**STAFF LEAVE MANAGEMENT SYSTEM**

Submitted to the

**BHARATHIDASAN UNIVERSITY, Tiruchirappalli**

**In partial fulfillment of the requirement for the award of the Degree of**

**BACHELOR OF COMPUTER SCIENCE**

SUBMITTED BY

**M.K.ABARNA (CB20S 201102)**

**K.KEERTHANA PRIYAM(CB20S 201105)**

Under the guidance of

**Mrs. P.INDHU MCA., M.Phil.,**



**DEPARTMENT OF COMPUTER SCIENCE**

**ANNAI WOMEN’S COLLEGE (ARTS AND SCIENCE)**

**Aurobindo Nagar, TNPL Road, Punnamchatram,**

**KARUR – 639 136.**

**APRIL – 2023**

**CERTIFICATE**

**ANNAI WOMEN’S COLLEGE**

**(Affilicated to Bharathidasan University,Tiruchirapalli)**

**Aurobindonagar, TNPL Road, Punnamchatram,**

**KARUR – 639 136.**

**CERTIFICATE**

This is to certify that the project entitled “**STAFF LEAVE MANAGEMENT SYSTEM”** is the bonafide record of original work done by **M.K.ABARNA(CB20S 201102),K.KEERTHANA PRIYAM(CB20S 201105)** in partial fulfillment of the requirement for the award of degree of Bachelor of Computer Science for the academic year 2022-2023 is the original work done by the candidate under my guidance and supervision.

**INTERNAL GUIDE HEAD OF THE DEPARTMENT**

Submitted for the Bharathidasan University, Tiruchirappalli project evaluation and viva-voce examination held on \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ in the Computer Science Department, **ANNAI WOMEN’S COLLEGE, PUNNAMCHATRAM, KARUR - 639 136.**

**INTERNAL EXAMINER EXTERNAL EXAMINER**

**+**

**DECLARATION**

**DECLARATION**

We hereby declare that this project entitled “**STAFF LEAVE MANAGEMENT SYSTEM**”, is the original work done by us and to the best of our knowledge, a similar work has not been submitted before by any candidate to the Bharathidasan University or any other, for fulfillment of requirements of the course of study Bachelor of Computer Science.

**Place :**

**Date :**

**Signature of the Candidate**

**M.K.ABARNA(CB20S 201102)**

**K. KEERHANA PRIYAM (CB20S 201105)**

**ACKNOWLEDGEMENT**

**ACKNOWLEDGEMENT**

We whole hearted gratitude to the Lord Almighty for having given us wisdom and good health, to complete this project work at all stages.

We have great pleasure in expressing our profound and whole hearted thanks to our Chairman, **Thiru. P. THANGARAJU** and all others members of Annai Sri Aurobindo Educational Trust for their benevolence in giving us most favourable moments to be students in Annai Women’s College.

We wish to express our sincere gratitude to our Principal, **Dr. I. SARUMATHI M.Com., M.Phil., M.B.M., M.B.A., PhD., NET.,** for her deep concern over each and every student in their good performance.

We very much thankful to **Dr. N. KRISHNAVENI M.Sc., M.Phil., B.Ed., Ph.D** Dean of science for the approval of the request to print the module in the college to lesson their financial group distribution

We feel delighted to thank **Mrs. S. LEELAVATHI M.Sc., M.Phil.,** Head of the department of computer science for her support.

We express my sincere thanks to **Mrs. P. INDHU MCA., M.Phil .,** Assistant Professor in Department of Computer Science, for her valuable guidance. Also We express our gratitude to all faculty members of computer science department for their cooperation and support.

Last but not the least we thanks our dear parents for their love and support who give us moral boost to complete the project work. Besides everybody, our other family members and friends were also there who indirectly helped us finish the project work. Thanks to them.

Finally the press and the printers also thanks for getting our project record neatly printed and bound.

**ABSTRACT**

# ABSTRACT

This Staff Leave Management System is aimed at implementing a leave management application for faculty that is of importance to any Educational Institute. This project has three types of login they are Faculty, HOD and Principal. Faculty needs to get registered by entering all the details in the registration form.

The faculty can get login by entering unique username and password. Then faculty can apply their leave application form. This leave application details first send to HOD. HOD verifies the faculty leave application then approve or reject the leave request. HOD approved leave application only send to Principal.

Principal verify the faculty leave request then decide approve or reject the leave request. Leave applied faculty can view their leave application status, whether it is approved or not by HOD and Principal.

**CONTENT**

**CONTENTS**

|  |  |  |
| --- | --- | --- |
| **S.NO** | **TITLE** | **PAGE.NO.** |
| **1.** | **INTRODUCTION**  1.1 ABOUT THE PROJECT | 13 |
| **2.** | **SYSTEM ANALYSIS**  2.1 EXISTING SYSTEM  2.2 PROPOSED SYSTEM  2.3 FEASIBILITY STUDY | 15  2  3 |
| **3.** | **SYSTEM SPECIFICATIONS**  3.1 HARDWARE SPECIFICATION  3.2 SOFTWARE SPECIFICATION | 4  4 |
| **4.** | **SOFTWARE DESCRIPTION**  4.1 FRONT END FEATURES  4.2 BACK END FEATURES | 5  8 |
| **5.** | **PROJECT DESCRIPTION**  5.1 INPUT DESIGN  5.2 OUTPUT DESIGN  5.3 MODULE DESCRIPTION  5.4 DATABASE DESIGN  5.5 DATA FLOW DIAGRAM | 10  10  11  13  15 |
| **6.** | **SYSTEM IMPLEMENTATION**  6.1 SOURCE CODE | 17 |
| **7.** | **FORM DESIGN** | 48 |
| **8.** | **SYSTEM TESTING**  8.1 VERIFICATION TESTING  8.2 VALIDATION TESTING  8.3 INTEGRATION TESTING  8.4 UNIT TESTING  8.5 OUTPUT TESTING | 59  60  60  60  60 |
| **9.** | **CONCLUSION** | 62 |
| **10.** | **FUTURE ENHANCEMENT** | 63 |
| **11.** | **BIBLIOGRAPHY** | 64 |

