnection.Close()
    MsgBox("Employee having employee's id " & cname & " is deleted
from Others")
    DataGridView1.Columns.Clear()
    getmanagernames()
End Sub

```

```

    Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnadd.Click
        frmadd.Show()
    End Sub
    Private Sub btndedit_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnshowdetails.Click
        If DataGridView1.CurrentRow.Index = -1 Then Exit Sub
        Dim sid As String
        frmadd.lbltitle.Text = "Employee's Detail's"
        frmadd.Show()
        sid = DataGridView1.Item(0,
DataGridView1.CurrentRow.Index).Value.ToString
        frmadd.txtid.Text = sid
        frmadd.disabletextbox()
        frmadd.btnclear.Enabled = False
        frmadd.btnSave.Enabled = False
        frmadd.btnselectimage.Enabled = False
        frmadd.lblpermonth.Enabled = False
        frmadd.lblperhour.Enabled = False
        If cboclass.SelectedItem = "Carpenter" Then
            showcardetailsintxt(sid)
        Else
            showmandetailsintxt(sid)
        End If
    End Sub
    Public Sub crefresh()
        'if carpenter is selected then carpenters names should be refreshed
        If cboclass.SelectedItem = "Carpenter" Then
            'open carpenter's names'
            DataGridView1.Columns.Clear()
            get_carpenter_names()
            'DataGridView1.CurrentRow.Selected = True
        Else
            'open managers's names'
            DataGridView1.Columns.Clear()
            getmanagernames()
        End If
    End Sub
    Private Sub Button1_Click_1(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Btnrefresh.Click
        crefresh()
    End Sub

    Private Sub btneditemp_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btneditemp.Click
        If DataGridView1.CurrentRow.Index = -1 Then Exit Sub
        Dim sid As String
        sid = DataGridView1.Item(0,
DataGridView1.CurrentRow.Index).Value.ToString
        'frmadd.txtid.Text = sid
        If cboclass.SelectedItem = "Carpenter" Then
            Editcardetailsintxt(sid)
            'cboclass.SelectedItem = "Carpenter"
        Else
            Editmandetailsintxt(sid)
            'cboclass.SelectedItem = "Manager"
        End If
        FrmEditEmployee.Show()
    End Sub
    Private Sub showcardetailsintxt(ByVal sCarpenterID As String)
        'declaring connection

```

```

        Dim databaseconnection As New OleDb.OleDbConnection
        Dim database_provider As String =
"Provider=Microsoft.ace.oledb.12.0;" 
        'declaring data source
        Dim Datasource As String = "Data Source =" & CurDir() &
"\database.accdb"
        databaseconnection.ConnectionString = database_provider &
Datasource
        'opening connection to database
        databaseconnection.Open()
        Dim databasecommand As New OleDb.OleDbCommand
        Dim dbreader As OleDb.OleDbDataReader
        databasecommand.CommandText = "SELECT * FROM Cname WHERE Emp_ID = "
& sCarpenterID
        databasecommand.Connection = databaseconnection
        dbreader = databasecommand.ExecuteReader
        dbreader.Read()
        With frmadd
            .txtid.Text = dbreader.Item(0)
            .txtename.Text = dbreader.Item(1)
            .cbogender.Text = dbreader.Item(2)
            .cbotype.Text = dbreader.Item(3)
            .txtrate.Text = dbreader.Item(4)
            .cboclass.Text = dbreader.Item(5)
            .txttempaddress.Text = dbreader.Item(6)
            .txtperaddress.Text = dbreader.Item(7)
            .txtmobileno.Text = dbreader.Item(8)
            .txtcitizen.Text = dbreader.Item(9)
            .txtlno.Text = dbreader.Item(10)
            .DateTimePicker1.Text = dbreader.Item(11)
            .DateTimePicker2.Text = dbreader.Item(12)
            .PictureBox1.Tag = dbreader.Item(13)
            If dbreader.Item(13) = "" Then
                .PictureBox1.Image = My.Resources._error
            Else
                .PictureBox1.ImageLocation = .PictureBox1.Tag
            End If
        End With
        databaseconnection.Close()
    End Sub
    Public Sub showmandetailsintxt(ByVal sManagerID As String)
        'declaring connection
        Dim databaseconnection As New OleDb.OleDbConnection
        Dim database_provider As String =
"Provider=Microsoft.ace.oledb.12.0;" 
        'declaring data source
        Dim Datasource As String = "Data Source =" & CurDir() &
"\database.accdb"
        databaseconnection.ConnectionString = database_provider &
Datasource
        'opening connection to database
        databaseconnection.Open()
        Dim databasecommand As New OleDb.OleDbCommand
        Dim dbreader As OleDb.OleDbDataReader
        databasecommand.CommandText = "SELECT * FROM Mname WHERE Emp_ID = "
& sManagerID
        databasecommand.Connection = databaseconnection
        dbreader = databasecommand.ExecuteReader
        dbreader.Read()
        With frmadd
            .txtid.Text = dbreader.Item(0)

```

```

        .txtename.Text = dbreader.Item(1)
        .cbogender.Text = dbreader.Item(2)
        .txtsalary.Text = dbreader.Item(3)
        .cboclass.Text = dbreader.Item(4)
        .txttempaddress.Text = dbreader.Item(5)
        .txtperaddress.Text = dbreader.Item(6)
        .txtmobileno.Text = dbreader.Item(7)
        .txtcitizen.Text = dbreader.Item(8)
        .txtlno.Text = dbreader.Item(9)
        .DateTimePicker1.Text = dbreader.Item(10)
        .DateTimePicker2.Text = dbreader.Item(11)
        .PictureBox1.Tag = dbreader.Item(12)
        .PictureBox1.ImageLocation = .PictureBox1.Tag
    End With
    databaseconnection.Close()
    show_respective_labels_texts()
End Sub
Private Sub show_respective_labels_texts()
    If cboclass.SelectedItem = "Carpenter" Then
        With frmadd
            .lblsalary.Visible = False
            .txtsalary.Visible = False
            .lblpermonth.Visible = False
            .lblrate.Visible = True
            .lblperhour.Visible = True
            .txtrate.Visible = True
            .cbotype.Visible = True
            .lbltype.Visible = True
        End With
    Else
        With frmadd
            .lblrate.Visible = False
            .lblperhour.Visible = False
            .txtrate.Visible = False
            .cbotype.Visible = False
            .lbltype.Visible = False
            .lblsalary.Visible = True
            .txtsalary.Visible = True
            .lblpermonth.Visible = True
        End With
    End If
End Sub
Private Sub Editcardetailsintxt(ByVal sCarpenterID As String)
    'declaring connection
    Dim databaseconnection As New OleDb.OleDbConnection
    Dim database_provider As String =
"Provider=Microsoft.ace.oledb.12.0;"
    'declaring data source
    Dim Datasource As String = "Data Source =" & CurDir() &
"\database.accdb"
    databaseconnection.ConnectionString = database_provider &
Datasource
    'opening connection to database
    databaseconnection.Open()
    Dim databasecommand As New OleDb.OleDbCommand
    Dim dbreader As OleDb.OleDbDataReader
    databasecommand.CommandText = "SELECT * FROM Cname WHERE Emp_ID = "
& sCarpenterID
    databasecommand.Connection = databaseconnection
    dbreader = databasecommand.ExecuteReader
    dbreader.Read()

```

```

        With FrmEditEmployee
            .txtid.Text = dbreader.Item(0)
            .txtename.Text = dbreader.Item(1)
            .cbogender.Text = dbreader.Item(2)
            .cbotype.Text = dbreader.Item(3)
            .txtrate.Text = dbreader.Item(4)
            .cboclass.Text = dbreader.Item(5)
            .txttempaddress.Text = dbreader.Item(6)
            .txtperaddress.Text = dbreader.Item(7)
            .txtmobileno.Text = dbreader.Item(8)
            .txtcitizen.Text = dbreader.Item(9)
            .txtlno.Text = dbreader.Item(10)
            .DateTimePicker1.Text = dbreader.Item(11)
            .DateTimePicker2.Text = dbreader.Item(12)
            .PictureBox1.Tag = dbreader.Item(13)
            If dbreader.Item(13) = "" Then
                .PictureBox1.Image = My.Resources._error
            Else
                .PictureBox1.ImageLocation = .PictureBox1.Tag
            End If
        End With
        databaseconnection.Close()
    End Sub
    Public Sub Editmandetailsintxt(ByVal sManagerID As String)
        'declaring connection
        Dim databaseconnection As New OleDb.OleDbConnection
        Dim database_provider As String =
"Provider=Microsoft.ace.oledb.12.0;" 
        'declaring data source
        Dim Datasource As String = "Data Source =" & CurDir() &
"\database.accdb"
        databaseconnection.ConnectionString = database_provider &
Datasource
        'opening connection to database
        databaseconnection.Open()
        Dim databasecommand As New OleDb.OleDbCommand
        Dim dbreader As OleDb.OleDbDataReader
        databasecommand.CommandText = "SELECT * FROM Mname WHERE Emp_ID = "
& sManagerID
        databasecommand.Connection = databaseconnection
        dbreader = databasecommand.ExecuteReader
        dbreader.Read()
        With FrmEditEmployee
            .txtid.Text = dbreader.Item(0)
            .txtename.Text = dbreader.Item(1)
            .cbogender.Text = dbreader.Item(2)
            .txtsalary.Text = dbreader.Item(3)
            .cboclass.Text = dbreader.Item(4)
            .txttempaddress.Text = dbreader.Item(5)
            .txtperaddress.Text = dbreader.Item(6)
            .txtmobileno.Text = dbreader.Item(7)
            .txtcitizen.Text = dbreader.Item(8)
            .txtlno.Text = dbreader.Item(9)
            .DateTimePicker1.Text = dbreader.Item(10)
            .DateTimePicker2.Text = dbreader.Item(11)
            .PictureBox1.Tag = dbreader.Item(12)
            .PictureBox1.ImageLocation = .PictureBox1.Tag
        End With
        databaseconnection.Close()
    End Sub

```

```

    Private Sub btnprint_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnprint.Click
        If Not DataGridView1.Rows.Count > 0 Then Exit Sub
        PageSetupDialog1.PageSettings = New
System.Drawing.Printing.PageSettings
        PageSetupDialog1.PrinterSettings = New
System.Drawing.Printing.PrinterSettings
        PageSetupDialog1.ShowNetwork = True
        Dim result As DialogResult = PageSetupDialog1.ShowDialog()
        If result = Windows.Forms.DialogResult.Cancel Then Exit Sub
        Dim MyPrintDialog As New PrintDialog
        Dim MyPrintPreviewDialog As New PrintPreviewDialog
        PrintDocument1.DocumentName = Me.Text
        PrintDocument1.PrinterSettings = PageSetupDialog1.PrinterSettings
'MyPrintDialog.PrinterSettings
        PrintDocument1.DefaultPageSettings = PageSetupDialog1.PageSettings
        PrintDocument1.DefaultPageSettings.Margins =
PageSetupDialog1.PageSettings.Margins
        MyDataGridViewPrinter = New
DataGridPrinter.DataGridPrinter(DataGridView1, PrintDocument1, False, True,
Me.Text, New Font("Tahoma", 14, FontStyle.Bold, GraphicsUnit.Point),
Color.Black, True)
        MyPrintPreviewDialog.Document = PrintDocument1
        MyPrintPreviewDialog.ShowDialog()
    End Sub
    Private Sub DataGridView1_MouseDown(ByVal sender As Object, ByVal e As
System.Windows.Forms.MouseEventArgs) Handles DataGridView1.MouseDown
        If e.Button = Windows.Forms.MouseButtons.Right Then
            Dim ht As DataGridView.HitTestInfo
            ht = Me.DataGridView1.HitTest(e.X, e.Y)
            If ht.Type = DataGridViewHitTestType.Cell Then
                DataGridView1.ContextMenuStrip = ctxmenu
            Else
                End If
            End If
        End Sub

    Private Sub txitemrefresh_Click(ByVal sender As System.Object, ByVal e As
As System.EventArgs) Handles txitemrefresh.Click
        crefresh()
    End Sub
    Private Sub txshowdetails_Click(ByVal sender As System.Object, ByVal e As
As System.EventArgs) Handles txshowdetails.Click
        If DataGridView1.CurrentRow.Index = -1 Then Exit Sub
        Dim sid As String
        frmadd.lbltitle.Text = "Employee's Detail's"
        frmadd.Show()
        sid = DataGridView1.Item(0,
DataGridView1.CurrentRow.Index).Value.ToString
        frmadd.txtid.Text = sid
        frmadd.disabletxtbox()
        frmadd.btnclear.Enabled = False
        frmadd.btnSave.Enabled = False
        frmadd.btnselectimage.Enabled = False
        frmadd.lblpermonth.Enabled = False
        frmadd.lblperhour.Enabled = False
        If cboclass.SelectedItem = "Carpenter" Then
            showcardetailsintxt(sid)
        Else
            showmandetailsintxt(sid)
        End If

```

```

End Sub

Private Sub txeditdetails_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles txeditdetails.Click
    If DataGridView1.CurrentRow.Index = -1 Then Exit Sub
    Dim sid As String
    sid = DataGridView1.Item(0, DataGridView1.CurrentRow.Index).Value.ToString
    If cboclass.SelectedItem = "Carpenter" Then
        Editcardetailsintxt(sid)
    Else
        Editmandetailsintxt(sid)
    End If
    FrmEditEmployee.Show()
End Sub

Private Sub txdelete_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles txdelete.Click
    If DataGridView1.Rows.Count = 0 Then Exit Sub
    If cboclass.SelectedItem = "Carpenter" Then
        'delete carpenter's names'
        If MessageBox.Show("Are you sure you want to delete this employee", "WARNING", MessageBoxButtons.YesNo, MessageBoxIcon.Warning) = Windows.Forms.DialogResult.Yes Then

            clname = DataGridView1.Item(0, DataGridView1.CurrentRow.Index).Value.ToString
            'delete names'
            deletecarpenter()
        Else
            'Dont delete
        End If
    ElseIf cboclass.SelectedItem = "Others" Then
        'delete managers's names'
        If MessageBox.Show("Are you sure you want to delete this employee", "WARNING", MessageBoxButtons.YesNo, MessageBoxIcon.Warning) = Windows.Forms.DialogResult.Yes Then

            clname = DataGridView1.Item(0, DataGridView1.CurrentRow.Index).Value.ToString
            'delete names'
            deleteManager()
            'DataGridView1.Columns.Clear()
        Else
            'Dont delete
        End If
    End If
End Sub

Private Sub PrintDocument1_PrintPage(ByVal sender As System.Object, ByVal e As System.Drawing.Printing.PrintPageEventArgs) Handles PrintDocument1.PrintPage
    If Not DataGridView1.Rows.Count > 0 Then Exit Sub
    Dim more As Boolean
    more = MyDataGridViewPrinter.DrawDataGridView(e.Graphics)
    If more = True Then
        e.HasMorePages = True
    End If
End Sub
'aligning datagrid columns to the least width of column
Private Sub aligndatagridviewfor_car()

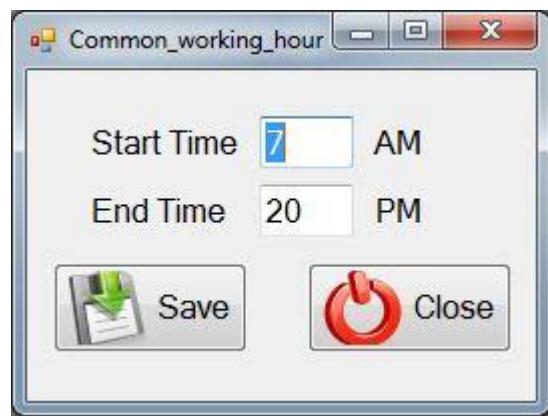
```

```

        DataGridView1.Columns(0).Width = 25
        DataGridView1.Columns(2).Width = 45
        DataGridView1.Columns(4).Width = 35
        DataGridView1.Columns(11).Width = 70
        DataGridView1.Columns(12).Width = 70
    End Sub
    'aligning datagrid columns to the least width of column
    Private Sub aligndatagridviewfor_man()
        DataGridView1.Columns(0).Width = 35
        'DataGridView1.Columns(0).AutoSizeMode =
        DataGridViewAutoSizeColumnsMode.DisplayedCells
        DataGridView1.Columns(1).AutoSizeMode =
        DataGridViewAutoSizeColumnsMode.DisplayedCells
        DataGridView1.Columns(2).AutoSizeMode =
        DataGridViewAutoSizeColumnsMode.DisplayedCells
        DataGridView1.Columns(3).AutoSizeMode =
        DataGridViewAutoSizeColumnsMode.DisplayedCells
        DataGridView1.Columns(4).AutoSizeMode =
        DataGridViewAutoSizeColumnsMode.DisplayedCells
        DataGridView1.Columns(5).AutoSizeMode =
        DataGridViewAutoSizeColumnsMode.DisplayedCells
        DataGridView1.Columns(6).AutoSizeMode =
        DataGridViewAutoSizeColumnsMode.DisplayedCells
        DataGridView1.Columns(7).AutoSizeMode =
        DataGridViewAutoSizeColumnsMode.DisplayedCells
        DataGridView1.Columns(8).AutoSizeMode =
        DataGridViewAutoSizeColumnsMode.DisplayedCells
        DataGridView1.Columns(9).AutoSizeMode =
        DataGridViewAutoSizeColumnsMode.DisplayedCells
        DataGridView1.Columns(10).AutoSizeMode =
        DataGridViewAutoSizeColumnsMode.DisplayedCells
        DataGridView1.Columns(11).AutoSizeMode =
        DataGridViewAutoSizeColumnsMode.DisplayedCells
    End Sub
End Class

```

Form: common working hour



```
Public Class Common_working_hour

    Private Sub Button2_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnclose.Click
        'closing this form
        Me.Close()
    End Sub

    Private Sub btnsave_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnsave.Click
        If txtam.Text = "" Then
            ErrorProvider1.SetError(txtam, "Time field cannot be empty")
        ElseIf txtpm.Text = "" Then
            ErrorProvider1.SetError(txtpm, "Time field cannot be empty")
        Else
            'delete work hours details
            deletecworkhours()
            'declaring connection
            Dim databaseconnection As New OleDb.OleDbConnection
            Dim database_provider As String =
"Provider=Microsoft.ace.oledb.12.0;"
            'declaring data source
            Dim Datasource As String = "Data Source =" & CurDir() &
"\database.accdb"
            databaseconnection.ConnectionString = database_provider &
Datasource
            'opening connection to database
            databaseconnection.Open()
            Dim databasecommand As New OleDb.OleDbCommand
            databasecommand.CommandText = "INSERT INTO ctime VALUES(?,?)"
            databasecommand.Parameters.AddWithValue("AM", txtam.Text)
            databasecommand.Parameters.AddWithValue("PM", txtpm.Text)
            databasecommand.Connection = databaseconnection
            databasecommand.ExecuteNonQuery()
            'Closing connection to database
            databaseconnection.Close()
            MsgBox("New common working hours is added ")
        End If
    End Sub
    'delete work hours
    Private Sub deletecworkhours()
```

```

    'declaring connection
    Dim databaseconnection As New OleDb.OleDbConnection
    Dim database_provider As String =
"Provider=Microsoft.ace.oledb.12.0;"
    'declaring data source
    Dim Datasource As String = "Data Source =" & CurDir() &
"\database.accdb"
    databaseconnection.ConnectionString = database_provider &
Datasource
    'opening connection to database
    databaseconnection.Open()
    Dim databasecommand As New OleDb.OleDbCommand
    databasecommand.CommandText = "DELETE * FROM ctime"
    databasecommand.Connection = databaseconnection
    databasecommand.ExecuteNonQuery()
    'Closing connection to database
    databaseconnection.Close()
End Sub

Private Sub Common_working_hour_Load(ByVal sender As System.Object,
 ByVal e As System.EventArgs) Handles MyBase.Load
    'loading common working hours
    loadcommonworkinghours()
End Sub
'loading common working hours
Private Sub loadcommonworkinghours()
    'declaring connection
    Dim databaseconnection As New OleDb.OleDbConnection
    Dim database_provider As String =
"Provider=Microsoft.ace.oledb.12.0;"
    'declaring data source
    Dim Datasource As String = "Data Source =" & CurDir() &
"\database.accdb"
    databaseconnection.ConnectionString = database_provider &
Datasource
    'opening connection to database
    databaseconnection.Open()
    Dim databasecommand As New OleDb.OleDbCommand
    Dim dbreader As OleDb.OleDbDataReader
    databasecommand.CommandText = "SELECT * FROM Ctime"
    databasecommand.Connection = databaseconnection
    dbreader = databasecommand.ExecuteReader
    dbreader.Read()
    txtam.Text = dbreader.Item(0)
    txtpm.Text = dbreader.Item(1)
    'Closing connection to database
    databaseconnection.Close()
End Sub

Private Sub txtam_KeyPress(ByVal sender As Object, ByVal e As
System.Windows.Forms.KeyPressEventArgs) Handles txtam.KeyPress
    If Asc(e.KeyChar) <> 8 Then
        If Asc(e.KeyChar) < 48 Or Asc(e.KeyChar) > 57 Then
            e.Handled = True
        End If
    End If
End Sub

Private Sub txtam_TextChanged(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles txtam.TextChanged
    txtam.MaxLength = 2

```

```
End Sub

Private Sub txtpm_KeyPress(ByVal sender As Object, ByVal e As
System.Windows.Forms.KeyPressEventArgs) Handles txtpm.KeyPress
    If Asc(e.KeyChar) <> 8 Then
        If Asc(e.KeyChar) < 48 Or Asc(e.KeyChar) > 57 Then
            e.Handled = True
        End If
    End If
End Sub

Private Sub txtpm_TextChanged(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles txtpm.TextChanged
    txtpm.MaxLength = 2
End Sub
End Class
```

**Form: FrmShow working details**

HMonth	HDay	HYear	Emp_Id	Employee_Name	Start Time	End Time	Expenses(RS)	Rate	Worked Hours	Income
3	2	2014	7	new one	7	20	0	100	13	1300
3	3	2014	7	new one	7	20	0	100	13	1300
3	5	2014	7	new one	7	20	0	100	13	1300
3	10	2014	7	new one	7	20	400	100	13	1300
					Total	400			52	5200
					Total Income	5200				
					Net Income	4800				

Total Hours Worked 52      Total Expenses 400  
 Total Income 5200      Net Income 4800

Print    Close

Logged in as sunil