**INTRODUCTION**

**1. INTRODUCTION**

**1.1ABOUT THE PROJECT**

The project presented here is detailed study and design of the**“STAFF LEAVE MANAGEMENT SYSTEM”**is designed using ASP .NET as Front End and Ms-SQL Server as back**.** It includes the following details**,**

The Staff leave management system is a web-based application that can access by all the faculty of the college. Their features like updating faculty records, approval of leave cancellations of leaves, track the staff leave application status.

The existing system has more draw backs. So it is a time-consuming process. So to save their time and minimize their application form process, we decided to design the online staff leave application form. Staff can apply their leave through online. HOD and principal verify the application form then they leave sanctioned or not. Applied staff can know their leave application status.

**SYSTEM ANALYSIS**

**2. SYSTEM ANALYSIS**

System Analysis is a very critical activity while developing a software application. This stage involves study of existing system and interacting with users, which determines user requirements and their expectation of the proposed system. Cost of incorporating changes required by the user is very less at this stage, which steeply increases as development advances.

**2.1 EXISTING SYSTEM**

This system is offline it requires a lot of effort. for applying for a leaving staff in the colleges go to the office and collect the leave application form and by filling up the details, type of leave, leave applied for how many days, the reason for leave, , etc. leave application form they have to submit it to the office. The office staff moves the application form to HOD, after HOD sanctioned then that application move to the Principal approval. Leave applied staff not able to know the application current status easily.

**2.2 PROPOSED SYSTEM**

The proposed system eliminates the drawback of existing system. The proposed system is developed to manage staff leave and their records in the colleges. It helps to record staff information, Approve or denies staff leave request. The purpose of the project is to build an application program to reduce the manual work for managing staff leave. Well-designed database to store faculty eave information. The main objective of this application is to automate as well as update the leave information of staff. If any faculty wants to know about their leave application status it will be very easy.

**2.3 FEASIBILITY STUDY**

Feasibility studies aim to objectively and rationally uncover the strengths and weaknesses of the existing business or proposed venture, opportunities and threats as presented by the environment, the resources required to carry through, and ultimately the prospects for success. In its simplest term, the two criteria to judge feasibility are cost required and value to be attained. As such, a well-designed feasibility study should provide a historical background of the project, description of the service, details of the operations and maintenance. Generally, feasibility studies precede technical development and project implementation.

**2.3.1 Technology and system feasibility**

The assessment is based on an outline design of system requirements in terms of Input, Processes, Output, Fields, Programs, and Procedures. This can be quantified in terms of volumes of data, trends, frequency of updating, etc. in order to estimate whether the new system will perform adequately or not. Technological feasibility is carried out to determine whether the company has the capability, in terms of software, hardware, personnel and expertise, to handle the completion of the project when writing a feasibility report, the following should be taken to consideration.

**2.3.2 Schedule feasibility**

A project will fail if it takes too long to be completed before it is useful. Typically this means estimating how long the system will take to develop, and if it can be completed in a given time period using some methods like payback period. Schedule feasibility is a measure of how reasonable the project timetable is. You need to determine whether the deadlines are mandatory or desirable.

**SYSTEM SPECIFICATIONS**

**3. SYSTEM SPECIFICATIONS**

**3.1 HARDWARE SPECIFICATION**

Processor : AMD PRO A4-3350B APU

Clock Rate : 2.00GHz

Hard Disk : 500 GB

RAM : 4 GB

**3.2 SOFTWARE SPECIFICATION**

Operating System : Windows 10

Front end Tool : ASP .NET

Back end Tool : Ms-SQL Server

Documentation : Microsoft Word 2010

**SOFTWARE DESCRIPTION**

**4. SOFTWARE DESCRIPTION**

**4.1 FRONT END FEATURES**

**.NET**

.NET is the Microsoft Web services strategy to connect information, people, systems, and devices through software. Integrated across the Microsoft platform, .NET technology provides the ability to quickly build, deploy, manage, and use connected, security-enhanced solutions with Web services. .NET-connected solutions enable businesses to integrate their systems more rapidly and in a more agile manner and help them realize the promise of information anytime, anywhere, on any device.

# ASP.NET Page Lifecycle

In ASP.NET, a web page has execution lifecycle that includes various phases. These phases include initialization, instantiation, restoring and maintaining state etc. it is required to understand the page lifecycle so that we can put custom code at any stage to perform our business logic.

ASP.NET is full of features and provides an awesome platform to create and develop web application. Here, we are discussing these features of Web Forms.

* Server Controls
* Master Pages
* Working with data
* Membership
* Client Script and Client Frameworks
* Routing
* State Management
* Security
* Performance
* Error Handling

## Page Lifecycle stages

The following table contains the lifecycle stages of ASP.NET web page.

|  |  |
| --- | --- |
| **Stage** | **Description** |
| Page request | This stage occurs before the lifecycle begins. When a page is requested by the user, ASP.NET parses and compiles that page. |
| Start | In this stage, page properties such as Request and response are set. It also determines the Request type. |
| Initialization | In this stage, each control's UniqueID property is set. Master page is applied to the page. |
| Load | During this phase, if page request is postback, control properties are loaded with information. |
| Postback event handling | In this stage, event handler is called if page request is postback. After that, the Validate method of all validator controls is called. |
| Rendering | Before rendering, view state is saved for the page and all controls. During the rendering stage, the page calls the Render method for each control, providing a text writer that writes its output to the OutputStream object of the page's Response property. |
| Unload | At this stage the requested page has been fully rendered and is ready to terminate.at this stage all properties are unloaded and cleanup is performed. |

A requested page first loaded into the server memory after that processes and sent to the bowser. At last it is unloaded from the server memory. ASP.NET provides methods and events at each stage of the page lifecycle that we can use in our application. In the following table, we are tabled events.