```

Imports DataGridPrinter.DataGridPrinter
Public Class frmShow_working_details
    Public month As Integer
    Public day As Integer
    Public year As Integer
    Public rate As Integer
    Public totalincome As Integer
    Public totalexpenses As Integer
    Public netincome As Integer
    Public totalworkedhrs As Integer
    Dim MyDataGridViewPrinter As DataGridPrinter.DataGridPrinter
    Private Sub frmShow_working_details_Load(ByVal sender As System.Object,
    ByVal e As System.EventArgs) Handles MyBase.Load
        'opening this form in maximum size
        Me.WindowState = FormWindowState.Maximized
        lblmsg.Visible = False
        cboname.Items.Add("ALL")
        'adding years in combobox
        cboyear.Items.Add(Today.Year)
        cboyear.Items.Add(Today.Year - 1)
        cboyear.Items.Add(Today.Year - 2)
        'getting names only
        getnameonly()
        cboyear.SelectedIndex = 0
        cbomonths.SelectedIndex = Today.Month
        cboname.SelectedIndex = 0
    End Sub
    Private Sub btnPrint_Click(ByVal sender As System.Object,
    ByVal e As System.EventArgs) Handles btnPrint.Click
        MyDataGridViewPrinter.Print(Me.DataGridView1)
    End Sub
    Private Sub btnClose_Click(ByVal sender As System.Object,
    ByVal e As System.EventArgs) Handles btnClose.Click
        Me.Close()
    End Sub
    Private Sub getnameonly()
        'getting names from database
        'code for getting names
    End Sub
    Private Sub cboyear_SelectedIndexChanged(ByVal sender As System.Object,
    ByVal e As System.EventArgs) Handles cboyear.SelectedIndexChanged
        'code for selecting year
    End Sub
    Private Sub cbomonths_SelectedIndexChanged(ByVal sender As System.Object,
    ByVal e As System.EventArgs) Handles cbomonths.SelectedIndexChanged
        'code for selecting month
    End Sub
    Private Sub cboname_SelectedIndexChanged(ByVal sender As System.Object,
    ByVal e As System.EventArgs) Handles cboname.SelectedIndexChanged
        'code for selecting name
    End Sub
End Class

```

```

End Sub
'getting datas from database
Private Sub getdatasfromdatabase()
    dghours.Rows.Clear()
    'declaring connection
    Dim databaseconnection As New OleDb.OleDbConnection
    Dim database_provider As String =
"Provider=Microsoft.ace.oledb.12.0;"
    'declaring data source
    Dim Datasource As String = "Data Source =" & CurDir() &
"\database.accdb"
    databaseconnection.ConnectionString = database_provider &
Datasource
    'opening connection to database
    databaseconnection.Open()
    Dim name As String = cboname.SelectedItem
    Dim databasecommand As New OleDb.OleDbCommand
    Dim dbreader As OleDb.OleDbDataReader
    month = cbomonths.SelectedIndex
    If cboyear.SelectedIndex = 0 Then
        year = Today.Year
    ElseIf cboyear.SelectedIndex = 1 Then
        year = Today.Year - 1
    ElseIf cboyear.SelectedIndex = 2 Then
        year = Today.Year - 2
    End If
    If cbomonths.SelectedIndex <> 0 Then
        If cboname.SelectedIndex = 0 Then
            databasecommand.CommandText = "SELECT * FROM
Emp_workingdetails WHERE HMonth =" & month & " AND HYear =" & year
        Else
            databasecommand.CommandText = "SELECT * FROM
Emp_workingdetails WHERE Emp_Name = " & "****" & name & "****" & " AND HMonth
=" & month & " AND HYear =" & year
        End If
    Else
        If cboname.SelectedIndex = 0 Then
            databasecommand.CommandText = "SELECT * FROM
Emp_workingdetails WHERE HYear =" & year
        Else
            databasecommand.CommandText = "SELECT * FROM
Emp_workingdetails WHERE Emp_Name = " & "****" & name & "****" & " AND HYear =" & year
        End If
    End If
    databasecommand.Connection = databaseconnection
    dbreader = databasecommand.ExecuteReader
    'initializing to zero
    totalworkedhrs = 0
    totalincome = 0
    totalexpenses = 0
    netincome = 0
    'reads until end of datas
    While dbreader.Read
        rate = dbreader.Item(8)
        Dim rowdata() As String = {dbreader.Item(0), dbreader.Item(1),
dbreader.Item(2), dbreader.Item(3), dbreader.Item(4), dbreader.Item(5),
dbreader.Item(6), dbreader.Item(7), dbreader.Item(8), dbreader.Item(6) -
dbreader.Item(5), (dbreader.Item(6) - dbreader.Item(5)) * rate}
        'adding rows to datagridview
        dghours.Rows.Add(rowdata)
    End While
End Sub

```

```

        'calculating totals
        totalworkedhrs = totalworkedhrs + dbreader.Item(6) -
dbreader.Item(5)
        totalincome = totalincome + (dbreader.Item(6) -
dbreader.Item(5)) * rate
        totalexpenses = totalexpenses + dbreader.Item(7)
        netincome = totalincome - totalexpenses
    End While
    databaseconnection.Close()
    lblworkedhrs.Text = totalworkedhrs
    lbltotalincome.Text = totalincome
    lblexpenses.Text = totalexpenses
    lblnetincome.Text = netincome
    'adding empty rows
    dghours.Rows.Add()
    Dim rowdatatotal() As String = {"", "", "", "", "", "", "", "Total",
totalexpenses, "", totalworkedhrs, totalincome}
    'adding totals to datagridview
    dghours.Rows.Add(rowdatatotal)
    'adding empty rows
    dghours.Rows.Add()
    Dim rowdata2() As String = {"", "", "", "", "Total Income",
totalincome}
    'adding total income to datagridview
    dghours.Rows.Add(rowdata2)
    Dim rowdata3() As String = {"", "", "", "", "Net Income",
totalincome - totalexpenses}
    'adding net income to datagridview
    dghours.Rows.Add(rowdata3)
    'colouring row of datagridview
    dghours.Rows(dghours.Rows.Count - 1).DefaultCellStyle.BackColor =
Color.Khaki
    dghours.Rows(dghours.Rows.Count - 2).DefaultCellStyle.BackColor =
Color.Khaki
    dghours.Rows(dghours.Rows.Count - 4).DefaultCellStyle.BackColor =
Color.Khaki
    If netincome < 0 Then
        lblnetincome.ForeColor = Color.Red
    ElseIf netincome > 0 Then
        lblnetincome.ForeColor = Color.Green
    Else
        lblnetincome.ForeColor = Color.Black
    End If
    'aligning coloumns
    aligndatagridview()
End Sub

Private Sub getnameonly()
    'declaring connection
    Dim databaseconnection As New OleDb.OleDbConnection
    Dim database_provider As String =
"Provider=Microsoft.ace.oledb.12.0;"
    'declaring data source
    Dim Datasource As String = "Data Source =" & CurDir() &
"\database.accdb"
    databaseconnection.ConnectionString = database_provider &
Datasource
    'opening connection to database
    databaseconnection.Open()
    Dim databasecommand As New OleDb.OleDbCommand
    Dim dbreader As OleDb.OleDbDataReader

```

```

        databasecommand.CommandText = "SELECT * FROM Cname "
        databasecommand.Connection = databaseconnection
        dbreader = databasecommand.ExecuteReader
        'reads until end of datas
        While dbreader.Read
            cboname.Items.Add(dbreader.Item(1))
        End While
        databaseconnection.Close()
    End Sub
    Private Sub btnclose_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnclose.Click
        Me.Close()
    End Sub
    Private Sub cboname_SelectedIndexChanged(ByVal sender As System.Object,
ByVal e As System.EventArgs) Handles cboname.SelectedIndexChanged
        getdatasfromdatabase()
        If dghours.Rows.Count = 0 Then
            lblmsg.Visible = True
        Else
            lblmsg.Visible = False
        End If
    End Sub
    Private Sub cbomonths_SelectedIndexChanged(ByVal sender As
System.Object, ByVal e As System.EventArgs) Handles cbomonths.SelectedIndexChanged
        getdatasfromdatabase()
        If dghours.Rows.Count = 0 Then
            lblmsg.Visible = True
        Else
            lblmsg.Visible = False
        End If
    End Sub
    Private Sub cboyear_SelectedIndexChanged(ByVal sender As System.Object,
ByVal e As System.EventArgs) Handles cboyear.SelectedIndexChanged
        getdatasfromdatabase()
        If dghours.Rows.Count = 0 Then
            lblmsg.Visible = True
        Else
            lblmsg.Visible = False
        End If
    End Sub
    'aligning datagrid columns to the least width of column
    Private Sub aligndatagridview()
        dghours.Columns(0).AutoSizeMode =
DataGridViewAutoSizeColumnMode.DisplayedCells
        dghours.Columns(1).AutoSizeMode =
DataGridViewAutoSizeColumnMode.DisplayedCells
        dghours.Columns(2).AutoSizeMode =
DataGridViewAutoSizeColumnMode.DisplayedCells
        dghours.Columns(3).AutoSizeMode =
DataGridViewAutoSizeColumnMode.DisplayedCells
        dghours.Columns(4).AutoSizeMode =
DataGridViewAutoSizeColumnMode.DisplayedCells
        dghours.Columns(5).AutoSizeMode =
DataGridViewAutoSizeColumnMode.DisplayedCells
        dghours.Columns(6).AutoSizeMode =
DataGridViewAutoSizeColumnMode.DisplayedCells
        dghours.Columns(7).AutoSizeMode =
DataGridViewAutoSizeColumnMode.DisplayedCells
    End Sub

```

```

        dghours.Columns(8).AutoSizeMode =
DataGridViewAutoSizeColumnMode.DisplayedCells
        dghours.Columns(9).AutoSizeMode =
DataGridViewAutoSizeColumnMode.DisplayedCells
        dghours.Columns(10).AutoSizeMode =
DataGridViewAutoSizeColumnMode.DisplayedCells
    End Sub
    Private Sub btnprint_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnprint.Click
        If Not dghours.Rows.Count > 0 Then Exit Sub
        PageSetupDialog1.PageSettings = New
System.Drawing.Printing.PageSettings
        PageSetupDialog1.PrinterSettings = New
System.Drawing.Printing.PrinterSettings
        PageSetupDialog1.ShowNetwork = True
        Dim result As DialogResult = PageSetupDialog1.ShowDialog()
        If result = Windows.Forms.DialogResult.Cancel Then Exit Sub
        Dim MyPrintDialog As New PrintDialog
        Dim MyPrintPreviewDialog As New PrintPreviewDialog
        PrintDocument1.DocumentName = Me.Text
        PrintDocument1.PrinterSettings = PageSetupDialog1.PrinterSettings
'MyPrintDialog.PrinterSettings
        PrintDocument1.DefaultPageSettings = PageSetupDialog1.PageSettings
        PrintDocument1.DefaultPageSettings.Margins =
PageSetupDialog1.PageSettings.Margins
        MyDataGridViewPrinter = New
DataGridViewPrinter(dghours, PrintDocument1, False, True,
Me.Text, New Font("Tahoma", 14, FontStyle.Bold, GraphicsUnit.Point),
Color.Black, True)
        MyPrintPreviewDialog.Document = PrintDocument1
        MyPrintPreviewDialog.ShowDialog()
    End Sub
    Private Sub PrintDocument1_PrintPage(ByVal sender As System.Object,
ByVal e As System.Drawing.Printing.PrintPageEventArgs) Handles
PrintDocument1.PrintPage
        If Not dghours.Rows.Count > 0 Then Exit Sub
        Dim more As Boolean
        more = MyDataGridViewPrinter.DrawDataGridView(e.Graphics)
        If more = True Then
            e.HasMorePages = True
        End If
    End Sub
End Class

```

### Form: Managers salary details

	HYear	HMonth	Emp_Id	Employee_Name	EClass	Salary	Allowance	Expenses(RS)
▶	2014	3	501	Sunil Thapa	Manager	30000	1500	70000
	2014	3	503	damai	Driver	30000	0	0
	2014	3	504	ram	assistant	75000	0	0
	2014	3	505	amit	driver	15000	0	0
	2014	3	506	sachin tendulkar	spider man	50000	0	0
					Total	200000	1500	70000
					Total Income	201500		
					Net Income	131500		

Basic Salary    200000    Total Expenses    70000  
 Total Allowance    1500    Net Income    131500  
 Total Income    201500

Print    
 Close

Logged in as sunil

```

Imports DataGridViewPrinter.DataGridViewPrinter

Public Class Manager_s_Salary_details
    'declaring public variable
    Public month As Integer
    Public day As Integer
    Public year As Integer
    Public basicsalary As Integer
    Public totalsalary As Integer
    Public totalincome As Integer
    Public totalexpenses As Integer
    Public netincome As Integer
    Public totalallowance As Integer
    Public allowance As Integer
    Public expenses As Integer
    Public sendcommand As String = ""
    Dim MyDataGridViewPrinter As DataGridViewPrinter.DataGridViewPrinter
    Private Sub Manager_s_Salary_details_Load(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles MyBase.Load
        Me.WindowState = FormWindowState.Maximized
        cboname.Items.Add("All")
        'getting names only
        getnameonly()
        'filling combobox by years
        fillyears()
        cboyear.SelectedIndex = 0
    End Sub
End Class

```

```

        cboname.SelectedIndex = 0
        cbomonths.SelectedIndex = Today.Month
    End Sub
    'aligning width of columns of datagridview to its minimum size
    Private Sub aligndatagridview()
        dghours.Columns(0).AutoSizeMode =
DataGridViewAutoSizeColumnMode.DisplayedCells
        dghours.Columns(1).AutoSizeMode =
DataGridViewAutoSizeColumnMode.DisplayedCells
        dghours.Columns(2).AutoSizeMode =
DataGridViewAutoSizeColumnMode.DisplayedCells
        dghours.Columns(3).AutoSizeMode =
DataGridViewAutoSizeColumnMode.DisplayedCells
        dghours.Columns(4).AutoSizeMode =
DataGridViewAutoSizeColumnMode.DisplayedCells
        dghours.Columns(5).AutoSizeMode =
DataGridViewAutoSizeColumnMode.DisplayedCells
        dghours.Columns(6).AutoSizeMode =
DataGridViewAutoSizeColumnMode.DisplayedCells
        dghours.Columns(7).AutoSizeMode =
DataGridViewAutoSizeColumnMode.DisplayedCells
    End Sub
    'adding years to combobox
    Private Sub fillyears()
        For j = 0 To 4
            cboyear.Items.Add(Today.Year - j)
        Next
    End Sub
    'getting today's year
    Private Sub getyears()
        year = Today.Year - cboyear.SelectedIndex
    End Sub
    Private Sub btnclose_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnclose.Click
        Me.Close()
    End Sub
    Private Sub cboyear_SelectedIndexChanged(ByVal sender As System.Object,
ByVal e As System.EventArgs) Handles cboyear.SelectedIndexChanged
        If cbomonths.SelectedIndex = -1 Or cboyear.SelectedIndex = -1 Then
Exit Sub
        load_datas_from_database()
        If dghours.Rows.Count = 0 Then
            'show lblmsg
            lblmsg.Visible = True
        Else
            'hide lblmsg
            lblmsg.Visible = False
        End If
    End Sub

    Private Sub cbomonths_SelectedIndexChanged(ByVal sender As
System.Object, ByVal e As System.EventArgs) Handles cbomonths.SelectedIndexChanged
        If cbomonths.SelectedIndex = -1 Or cboyear.SelectedIndex = -1 Then
Exit Sub
        load_datas_from_database()
        If dghours.Rows.Count = 0 Then
            'show lblmsg
            lblmsg.Visible = True
        Else
            'hide lblmsg

```

```

        lblmsg.Visible = False
    End If
End Sub
Private Sub cboname_SelectedIndexChanged(ByVal sender As System.Object,
ByVal e As System.EventArgs) Handles cboname.SelectedIndexChanged
    If cbomonths.SelectedIndex = -1 Or cboyear.SelectedIndex = -1 Then
Exit Sub
    load_datas_from_database()
    If dghours.Rows.Count = 0 Then
        'show lblmsg
        lblmsg.Visible = True
    Else
        'hide lblmsg
        lblmsg.Visible = False
    End If
End Sub
Private Sub getnameonly()
    'declaring connection
    Dim databaseconnection As New OleDb.OleDbConnection
    Dim database_provider As String =
"Provider=Microsoft.ace.oledb.12.0;"
    'declaring data source
    Dim Datasource As String = "Data Source =" & CurDir() &
"\database.accdb"
    databaseconnection.ConnectionString = database_provider &
Datasource
    'opening connection to database
    databaseconnection.Open()
    Dim databasecommand As New OleDb.OleDbCommand
    Dim dbreader As OleDb.OleDbDataReader
    databasecommand.CommandText = "SELECT * FROM mname "
    databasecommand.Connection = databaseconnection
    dbreader = databasecommand.ExecuteReader
    'reads until end
    While dbreader.Read
        cboname.Items.Add(dbreader.Item(1))
    End While
    databaseconnection.Close()
End Sub
Private Sub getdatasfromdatabase()
    dghours.Rows.Clear()
    Dim empid As Integer
    Dim empname As String = ""
    Dim eclass As String = ""
    'declaring connection
    Dim databaseconnection As New OleDb.OleDbConnection
    Dim database_provider As String =
"Provider=Microsoft.ace.oledb.12.0;"
    'declaring data source
    Dim Datasource As String = "Data Source =" & CurDir() &
"\database.accdb"
    databaseconnection.ConnectionString = database_provider &
Datasource
    'opening connection to database
    databaseconnection.Open()
    Dim ename As String = cboname.SelectedItem
    Dim databasecommand As New OleDb.OleDbCommand
    Dim dbreader As OleDb.OleDbDataReader
    month = cbomonths.SelectedIndex
    getyears()
    If cbomonths.SelectedIndex <> 0 Then

```

```

        databasecommand.CommandText = "SELECT * FROM Others_Expenses
WHERE Emp_Name = " & """" & ename & """" & " AND HMonth =" & month & " AND
HYear =" & year
    Else
        databasecommand.CommandText = "SELECT * FROM Others_Expenses
WHERE Emp_Name = " & """" & ename & """" & " AND HYear =" & year
    End If
    databasecommand.Connection = databaseconnection
    dbreader = databasecommand.ExecuteReader
    Dim rresult As Boolean = False
    'initialising variables to 0
    totalincome = 0
    totalexpenses = 0
    totalallowance = 0
    netincome = 0
    'reads until end
    While dbreader.Read
        totalincome = basicsalary
        totalexpenses = totalexpenses + dbreader.Item(6)
        netincome = totalincome - totalexpenses
        empid = dbreader.Item(3)
        empname = dbreader.Item(4)
        eclass = dbreader.Item(5)
        rresult = True
    End While
    If rresult = True Then
        Dim rowdata() As String = {cboyear.SelectedItem,
cbomonths.SelectedIndex, empid, empname, eclass, totalexpenses,
basicsalary, totalallowance}
        'adding a row to datagridview
        dghours.Rows.Add(rowdata)
    End If
    databaseconnection.Close()
    'colour income
    getnetincomecolors()
End Sub

Private Sub getnetincomecolors()
    'filling with colours
    If netincome < 0 Then
        lblnetincome.ForeColor = Color.Red
    ElseIf netincome > 0 Then
        lblnetincome.ForeColor = Color.Green
    Else
        lblnetincome.ForeColor = Color.Black
    End If
End Sub

Private Sub getexpensesofthat_month(ByVal pname As String, ByVal pmonth
As Integer, ByVal pyear As Integer)
    'declaring connection
    Dim databaseconnection As New OleDb.OleDbConnection
    Dim database_provider As String =
"Provider=Microsoft.ace.oledb.12.0;"
    'declaring data source
    Dim Datasource As String = "Data Source =" & CurDir() &
"\database.accdb"
    databaseconnection.ConnectionString = database_provider &
Datasource
    'opening connection to database
    databaseconnection.Open()
    Dim databasecommand As New OleDb.OleDbCommand

```

```

        Dim dbreader As OleDb.OleDbDataReader
        databasecommand.CommandText = "SELECT * FROM Others_Expenses WHERE
Emp_Name = '" & pname & "' AND HMonth = " & pmonth & " AND HYear = " &
pyear
        databasecommand.Connection = databaseconnection
        dbreader = databasecommand.ExecuteReader
        expenses = 0
        'reads until end
        While dbreader.Read()
            expenses = expenses + dbreader.Item(6)
        End While
        databaseconnection.Close()
    End Sub
    Private Sub getallowanceofthat_month(ByVal gname As String, ByVal gyear
As Integer, ByVal gmonth As Integer)
        'declaring connection
        Dim databaseconnection As New OleDb.OleDbConnection
        Dim database_provider As String =
"Provider=Microsoft.ace.oledb.12.0;" 
        'declaring data source
        Dim Datasource As String = "Data Source =" & CurDir() &
"\database.accdb"
        databaseconnection.ConnectionString = database_provider &
Datasource
        'opening connection to database
        databaseconnection.Open()
        Dim databasecommand As New OleDb.OleDbCommand
        Dim dbreader As OleDb.OleDbDataReader
        databasecommand.CommandText = "SELECT * FROM Allowances WHERE
Emp_Name = '" & gname & "' AND Fyear = " & gyear & "And FMonth = " & gmonth
        databasecommand.Connection = databaseconnection
        dbreader = databasecommand.ExecuteReader
        allowance = 0
        'reads until end
        While dbreader.Read()
            allowance = allowance + dbreader.Item(5)
        End While
        databaseconnection.Close()
    End Sub
    Private Sub dghours_MouseDown(ByVal sender As Object, ByVal e As
System.Windows.Forms.MouseEventArgs) Handles dghours.MouseDown
        If e.Button = Windows.Forms.MouseButtons.Right Then
            Dim ht As DataGridView.HitTestInfo
            ht = Me.dghours.HitTest(e.X, e.Y)
            If ht.Type = DataGridViewHitTestType.Cell Then
                dghours.ContextMenuStrip = ctxmenu
                ctxmenu.Visible = True
            Else
            End If
        End If
    End Sub