**FEATURES OF .NET**

This section contains information about some of the latest features available in this release of Visual Studio.

**Language Enhancements**

Microsoft Visual Basic, Microsoft C++, and Microsoft JScript have all been updated to meet the development needs. Additionally, a new language, Microsoft C#, has been introduced. These languages leverage the functionality of the .NET Framework, which provides access to key technologies that simplify the development of ASP Web applications and XML Web services.

**Visual Basic**

Visual Basic has been updated to include many new and improved language features that make it a powerful object-oriented programming language. These features include inheritance, interfaces, and overloading, among others. Visual Basic also now supports structured exception handling, and custom attributes.

**Web Forms**

Web Forms are an ASP.NET technology that are use to create programmable Web pages. Web Forms render themselves as browser-compatible HTML and script, which allows any browser on any platform to view the pages. Using Web Forms, we can create Web pages by dragging and dropping controls onto the designer and then adding code, similar to the way that we create Visual Basic forms.

**Windows Forms**

Windows Forms is the new platform for Microsoft Windows application development, based on the .NET Framework. This framework provides a clear, object-oriented, extensible set of classes that enables to develop rich Windows applications.

**4.2 BACK END FEATURES**

**MS – SQL SERVER 2008**

## **Characteristics of SQL**

* **High Performance**

SQL provides high-performance programming capability for highly transactional, heavy workload, and high usage [database systems](https://intellipaat.com/blog/what-is-database/). SQL programming gives various ways to describe the data more analytically.

* **High Availability**

SQL is compatible with databases like MS Access, Microsoft SQL Server, MySQL, Oracle Database, SAP HANA, SAP Adaptive Server, etc. All of these [relational database management systems](https://intellipaat.com/blog/tutorial/sql-tutorial/rdbms/) support SQL and it is easy to create an application extension for procedural programming and various other [SQL functions](https://intellipaat.com/blog/tutorial/sql-tutorial/sql-functions/) which are additional features thus converting SQL into a powerful tool.

### **Scalability and Flexibility**

### [SQL](https://intellipaat.com/blog/tutorial/sql-tutorial/introduction-to-sql/) provides Scalability and Flexibility. It is very easy to create new tables and previously created or not used tables can be dropped or deleted in a database.

* **Robust Transactional Support**SQL programming can handle large records and manage numerous transactions.

**SQL Constraints**

Constraints are the rules enforced on data columns on table. These are used to limit the type of data that can go into a table. This ensures the accuracy and reliability of the data in the database. Constraints could be column level or table level. Column level constraints are applied only to one column, whereas table level constraints are applied to the whole table. Following are commonly used constraints available in SQL:

• NOT NULL Constraint: Ensures that a column cannot have NULL value.

• UNIQUE Constraint: Ensures that all values in a column are different.

• PRIMARY Key: Uniquely identified each rows/records in a database table.

• FOREIGN Key: Uniquely identified a rows/records in any another database table.

**NOT NULL Constraint**

By default, a column can hold NULL values. If you do not want a column to have a NULL value, then you need to define such constraint on this column specifying that NULL is now not allowed for that column.

**UNIQUE Constraint**

The UNIQUE Constraint prevents two records from having identical values in a particular column. In the CUSTOMERS table, for example, you might want to prevent two or more people from having identical age.

**PRIMARY Key**

A primary key is a field in a table which uniquely identifies each row/record in a database table. Primary keys must contain unique values. A primary key column cannot have NULL values. A table can have only one primary key, which may consist of single or multiple fields. When multiple fields are used as a primary key, they are called a composite key. If a table has a primary key defined on any field(s), then you cannot have two records having the same value of that field(s).

**FOREIGN Key**

A foreign key is a key used to link two tables together. This is sometimes called a referencing key. Foreign Key is a column or a combination of columns whose values match a Primary Key in a different table. The relationship between 2 tables matches the Primary Key in one of the tables with a Foreign Key in the second table. If a table has a primary key defined on any field(s), then you cannot have two records having the same value of that field(s).

**PROJECT DESCRIPTION**

**5. PROJECT DESCRIPTION**

**5.1 INPUT DESIGN**

Input design is the part of overall system design which requires very careful attention. Often the collection of input data is the most expensive part of the system, in terms of both the equipment used and the number of people involved; it is the point of most contact for the users with the computer system, and it is prone to error. If data going into the system are incorrect, then the processing and output will magnify this error. Validations are set in every form, so not able to feed wrong data.

Input design is the very important part in the project and should be concentrated well as it is prone to error. The data that are to be inserted with care as this plays a very important role. In order to get the meaningful output and to achieve good accuracy the input should be acceptable and understandable by the user.

**5.2 OUTPUT DESIGN**

Output design plays a very important role in a system. Getting a correct output is a task that has to be concentrated, as a system is validated as a correct one only if it gives the correct output according to the input.

[Graphic design](http://en.wikipedia.org/wiki/Graphic_design) may be utilized to support its [usability](http://en.wikipedia.org/wiki/Usability). The design process must balance technical functionality and visual elements to create a system that is not only operational but also usable and adaptable to changing user needs.