    Private Sub txshowdetails_Click(ByVal sender As System.Object, ByVal e
As System.EventArgs) Handles txshowdetails.Click
        'sending sendcommand as open
        sendcommand = "Open"
        Show_Manager_expenses.Show()
    End Sub
    Private Sub btnprint_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnprint.Click
        If Not dghours.Rows.Count > 0 Then Exit Sub

```

```

        PageSetupDialog1.PageSettings = New
System.Drawing.Printing.PageSettings
        PageSetupDialog1.PrinterSettings = New
System.Drawing.Printing.PrinterSettings
        PageSetupDialog1.ShowNetwork = True
        Dim result As DialogResult = PageSetupDialog1.ShowDialog()
        If result = Windows.Forms.DialogResult.Cancel Then Exit Sub
        Dim MyPrintDialog As New PrintDialog
        Dim MyPrintPreviewDialog As New PrintPreviewDialog
        PrintDocument1.DocumentName = Me.Text
        PrintDocument1.PrinterSettings = PageSetupDialog1.PrinterSettings
'MyPrintDialog.PrinterSettings
        PrintDocument1.DefaultPageSettings = PageSetupDialog1.PageSettings
        PrintDocument1.DefaultPageSettings.Margins =
PageSetupDialog1.PageSettings.Margins
        MyDataGridViewPrinter = New
DataGridPrinter.DataGridPrinter(dghours, PrintDocument1, False, True,
Me.Text, New Font("Tahoma", 14, FontStyle.Bold, GraphicsUnit.Point),
Color.Black, True)
        MyPrintPreviewDialog.Document = PrintDocument1
        MyPrintPreviewDialog.ShowDialog()
    End Sub
    Private Sub load_datas_from_database()
        dghours.Rows.Clear()
        Dim empid As Integer
        Dim empname As String = ""
        Dim eclass As String = ""
        'declaring connection
        Dim databaseconnection As New OleDb.OleDbConnection
        Dim database_provider As String =
"Provider=Microsoft.ace.oledb.12.0;"
        'declaring data source
        Dim Datasource As String = "Data Source =" & CurDir() &
"\database.accdb"
        databaseconnection.ConnectionString = database_provider &
Datasource
        'opening connection to database
        databaseconnection.Open()
        Dim ename As String = cboname.SelectedItem
        Dim databasecommand As New OleDb.OleDbCommand
        Dim dbreader As OleDb.OleDbDataReader
        month = cbomonths.SelectedIndex
        getyears()
        If cbomonths.SelectedIndex = 0 Then
            If cboname.SelectedIndex = 0 Then
                databasecommand.CommandText = "SELECT * FROM Others_salary
WHERE HYear =" & year
            Else
                databasecommand.CommandText = "SELECT * FROM Others_salary
WHERE Emp_Name = '" & ename & "' AND HYear =" & year
            End If
        Else
            If cboname.SelectedIndex = 0 Then
                databasecommand.CommandText = "SELECT * FROM Others_salary
WHERE HMonth =" & month & " AND HYear =" & year
            Else
                databasecommand.CommandText = "SELECT * FROM Others_salary
WHERE Emp_Name = '" & ename & "' AND HMonth =" & month & " AND HYear
=" & year
            End If
        End If
    End Sub

```

```

databasecommand.Connection = databaseconnection
dbreader = databasecommand.ExecuteReader
Dim rresult As Boolean = False
'initialising variables to 0
totalsalary = 0
totalincome = 0
totalallowance = 0
totalexpenses = 0
While dbreader.Read
    Dim eyear As Integer
    Dim emonth As Integer
    eyear = dbreader.Item(0)
    emonth = dbreader.Item(1)
    empid = dbreader.Item(2)
    empname = dbreader.Item(3)
    eclass = dbreader.Item(4)
    getexpensesofthat_month(empname, emonth, eyear)
    getallowanceofthat_month(empname, eyear, emonth)
    Dim rowdata() As String = {dbreader.Item(0), dbreader.Item(1),
dbreader.Item(2), dbreader.Item(3), dbreader.Item(4), dbreader.Item(5),
allowance, expenses}
        'adding a row to datagridview
        dghours.Rows.Add(rowdata)
        'calculating totals
        totalsalary = totalsalary + dbreader.Item(5)
        totalallowance = totalallowance + allowance
        totalexpenses = totalexpenses + expenses
        totalincome = totalsalary + totalallowance
        rresult = True
End While
'adding an empty row
dghours.Rows.Add()
Dim rowdatatotal() As String = {"", "", "", "", "", "Total",
totalsalary, totalallowance, totalexpenses}
'adding a row with totals
dghours.Rows.Add(rowdatatotal)
'adding an empty row
dghours.Rows.Add()
Dim rowdata2() As String = {"", "", "", "", "", "Total Income",
totalincome}
'adding a row with total income
dghours.Rows.Add(rowdata2)
Dim rowdata3() As String = {"", "", "", "", "", "Net Income",
totalincome - totalexpenses}
'adding a row with net income
dghours.Rows.Add(rowdata3)
'colouring rows
dghours.Rows(dghours.Rows.Count - 1).DefaultCellStyle.BackColor =
Color.Khaki
dghours.Rows(dghours.Rows.Count - 2).DefaultCellStyle.BackColor =
Color.Khaki
dghours.Rows(dghours.Rows.Count - 4).DefaultCellStyle.BackColor =
Color.Khaki
databaseconnection.Close()
gettexts()
getnetincomecolors()
End Sub
'showing totals in labels
Private Sub gettexts()
lblbasicsalary.Text = totalsalary
lblallowance.Text = totalallowance

```

```

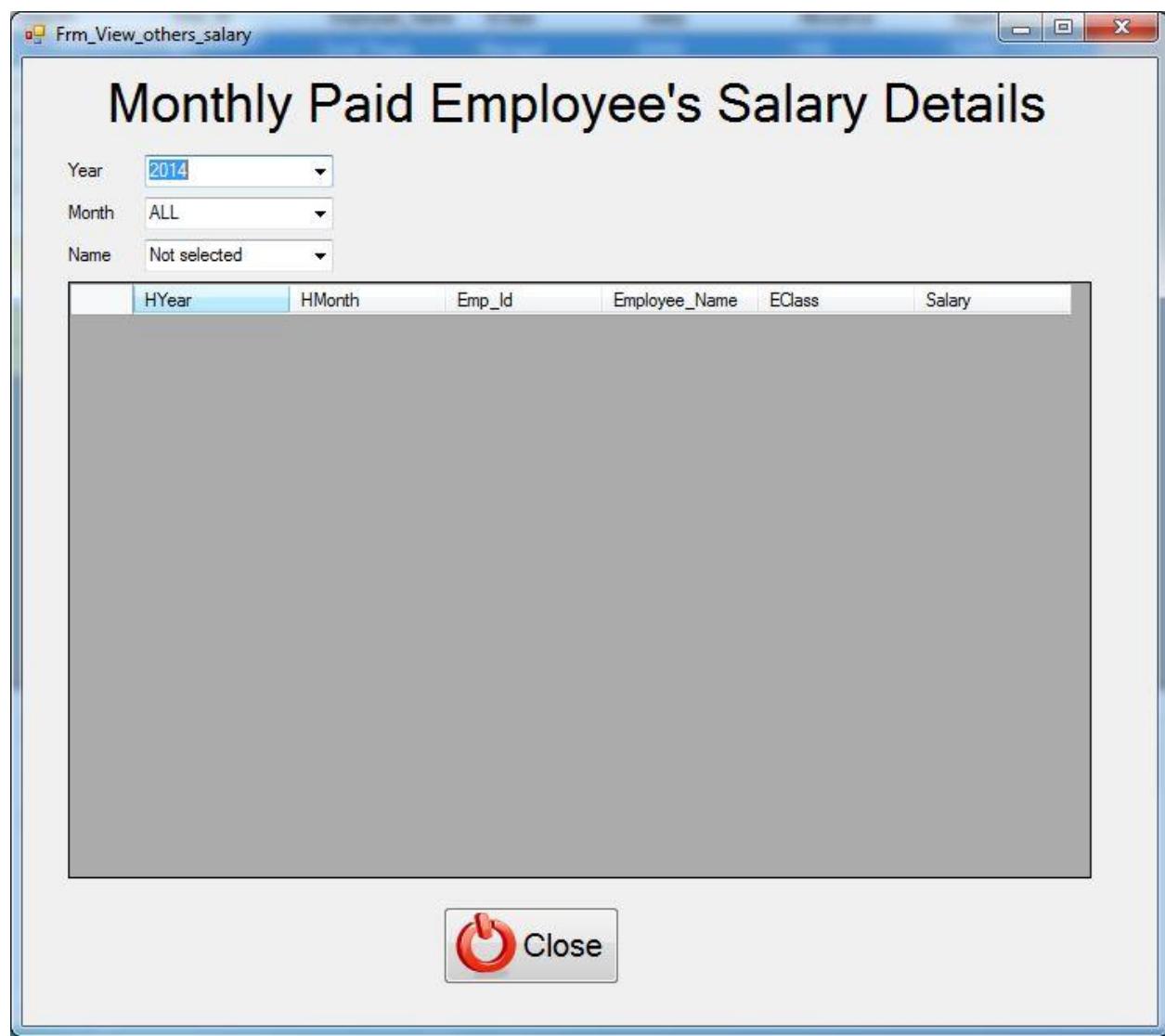
        lbltotalincome.Text = totalsalary + totalallowance
        lblexpenses.Text = totalexpenses
        netincome = totalsalary + totalallowance - totalexpenses
        lblnetincome.Text = netincome
    End Sub

    Private Sub ShowAllowanceDetailsToolStripMenuItem_Click(ByVal sender As
System.Object, ByVal e As System.EventArgs) Handles
ShowAllowanceDetailsToolStripMenuItem.Click
        'sending sendcommand = proceed
        sendcommand = "Proceed"
        Frm_View_allowances.Show()
    End Sub

    Private Sub PrintDocument1_PrintPage(ByVal sender As System.Object,
ByVal e As System.Drawing.Printing.PrintPageEventArgs) Handles
PrintDocument1.PrintPage
        If Not dghours.Rows.Count > 0 Then Exit Sub
        Dim more As Boolean
        more = MyDataGridViewPrinter.DrawDataGridView(e.Graphics)
        If more = True Then
            e.HasMorePages = True
        End If
    End Sub
End Class

```

**Form: Frm\_View\_others\_Salary**



```
Public Class Frm_View_others_salary
    Public found As Boolean
    Public oyear As Integer
    Public omonth As Integer
    Public oclass As String
    Public salary As Integer
    Public oname As String
    Public oid As Integer
    Private Sub checkifdataalreadypresent()
        dghours.Rows.Clear()
        'declaring connection
        Dim databaseconnection As New OleDb.OleDbConnection
        Dim database_provider As String =
"Provider=Microsoft.ace.oledb.12.0;"
        'declaring data source
        Dim Datasource As String = "Data Source =" & CurDir() &
"\database.accdb"
        databaseconnection.ConnectionString = database_provider &
Datasource
        'opening connection to database
```

```

        databaseconnection.Open()
        Dim databasecommand As New OleDb.OleDbCommand
        Dim dbreader As OleDb.OleDbDataReader
        If cbomonths.SelectedIndex <> 0 Then
            databasecommand.CommandText = "SELECT * FROM Others_salary
WHERE HMonth =" & cbomonths.SelectedIndex & " AND HYear =" &
cboyear.SelectedItem & " AND Emp_Name = '" & cboname.SelectedItem & "'"
        Else
            databasecommand.CommandText = "SELECT * FROM Others_salary
WHERE HYear =" & cboyear.SelectedItem & " AND Emp_Name = '" &
cboname.SelectedItem & "'"
        End If

        databasecommand.Connection = databaseconnection
        dbreader = databasecommand.ExecuteReader
        'reading details until database is reading datas
        While dbreader.Read
            Dim rowdata() As String = {dbreader.Item(0), dbreader.Item(1),
dbreader.Item(2), dbreader.Item(3), dbreader.Item(4), dbreader.Item(5)}
            dghours.Rows.Add(rowdata)
            found = True
        End While
        'Closing connection to database
        databaseconnection.Close()
    End Sub
    Private Sub getnames_salary()
        dghours.Rows.Clear()
        'declaring connection
        Dim databaseconnection As New OleDb.OleDbConnection
        Dim database_provider As String =
"Provider=Microsoft.ace.oledb.12.0;"
        'declaring data source
        Dim Datasource As String = "Data Source =" & CurDir() &
"\database.accdb"
        databaseconnection.ConnectionString = database_provider &
Datasource
        'opening connection to database
        databaseconnection.Open()
        Dim databasecommand1 As New OleDb.OleDbCommand
        Dim dbreader1 As OleDb.OleDbDataReader
        databasecommand1.CommandText = "SELECT * FROM Mname"
        databasecommand1.Connection = databaseconnection
        dbreader1 = databasecommand1.ExecuteReader
        salary = 0
        'reading details until database is reading datas
        While dbreader1.Read
            oid = dbreader1.Item(0)
            oname = dbreader1.Item(1)
            oclass = dbreader1.Item(4)
            salary = dbreader1.Item(3)
            Dim rowdata() As String = {cboyear.SelectedItem,
cbomonths.SelectedItem, oid, oname, oclass, salary}
            dghours.Rows.Add(rowdata)
            checkifdataalreadypresent()

        End While
        'Closing connection to database
        databaseconnection.Close()
    End Sub
    'saving salaries automatically
    Private Sub saveautomatically()

```

```

        Dim year As Integer = Today.Year
        Dim month As Integer = Today.Month
        'declaring connection
        Dim databaseconnection As New OleDb.OleDbConnection
        Dim database_provider As String =
"Provider=Microsoft.ace.oledb.12.0;" 
        'declaring data source
        Dim Datasource As String = "Data Source =" & CurDir() &
"\database.accdb"
        databaseconnection.ConnectionString = database_provider &
Datasource
        'opening connection to database
        databaseconnection.Open()
        Dim databasecommand As New OleDb.OleDbCommand
        databasecommand.CommandText = "INSERT INTO Others_Salary
VALUES (?,?,?,?,?,?)"
        With databasecommand.Parameters
            .AddWithValue("year", year)
            .AddWithValue("Month", month)
            .AddWithValue("Eid", oid)
            .AddWithValue("Name", oname)
            .AddWithValue("Class", oclass)
            .AddWithValue("salary", salary)
        End With
        databasecommand.Connection = databaseconnection
        databasecommand.ExecuteNonQuery()
        'Closing connection to database
        databaseconnection.Close()
        MsgBox("This month's salary of are saved automatically")
    End Sub
    Private Sub getnames()
        'declaring connection
        Dim databaseconnection As New OleDb.OleDbConnection
        Dim database_provider As String =
"Provider=Microsoft.ace.oledb.12.0;" 
        'declaring data source
        Dim Datasource As String = "Data Source =" & CurDir() &
"\database.accdb"
        databaseconnection.ConnectionString = database_provider &
Datasource
        'opening connection to database
        databaseconnection.Open()
        Dim databasecommand1 As New OleDb.OleDbCommand
        Dim dbreader1 As OleDb.OleDbDataReader
        databasecommand1.CommandText = "SELECT * FROM Mname"
        databasecommand1.Connection = databaseconnection
        dbreader1 = databasecommand1.ExecuteReader
        'reading details until database is reading datas
        While dbreader1.Read
            cbename.Items.Add(dbreader1.Item(1))
        End While
        'Closing connection to database
        databaseconnection.Close()
    End Sub
    Private Sub Frm_save_others_salary_Load(ByVal sender As System.Object,
 ByVal e As System.EventArgs) Handles MyBase.Load
        addyears()
        cbename.Items.Add("Not selected")
        getnames()
        cbomonths.SelectedItem = "ALL"
        cboyear.SelectedIndex = 0
    End Sub

```

```

        cboname.SelectedIndex = 0
    End Sub
    Private Sub addyears()
        For j = 0 To 4
            cboyear.Items.Add(Today.Year - j)
        Next
    End Sub
    Private Sub btnclose_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnclose.Click
        Me.Close()
    End Sub

    Private Sub Btnsavechanges_Click(ByVal sender As System.Object, ByVal e
As System.EventArgs) Handles Btnsavechanges.Click
        saveworkingdetails()
    End Sub

    Private Sub cboyear_SelectedIndexChanged(ByVal sender As System.Object,
ByVal e As System.EventArgs) Handles cboyear.SelectedIndexChanged
        If cboyear.SelectedIndex <> -1 And cboname.SelectedIndex <> -1 And
cboname.SelectedIndex <> -1 Then
            getdatasfromdb()
        End If
    End Sub
    Private Sub cbomonths_SelectedIndexChanged(ByVal sender As
System.Object, ByVal e As System.EventArgs) Handles cbomonths.SelectedIndexChanged
        If cboyear.SelectedIndex <> -1 And cboname.SelectedIndex <> -1 And
cboname.SelectedIndex <> -1 Then
            getdatasfromdb()
        End If
    End Sub

    Private Sub cboname_SelectedIndexChanged(ByVal sender As System.Object,
ByVal e As System.EventArgs) Handles cboname.SelectedIndexChanged
        If cboyear.SelectedIndex <> -1 And cboname.SelectedIndex <> -1 And
cboname.SelectedIndex <> -1 Then
            getdatasfromdb()
        End If
    End Sub

    Private Sub getdatasfromdb()
        dghours.Rows.Clear()
        'declaring connection
        Dim databaseconnection As New OleDb.OleDbConnection
        Dim database_provider As String =
"Provider=Microsoft.aceoledb.12.0;"
        'declaring data source
        Dim Datasource As String = "Data Source =" & CurDir() &
"\database.accdb"
        databaseconnection.ConnectionString = database_provider &
Datasource
        'opening connection to database
        Dim name As String = cboname.SelectedItem
        Dim month As Integer = cbomonths.SelectedIndex
        Dim year As String = cboyear.SelectedItem
        databaseconnection.ConnectionString = database_provider &
Datasource
        databaseconnection.Open()
        Dim databasecommand As New OleDb.OleDbCommand
        Dim dbreader As OleDb.OleDbDataReader
        If cbomonths.SelectedIndex = 0 Then

```

```

        databasecommand.CommandText = "SELECT * FROM Others_salary
WHERE HYear = " & year & " AND Emp_Name = '" & name & "'"
    Else
        databasecommand.CommandText = "SELECT * FROM Others_salary
WHERE HYear = " & year & " AND HMonth = " & month & " AND Emp_Name = '" &
name & "'"
    End If
    databasecommand.Connection = databaseconnection
    dbreader = databasecommand.ExecuteReader
    While dbreader.Read
        Dim rowdata() As String = {dbreader.Item(0), dbreader.Item(1),
dbreader.Item(2), dbreader.Item(3), dbreader.Item(4), dbreader.Item(5)}
        dghours.Rows.Add(rowdata)
        found = True
    End While
    'Closing connection to database
    databaseconnection.Close()
End Sub
'saving all working details
Private Sub saveworkingdetails()
    Dim nrows As Integer
    'declaring connection
    Dim databaseconnection As New OleDb.OleDbConnection
    Dim database_provider As String =
"Provider=Microsoft.ace.oledb.12.0;"
    'declaring data source
    Dim Datasource As String = "Data Source =" & CurDir() &
"\database.accdb"
    databaseconnection.ConnectionString = database_provider &
Datasource
    'opening connection to database
    databaseconnection.Open()
    databaseconnection.ConnectionString = database_provider &
Datasource
    Dim month As Integer = cbomonths.SelectedIndex
    Dim year As String = cboyear.SelectedItem
    If cbomonths.SelectedIndex = 0 Then
        Dim dbdel As New OleDb.OleDbCommand("DELETE FROM Others_salary
WHERE HYear = ? AND Emp_name = ?", databaseconnection)
        dbdel.Parameters.AddWithValue("pyear", year)
        dbdel.Parameters.AddWithValue("pname", cboname.SelectedItem)
        dbdel.ExecuteNonQuery()
    Else
        Dim dbdel As New OleDb.OleDbCommand("DELETE FROM Others_salary
WHERE HMonth=? AND HYear = ? AND Emp_name = ?", databaseconnection)
        dbdel.Parameters.AddWithValue("pmonth", month)
        dbdel.Parameters.AddWithValue("pyear", year)
        dbdel.Parameters.AddWithValue("pname", cboname.SelectedItem)
        dbdel.ExecuteNonQuery()
    End If
    'Closing connection to database
    databaseconnection.Close()
    For nrows = 0 To dghours.Rows.Count - 1
        databaseconnection.Open()
        Dim databasecommand As New OleDb.OleDbCommand
        databasecommand.CommandText = "INSERT INTO Others_salary
VALUES (?,?,?,?,?,?)"
        With databasecommand.Parameters
            .AddWithValue("year", dghours.Item(0, nrows).Value)
            .AddWithValue("Month", dghours.Item(1, nrows).Value)
            .AddWithValue("E_id", dghours.Item(2, nrows).Value)

```

```
    .AddWithValue("E_name", dghours.Item(3, nrows).Value)
    .AddWithValue("EClass", dghours.Item(4, nrows).Value)
    .AddWithValue("salary", dghours.Item(5, nrows).Value)
End With
databasecommand.Connection = databaseconnection
databasecommand.ExecuteNonQuery()
'Closing connection to database
databaseconnection.Close()
Next
MsgBox(cboname.SelectedItem & "'s salary are saved to the
database")
End Sub
End Class
```

Form: Frm\_View\_allowance



```
Public Class Frm_View_allowances
    Public month As Integer
    Public year As Integer
    Public allowance As Integer
    Private Sub btnclose_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnclose.Click
        Me.Close()
    End Sub
    Private Sub Frm_View_allowances_Load(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles MyBase.Load
        'selecting name with index 0
        cboname.SelectedIndex = 0
        'getting names
        getname()
        'adding years
        addyears()
        'selecting year and month having index 0
        cboyear.SelectedIndex = 0
        cbomonths.SelectedIndex = 0
    End Sub
    'getting years
    Private Sub addyears()
        For j = 0 To 4
            'adding years in combobox
            cboyear.Items.Add(Today.Year - j)
        Next
    End Sub
    'getting names from database
```

```

Private Sub getname()
    'declaring connection
    Dim databaseconnection As New OleDb.OleDbConnection
    Dim database_provider As String =
"Provider=Microsoft.ace.oledb.12.0;"
    'declaring data source
    Dim Datasource As String = "Data Source =" & CurDir() &
"\database.accdb"
    databaseconnection.ConnectionString = database_provider &
Datasource
    'opening connection to database
    databaseconnection.Open()
    Dim databasecommand As New OleDb.OleDbCommand
    Dim dbreader As OleDb.OleDbDataReader
    databasecommand.CommandText = "SELECT * FROM mname "
    databasecommand.Connection = databaseconnection
    dbreader = databasecommand.ExecuteReader
    'frmadd.txtid.Text = dbreader.Item(0)
    While dbreader.Read
        cboname.Items.Add(dbreader.Item(1))
    End While
    'Closing connection to database
    databaseconnection.Close()
End Sub

Private Sub cboname_SelectedIndexChanged(ByVal sender As System.Object,
ByVal e As System.EventArgs) Handles cboname.SelectedIndexChanged
    getdatasfromdatabase()
End Sub
Private Sub getdatasfromdatabase()
    DataGridView1.Rows.Clear()
    Dim name As String = cboname.SelectedItem
    If Manager_s_Salary_details.sendcommand = "" Then
        'do nothing
    Else
        month = Manager_s_Salary_details.cbomonths.SelectedIndex
        year = Manager_s_Salary_details.cboyear.SelectedItem
        name = Manager_s_Salary_details.cboname.SelectedItem
        cboyear.SelectedItem = year
        cbomonths.SelectedIndex = month
        cboname.SelectedItem = name
        cboname.Enabled = False
        cboyear.Enabled = False
        cbomonths.Enabled = False
    End If
    Dim i As Integer
    'declaring connection
    Dim databaseconnection As New OleDb.OleDbConnection
    Dim database_provider As String =
"Provider=Microsoft.ace.oledb.12.0;"
    'declaring data source
    Dim Datasource As String = "Data Source =" & CurDir() &
"\database.accdb"
    databaseconnection.ConnectionString = database_provider &
Datasource
    'opening connection to database
    databaseconnection.Open()
    Dim databasecommand As New OleDb.OleDbCommand
    Dim dbreader As OleDb.OleDbDataReader
    month = cbomonths.SelectedIndex
    For i = 0 To 4

```

```

        If cboyear.SelectedIndex = i Then
            year = Today.Year - i
        End If
    Next
    If cbomonths.SelectedIndex <> 0 Then
        databasecommand.CommandText = "SELECT * FROM Allowances WHERE
Emp_Name = " & "****" & name & "****" & " AND FMonth =" & month & " AND FYear
= " & year
    Else
        databasecommand.CommandText = "SELECT * FROM Allowances WHERE
Emp_Name = " & "****" & name & "****" & " AND FYear =" & year
    End If
    databasecommand.Connection = databaseconnection
    dbreader = databasecommand.ExecuteReader
    allowance = 0
    DataGridView1.Rows.Clear()
    While dbreader.Read
        Dim rowdata() As String = {dbreader.Item(0), dbreader.Item(1),
dbreader.Item(2), dbreader.Item(3), dbreader.Item(4), dbreader.Item(5)}
        DataGridView1.Rows.Add(rowdata)
        allowance = allowance + dbreader.Item(5)
    End While
    lblallowance.Text = allowance
    'Closing connection to database
    databaseconnection.Close()
End Sub

Private Sub cboyear_SelectedIndexChanged(ByVal sender As System.Object,
 ByVal e As System.EventArgs) Handles cboyear.SelectedIndexChanged
    getdatasfromdatabase()
End Sub

Private Sub cbomonths_SelectedIndexChanged(ByVal sender As
System.Object, ByVal e As System.EventArgs) Handles
cbomonths.SelectedIndexChanged
    getdatasfromdatabase()
End Sub
End Class

```

**Form: Show manager expenses**



```
Public Class Show_Manager_expenses
    Public month As String
    Public year As String
    Public ename As String
    Public totalexpenses As Integer

    Private Sub btnclose_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnclose.Click
        'close this form
        Me.Close()
    End Sub

    Private Sub Show_Manager_expenses_Load(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles MyBase.Load
        cboname.SelectedIndex = 0
        getname()
        addyears()
        cboyear.SelectedIndex = 0
        cbomonths.SelectedIndex = 0
    End Sub

    Private Sub loadaddatas()
        dghours.Rows.Clear()
        Dim name As String = cboname.SelectedItem
        If Manager_s_Salary_details.sendcommand = "" Then
            'do nothing
        Else
            month = Manager_s_Salary_details.cbomonths.SelectedIndex
            year = Manager_s_Salary_details.cboyear.SelectedItem
            name = Manager_s_Salary_details.cboname.SelectedItem
        End If
    End Sub
```

```

cboyear.SelectedItem = year
cbomonths.SelectedIndex = month
cboname.SelectedItem = name
cboname.Enabled = False
cboyear.Enabled = False
cbomonths.Enabled = False
End If
month = cbomonths.SelectedIndex
year = cboyear.SelectedItem
ename = cboname.SelectedItem
If cbomonths.SelectedIndex = -1 Then Exit Sub
'declaring connection
Dim databaseconnection As New OleDb.OleDbConnection
Dim database_provider As String =
"Provider=Microsoft.ace.oledb.12.0;"
'declaring data source
Dim Datasource As String = "Data Source =" & CurDir() &
"\database.accdb"
databaseconnection.ConnectionString = database_provider &
Datasource
'opening connection to database
databaseconnection.Open()
Dim databasecommand As New OleDb.OleDbCommand
Dim dbreader As OleDb.OleDbDataReader
If month = 0 Then
    databasecommand.CommandText = "SELECT * FROM Others_Expenses
WHERE Emp_Name = '" & ename & "' AND HYear =" & year
Else
    databasecommand.CommandText = "SELECT * FROM Others_Expenses
WHERE Emp_Name = '" & ename & "' AND HMonth =" & month & " AND HYear
=" & year
End If
databasecommand.Connection = databaseconnection
dbreader = databasecommand.ExecuteReader
totalexpenses = 0
dghours.Rows.Clear()
While dbreader.Read
    Dim rowdata() As String = {dbreader.Item(0), dbreader.Item(1),
dbreader.Item(2), dbreader.Item(3), _           dbreader.Item(4), dbreader.Item(5),
dbreader.Item(6)}
    'adding row to datagridview
    dghours.Rows.Add(rowdata)
    totalexpenses = totalexpenses + dbreader.Item(6)
End While
databaseconnection.Close()
lblexpenses.Text = totalexpenses
End Sub
'adding years to combobox
Private Sub addyears()
    For j = 0 To 4
        cboyear.Items.Add(Today.Year - j)
    Next
End Sub
Private Sub getname()
'declaring connection
Dim databaseconnection As New OleDb.OleDbConnection
Dim database_provider As String =
"Provider=Microsoft.ace.oledb.12.0;"
'declaring data source

```

```

        Dim Datasource As String = "Data Source =" & CurDir() &
"\database.accdb"
        databaseconnection.ConnectionString = database_provider &
Datasource
        'opening connection to database
        databaseconnection.Open()
        Dim databasecommand As New OleDb.OleDbCommand
        Dim dbreader As OleDb.OleDbDataReader
        databasecommand.CommandText = "SELECT * FROM mname "
        databasecommand.Connection = databaseconnection
        dbreader = databasecommand.ExecuteReader
        'reads datas until end
        While dbreader.Read
            cboname.Items.Add(dbreader.Item(1))
        End While
        databaseconnection.Close()
    End Sub
    Private Sub cboyear_SelectedIndexChanged(ByVal sender As System.Object,
 ByVal e As System.EventArgs) Handles cboyear.SelectedIndexChanged
        'if month or year has selected index = -1 then exit this sub
        If cbomonths.SelectedIndex = -1 Or cboyear.SelectedIndex = -1 Then
Exit Sub
        loaddatas()
    End Sub
    Private Sub cbomonths_SelectedIndexChanged(ByVal sender As
System.Object, ByVal e As System.EventArgs) Handles
cbomonths.SelectedIndexChanged
        'if month or year has selected index = -1 then exit this sub
        If cbomonths.SelectedIndex = -1 Or cboyear.SelectedIndex = -1 Then
Exit Sub
        loaddatas()
    End Sub
    Private Sub cboname_SelectedIndexChanged(ByVal sender As System.Object,
 ByVal e As System.EventArgs) Handles cboname.SelectedIndexChanged
        'if month or year has selected index = -1 then exit this sub
        If cbomonths.SelectedIndex = -1 Or cboyear.SelectedIndex = -1 Then
Exit Sub
        loaddatas()
    End Sub
End Class

```

### Form: Frmworking\_detail

The screenshot shows a Windows application window titled "Main - [frmWorking\_detail]". The menu bar includes File, Employee, Edit, View, Upload, User, and Help. The main content area displays a title "All workers working hours update" above a DataGridView. The DataGridView has columns: HMonth, HDay, HYear, Emp\_Id, Employee\_Name, Start Time, End Time, Expenses(RS), and Rate. The data shows various employees (Sunil, Sunil Thapa, Ram Thapa, Sharukh Khan, Salman Khan, rahul singh, new one) with their respective details. At the bottom of the window are two buttons: "Save" (with a save icon) and "Close" (with a close icon). A status bar at the bottom left shows "Logged in as sunil".

```

Public Class frmWorking_detail
    Public day As String
    Public month As String
    Public year As String
    Public status As Boolean
    Public dte As String
    Public rate As Integer
    Public cname As String
    Private Sub getnamesanddate()
        dghours.Rows.Clear()
        Dim start, over As Integer
        'declaring connection
        Dim databaseconnection As New OleDb.OleDbConnection
        Dim database_provider As String =
"Provider=Microsoft.ace.oledb.12.0;"
        'declaring data source
        Dim Datasource As String = "Data Source =" & CurDir() &
"\database.accdb"
        databaseconnection.ConnectionString = database_provider &
Datasource
        'opening connection to database
        databaseconnection.Open()
        Dim databasecommand1, databasecommand2 As New OleDb.OleDbCommand
        Dim dbreader1, dbreader2 As OleDb.OleDbDataReader
        databasecommand1.CommandText = "SELECT * FROM Cname"
        databasecommand2.CommandText = "SELECT * FROM Ctime"
        databasecommand1.Connection = databaseconnection
        databasecommand2.Connection = databaseconnection
        dbreader1 = databasecommand1.ExecuteReader
        dbreader2 = databasecommand2.ExecuteReader
        dbreader2.Read()
        start = dbreader2.Item(0)
    End Sub
End Class

```

```

        over = dbreader2.Item(1)
    While dbreader1.Read
        cname = dbreader1.Item(1)
        'get rate of an employee first
        getrate()
        Dim rowdata() As String = {month, day, year, dbreader1.Item(0),
dbreader1.Item(1), start, over, "0", rate}
        'add row to datagridview
        dghours.Rows.Add(rowdata)
    End While
    databaseconnection.Close()
End Sub
Private Sub checkifdataalreadypresent()
    dghours.Rows.Clear()
    'declaring connection
    Dim databaseconnection As New OleDb.OleDbConnection
    Dim database_provider As String =
"Provider=Microsoft.ace.oledb.12.0;"
    'declaring data source
    Dim Datasource As String = "Data Source =" & CurDir() &
"\database.accdb"
    databaseconnection.ConnectionString = database_provider &
Datasource
    'opening connection to database
    databaseconnection.Open()
    Dim databasecommand As New OleDb.OleDbCommand
    Dim dbreader As OleDb.OleDbDataReader
    databasecommand.CommandText = "SELECT * FROM Emp_workingdetails
WHERE HDay = " & day & " AND HMonth =" & month & " AND HYear =" & year
    databasecommand.Connection = databaseconnection
    dbreader = databasecommand.ExecuteReader
    status = False
    While dbreader.Read
        Dim rowdata() As String = {dbreader.Item(0), dbreader.Item(1),
dbreader.Item(2), dbreader.Item(3), dbreader.Item(4), dbreader.Item(5),
dbreader.Item(6), dbreader.Item(7), dbreader.Item(8)}
        'add row to datagridview
        dghours.Rows.Add(rowdata)
        status = True
    End While
    databaseconnection.Close()
End Sub
Private Sub loadfrm()
    'extract dte from main.dte
    dte = Main.dte
    Dim lengthofdte As Integer
    Dim sfpos As Integer
    Dim slpos As Integer
    'get length of dte
    lengthofdte = dte.Length
    'get position of / in dte
    sfpos = dte.IndexOf("/") + 1
    slpos = dte.LastIndexOf("//") + 1
    'get month
    month = dte.Substring(0, sfpos - 1)
    'get day
    day = dte.Substring(sfpos, slpos - sfpos - 1)
    'get year
    year = dte.Substring(slpos, 4)
    Me.WindowState = FormWindowState.Maximized
    checkifdataalreadypresent()

```

```

        If status = True Then
        Else
            getnamesanddate()
        End If
    End Sub
    Private Sub frmWorking_detail_Load(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles MyBase.Load
        loadfrm()
    End Sub
    Private Sub Btnsave_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Btnsave.Click
        saveworkingdetails()
        'message! do you want to add another record. if yes then continue
    else dont
        If MessageBox.Show("Do you want to upload another record.", "Upload
Next Record", MessageBoxButtons.YesNo, MessageBoxIcon.Question) =
Windows.Forms.DialogResult.Yes Then
            'open upload form
            dte = InputBox("Enter Date of that day you want to edit", ,
Today)
            Main.dte = dte
            If Main.dte <> "" Then
                loadfrm()
                Me.Show()
                Me.WindowState = FormWindowState.Maximized
            End If
        Else
            Me.Close()
        End If
    End Sub
    Private Sub saveworkingdetails()
        Dim nrows As Integer
        'declaring connection
        Dim databaseconnection As New OleDb.OleDbConnection
        Dim database_provider As String =
"Provider=Microsoft.ace.oledb.12.0;"
        'declaring data source
        Dim Datasource As String = "Data Source =" & CurDir() &
"\database.accdb"
        databaseconnection.ConnectionString = database_provider &
Datasource
        'opening connection to database
        databaseconnection.Open()
        Dim dbdel As New OleDb.OleDbCommand("DELETE FROM Emp_Workingdetails
WHERE HMonth=? AND HDay=? AND HYear = ?", databaseconnection)
        dbdel.Parameters.AddWithValue("pmonth", month)
        dbdel.Parameters.AddWithValue("pday", day)
        dbdel.Parameters.AddWithValue("pyear", year)
        dbdel.ExecuteNonQuery()
        databaseconnection.Close()
        For nrows = 0 To dghours.Rows.Count - 1
            databaseconnection.Open()
            Dim databasecommand As New OleDb.OleDbCommand
            databasecommand.CommandText = "INSERT INTO Emp_workingdetails
VALUES(?,?,?,?,?,?)"
            With databasecommand.Parameters
                .AddWithValue("Month", dghours.Item(0, nrows).Value)
                .AddWithValue("Day", dghours.Item(1, nrows).Value)
                .AddWithValue("year", dghours.Item(2, nrows).Value)
                .AddWithValue("e_id", dghours.Item(3, nrows).Value)
                .AddWithValue("E_name", dghours.Item(4, nrows).Value)
            End With
        Next
    End Sub

```

```

        .AddWithValue("stime", dghours.Item(5, nrows).Value)
        .AddWithValue("etime", dghours.Item(6, nrows).Value)
        .AddWithValue("expenses", dghours.Item(7, nrows).Value)
        .AddWithValue("rate", dghours.Item(8, nrows).Value)
    End With
    databasecommand.Connection = databaseconnection
    databasecommand.ExecuteNonQuery()
    databaseconnection.Close()
Next
MsgBox("Employee's working hours details are added to the
database")
End Sub

Private Sub btnclose_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnclose.Click
    Me.Close()
End Sub
Private Sub dghours_CellContentClick(ByVal sender As System.Object,
ByVal e As System.Windows.Forms.DataGridViewCellEventArgs) Handles
dghours.CellContentClick
    If e.ColumnIndex <> 8 Then Exit Sub
End Sub
Private Sub getrate()
    'declaring connection
    Dim databaseconnection As New OleDb.OleDbConnection
    Dim database_provider As String =
"Provider=Microsoft.ace.oledb.12.0;"
    'declaring data source
    Dim Datasource As String = "Data Source =" & CurDir() &
"\database.accdb"
    databaseconnection.ConnectionString = database_provider &
Datasource
    'opening connection to database
    databaseconnection.Open()
    Dim databasecommand As New OleDb.OleDbCommand
    Dim dbreader As OleDb.OleDbDataReader
    databasecommand.CommandText = "SELECT * FROM Cname WHERE
Carpenter_Name=''" & cname & "'"
    databasecommand.Connection = databaseconnection
    dbreader = databasecommand.ExecuteReader
    Dim hasrows As Boolean = dbreader.Read()
    If hasrows = True Then
        rate = dbreader.Item(4)
    Else
        rate = 0
    End If
    databaseconnection.Close()
End Sub
End Class

```

Form: Frm others salary upload

Main - [Frm\_others\_salary\_Upload]

File Employee Edit View Upload User Help

## Monthly Paid Employee's Salary Upload

HYear	HMonth	Emp_Id	Employee_Name	EClass	Salary
2014	3	501	Sunil Thapa	Manager	30000
2014	3	503	damai	Driver	30000
2014	3	504	ram	assistant	75000
2014	3	505	amit	driver	15000
2014	3	506	sachin tendulkar	spider man	50000

 Save     Close

Logged in as sunil

```
Public Class Frm_others_salary_Upload
    Public month As Integer
    Public year As Integer
    Public status As Boolean
    Public oid As Integer
    Public oname As String
    Public salary As Integer
    Public oclass As String
    Public ename As String
    Public dte As String

    Private Sub loadfrm()
        Me.WindowState = FormWindowState.Maximized
        dte = Main.dte
        Dim lengthofdte As Integer
        Dim sfpos As Integer
        Dim slpos As Integer
        'getting length of dte
        lengthofdte = dte.Length
        'extracting position of /
        sfpos = dte.IndexOf("/") + 1
        slpos = dte.LastIndexOf("/") + 1
        'getting month
        month = dte.Substring(0, sfpos - 1)
        'getting year
        year = dte.Substring(slpos, 4)
```

```

        'showing this form in maximum size
        Me.WindowState = FormWindowState.Maximized
        checkdata_for_that_name_only()
    End Sub
    Private Sub Frm_others_salary_Load(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles MyBase.Load
        loadfrm()
    End Sub
    Private Sub btnclose_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnclose.Click
        'closing this form
        Me.Close()
    End Sub
    Private Sub Btnsave_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Btnsave.Click
        'saving all working details
        saveworkingdetails()
        If MessageBox.Show("Do you want to upload another record.", "Upload Next Record", MessageBoxButtons.YesNo, MessageBoxIcon.Question) =
        Windows.Forms.DialogResult.Yes Then
            'getting an inputbox
            dte = InputBox("Enter Month and year of which you want to upload salary of monthly paid employee. (Month/Year) (MM/YYYY)", ,
            Today.Month & "/" & Today.Year)
            Main.dte = dte
            If Main.dte <> "" Then
                loadfrm()
                Me.Show()
                'showing the form in maximum size
                Me.WindowState = FormWindowState.Maximized
            End If
        Else
            Me.Close()
        End If
    End Sub
    Private Sub getnames_salary()
        'declaring connection
        Dim databaseconnection As New OleDb.OleDbConnection
        Dim database_provider As String =
        "Provider=Microsoft.ace.oledb.12.0;"
        'declaring data source
        Dim Datasource As String = "Data Source =" & CurDir() &
        "\database.accdb"
        databaseconnection.ConnectionString = database_provider &
        Datasource
        'opening connection to database
        databaseconnection.Open()
        Dim databasecommand1 As New OleDb.OleDbCommand
        Dim dbreader1 As OleDb.OleDbDataReader
        databasecommand1.CommandText = "SELECT * FROM Mname WHERE Emp_Name
        = '" & ename & "'"
        databasecommand1.Connection = databaseconnection
        dbreader1 = databasecommand1.ExecuteReader
        'initializing salary as 0
        salary = 0
        'reading details until database is reading datas
        While dbreader1.Read
            Dim rowdata() As String = {year, month, dbreader1.Item(0),
            dbreader1.Item(1), dbreader1.Item(4), dbreader1.Item(3)}
            dghours.Rows.Add(rowdata)
        End While
    End Sub

```

```

        'Closing connection to database
        databaseconnection.Close()
    End Sub
    Private Sub checkifdataalreadypresent()
        'declaring connection
        Dim databaseconnection As New OleDb.OleDbConnection
        Dim database_provider As String =
"Provider=Microsoft.ace.oledb.12.0;"
        'declaring data source
        Dim Datasource As String = "Data Source =" & CurDir() &
"\database.accdb"
        databaseconnection.ConnectionString = database_provider &
Datasource
        'opening connection to database
        databaseconnection.Open()
        Dim databasecommand As New OleDb.OleDbCommand
        Dim dbreader As OleDb.OleDbDataReader
        databasecommand.CommandText = "SELECT * FROM Others_salary WHERE
HMonth =" & month & " AND HYear =" & year & " AND Emp_Name = '" & ename &
"'
        databasecommand.Connection = databaseconnection
        dbreader = databasecommand.ExecuteReader
        'setting status = false
        status = False
        'reading details until database is reading datas
        While dbreader.Read
            Dim rowdata() As String = {dbreader.Item(0), dbreader.Item(1),
dbreader.Item(2), dbreader.Item(3), dbreader.Item(4), dbreader.Item(5)}
            'adding 1 row to datagridview
            dghours.Rows.Add(rowdata)
            status = True
        End While
        If status = False Then
            getnames_salary()
        End If
        'Closing connection to database
        databaseconnection.Close()
    End Sub
    Private Sub saveworkingdetails()
        Dim nrows As Integer
        'declaring connection
        Dim databaseconnection As New OleDb.OleDbConnection
        Dim database_provider As String =
"Provider=Microsoft.ace.oledb.12.0;"
        'declaring data source
        Dim Datasource As String = "Data Source =" & CurDir() &
"\database.accdb"
        databaseconnection.ConnectionString = database_provider &
Datasource
        'opening connection to database
        databaseconnection.Open()
        Dim dbdel As New OleDb.OleDbCommand("DELETE FROM Others_salary
WHERE HMonth=? AND HYear = ? ", databaseconnection)
        dbdel.Parameters.AddWithValue("pmonth", month)
        dbdel.Parameters.AddWithValue("pyear", year)
        dbdel.ExecuteNonQuery()
        databaseconnection.Close()
        For nrows = 0 To dghours.Rows.Count - 1
            databaseconnection.Open()
            Dim databasecommand As New OleDb.OleDbCommand
            'saving datas from datagridview to database