**5.3 MODULE DESCRIPTION**

* **Home**
* **Registration**
* **Login**
  + **Staff**
  + **HOD**
  + **Principal**
* **Staff Login**
  + **Leave Application**
  + **Leave Application Status**
* **HOD View Application Form**
* **Principal View Leave Application**

**Home**

This is the first page in this project. It contains Annai College details. Here the following links are available, Staff Login, HOD Login, Principal Login and Registration

**Registration**

Registration form is used to register faculty details. All college faculties should register in this page. Here enter the following faculty details like faculty id, name, designation, department email id and phone number. Here faculty creates their own user name and password for login.

**Staff Login**

Here staff enters their user name and password, if it is correct then only they can enter staff leave application form and view leave application current status.

**Leave Application**

This page is used to for staff. Staff can able to feed their leave application details. Faculty name, designation and department details are derived from staff registration page. Other details should be filled by staff like leave applied days, leave category, purpose of leave and Assignment details. This leave application details first send to HOD for approval

**HOD Login**

HOD enters their unique user name and password then only they can enter View Staff Application form page.

**HOD View Application form**

HOD can only enter this page. Staff applied leave applications are displayed here. The application is filtered by Department wise. Every department HOD can only view their department staff leave application only. Other department staff leave application cannot view. HOD views the application form and then they decide to accept or reject the leave application. Only leave application approved by HOD will go to principal.

**Principal Login**

Principal enter their unique user name and password, if it is correct then they enter to Staff Leave Application Page.

**Principal View Leave Application**

This page is authenticated by admin (Principal). Only leave application approved by HOD will be displayed here. After verification principal decide to accept or reject the staff leave application.

**Leave Application Status.**

This page is for staff. Logged staff can only enter this page. Leave applied staff can able to known their leave application status whether it is accept or rejected by HOD and Principal.

**5.4 DATABASE DESIGN**

**TABLES**

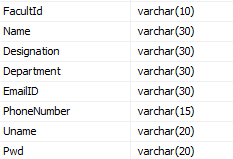
The following tables and their respective fields used in this project.

**Data Base Name:LEAVE**

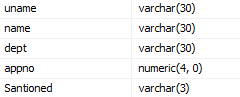
**Table Name:Admin**

****

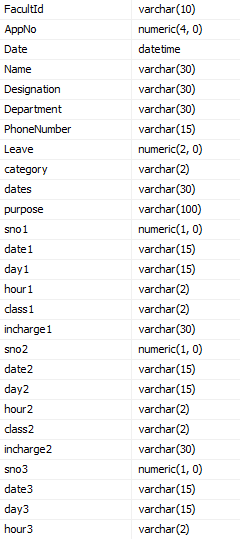
**Table Name:Regis**

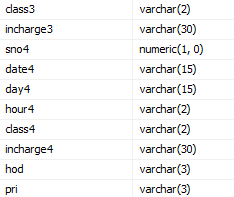
****

**Table Name :feed**

****

**Table Name: Appli**

****

****

**5.5 DATA FLOW DIAGRAM**

###### HOME PAGE

Registration

Login

HOD

Principal

Staff

Leave Application

View Leave Application

View Leave Application

Application Status

Database

**SYSTEM IMPLEMENTATION**

**6. SYSTEM IMPLEMENTATION**

**IMPLEMENTATION PLAN**

Once a program has been coded it is time to run it. Testing and Debugging refer the task of detecting and removing errors in a program, so that the program procedure the derived results in all occasions.

Every programmer should be aware of the fact that rarely does a program that run perfect the first time.

A project has been fully started in terms of sequence of steps, correctness, than it is time to implement the system that is to code it into a computer.

Before the implementation we can begin to write code we must design entire system of computer data structure to represent name aspects of this project.

Once the user is satisfied with the system it is implemented and made online. Now the system is ready for implementation.

**IMPLEMENTATION PROCESS**

* Careful planning
* Investigation of system
* Design of methods to achieve the changeover.
* Evaluation of changeover method.

**6.1 SOURCE CODE**

**Registration Form**

Imports System.Data.SqlClient

Partial Class \_Default

Inherits System.Web.UI.Page

Private con As New SqlConnection("server=DESKTOP\SQLEXPRESS8;uid=sa;pwd=.;database=Leave")

Private drAsSqlDataReader

Protected Sub Button1\_Click(ByVal sender As Object, ByVal e As System.EventArgs) Handles Button1.Click

Dim m

Dim r As String

m = MsgBox("Do u want to Register?", MsgBoxStyle.YesNo + MsgBoxStyle.Question, "Register")

If m = MsgBoxResult.Yes Then

If Trim(TextBox5.Text) = "" Then

MsgBox("Enter User Name")

TextBox4.Focus()

Exit Sub

End If

If Trim(TextBox6.Text) = "" Then

MsgBox("Enter Password")

TextBox5.Focus()

Exit Sub

End If

con.Open()

Dim co As New SqlCommand("select \* from regis where uname='" & TextBox5.Text & "'", con)

dr = co.ExecuteReader

If dr.Read = True Then

MsgBox("User Name already found")

TextBox5.Focus()

con.Close()

Exit Sub

End If

dr.Close()

Dim com As New SqlCommand("insert into regis values('" & Trim(UCase(TextBox1.Text)) & "', '" & Trim(UCase(TextBox2.Text)) & "','" & Trim(UCase(DropDownList1.Text)) & "','" & Trim(UCase(DropDownList2.Text)) & "', '" & Trim(UCase(TextBox3.Text)) & "','" & Trim(UCase(TextBox4.Text)) & "','" & Trim(TextBox5.Text) & "','" & Trim(TextBox6.Text) & "')", con)

com.ExecuteNonQuery()

con.Close()

MsgBox("Username Created.")