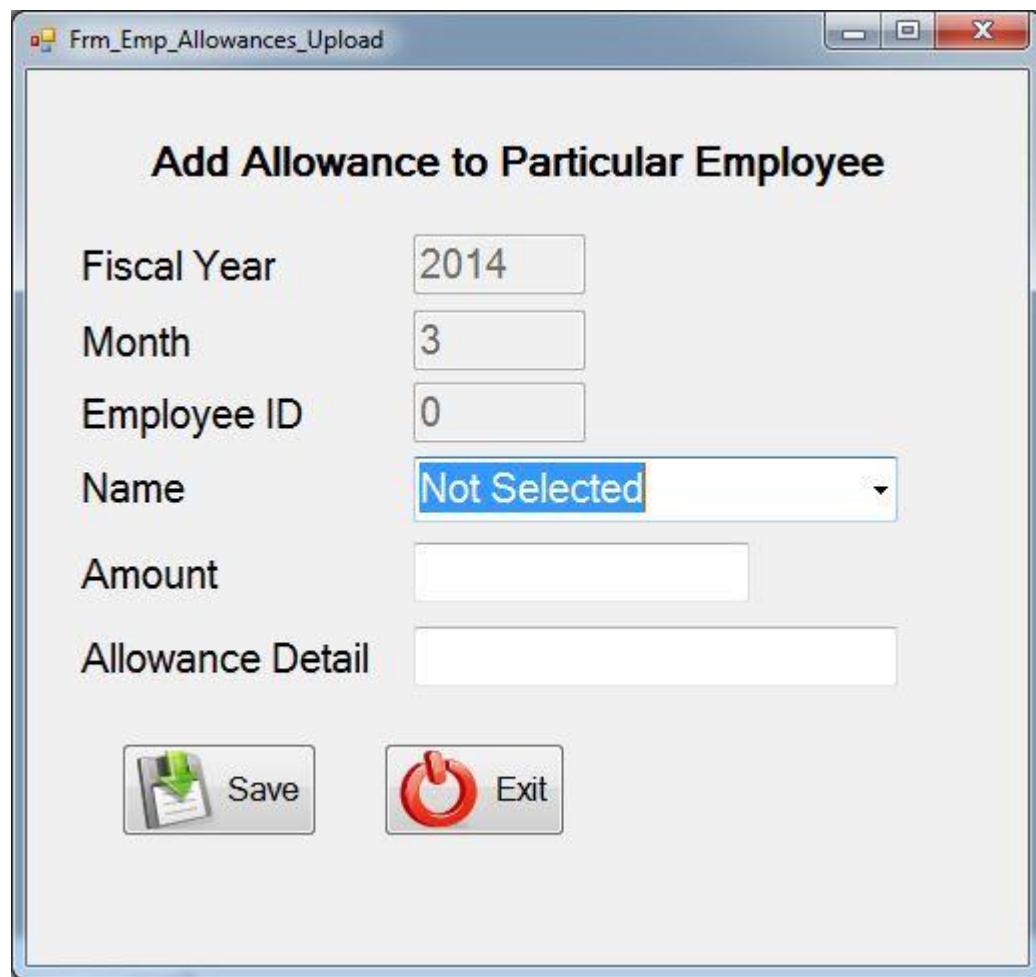
```

```

        databasecommand.CommandText = "INSERT INTO Others_salary
VALUES (?,?,?,?,?,?)"
        With databasecommand.Parameters
            .AddWithValue("year", dghours.Item(0, nrows).Value)
            .AddWithValue("Month", dghours.Item(1, nrows).Value)
            .AddWithValue("E_id", dghours.Item(2, nrows).Value)
            .AddWithValue("E_name", dghours.Item(3, nrows).Value)
            .AddWithValue("EClass", dghours.Item(4, nrows).Value)
            .AddWithValue("salary", dghours.Item(5, nrows).Value)
        End With
        databasecommand.Connection = databaseconnection
        databasecommand.ExecuteNonQuery()
        'Closing connection to database
        databaseconnection.Close()
    Next
    MsgBox("All data are saved to the database")
End Sub
Private Sub checkdata_for_that_name_only()
    dghours.Rows.Clear()
    'declaring connection
    Dim databaseconnection As New OleDb.OleDbConnection
    Dim database_provider As String =
"Provider=Microsoft.ace.oledb.12.0;"
    'declaring data source
    Dim Datasource As String = "Data Source =" & CurDir() &
"\database.accdb"
    databaseconnection.ConnectionString = database_provider &
Datasource
    'opening connection to database
    databaseconnection.Open()
    Dim databasecommand1 As New OleDb.OleDbCommand
    Dim dbreader1 As OleDb.OleDbDataReader
    databasecommand1.CommandText = "SELECT * FROM Mname"
    databasecommand1.Connection = databaseconnection
    dbreader1 = databasecommand1.ExecuteReader
    'reading details until database is reading datas
    While dbreader1.Read
        ename = dbreader1.Item(1)
        checkifdataalreadypresent()
    End While
    'Closing connection to database
    databaseconnection.Close()
End Sub
End Class

```

Form: Frm\_emp\_allowances\_upload



```
Public Class Frm_Emp_Allowances_upload
    Public month As Integer
    Public year As Integer
    Public eid As Integer

    Private Sub Frm_Emp_Allowances_upload_Load(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles MyBase.Load
        'disabling text boxes
        txtmonth.Enabled = False
        txtyear.Enabled = False
        txtid.Enabled = False
        'selecting item 'not selected'
        cboname.Items.Add("Not Selected")
        cboname.SelectedIndex = 0
        Dim tdate As String
        'getting an inputbox
        tdate = InputBox("Enter the date of which you want to upload allowances", "Date", Today.Month & "/" & Today.Year)
        If tdate = "" Then
            Me.Close()
        Else
            Dim lengthofdte As Integer
            Dim sfpos As Integer
```

```

        Dim slpos As Integer
        'getting length of tdate
        lengthofdte = tdate.Length
        'extracting position of /
        sfpos = tdate.IndexOf("/") + 1
        slpos = tdate.LastIndexOf("/") + 1
        'getting month
        month = tdate.Substring(0, sfpos - 1)
        'getting year
        year = tdate.Substring(slpos, 4)
        txtyear.Text = year
        txtmonth.Text = month
    End If
    get_names()
End Sub

Private Sub Btnexit_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)
    'closing this form
    Me.Close()
End Sub

Private Sub btnsave_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnsave.Click
    'saving details
    savedetails()
End Sub
Private Sub savedetails()
    'declaring connection
    Dim databaseconnection As New OleDb.OleDbConnection
    Dim database_provider As String =
"Provider=Microsoft.ace.oledb.12.0;" 
    'declaring data source
    Dim Datasource As String = "Data Source =" & CurDir() &
"\database.accdb"
    databaseconnection.ConnectionString = database_provider &
Datasource
    'opening connection to database
    databaseconnection.Open()
    Dim databasecommand As New OleDb.OleDbCommand
    databasecommand.CommandText = "INSERT INTO Allowances
VALUES (?,?,?,?,?,?)"
    databasecommand.Parameters.AddWithValue("Fyear", year)
    databasecommand.Parameters.AddWithValue("Month", month)
    databasecommand.Parameters.AddWithValue("Eid", txtid.Text)
    databasecommand.Parameters.AddWithValue("Ename",
cboname.SelectedItem)
    databasecommand.Parameters.AddWithValue("altype",
txtallodetails.Text)
    databasecommand.Parameters.AddWithValue("alamt", txtamount.Text)
    databasecommand.Connection = databaseconnection
    databasecommand.ExecuteNonQuery()
    'Closing connection to database
    databaseconnection.Close()
    MsgBox(cboname.SelectedItem & "'s allowance is added ")
End Sub

Private Sub cboname_SelectedIndexChanged(ByVal sender As System.Object,
ByVal e As System.EventArgs) Handles cboname.SelectedIndexChanged
    'extract id form database
    getid()

```

```

End Sub
'getting names form database
Private Sub get_names()
    'declaring connection
    Dim databaseconnection As New OleDb.OleDbConnection
    Dim database_provider As String =
"Provider=Microsoft.ace.oledb.12.0;"
    'declaring data source
    Dim Datasource As String = "Data Source =" & CurDir() &
"\database.accdb"
    databaseconnection.ConnectionString = database_provider &
Datasource
    'opening connection to database
    databaseconnection.Open()
    Dim databasecommand As New OleDb.OleDbCommand
    Dim dbreader As OleDb.OleDbDataReader
    databasecommand.CommandText = "SELECT * FROM Mname"
    databasecommand.Connection = databaseconnection
    dbreader = databasecommand.ExecuteReader
    'reading details until database is reading datas
    While dbreader.Read
        cboname.Items.Add(dbreader.Item(1))
        eid = dbreader.Item(0)
    End While
    'Closing connection to database
    databaseconnection.Close()
End Sub
Private Sub getid()
    eid = 0
    txtid.Clear()
    'declaring connection
    Dim databaseconnection As New OleDb.OleDbConnection
    Dim database_provider As String =
"Provider=Microsoft.ace.oledb.12.0;"
    'declaring data source
    Dim Datasource As String = "Data Source =" & CurDir() &
"\database.accdb"
    databaseconnection.ConnectionString = database_provider &
Datasource
    'opening connection to database
    databaseconnection.Open()
    Dim databasecommand As New OleDb.OleDbCommand
    Dim dbreader As OleDb.OleDbDataReader
    databasecommand.CommandText = "SELECT * FROM Mname WHERE Emp_Name =
" & cboname.SelectedItem & """
    databasecommand.Connection = databaseconnection
    dbreader = databasecommand.ExecuteReader
    'reading details until database is reading datas
    While dbreader.Read
        eid = dbreader.Item(0)
    End While
    txtid.Text = eid
    'Closing connection to database
    databaseconnection.Close()
End Sub

Private Sub Btnexit_Click_1(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Btnexit.Click
    'closing this form
    Me.Close()
End Sub

```

```

    Private Sub txtamount_KeyPress(ByVal sender As Object, ByVal e As
System.Windows.Forms.KeyPressEventArgs) Handles txtamount.KeyPress
        If Asc(e.KeyChar) <> 8 Then
            If Asc(e.KeyChar) < 48 Or Asc(e.KeyChar) > 57 Then
                e.Handled = True
            End If
        End If
    End Sub

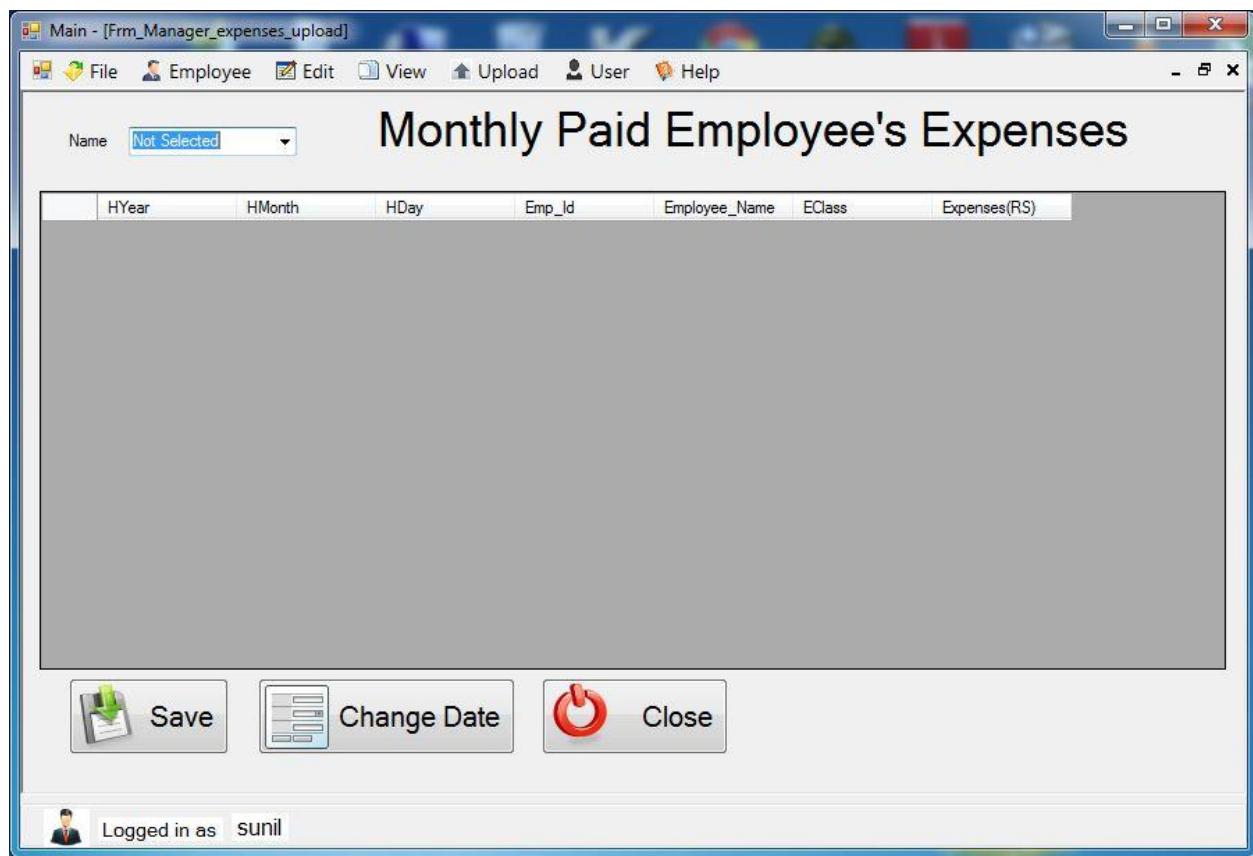
    Private Sub txtamount_TextChanged(ByVal sender As System.Object, ByVal
e As System.EventArgs) Handles txtamount.TextChanged
        txtamount.MaxLength = 10
    End Sub

    Private Sub txtallodetails_KeyPress(ByVal sender As Object, ByVal e As
System.Windows.Forms.KeyPressEventArgs) Handles txtallodetails.KeyPress
        Select Case Asc(e.KeyChar)
            Case 8, 32, 65 To 90, 97 To 122
                e.Handled = False
            Case Else
                e.Handled = True
        End Select
    End Sub

    Private Sub txtallodetails_TextChanged(ByVal sender As System.Object,
ByVal e As System.EventArgs) Handles txtallodetails.TextChanged
        txtallodetails.MaxLength = 40
    End Sub
End Class

```

**Form: Frm\_manager\_expenses\_upload**



```
Public Class Frm_Manager_expenses_upload
    Public day As String
    Public month As String
    Public year As String
    Public status As Boolean
    Public dte As String
    Public mname As String
    Public eclass As String

    Private Sub loadfrm()
        'getting dte from form main
        dte = Main.dte
        lblmsg.Visible = False
        Dim lengthofdte As Integer
        Dim sfpos As Integer
        Dim slpos As Integer
        cboname.Items.Add("Not Selected")
        getnames()
        'getting length of dte
        lengthofdte = dte.Length
        'extracting position of /
        sfpos = dte.IndexOf("/") + 1
        slpos = dte.LastIndexOf("/") + 1
        'Extracting month
        month = dte.Substring(0, sfpos - 1)
        'Extracting day
        day = dte.Substring(sfpos, slpos - sfpos - 1)
        'Extracting year
        year = dte.Substring(slpos, 4)
```

```

        cboname.SelectedIndex = 0
        checkifdataalreadypresent()
        'opening this is in maximum size
        Me.WindowState = FormWindowState.Maximized
    End Sub
    Private Sub Frm_Manager_expenses_upload_Load(ByVal sender As
System.Object, ByVal e As System.EventArgs) Handles MyBase.Load
        loadfrm()
    End Sub
    'getting names and dates
    Private Sub getnamesanddate()
        'declaring connection
        Dim databaseconnection As New OleDb.OleDbConnection
        Dim database_provider As String =
"Provider=Microsoft.ace.oledb.12.0;"
        'declaring data source
        Dim Datasource As String = "Data Source =" & CurDir() &
"\database.accdb"
        databaseconnection.ConnectionString = database_provider &
Datasource
        'opening connection to database
        databaseconnection.Open()
        Dim databasecommand As New OleDb.OleDbCommand
        Dim dbreader As OleDb.OleDbDataReader
        Dim manname = cboname.SelectedItem
        databasecommand.CommandText = "SELECT * FROM Mname WHERE Emp_Name =
" & manname & """
        databasecommand.Connection = databaseconnection
        dbreader = databasecommand.ExecuteReader
        'reading details until database is reading datas
        While dbreader.Read
            eclass = dbreader.Item(4)
            Dim rowdata() As String = {year, month, day, dbreader.Item(0),
dbreader.Item(1), dbreader.Item(4), "0"}
            'adding rows to datagridview
            dghours.Rows.Add(rowdata)
        End While
        'Closing connection to database
        databaseconnection.Close()
    End Sub
    Private Sub checkifdataalreadypresent()
        dghours.Rows.Clear()
        'declaring connection
        Dim databaseconnection As New OleDb.OleDbConnection
        Dim database_provider As String =
"Provider=Microsoft.ace.oledb.12.0;"
        'declaring data source
        Dim Datasource As String = "Data Source =" & CurDir() &
"\database.accdb"
        databaseconnection.ConnectionString = database_provider &
Datasource
        'opening connection to database
        databaseconnection.Open()
        Dim databasecommand As New OleDb.OleDbCommand
        Dim dbreader As OleDb.OleDbDataReader
        Dim name As String = cboname.SelectedItem
        databasecommand.CommandText = "SELECT * FROM Others_Expenses WHERE
Emp_Name = " & name & """ & "AND HDay = " & day & " AND HMonth =" & month
& " AND HYear =" & year
        databasecommand.Connection = databaseconnection
        dbreader = databasecommand.ExecuteReader

```

```

    'set status = false
    status = False
    'reading details until database is reading datas
    While dbreader.Read
        Dim rowdata() As String = {dbreader.Item(0), dbreader.Item(1),
dbreader.Item(2), dbreader.Item(3), dbreader.Item(4), dbreader.Item(5),
dbreader.Item(6)}
        eclass = dbreader.Item(4)
        'adding rows to datagridview
        dghours.Rows.Add(rowdata)
        status = True
    End While
    If status = True Then
        lblmsg.Text = "Saved"
        lblmsg.ForeColor = Color.Green
        lblmsg.Visible = True
    Else
        status = False
        If cboname.SelectedIndex = 0 Then
            lblmsg.Visible = False
        Else
            lblmsg.Text = "Not Saved"
            lblmsg.ForeColor = Color.Red
            lblmsg.Visible = True
        End If
    End If
    'Closing connection to database
    databaseconnection.Close()
End Sub

Private Sub btnclose_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnclose.Click
    Me.Close()
End Sub

Private Sub Btnsave_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Btnsave.Click
    'save all details
    saveworkingdetails()
    'checkif data is already present in database
    checkifdataalreadypresent()
End Sub
Private Sub saveworkingdetails()
    Dim nrows As Integer
    'declaring connection
    Dim databaseconnection As New OleDb.OleDbConnection
    Dim database_provider As String =
"Provider=Microsoft.ace.oledb.12.0;"
    'declaring data source
    Dim Datasource As String = "Data Source =" & CurDir() &
"\database.accdb"
    databaseconnection.ConnectionString = database_provider &
Datasource
    'opening connection to database
    databaseconnection.Open()
    'first deleting previous data from database
    Dim dbdel As New OleDb.OleDbCommand("DELETE FROM Others_Expenses
WHERE HMonth=? AND HDay=? AND HYear = ? AND Emp_name = ?",
databaseconnection)
    dbdel.Parameters.AddWithValue("pmonth", month)

```

```

        dbdel.Parameters.AddWithValue("pday", day)
        dbdel.Parameters.AddWithValue("pyear", year)
        dbdel.Parameters.AddWithValue("pname", cboname.SelectedItem)
        dbdel.ExecuteNonQuery()
        databaseconnection.Close()
    For nrows = 0 To dghours.Rows.Count - 1
        databaseconnection.Open()
        Dim databasecommand As New OleDb.OleDbCommand
        'saving datagridview details to database
        databasecommand.CommandText = "INSERT INTO Others_Expenses
VALUES (?,?,?,?,?,?)"
        With databasecommand.Parameters
            .AddWithValue("Month", dghours.Item(0, nrows).Value)
            .AddWithValue("Day", dghours.Item(1, nrows).Value)
            .AddWithValue("year", dghours.Item(2, nrows).Value)
            .AddWithValue("E_id", dghours.Item(3, nrows).Value)
            .AddWithValue("E_name", dghours.Item(4, nrows).Value)
            .AddWithValue("EClass", dghours.Item(5, nrows).Value)
            .AddWithValue("expenses", dghours.Item(6, nrows).Value)
        End With
        databasecommand.Connection = databaseconnection
        databasecommand.ExecuteNonQuery()
        'Closing connection to database
        databaseconnection.Close()
    Next
    MsgBox(eclass & "'s Expenses are saved to the database")
End Sub
Private Sub deleteemp()
    'declaring connection
    Dim databaseconnection As New OleDb.OleDbConnection
    Dim database_provider As String =
"Provider=Microsoft.ace.oledb.12.0;"
    'declaring data source
    Dim Datasource As String = "Data Source =" & CurDir() &
"\database.accdb"
    databaseconnection.ConnectionString = database_provider &
Datasource
    'opening connection to database
    databaseconnection.Open()
    Dim dbdel As New OleDb.OleDbCommand("DELETE FROM Others_Expenses
WHERE HMonth=? AND HDay=? AND HYear = ?", databaseconnection)
    dbdel.Parameters.AddWithValue("pmonth", month)
    dbdel.Parameters.AddWithValue("pday", day)
    dbdel.Parameters.AddWithValue("pyear", year)
    dbdel.ExecuteNonQuery()
    'Closing connection to database
    databaseconnection.Close()
End Sub
Private Sub getsalary()
    'declaring connection
    Dim databaseconnection As New OleDb.OleDbConnection
    Dim database_provider As String =
"Provider=Microsoft.ace.oledb.12.0;"
    'declaring data source
    Dim Datasource As String = "Data Source =" & CurDir() &
"\database.accdb"
    databaseconnection.ConnectionString = database_provider &
Datasource
    'opening connection to database
    databaseconnection.Open()
    Dim databasecommand As New OleDb.OleDbCommand

```