End If

End Sub

Protected Sub Page\_Init(ByVal sender As Object, ByVal e As System.EventArgs) Handles Me.Init

DropDownList1.Items.Add("ASST.PROF")

DropDownList1.Items.Add("PROF")

DropDownList1.Items.Add("A.H.O.D")

DropDownList1.Items.Add("H.O.D")

DropDownList2.Items.Add("COMPUTER SCIENCE")

DropDownList2.Items.Add("TAMIL")

DropDownList2.Items.Add("ENGLISH")

DropDownList2.Items.Add("COMMERCE")

DropDownList2.Items.Add("PHYSICS")

TextBox6.TextMode = TextBoxMode.Password

LinkButton1.PostBackUrl = "default9.aspx"

LinkButton2.PostBackUrl = "default5.aspx"

End Sub

Protected Sub Page\_Load(ByVal sender As Object, ByVal e As System.EventArgs) Handles Me.Load

End Sub

Protected Sub Button2\_Click(ByVal sender As Object, ByVal e As System.EventArgs) Handles Button2.Click

TextBox1.Text = ""

TextBox2.Text = ""

TextBox3.Text = ""

TextBox4.Text = ""

TextBox5.Text = ""

TextBox6.Text = ""

TextBox1.Focus()

End Sub

End Class

**Leave Application Form**

Imports System.Data.SqlClient

Partial Class Default2

Inherits System.Web.UI.Page

Private con As New SqlConnection("server=DESKTOP\SQLEXPRESS8;uid=sa;pwd=.;database=Leave")

Private drAsSqlDataReader

Private sno1, sno2, sno3, sno4 As Integer

Private date1, date2, date3, date4, day1, day2, day3, day4, hour1, hour2, hour3, hour4, class1, class2, class3, class4, inc1, inc2, inc3, inc4 As String

Protected Sub Button1\_Click(ByVal sender As Object, ByVal e As System.EventArgs) Handles Button1.Click

If Trim(TextBox10.Text) = "" Then

sno1 = 0

date1 = "-"

day1 = "-"

hour1 = "-"

class1 = "-"

inc1 = "-"

Else

sno1 = TextBox10.Text

date1 = TextBox11.Text

day1 = DropDownList2.Text

hour1 = DropDownList3.Text

class1 = DropDownList4.Text

inc1 = TextBox12.Text

End If

If Trim(TextBox13.Text) = "" Then

sno2 = 0

date2 = "-"

day2 = "-"

hour2 = "-"

class2 = "-"

inc2 = "-"

Else

sno2 = TextBox13.Text

date2 = TextBox14.Text

day2 = DropDownList5.Text

hour2 = DropDownList6.Text

class2 = DropDownList7.Text

inc2 = TextBox15.Text

End If

If Trim(TextBox16.Text) = "" Then

sno3 = 0

date3 = "-"

day3 = "-"

hour3 = "-"

class3 = "-"

inc3 = "-"

Else

sno3 = TextBox16.Text

date3 = TextBox17.Text

day3 = DropDownList8.Text

hour3 = DropDownList9.Text

class3 = DropDownList10.Text

inc3 = TextBox18.Text

End If

If Trim(TextBox19.Text) = "" Then

sno4 = 0

date4 = "-"

day4 = "-"

hour4 = "-"

class4 = "-"

inc4 = "-"

Else

sno4 = TextBox19.Text

date4 = TextBox20.Text

day4 = DropDownList11.Text

hour4 = DropDownList12.Text

class4 = DropDownList13.Text

inc4 = TextBox21.Text

End If

Dim m

Dim tot As Integer

Dim r As String

m = MsgBox("Do u want to Submit?", MsgBoxStyle.YesNo + MsgBoxStyle.Question, "Leave")

If m = MsgBoxResult.Yes Then

If Trim(TextBox7.Text) = "" Then

MsgBox("Enter Leave Days")

TextBox7.Focus()

Exit Sub

End If

con.Open()

Dim com As New SqlCommand("insert into appli values('" & Label1.Text & "'," & TextBox1.Text & ", '" & TextBox2.Text & "','" & Trim(UCase(TextBox3.Text)) & "','" & Trim(UCase(TextBox4.Text)) & "','" & Trim(UCase(TextBox5.Text)) & "','" & Trim(UCase(TextBox6.Text)) & "'," & TextBox7.Text & ",'" & Trim(UCase(DropDownList1.Text)) & "','" & Trim(UCase(TextBox8.Text)) & "','" & Trim(UCase(TextBox9.Text)) & "'," & sno1 & ",'" & date1 & "','" & day1 & "','" & hour1 & "','" & class1 & "','" & inc1 & "'," & sno2 & ",'" & date2 & "','" & day2 & "','" & hour2 & "','" & class2 & "','" & inc2 & "'," & sno3 & ",'" & date3 & "','" & day3 & "','" & hour3 & "','" & class3 & "','" & inc3 & "'," & sno4 & ",'" & date4 & "','" & day4 & "','" & hour4 & "','" & class4 & "','" & inc4 & "','-','-')", con)

com.ExecuteNonQuery()

con.Close()

MsgBox("Application Submitted")

con.Open()

Dim co As New SqlCommand("select \* from regis where facultid='" & Label1.Text & "'", con)

dr = co.ExecuteReader

If dr.Read = True Then

TextBox3.Text = dr(1)

TextBox4.Text = dr(2)

TextBox5.Text = dr(3)

TextBox6.Text = dr(5)

End If

con.Close()

End If

End Sub

Protected Sub Page\_Init(ByVal sender As Object, ByVal e As System.EventArgs) Handles Me.Init

Label1.Text = Session("fid")

'Label1.Text = "C01"

LinkButton1.PostBackUrl = "default9.aspx"

LinkButton2.PostBackUrl = "default6.aspx"

con.Open()

Dim co As New SqlCommand("select \* from regis where facultid='" & Label1.Text & "'", con)

dr = co.ExecuteReader

If dr.Read = True Then

TextBox3.Text = dr(1)

TextBox4.Text = dr(2)

TextBox5.Text = dr(3)

TextBox6.Text = dr(5)

End If

con.Close()

con.Open()

Dim co2 As New SqlCommand("select max(appno) from appli", con)

dr = co2.ExecuteReader

If dr.Read = True Then

TextBox1.Text = dr(0) + 1

End If

con.Close()

TextBox2.Text = Format(Now.Date, "dd-MMM-yyyy")

TextBox1.Enabled = False

TextBox2.Enabled = False

TextBox3.Enabled = False

TextBox4.Enabled = False

TextBox5.Enabled = False

TextBox6.Enabled = False

DropDownList1.Items.Add("CL")

DropDownList1.Items.Add("SL")

DropDownList1.Items.Add("OD")

DropDownList2.Items.Add("-")

DropDownList2.Items.Add("1ST")

DropDownList2.Items.Add("2ND")

DropDownList2.Items.Add("3RD")

DropDownList2.Items.Add("4TH")

DropDownList2.Items.Add("5TH")

DropDownList3.Items.Add("-")

DropDownList3.Items.Add("1")

DropDownList3.Items.Add("2")

DropDownList3.Items.Add("3")

DropDownList3.Items.Add("4")

DropDownList3.Items.Add("5")

DropDownList4.Items.Add("-")

DropDownList4.Items.Add("1")

DropDownList4.Items.Add("2")

DropDownList4.Items.Add("3")

DropDownList4.Items.Add("4")

DropDownList4.Items.Add("5")

DropDownList5.Items.Add("-")

DropDownList5.Items.Add("1ST")

DropDownList5.Items.Add("2ND")

DropDownList5.Items.Add("3RD")

DropDownList5.Items.Add("4TH")

DropDownList5.Items.Add("5TH")

DropDownList6.Items.Add("-")

DropDownList6.Items.Add("1")

DropDownList6.Items.Add("2")

DropDownList6.Items.Add("3")

DropDownList6.Items.Add("4")

DropDownList6.Items.Add("5")

DropDownList7.Items.Add("-")

DropDownList7.Items.Add("1")

DropDownList7.Items.Add("2")

DropDownList7.Items.Add("3")

DropDownList7.Items.Add("4")

DropDownList7.Items.Add("5")

DropDownList8.Items.Add("-")

DropDownList8.Items.Add("1ST")

DropDownList8.Items.Add("2ND")

DropDownList8.Items.Add("3RD")

DropDownList8.Items.Add("4TH")

DropDownList8.Items.Add("5TH")

DropDownList9.Items.Add("-")

DropDownList9.Items.Add("1")

DropDownList9.Items.Add("2")

DropDownList9.Items.Add("3")

DropDownList9.Items.Add("4")

DropDownList9.Items.Add("5")

DropDownList10.Items.Add("-")

DropDownList10.Items.Add("1")

DropDownList10.Items.Add("2")

DropDownList10.Items.Add("3")

DropDownList10.Items.Add("4")

DropDownList10.Items.Add("5")

DropDownList11.Items.Add("-")

DropDownList11.Items.Add("1ST")

DropDownList11.Items.Add("2ND")

DropDownList11.Items.Add("3RD")

DropDownList11.Items.Add("4TH")

DropDownList11.Items.Add("5TH")

DropDownList12.Items.Add("-")

DropDownList12.Items.Add("1")

DropDownList12.Items.Add("2")

DropDownList12.Items.Add("3")

DropDownList12.Items.Add("4")

DropDownList12.Items.Add("5")

DropDownList13.Items.Add("-")

DropDownList13.Items.Add("1")

DropDownList13.Items.Add("2")

DropDownList13.Items.Add("3")

DropDownList13.Items.Add("4")

DropDownList13.Items.Add("5")

End Sub

Protected Sub Page\_Load(ByVal sender As Object, ByVal e As System.EventArgs) Handles Me.Load

End Sub

Protected Sub Button2\_Click(ByVal sender As Object, ByVal e As System.EventArgs) Handles Button2.Click

con.Open()

Dim co2 As New SqlCommand("select max(appno) from appli", con)

dr = co2.ExecuteReader

If dr.Read = True Then

TextBox1.Text = dr(0) + 1

End If

con.Close()

con.Open()

Dim co As New SqlCommand("select \* from regis where facultid='" & Label1.Text & "'", con)

dr = co.ExecuteReader

If dr.Read = True Then

TextBox3.Text = dr(1)

TextBox4.Text = dr(2)

TextBox5.Text = dr(3)

TextBox6.Text = dr(5)

End If

con.Close()

TextBox7.Text = ""

TextBox8.Text = ""

TextBox9.Text = ""

TextBox10.Text = ""

TextBox11.Text = ""

TextBox12.Text = ""

TextBox13.Text = ""

TextBox14.Text = ""

TextBox15.Text = ""

TextBox16.Text = ""

TextBox17.Text = ""

TextBox18.Text = ""

TextBox19.Text = ""

TextBox20.Text = ""

TextBox21.Text = ""

DropDownList2.Text = "-"

DropDownList3.Text = "-"

DropDownList4.Text = "-"

DropDownList5.Text = "-"

DropDownList6.Text = "-"

DropDownList7.Text = "-"

DropDownList8.Text = "-"

DropDownList9.Text = "-"

DropDownList10.Text = "-"

DropDownList11.Text = "-"

DropDownList12.Text = "-"

DropDownList13.Text = "-"

con.Open()