```

        Dim dbreader As OleDb.OleDbDataReader
        databasecommand.CommandText = "SELECT * FROM mname WHERE
Manager_Name = '" & mname & "'"
        databasecommand.Connection = databaseconnection
        dbreader = databasecommand.ExecuteReader
        'reading database
        dbreader.Read()
        'Closing connection to database
        databaseconnection.Close()
    End Sub
    Private Sub getnames()
        'declaring connection
        Dim databaseconnection As New OleDb.OleDbConnection
        Dim database_provider As String =
"Provider=Microsoft.ace.oledb.12.0;"
        'declaring data source
        Dim Datasource As String = "Data Source =" & CurDir() &
"\database.accdb"
        databaseconnection.ConnectionString = database_provider &
Datasource
        'opening connection to database
        databaseconnection.Open()
        Dim databasecommand1 As New OleDb.OleDbCommand
        Dim dbreader1 As OleDb.OleDbDataReader
        databasecommand1.CommandText = "SELECT * FROM Mname"
        databasecommand1.Connection = databaseconnection
        dbreader1 = databasecommand1.ExecuteReader
        'reading details until database is reading datas
        While dbreader1.Read
            cboname.Items.Add(dbreader1.Item(1))
            eclass = dbreader1.Item(4)
        End While
        'Closing connection to database
        databaseconnection.Close()
    End Sub

    Private Sub cboname_SelectedIndexChanged(ByVal sender As System.Object,
 ByVal e As System.EventArgs) Handles cboname.SelectedIndexChanged
        checkifdataalreadypresent()
        If status = True Then
            'do nothing
        Else
            'get names and dates to datagridview from database
            getnamesanddate()
        End If
    End Sub
    Private Sub btnaddanotherrecord_Click(ByVal sender As System.Object,
 ByVal e As System.EventArgs) Handles btnaddanotherrecord.Click
        'getting an inputbox
        dte = InputBox("Enter Date of that day you want to Keep Expenses of
monthly paid employee", , Today)
        Main.dte = dte
        If Main.dte <> "" Then
            cboname.Items.Clear()
            loadfrm()
            Me.Show()
            'showing this form in maximum size
            Me.WindowState = FormWindowState.Maximized
        End If
    End Sub
End Class

```

### Form: Add\_Users



```
Public Class Add_Users
    'declaring public variables
    Public usernamefromdb As String
    Public passwordfromdb As String
    Public typefromdb As String
    Private Sub Add_Users_Load(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles MyBase.Load
        'selecting type with index 1
        cbotype.SelectedIndex = 1
    End Sub
    Private Sub Btnexit_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Btnexit.Click
        'closing this form
        Me.Close()
    End Sub
    Private Sub btnenter_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnenter.Click
        'clearing errorprovider
        ErrorProvider1.Clear()
        If txtusername.Text = "" Then
            'getting error in username
            ErrorProvider1.SetError(txtusername, "Please Enter User Name")
        ElseIf txtpassword.Text = "" Then
            'getting error in password
            ErrorProvider1.SetError(txtpassword, "Please Enter Password")
        Else
            'clearing errorprovider
            ErrorProvider1.Clear()
            'checking id user is already present in database
            isuserpresentindb()
        End If
    End Sub
    'saving username and password
    Private Sub saveusernameandpassword()
        'declaring connection
```

```

        Dim databaseconnection As New OleDb.OleDbConnection
        Dim database_provider As String =
"Provider=Microsoft.ace.oledb.12.0;" 
        'declaring data source
        Dim Datasource As String = "Data Source =" & CurDir() &
"\database.accdb"
        databaseconnection.ConnectionString = database_provider &
Datasource
        'opening connection to database
        databaseconnection.Open()
        Dim databasecommand As New OleDb.OleDbCommand
        databasecommand.CommandText = "INSERT INTO Userpass VALUES(?, ?, ?)"
        databasecommand.Parameters.AddWithValue("Username",
txtusername.Text)
        databasecommand.Parameters.AddWithValue("Password",
txtpassword.Text)
        databasecommand.Parameters.AddWithValue("type",
cbotype.SelectedItem)
        databasecommand.Connection = databaseconnection
        databasecommand.ExecuteNonQuery()
        'Closing connection to database
        databaseconnection.Close()
        MsgBox("New UserName and Password is added to " &
cbotype.SelectedItem)
        'clearing all details
        clearall()
    End Sub
    'checking if user is already in database
    Private Sub isuserpresentindb()
        'declaring connection
        Dim databaseconnection As New OleDb.OleDbConnection
        Dim database_provider As String =
"Provider=Microsoft.ace.oledb.12.0;" 
        'declaring data source
        Dim Datasource As String = "Data Source =" & CurDir() &
"\database.accdb"
        databaseconnection.ConnectionString = database_provider &
Datasource
        'opening connection to database
        databaseconnection.Open()
        Dim databasecommand As New OleDb.OleDbCommand
        Dim dbreader As OleDb.OleDbDataReader
        databasecommand.CommandText = "SELECT * FROM Userpass WHERE
UserName = " & "****" & txtusername.Text & "****"
        databasecommand.Connection = databaseconnection
        dbreader = databasecommand.ExecuteReader
        While dbreader.Read()
            usernamefromdb = dbreader.Item(0)
        End While
        'Closing connection to database
        databaseconnection.Close()
        If txtusername.Text = usernamefromdb Then
            'messaging username already present try another one
            ErrorProvider1.SetError(txtusername, "Username already present
try another username")
        Else
            'save all details
            saveusernameandpassword()
        End If
    End Sub
    'clearin all details

```

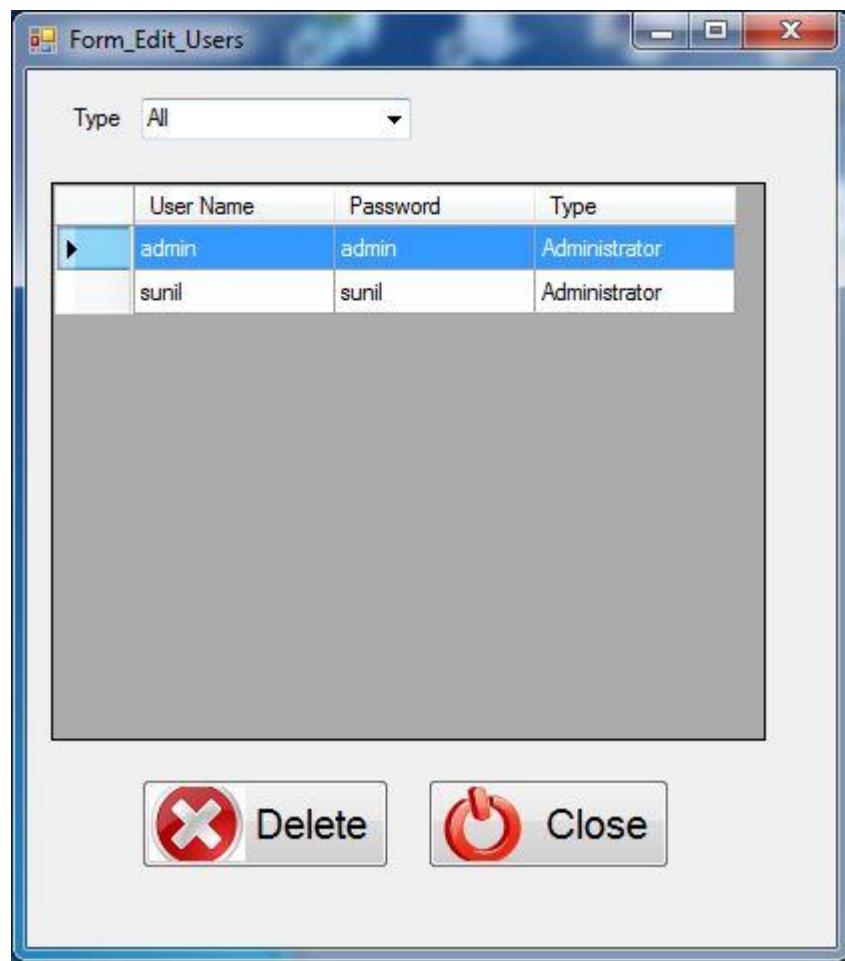
```
Private Sub clearall()
    txtpassword.Clear()
    txtusername.Clear()
    cbotype.SelectedIndex = 1
End Sub

Private Sub txtusername_KeyPress(ByVal sender As Object, ByVal e As System.Windows.Forms.KeyPressEventArgs) Handles txtusername.KeyPress
    Select Case Asc(e.KeyChar)
        Case 8, 32, 65 To 90, 97 To 122
            e.Handled = False
        Case Else
            e.Handled = True
    End Select
End Sub

Private Sub txtusername_TextChanged(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles txtusername.TextChanged
    txtusername.MaxLength = 10
End Sub

Private Sub txtpassword_TextChanged(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles txtpassword.TextChanged
    txtpassword.MaxLength = 10
End Sub
End Class
```

**Form: Form\_edit\_Users**



```
Public Class Form_Edit_Users
    Public type As String
    Public cname As String
    Private Sub Button3_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnclose.Click
        'closing this form
        Me.Close()
    End Sub

    Private Sub btnedit_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)
        'show another form
        Change_User_Password.Show()
    End Sub

    Private Sub Form_Edit_Users_Load(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles MyBase.Load
        'selecting type having index 0
        cbotype.SelectedIndex = 0
        'getting datas from database
        getdatasfromdatabase()
    End Sub
    Private Sub deleteusers()
```

```

'declaring connection
Dim databaseconnection As New OleDb.OleDbConnection
Dim database_provider As String =
"Provider=Microsoft.ace.oledb.12.0;"
'declaring data source
Dim Datasource As String = "Data Source =" & CurDir() &
"\database.accdb"
databaseconnection.ConnectionString = database_provider &
Datasource
'opening connection to database
databaseconnection.Open()
'Send delete command
Dim databasecommand As New OleDb.OleDbCommand
databasecommand.CommandText = "DELETE FROM Userpass WHERE UserName
= " & """" & cname & """
databasecommand.Connection = databaseconnection
databasecommand.ExecuteNonQuery()
'Closing connection to database
databaseconnection.Close()
MsgBox("User having username " & cname & " is deleted from
database")
DataGridView1.Rows.Clear()
getdatasfromdatabase()
End Sub
Private Sub getdatasfromdatabase()
DataGridView1.Refresh()
DataGridView1.Rows.Clear()
type = cbotype.SelectedItem
'declaring connection
Dim databaseconnection As New OleDb.OleDbConnection
Dim database_provider As String =
"Provider=Microsoft.ace.oledb.12.0;"
'declaring data source
Dim Datasource As String = "Data Source =" & CurDir() &
"\database.accdb"
databaseconnection.ConnectionString = database_provider &
Datasource
'opening connection to database
databaseconnection.Open()
Dim databasecommand As New OleDb.OleDbCommand
Dim dbreader As OleDb.OleDbDataReader
If type = "All" Then
    databasecommand.CommandText = "SELECT * FROM Userpass"
ElseIf type = "Administrator" Then
    databasecommand.CommandText = "SELECT * FROM Userpass WHERE
Type = " & """Administrator"""
ElseIf type = "Other Users" Then
    databasecommand.CommandText = "SELECT * FROM Userpass WHERE
Type = " & """Other Users"""
End If
databasecommand.Connection = databaseconnection
dbreader = databasecommand.ExecuteReader
While dbreader.Read
    Dim rowdata() As String = {dbreader.Item(0), dbreader.Item(1),
dbreader.Item(2)}
    'adding datagrid rows
    DataGridView1.Rows.Add(rowdata)
End While
'Closing connection to database
databaseconnection.Close()
End Sub

```

```

    Private Sub cbotype_SelectedIndexChanged(ByVal sender As System.Object,
    ByVal e As System.EventArgs) Handles cbotype.SelectedIndexChanged
        'getting datas from database
        getdatasfromdatabase()
    End Sub
    Private Sub DataGridView1_CellClick(ByVal sender As Object, ByVal e As
System.Windows.Forms.DataGridViewCellEventArgs) Handles
DataGridView1.CellClick
        'getting id of that row which is clicked
        clname = DataGridView1.Item(0,
        DataGridView1.CurrentRow.Index).Value.ToString
    End Sub
    Private Sub btndelete_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btndelete.Click
        If DataGridView1.CurrentCell.Value Is Nothing Or
DataGridView1.CurrentCell.Value.ToString = "" Then
            MsgBox("You have nothing to delete")
        Else
            'deleting users
            deleteusers()
        End If
    End Sub
    Private Sub DataGridView1_CellEnter(ByVal sender As Object, ByVal e As
System.Windows.Forms.DataGridViewCellEventArgs) Handles
DataGridView1.CellEnter
        'getting id of that row which is clicked
        clname = DataGridView1.Item(0,
        DataGridView1.CurrentRow.Index).Value.ToString
    End Sub
End Class

```

**Form: Change user password**



```
Public Class Change_User_Password
    'declaring all public variables
    Public oldpasswordfromdb As String
    Public oldusernamefromdb
    Public typefromdb As String
    'clearing all textboxes
    Private Sub cleartext()
        txtnewpassword.Clear()
        txtoldpassword.Clear()
        txtusername.Clear()
    End Sub
    Private Sub Btnsave_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Btnsave.Click
        'Clearing error provider
        ErrorProvider1.Clear()
        If txtusername.Text = "" Then
            'get error message
            ErrorProvider1.SetError(txtusername, "Please Enter User Name")
        ElseIf txtoldpassword.Text = "" Then
            'get error message
            ErrorProvider1.SetError(txtoldpassword, "Old Password cannot be
empty")
        ElseIf txtnewpassword.Text = "" Then
            'get error message
            ErrorProvider1.SetError(txtnewpassword, "New Password cannot be
empty")
        ElseIf txtoldpassword.Text = txtnewpassword.Text Then
            'get error message
            ErrorProvider1.SetError(txtnewpassword, "New Password cannot be
same as old password")
        End If
    End Sub
End Class
```

```

    Else
        'Clearing error provider
        ErrorProvider1.Clear()
        checkpassword()
    End If
End Sub
'getting old username
Private Sub getoldusername()
    'declaring connection
    Dim databaseconnection As New OleDb.OleDbConnection
    Dim database_provider As String =
"Provider=Microsoft.ace.oledb.12.0;"
    'declaring data source
    Dim Datasource As String = "Data Source =" & CurDir() &
"\database.accdb"
    databaseconnection.ConnectionString = database_provider &
Datasource
    'opening connection to database
    databaseconnection.Open()
    Dim databasecommand As New OleDb.OleDbCommand
    Dim dbreader As OleDb.OleDbDataReader
    databasecommand.CommandText = "SELECT * FROM Userpass WHERE
UserName = " & "****" & Log_In.txtusername.Text & "****"
    databasecommand.Connection = databaseconnection
    dbreader = databasecommand.ExecuteReader
    While dbreader.Read()
        oldusernamefromdb = dbreader.Item(0)
        txtusername.Text = oldusernamefromdb
        oldpasswordfromdb = dbreader.Item(1)
        typefromdb = dbreader.Item(2)
    End While
    'Closing connection to database
    databaseconnection.Close()
End Sub
Private Sub getusername()
    'declaring connection
    Dim databaseconnection As New OleDb.OleDbConnection
    Dim database_provider As String =
"Provider=Microsoft.ace.oledb.12.0;"
    'declaring data source
    Dim Datasource As String = "Data Source =" & CurDir() &
"\database.accdb"
    databaseconnection.ConnectionString = database_provider &
Datasource
    'opening connection to database
    databaseconnection.Open()
    Dim databasecommand As New OleDb.OleDbCommand
    Dim dbreader As OleDb.OleDbDataReader
    databasecommand.CommandText = "SELECT * FROM Userpass WHERE
UserName = " & "****" & txtusername.Text & "****"
    databasecommand.Connection = databaseconnection
    dbreader = databasecommand.ExecuteReader
    While dbreader.Read()
        oldusernamefromdb = dbreader.Item(0)
        oldpasswordfromdb = dbreader.Item(1)
        typefromdb = dbreader.Item(2)
    End While
    'Closing connection to database
    databaseconnection.Close()
End Sub
Private Sub deletealldatasfromdb()

```

```

'declaring connection
Dim databaseconnection As New OleDb.OleDbConnection
Dim database_provider As String =
"Provider=Microsoft.ace.oledb.12.0;"
'declaring data source
Dim Datasource As String = "Data Source =" & CurDir() &
"\database.accdb"
databaseconnection.ConnectionString = database_provider &
Datasource
'opening connection to database
databaseconnection.Open()
Dim databasecommand As New OleDb.OleDbCommand
Dim dbreader As OleDb.OleDbDataReader
databasecommand.CommandText = "DELETE * FROM Userpass WHERE
UserName = " & "****" & txtusername.Text & "****"
databasecommand.Connection = databaseconnection
dbreader = databasecommand.ExecuteReader
'Closing connection to database
databaseconnection.Close()

End Sub
Private Sub saveusernameandpassword()
'declaring connection
Dim databaseconnection As New OleDb.OleDbConnection
Dim database_provider As String =
"Provider=Microsoft.ace.oledb.12.0;"
'declaring data source
Dim Datasource As String = "Data Source =" & CurDir() &
"\database.accdb"
databaseconnection.ConnectionString = database_provider &
Datasource
'opening connection to database
databaseconnection.Open()
Dim databasecommand As New OleDb.OleDbCommand
databasecommand.CommandText = "INSERT INTO Userpass VALUES(?, ?, ?)"
databasecommand.Parameters.AddWithValue("Username",
txtusername.Text)
databasecommand.Parameters.AddWithValue("Password",
txtnewpassword.Text)
databasecommand.Parameters.AddWithValue("type", typefromdb)
databasecommand.Connection = databaseconnection
databasecommand.ExecuteNonQuery()
'Closing connection to database
databaseconnection.Close()
MsgBox("New UserName and Password is added ")
End Sub
Private Sub checkpassword()
getusername()
If txtoldpassword.Text = oldpasswordfromdb And txtusername.Text =
oldusernamefromdb Then
    If MessageBox.Show("Are you sure you want to change username or
password ?", "WARNING", MessageBoxButtons.YesNo, MessageBoxIcon.Warning) =
Windows.Forms.DialogResult.Yes Then
        'change username and password
        'delete old data of password from database
        deletealldatasfromdb()
        'saving new username and password
        saveusernameandpassword()
        'clearing textboxes
        cleartext()
    Else
        'Dont change
    End If
End If
End Sub

```

```

        End If
    ElseIf txtoldpassword.Text <> oldpasswordfromdb And
txtusername.Text <> oldusernamefromdb Then
        'get error message
        ErrorProvider1.SetError(txtnewpassword, "Sorry your Username
and Old Password is incorrect")
    ElseIf txtusername.Text <> oldusernamefromdb Then
        'get error message
        ErrorProvider1.SetError(txtusername, "Username is not correct")
    ElseIf txtoldpassword.Text <> oldpasswordfromdb Then
        'get error message
        ErrorProvider1.SetError(txtoldpassword, "Old Password is not
correct")
    End If
End Sub

Private Sub Btnexit_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Btnexit.Click
    'closing this form
    Me.Close()
End Sub

Private Sub Change_User_Password_Load(ByVal sender As System.Object,
ByVal e As System.EventArgs) Handles MyBase.Load
    'getting old username
    getoldusername()
End Sub

Private Sub txtoldpassword_TextChanged(ByVal sender As System.Object,
ByVal e As System.EventArgs) Handles txtoldpassword.TextChanged
    txtoldpassword.MaxLength = 10
End Sub

Private Sub txtnewpassword_TextChanged(ByVal sender As System.Object,
ByVal e As System.EventArgs) Handles txtnewpassword.TextChanged
    txtnewpassword.MaxLength = 10
End Sub

Private Sub txtusername_KeyPress(ByVal sender As Object, ByVal e As
System.Windows.Forms.KeyPressEventArgs) Handles txtusername.KeyPress
    Select Case Asc(e.KeyChar)
        Case 8, 32, 65 To 90, 97 To 122
            e.Handled = False
        Case Else
            e.Handled = True
    End Select
End Sub