Dim co As New SqlCommand("select \* from regis where facultid='" & Label1.Text & "'", con)

dr = co.ExecuteReader

If dr.Read = True Then

TextBox3.Text = dr(1)

TextBox4.Text = dr(2)

TextBox5.Text = dr(3)

TextBox6.Text = dr(5)

End If

con.Close()

End Sub

End Class

**HOD View Application**

Imports System.Data.SqlClient

Partial Class Default3

Inherits System.Web.UI.Page

Private con As New SqlConnection("server=DESKTOP\SQLEXPRESS8;uid=sa;pwd=.;database=Leave")

Private drAsSqlDataReader

Protected Sub Page\_Init(ByVal sender As Object, ByVal e As System.EventArgs) Handles Me.Init

Label1.Text = Session("uname")

Label2.Text = ""

Label3.Text = ""

'Label1.Text = "C08"

con.Open()

Dim co As New SqlCommand("select \* from regis where uname='" & Label1.Text & "'", con)

dr = co.ExecuteReader

If dr.Read = True Then

Label2.Text = dr(1)

Label3.Text = dr(3)

End If

con.Close()

ListBox1.Items.Clear()

con.Open()

Dim co1 As New SqlCommand("select \* from appli where hod='-' and department='" & Label3.Text & "'", con)

dr = co1.ExecuteReader

Do While dr.Read = True

ListBox1.Items.Add(dr(1))

Loop

con.Close()

SqlDataSource1.SelectCommand = "select \* from appli where appno=0"

LinkButton1.PostBackUrl = "default9.aspx"

End Sub

Protected Sub Page\_Load(ByVal sender As Object, ByVal e As System.EventArgs) Handles Me.Load

End Sub

Protected Sub ListBox1\_SelectedIndexChanged(ByVal sender As Object, ByVal e As System.EventArgs) Handles ListBox1.SelectedIndexChanged

SqlDataSource1.SelectCommand = "select \* from appli where appno=" & ListBox1.Text & " and department = '" & Label3.Text & "'"

End Sub

Protected Sub Button1\_Click(ByVal sender As Object, ByVal e As System.EventArgs) Handles Button1.Click

Dim chAs String

Dim m

m = MsgBox("Do u want to Submit?", MsgBoxStyle.YesNo + MsgBoxStyle.Question, "Leave")

If m = MsgBoxResult.Yes Then

If CheckBox1.Checked = True Then

ch = "YES"

Else

ch = "NO"

End If

con.Open()

Dim com As New SqlCommand("update appli set hod='" &ch& "' where appno=" & ListBox1.Text & "", con)

com.ExecuteNonQuery()

Dim com1 As New SqlCommand("insert into hod values('" & Label1.Text & "','" & Label2.Text & "','" & Label3.Text & "'," & ListBox1.Text & ",'" &ch& "')", con)

com1.ExecuteNonQuery()

con.Close()

MsgBox("Data Saved")

con.Open()

Dim co As New SqlCommand("select \* from regis where facultid='" & Label1.Text & "'", con)

dr = co.ExecuteReader

If dr.Read = True Then

TextBox3.Text = dr(1)

TextBox4.Text = dr(2)

TextBox5.Text = dr(3)

TextBox6.Text = dr(5)

End If

con.Close()

ListBox1.Items.Clear()

con.Open()

Dim co1 As New SqlCommand("select \* from appli where hod='-' and department='" & Label3.Text & "'", con)

dr = co1.ExecuteReader

Do While dr.Read = True

ListBox1.Items.Add(dr(1))

Loop

con.Close()

CheckBox1.Checked = False

SqlDataSource1.SelectCommand = "select \* from appli where appno=0"

End If

End Sub

End Class

**Principal View Application**

Imports System.Data.SqlClient

Partial Class Default4

Inherits System.Web.UI.Page

Private con As New SqlConnection("server=DESKTOP\SQLEXPRESS8;uid=sa;pwd=.;database=Leave")

Private drAsSqlDataReader

Protected Sub ListBox1\_SelectedIndexChanged(ByVal sender As Object, ByVal e As System.EventArgs) Handles ListBox1.SelectedIndexChanged

Dim s As String

SqlDataSource1.SelectCommand = "select \* from appli where appno=" & ListBox1.Text & ""

con.Open()

Dim co1 As New SqlCommand("select \* from appli where appno=" & ListBox1.Text & "", con)

dr = co1.ExecuteReader

If dr.Read = True Then

s = dr(0)

End If

con.Close()

SqlDataSource2.SelectCommand = "select \* from appli where facultid='" & s & "' and hod='YES' and pri='YES'"

End Sub

Protected Sub Page\_Init(ByVal sender As Object, ByVal e As System.EventArgs) Handles Me.Init

con.Open()

Dim co As New SqlCommand("select \* from regis where facultid='" & Label1.Text & "'", con)

dr = co.ExecuteReader

If dr.Read = True Then

TextBox3.Text = dr(1)

TextBox4.Text = dr(2)

TextBox5.Text = dr(3)

TextBox6.Text = dr(5)

End If

con.Close()

ListBox1.Items.Clear()

con.Open()

Dim co1 As New SqlCommand("select \* from appli where hod='YES' and pri='-'", con)

dr = co1.ExecuteReader

Do While dr.Read = True

ListBox1.Items.Add(dr(1))

Loop

con.Close()

SqlDataSource1.SelectCommand = "select \* from appli where appno=0"

SqlDataSource2.SelectCommand = "select \* from appli where appno=0"

LinkButton1.PostBackUrl = "default9.aspx"

End Sub

Protected Sub Page\_Load(ByVal sender As Object, ByVal e As System.EventArgs) Handles Me.Load