Private Sub txtusername_TextChanged(ByVal sender As System.Object,
ByVal e As System.EventArgs) Handles txtusername.TextChanged
    txtusername.MaxLength = 10
End Sub
End Class

```

## **TESTING**

### **Testing Plan and Evidence**

#### **Validation Test**

**Test Number: 1**

**Test Objective:** Not to allow date of birth as today

**Test Condition:** Selecting date of birth as today

**Expected Outcome:** An error should be shown by 'error provider'.

**Actual Outcome in picture :**

The screenshot shows the 'ADD EMPLOYEE' window with the title 'ADD Employee's Details'. The window contains various input fields and a placeholder image area. The fields include:

- Employee ID: 5
- Name: Rahul
- Gender: Male
- Class: Carpenter
- Type: Head
- Rate: 75 Per Hour
- Temporary Address: [empty]
- Permanent Address: [empty]
- Mobile No: [empty]
- Citizenship No: [empty]
- Licence No: [empty]
- Date OF Birth: Friday, March 28, 2014 (with a red error icon)
- Date Hired: Friday, March 28, 2014

On the right side, there is a placeholder image box containing a dark silhouette of a person with the text 'No Photo Available'. Below this box is a 'Select Image' button with a magnifying glass icon. At the bottom of the window are three buttons: 'Save' (with a save icon), 'Clear' (with a brush icon), and 'Close' (with a close icon).

**Remark:** Test was successful.

#### **Validation Test**

**Test Number: 2**

**Test Objective:** Not to type invalid characters

**Test Condition:** Try to type invalid characters. like alphabets in rate or salary

**Expected Outcome:** Those characters should not be typed

**Actual Outcome :** It did not typed any invalid characters.

**Note:** Since it cannot be shown in picture. It is only in code of program.

**Remark:** Test was successful.

**Note:** Similarly, this validation was also checked with other forms and other textboxes which worked well.

## **Validation Test**

**Test Number:** 3

**Test Objective:** Not to allow old password and new password to be same.

**Test Condition:** Both old password and new password was typed same.

**Expected Outcome:** An error should be shown by 'error provider'.

**Actual Outcome:**



**Remark:** Test was successful.

## **Field Size Test**

**Test Number: 4**

**Test Objective:** : Not to allow data having size greater than the specified field size.

**Test Condition:** Trying to type more characters than its field size

**Expected Outcome:** More characters should not be typed.

**Actual Outcome in picture :** It did not typed any characters more than field size.

**Note:** Since it cannot be shown in picture. It is only in code of program.

**Remark:** Test was successful.

**Note:** Similarly, this test was also done with other forms and other textboxes which worked fine.

## **Testing Plan and Evidence**

### **Null Value Test**

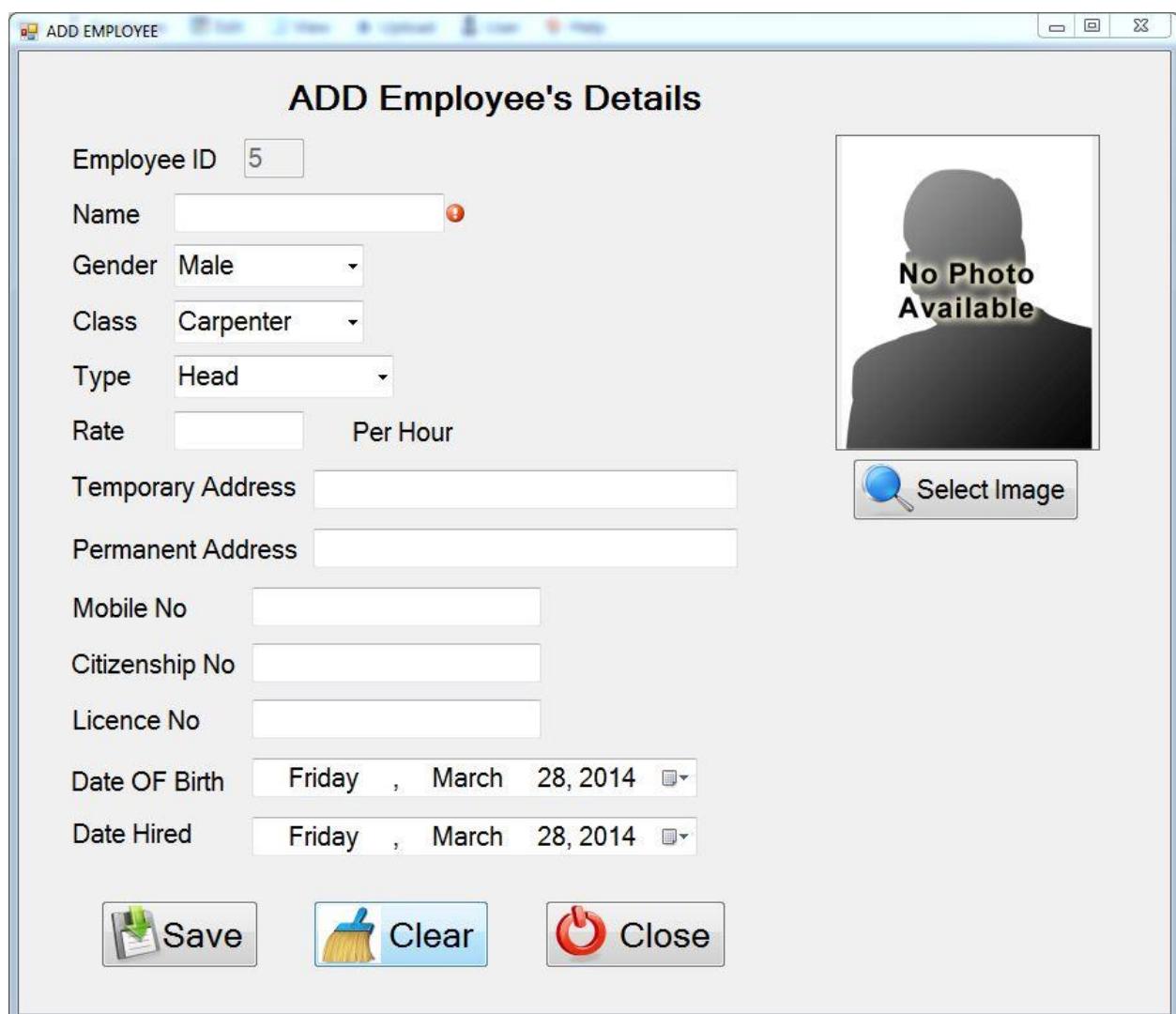
**Test Number:** 5

**Test Objective:** Not to allow empty textbox

**Test Condition:** Leaving textbox empty

**Expected Outcome:** An error should be shown by 'error provider'.

**Actual Outcome in picture :**



**ADD Employee's Details**

Employee ID	5	
Name	Rahul	
Gender	Male	
Class	Carpenter	
Type	Head	
Rate	<input type="text"/> Per Hour	
Temporary Address		
Permanent Address		
Mobile No	<input type="text"/>	
Citizenship No	<input type="text"/>	
Licence No	<input type="text"/>	
Date OF Birth	Friday , March 28, 2014	
Date Hired	Friday , March 28, 2014	

Save   Clear   Close

**Common\_working\_hour**

Start Time	<input type="text"/> AM
End Time	20 PM

Save   Close

**Remark:** Test was successful.

**Note:** Similarly, this validation was also checked with other forms and other textboxes which worked well.

## **Functional Testing: Adding Employee**

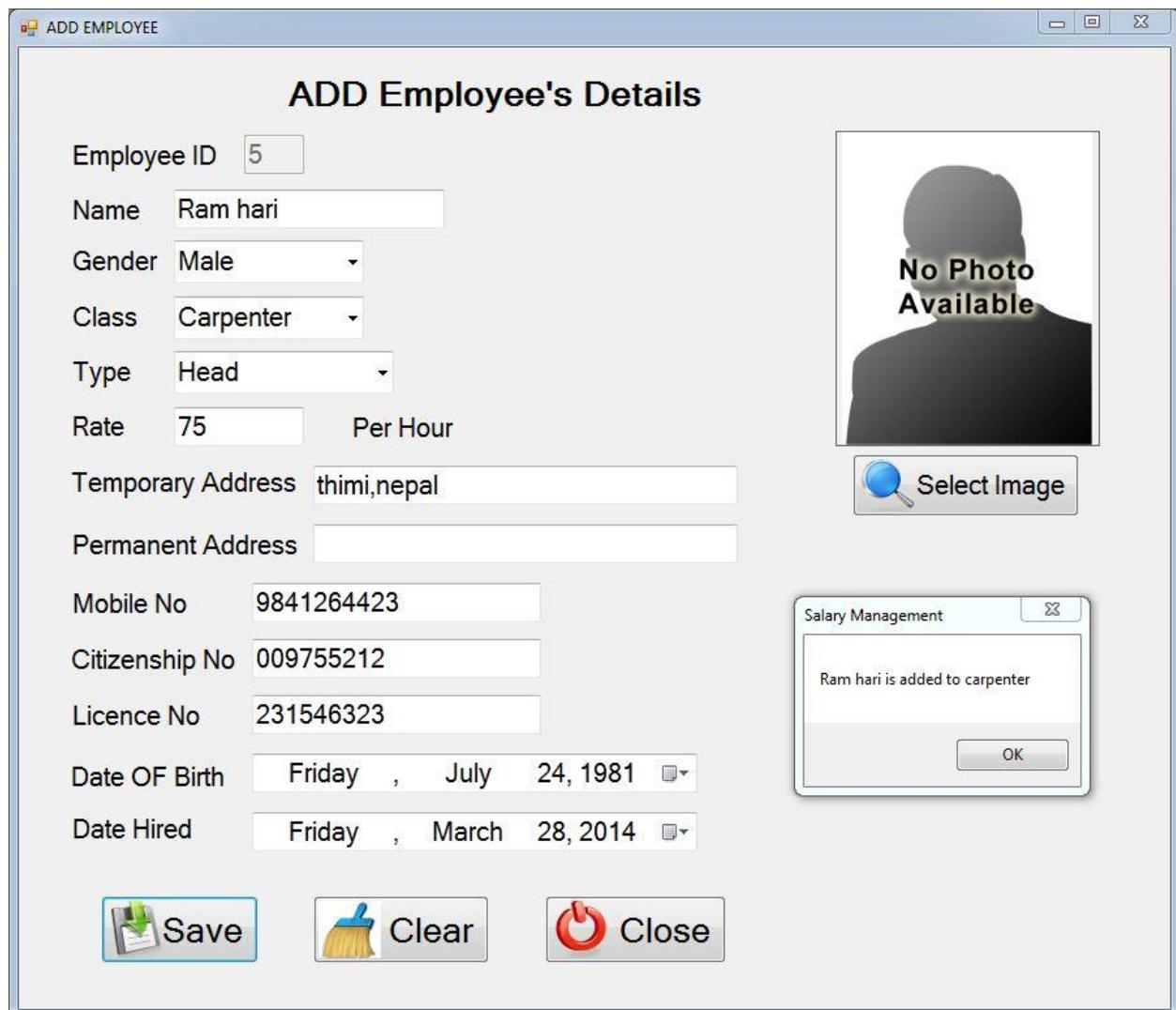
**Test Number:** 6

**Test Objective:** To add an employees of selected class

**Test Condition:** Add all details and click save.

**Expected Outcome:** A message box should be shown with employee' name that is added.

**Actual Outcome:**



**Remark:** Test is successful.

## **Functional Testing: Viewing All Employee**

**Test Number: 7**

**Test Objective: To list all the employees of selected class**

**Test Condition: Select carpenter**

**Expected Outcome: List all the details of carpenter.**

**Actual Outcome:**

Employee Details													
Select Class		Carpenter											
		Print	ADD Employee	Show Details	EDIT Employee	DELETE Employee							
1	EMP_ID	EMP_Name	Gender	Type	Rate	Class	Temporary_Address	Permanent_Address	Mobile_No	Citizenship_No	License_No	Date_OF_Birth	Date_Hired
2		Indramoni Bhujel	Male	Head	75	Carpenter	Lokanthal, bhakt...	Bhotsipa, Sindhuachowk,	984159426	00693006845	489475221	6/13/1974	3/1/2014
3		Shiva	Male	Designer(Head)	85	Carpenter	Bhaktapur,nep...		9813459865	00945689754	990085742	7/28/1979	3/1/2014
4		Sharukh	Male	Designer(Head)	85	Carpenter	bhaktapur,nep...		9851075426	09421100695	09840485	12/10/1979	3/1/2014
		Mine	Male	Head	75	Carpenter	Lokanthal,bhakt...		9841657812	009452160095		12/18/1980	3/1/2014

Employee Details													
Select Class		Carpenter											
		Print	ADD Employee	Show Details	EDIT Employee	DELETE Employee							
1	Emp_ID	Carpenter_Name	Gender	EType	ERate	EClass	Temporary_	Permanent_	Mobile_no	Citizenship_	licence_No	Date_of_Bir	Date_Hired
2		Indramoni Bhujel	Male	Head	75.00	Carpenter	Lokanthal, bha	Bhotsipa, Sindhi	9841589426	00693006845	489475221	6/13/1974	3/1/2014
3		Shiva	Male	Designer(Head)	85.00	Carpenter	Bhaktapur,nep		9813459865	00945689754	990085742	7/28/1979	3/1/2014
4		Sharukh	Male	Designer(Head)	85.00	Carpenter	bhaktapur,nep		9851075426	09421100695	09840485	12/10/1979	3/1/2014
*		Mine	Male	Head	75.00	Carpenter	Lokanthal,bha		9841657812	009452160095		12/18/1980	3/1/2014

**Remark: Test is successful.**

## ***Functional Testing: Adding common working hours***

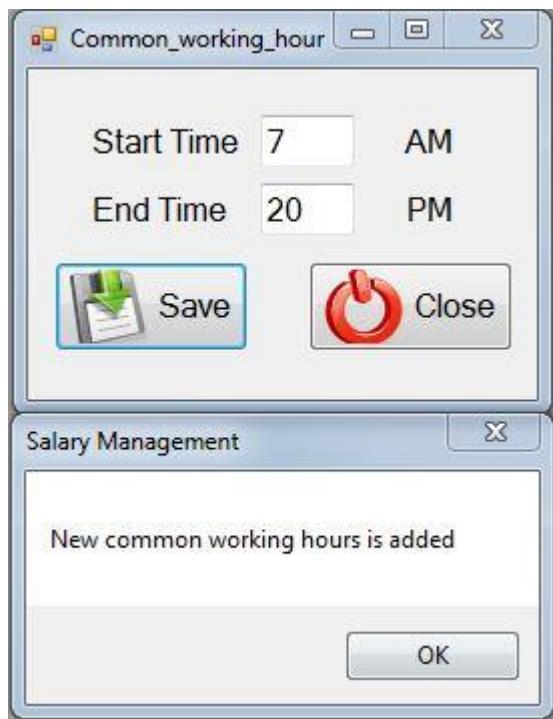
**Test Number:** 8

**Test Objective:** To add new common working hours

**Test Condition:** New details are added and clicked on save button

**Expected Outcome:** A message box should be shown 'common working hours is added'.

**Actual Outcome:**



**Remark:** Test is successful.

## **Functional Testing: Editing salary**

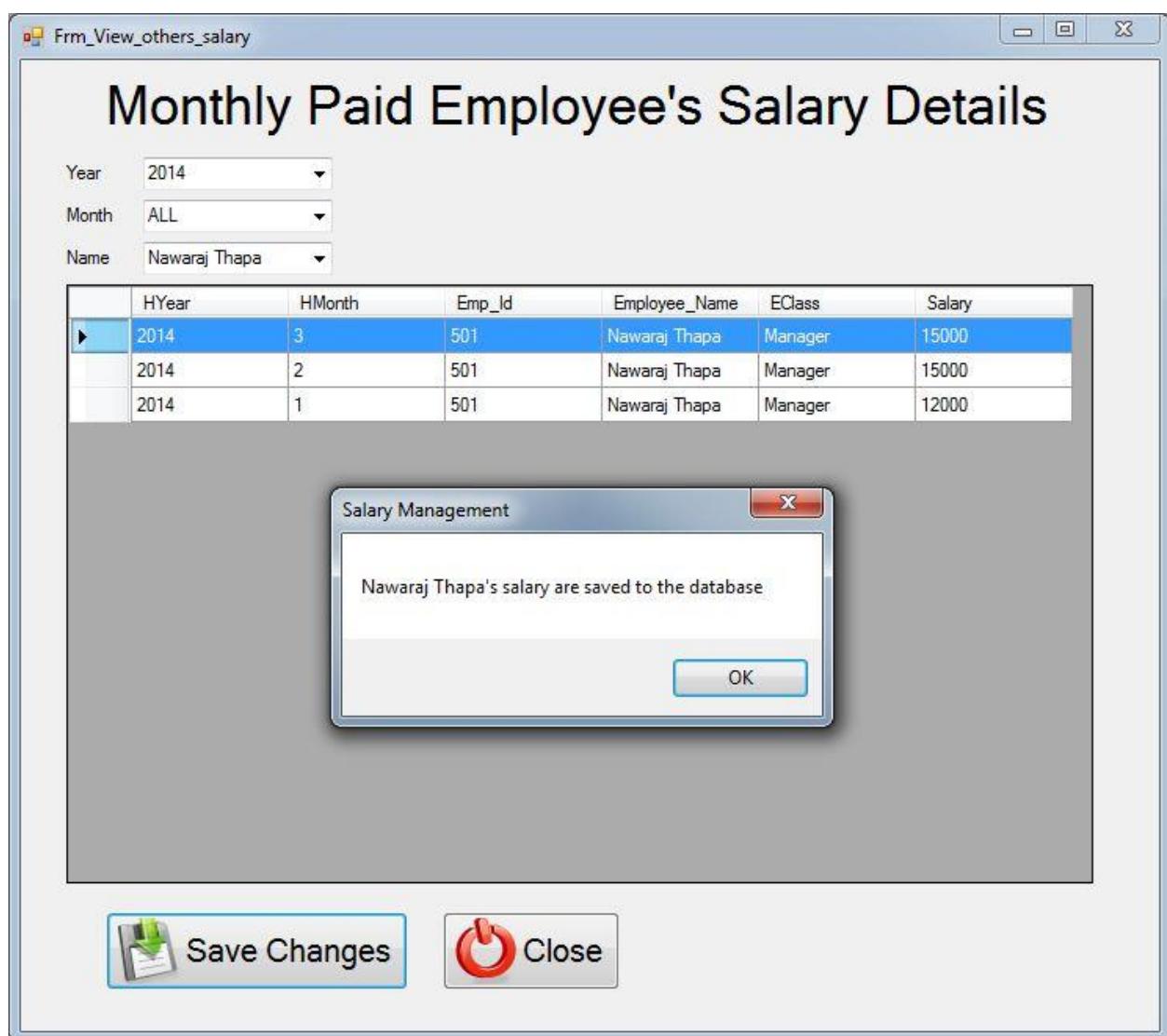
**Test Number:** 9

**Test Objective:** To edit salary of monthly paid employee

**Test Condition:** Salary is changed and saved.

**Expected Outcome:** A message box should be shown salary is added.

**Actual Outcome:**



Others_Salary					
HYear	HMonth	Emp_ID	Emp_Name	Class	Salary
2014	1	501	Nawaraj Thapa	Manager	12000
2014	2	501	Nawaraj Thapa	Manager	15000
2014	3	501	Nawaraj Thapa	Manager	15000
2014	1	502	Ritte Rijal	Driver	15000
2014	2	502	Ritte Rijal	Driver	15000
2014	3	502	Ritte Rijal	Driver	15000
*					

**Remark:** Test is successful.

## **Functional Testing: Uploading Working hours**

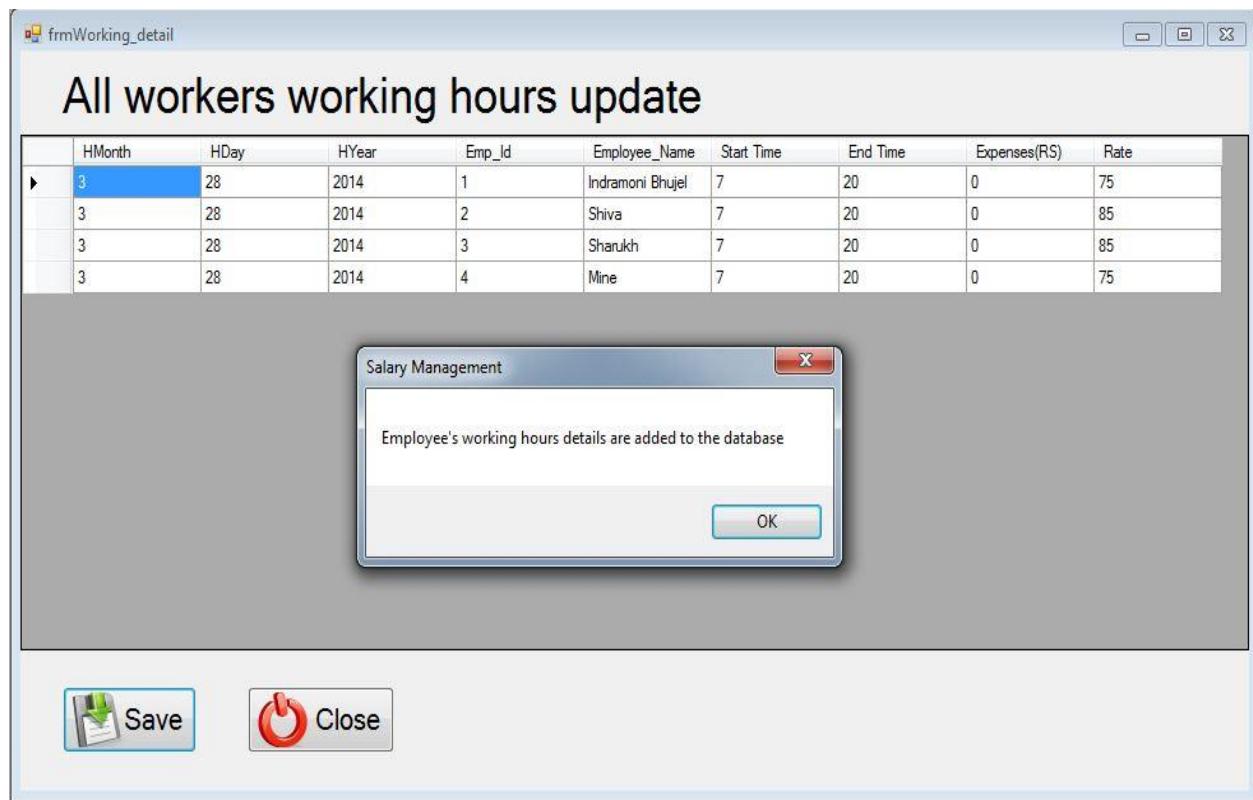
**Test Number: 10**

**Test Objective:** To upload working hours of an employee

**Test Condition:** Working hours and expenses are changed and saved.

**Expected Outcome:** A message box should be shown 'Working details are saved'.

**Actual Outcome:**



**Remark:** Test is successful.

## **Functional Testing: Uploading salary**

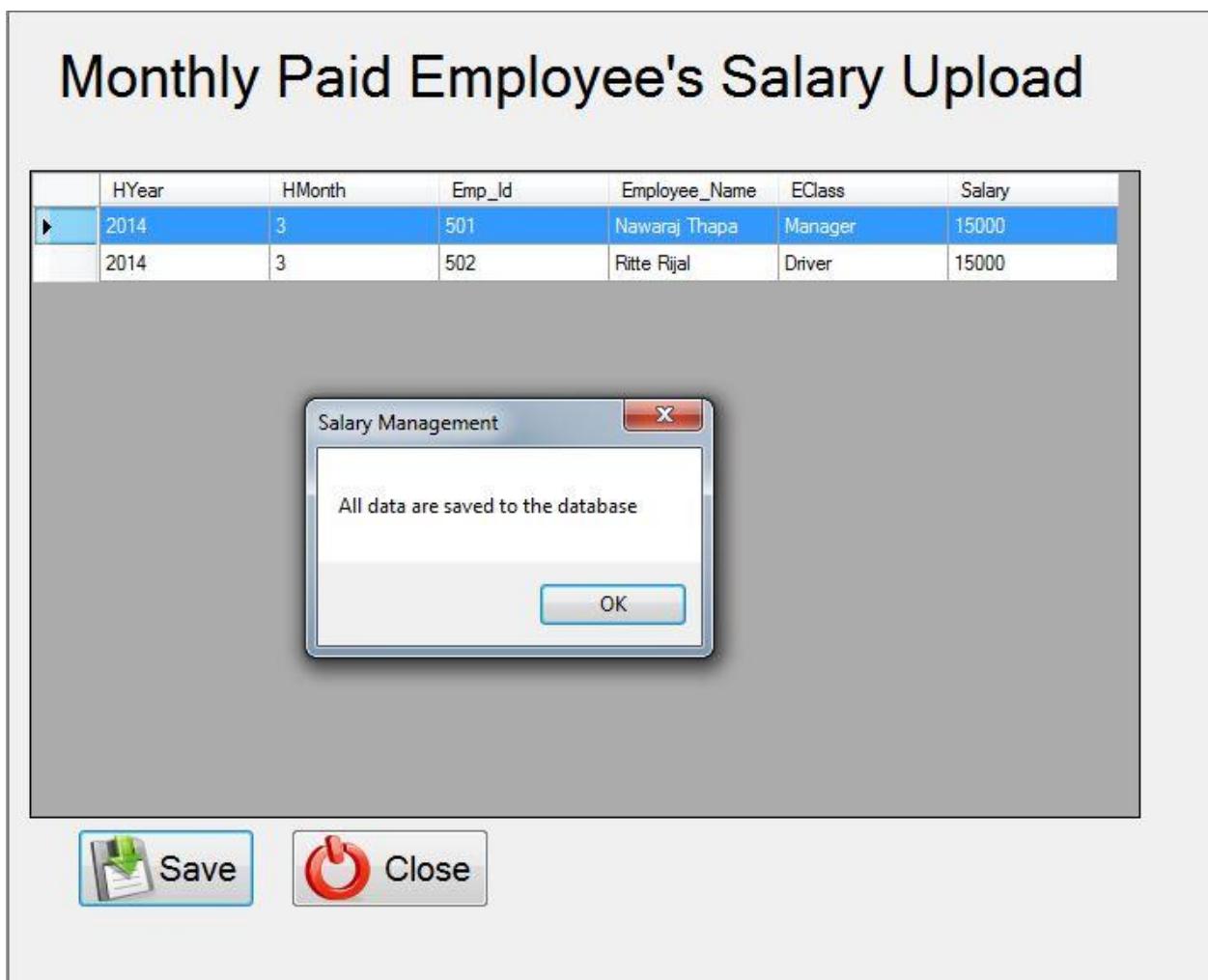
**Test Number:** 11

**Test Objective:** To upload or save salary of monthly paid employee

**Test Condition:** Salary is saved.

**Expected Outcome:** A message box should be shown 'salary is added'.

**Actual Outcome:**



**Remark:** Test is successful.

## **Functional Testing: Uploading Allowance**

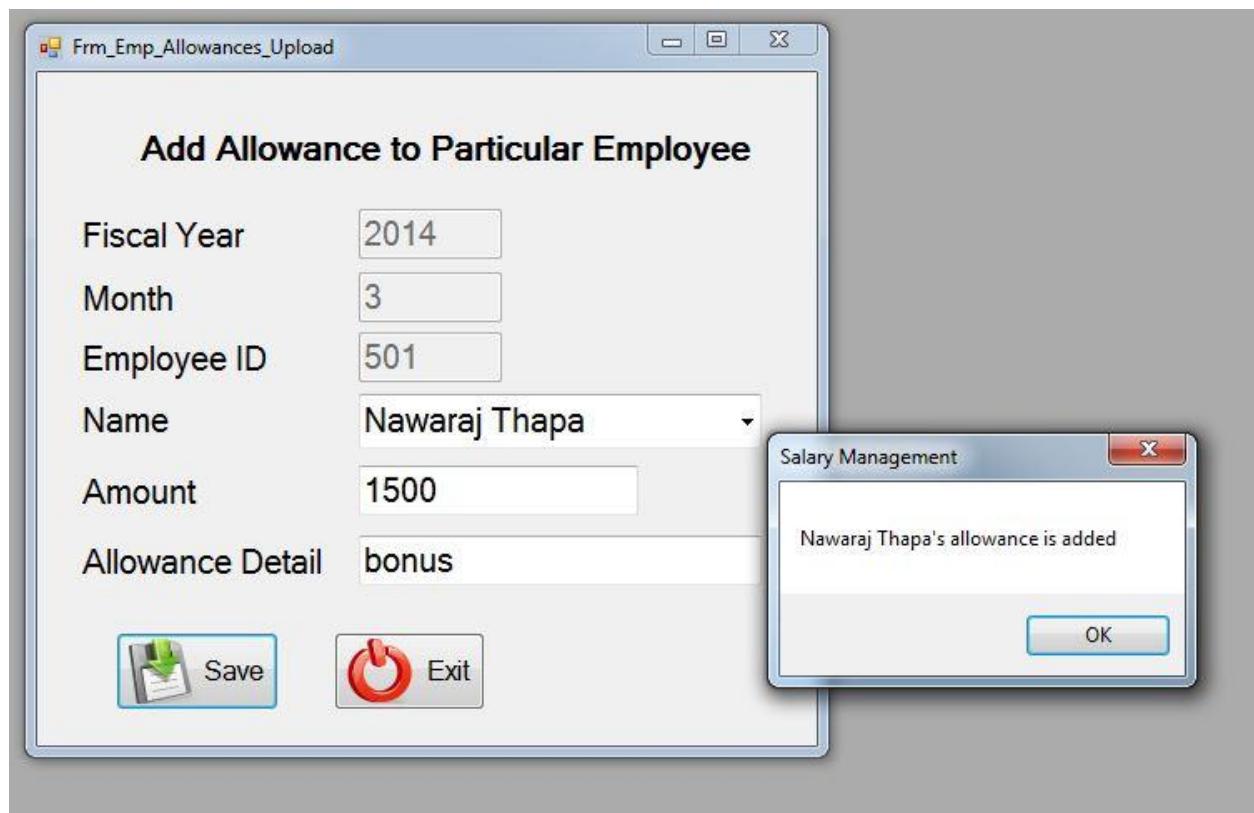
**Test Number:** 12

**Test Objective:** To save allowance of monthly paid employee

**Test Condition:** Allowance details are filled and saved.

**Expected Outcome:** A message box should be shown 'Allowance is added'.

**Actual Outcome:**



**Remark:** Test is successful.

## **Functional Testing: Uploading Expenses**

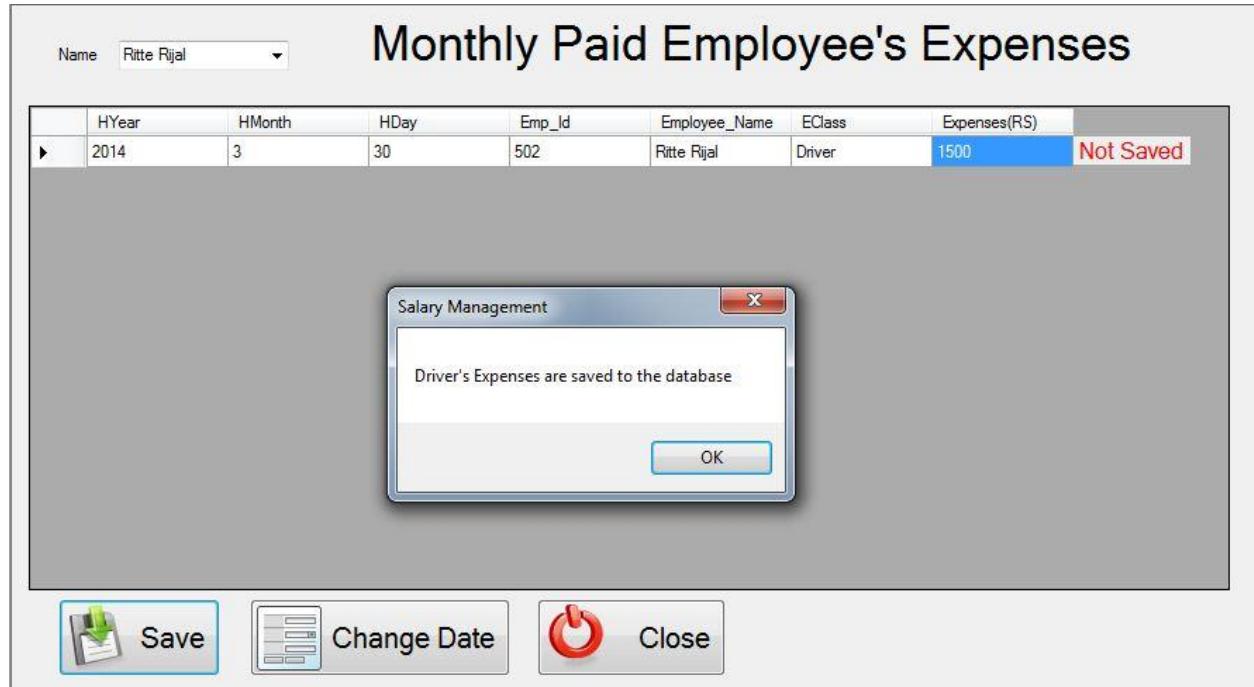
**Test Number:** 13

**Test Objective:** To save expenses of monthly paid employee to that day

**Test Condition:** Expenses are filled and saved.

**Expected Outcome:** A message box should be shown 'Expense is added'.

**Actual Outcome:**



**Remark:** Test is successful.

## **Functional Testing: Calculating total income of an employee**

**Test Number:** 14

**Test Objective:** To calculate income of an employee

**Test Condition:** Employee's income details is viewed.

**Expected Outcome:** All calculation is to be done and total is shown in datagridview.

**Actual Outcome:**

### Monthly Paid Employee's Income Details

	HYear	HMonth	Emp_Id	Employee_Name	EClass	Salary	Allowance	Expenses(RS)
▶	2014	3	502	Ritte Rijal	Driver	15000	2500	25000
	2014	2	502	Ritte Rijal	Driver	15000	0	0
	2014	1	502	Ritte Rijal	Driver	15000	0	0
	2014	3	501	Nawaraj Thapa	Manager	15000	2000	15000
	2014	2	501	Nawaraj Thapa	Manager	15000	0	0
	2014	1	501	Nawaraj Thapa	Manager	12000	0	0
					Total	87000	4500	40000
					Total Income	91500		
					Net Income	51500		

Basic Salary      **87000**      Total Expenses      **40000**  
Total Allowance    **4500**      Net Income          **51500**  
Total Income        **91500**

 Print     Close

Year

Month

Name

## Carpenter working hours Details

HMonth	HDay	HYear	Emp_Id	Employee_Name	Start Time	End Time	Expenses(RS)	Rate	Worked Hours	Income
3	12	2014	4	Mine	7	20	0	75	13	975
3	13	2014	4	Mine	7	20	0	75	13	975
3	14	2014	4	Mine	7	20	0	75	13	975
3	15	2014	4	Mine	7	20	0	75	13	975
3	16	2014	4	Mine	7	20	0	75	13	975
3	17	2014	4	Mine	7	20	0	75	13	975
3	18	2014	4	Mine	7	20	20000	75	13	975
3	19	2014	4	Mine	7	20	0	75	13	975
3	20	2014	4	Mine	7	20	0	75	13	975
3	21	2014	4	Mine	7	20	0	75	13	975
3	22	2014	4	Mine	7	20	0	75	13	975
3	23	2014	4	Mine	7	20	0	75	13	975
3	24	2014	4	Mine	7	20	0	75	13	975
3	25	2014	4	Mine	7	20	0	75	13	975
3	26	2014	4	Mine	7	20	0	75	13	975
3	27	2014	4	Mine	7	20	0	75	13	975
3	29	2014	4	Mine	7	20	0	75	13	975
3	30	2014	4	Mine	7	20	0	75	13	975
3	31	2014	4	Mine	7	20	9250	75	13	975
3	28	2014	4	Mine	7	20	0	75	13	975
						Total	29250		403	30225
						Total Income	30225			
						Net Income	975			

Total Hours Worked **403** Total Expenses **29250**  
 Total Income **30225** Net Income **975**



**Remark: Test is successful.**

## **Functional Testing: Adding new user**

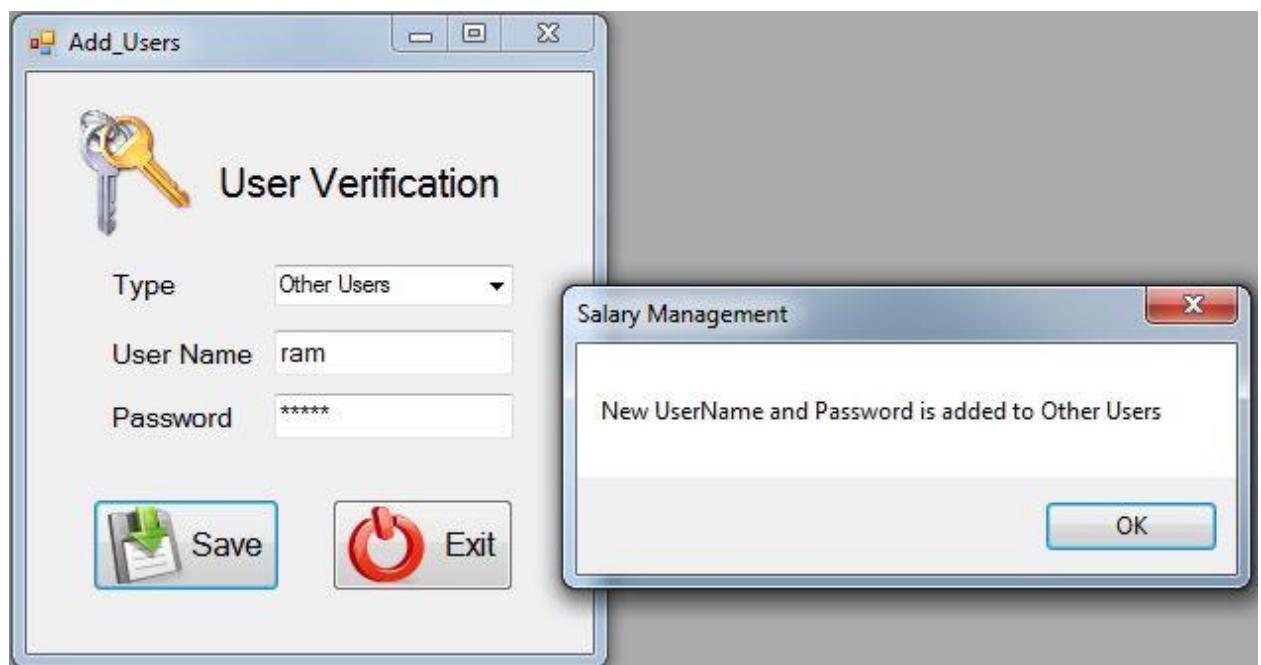
**Test Number:** 15

**Test Objective:** To Add new user to use this application

**Test Condition:** All details are filled and saved

**Expected Outcome:** A message box should be shown 'New user is added'.

**Actual Outcome:**



**Remark:** Test is successful.

## **Functional Testing: Changing Username or password**

**Test Number: 16**

**Test Objective:** To edit username or password of a user

**Test Condition:** Username or Password is changed and saved.

**Expected Outcome:** A message box should be shown 'New username and password is added'.

**Actual Outcome:**



**Remark:** Test is successful.

## **Functional Testing: Printing Report**

**Test Number: 17**

**Test Objective:** To print working Details of employee of march of Shiva.

**Test Condition:** Print button is pressed.

**Expected Outcome:** It should print these details to paper.

**Actual Outcome:**

frmShow_working_details											3/28/2014 6:50:00 PM   Page 1
HMonth	HDay	HYear	Emp_Id	Employee_Name	Start Time	End Time	Expenses(RS)	Rate	Worked Hours	Income	
3	1	2014	2	Shiva	7	20	0	85	13	1105	
3	2	2014	2	Shiva	7	20	0	85	13	1105	
3	3	2014	2	Shiva	7	20	0	85	13	1105	
3	4	2014	2	Shiva	7	20	0	85	13	1105	
3	5	2014	2	Shiva	7	20	0	85	13	1105	
3	6	2014	2	Shiva	7	20	0	85	13	1105	
3	7	2014	2	Shiva	7	20	0	85	13	1105	
3	8	2014	2	Shiva	7	20	0	85	13	1105	
3	9	2014	2	Shiva	7	20	0	85	13	1105	
3	10	2014	2	Shiva	7	20	0	85	13	1105	
3	11	2014	2	Shiva	7	20	0	85	13	1105	
3	12	2014	2	Shiva	7	20	0	85	13	1105	
3	13	2014	2	Shiva	7	20	0	85	13	1105	
3	14	2014	2	Shiva	7	20	0	85	13	1105	
3	15	2014	2	Shiva	7	20	15000	85	13	1105	
3	16	2014	2	Shiva	7	20	0	85	13	1105	
3	17	2014	2	Shiva	7	20	0	85	13	1105	
3	18	2014	2	Shiva	7	20	0	85	13	1105	
3	19	2014	2	Shiva	7	20	0	85	13	1105	
3	20	2014	2	Shiva	7	20	0	85	13	1105	
3	21	2014	2	Shiva	7	20	0	85	13	1105	
3	22	2014	2	Shiva	7	20	0	85	13	1105	
3	23	2014	2	Shiva	7	20	0	85	13	1105	
3	24	2014	2	Shiva	7	20	0	85	13	1105	
3	25	2014	2	Shiva	7	20	0	85	13	1105	
3	26	2014	2	Shiva	7	20	0	85	13	1105	
3	27	2014	2	Shiva	7	20	0	85	13	1105	
3	29	2014	2	Shiva	7	20	0	85	13	1105	
3	30	2014	2	Shiva	7	20	0	85	13	1105	
3	31	2014	2	Shiva	7	20	18150	85	13	1105	
3	28	2014	2	Shiva	7	20	0	85	13	1105	
					Total		33150		403	34255	
				Total Income			34255				
				Net Income			1105				

**Remark:** Test is successful.

## **Problems Encountered**

**S.N. = 1**

**Problem Description = Error in deleting employee when datagridview is empty**

**Solution = By using the following code problem was solved.**

```
If DataGridView1.Rows.Count = 0 Then Exit Sub
```

**Result = Solved**

**S.N. = 2**

**Problem Description = Error in updating employee's photo**

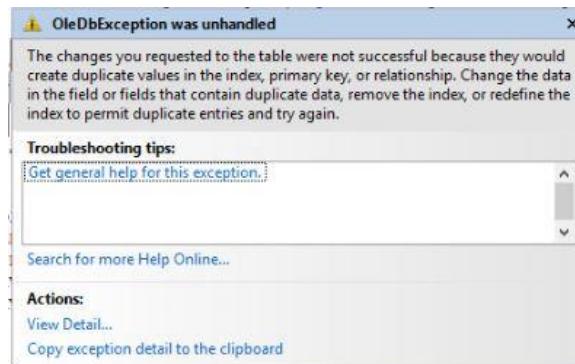
**Cause = By using field name as 'image' is database**

**Solution = It was changed to 'imagepic'**

**Result = Solved**

**S.N. = 3**

**Problem Description = Error in saving common working. This was shown**



**Solution = By deleting all previous data and saving new data solved the problem as we only have one common working hour.**

**Result = Solved**

## ***Installation***

### ***System Changeover***

**My client installed windows 7 operating system in his desktop computer. So this software had been tested in windows 7 operating system and it shows no trouble. Various data testing were conducted which are presented in section *Testing*.**

**After thorough testing of the software was done, it was time to install it in the computer system of B.D.Multipurpose. Since all details are handled manually, I recommended direct implementation of the created software.**

**However, Mr. Badri Thapa preferred pilot implementation as he was not fully convinced of the new system before using it. Pilot Implementation or Parallel Implementation includes the operator or user of the system to maintain and use both the systems parallel.**

**Most of description on how to use the new software was briefed to Mr. Badri Thapa by me and rest of the functions were taught to him through the User Manual.**

### ***Implementation***

**The system is flawless and working for more than two or more weeks, the manager of the system has plans to completely replace the manual system. Since he did not had any trouble caused in the system. So he agreed to go completely for the computerized solution and stop handling the details manually.**

## **4. DOCUMENTATION**

### ***Technical Documentation***

The Technical Documentation has been published as a booklet but it has not been attached with this project report. The reason is because it contains the same topics present in this project report. To view the contents of the Technical Documentation, please refer to the section *System Design* and the section *Software Development and Programming*.

## ***User Documentation***

**The User Documentation has been published as a booklet which is shown in the following pages. The User Manual has its own Table of Contents.**

## **5. EVALUATION**

### **Degree of Successful Implementation**

Once the new system was operated by the organization. Manager was very much satisfied by the new system as it makes all works much easier. The problems faced by manager previously are solved now. The problems which were solved are listed below :

**1) Easier and faster service**

With the new system it is easier to see all earning details. It is faster as well because all calculations are done automatically.

**2) Workload has been decreased**

Since new system does all calculations automatically manager has less work load.

**3) Easier search**

With new system it is easier to search all data of history, in less time.

**4) Protected Data**

It is computerized database protected with password so data is a strongly protected.

**5) Easier to extend for future developments**

New system is a computerized system, so updates can be placed without losing data.

### ***Client Feedback/Response***

**Mr. Badri Thapa was satisfied with the new computerized system and had no complains about any feature.**

- ✓ Manager loved the user interface that is much user friendly.
- ✓ He was happy with new way to save working hour detail of a worker.
- ✓ He was also very happy to store all the details of an employee with their photos.
- ✓ He also loved that all calculations were done automatically and now it was possible to generate earnings of each day as well.
- ✓ Printing facility was also praised by the manager that made his work much easier.

## **Possible Program Extensions**

All the requirement that were set aside are fully met, but further extension can be made in this software to make it more better. Some of the future plans are described below:

- Options to save the attendance of an employee.
- *Flexible customizable* option for the whole user interface like tabs, toolbar and others.
- Stock management modules can be added so that it would handle stocks of this organization.
- Use of *more efficient database* such as MySQL if native Linux build is made.