End Sub

Protected Sub Button1\_Click(ByVal sender As Object, ByVal e As System.EventArgs) Handles Button1.Click

Dim chAs String

Dim m

m = MsgBox("Do u want to Submit?", MsgBoxStyle.YesNo + MsgBoxStyle.Question, "Leave")

If m = MsgBoxResult.Yes Then

If CheckBox1.Checked = True Then

ch = "YES"

Else

ch = "NO"

End If

con.Open()

Dim com As New SqlCommand("update appli set pri='" &ch& "' where appno=" & ListBox1.Text & "", con)

com.ExecuteNonQuery()

con.Close()

MsgBox("Data Saved")

ListBox1.Items.Clear()

con.Open()

Dim co1 As New SqlCommand("select \* from appli where hod='YES' and pri='-'", con)

dr = co1.ExecuteReader

Do While dr.Read = True

ListBox1.Items.Add(dr(1))

Loop

con.Close()

CheckBox1.Checked = False

con.Open()

Dim co As New SqlCommand("select \* from regis where facultid='" & Label1.Text & "'", con)

dr = co.ExecuteReader

If dr.Read = True Then

TextBox3.Text = dr(1)

TextBox4.Text = dr(2)

TextBox5.Text = dr(3)

TextBox6.Text = dr(5)

End If

con.Close()

SqlDataSource1.SelectCommand = "select \* from appli where appno=0"

con.Open()

Dim co As New SqlCommand("select \* from regis where facultid='" & Label1.Text & "'", con)

dr = co.ExecuteReader

If dr.Read = True Then

TextBox3.Text = dr(1)

TextBox4.Text = dr(2)

TextBox5.Text = dr(3)

TextBox6.Text = dr(5)

End If

con.Close()

End If

End Sub

End Class

**Staff Login**

Imports System.Data.SqlClient

Partial Class Default5

Inherits System.Web.UI.Page

Private con As New SqlConnection("server=DESKTOP\SQLEXPRESS8;uid=sa;pwd=.;database=Leave")

Private drAsSqlDataReader

Protected Sub Page\_Init(ByVal sender As Object, ByVal e As System.EventArgs) Handles Me.Init

TextBox2.TextMode = TextBoxMode.Password

Label1.Text = ""

LinkButton4.Visible = False

LinkButton5.Visible = False

LinkButton4.PostBackUrl = "default2.aspx"

LinkButton5.PostBackUrl = "default6.aspx"

End Sub

Protected Sub Page\_Load(ByVal sender As Object, ByVal e As System.EventArgs) Handles Me.Load

End Sub

Protected Sub LinkButton1\_Click(ByVal sender As Object, ByVal e As System.EventArgs) Handles LinkButton1.Click

con.Open()

Dim co1 As New SqlCommand("select \* from regis where uname='" & TextBox1.Text & "' and pwd='" & TextBox2.Text & "'", con)

dr = co1.ExecuteReader

If dr.Read = True Then

Label1.Text = "Login Success"

Session("fid") = dr(0)

LinkButton4.Visible = True

LinkButton5.Visible = True

con.Open()

Dim co As New SqlCommand("select \* from regis where facultid='" & Label1.Text & "'", con)

dr = co.ExecuteReader

If dr.Read = True Then

TextBox3.Text = dr(1)

TextBox4.Text = dr(2)

TextBox5.Text = dr(3)

TextBox6.Text = dr(5)

End If

con.Close()

Else

Label1.Text = "Login Failed"

LinkButton4.Visible = False

LinkButton5.Visible = False

con.Open()

Dim co As New SqlCommand("select \* from regis where facultid='" & Label1.Text & "'", con)

dr = co.ExecuteReader

If dr.Read = True Then

TextBox3.Text = dr(1)

TextBox4.Text = dr(2)

TextBox5.Text = dr(3)

TextBox6.Text = dr(5)

End If

con.Close()

End If

con.Close()

End Sub

Protected Sub LinkButton2\_Click(ByVal sender As Object, ByVal e As System.EventArgs) Handles LinkButton2.Click

TextBox1.Text = ""

TextBox2.Text = ""

Label1.Text = ""

TextBox1.Focus()

End Sub

End Class

**Application Status**

Imports System.Data.SqlClient

Partial Class Default6

Inherits System.Web.UI.Page

Private con As New SqlConnection("server=DESKTOP\SQLEXPRESS8;uid=sa;pwd=.;database=Leave")

Private drAsSqlDataReader

Protected Sub Page\_Init(ByVal sender As Object, ByVal e As System.EventArgs) Handles Me.Init

Label1.Text = Session("fid")

'Label1.Text = "C01"

con.Open()

Dim co As New SqlCommand("select \* from appli where facultid='" & Label1.Text & "'", con)

dr = co.ExecuteReader

Do While dr.Read = True

ListBox1.Items.Add(dr(1))

Loop

con.Close()

Label2.Text = ""

Label3.Text = ""

Label4.Text = ""

LinkButton3.PostBackUrl = "default9.aspx"

LinkButton4.PostBackUrl = "default2.aspx"

con.Open()

Dim co As New SqlCommand("select \* from regis where facultid='" & Label1.Text & "'", con)

dr = co.ExecuteReader

If dr.Read = True Then

TextBox3.Text = dr(1)

TextBox4.Text = dr(2)

TextBox5.Text = dr(3)

TextBox6.Text = dr(5)

End If

con.Close()

End Sub

Protected Sub Page\_Load(ByVal sender As Object, ByVal e As System.EventArgs) Handles Me.Load

con.Open()

Dim co As New SqlCommand("select \* from regis where facultid='" & Label1.Text & "'", con)

dr = co.ExecuteReader

If dr.Read = True Then

TextBox3.Text = dr(1)

TextBox4.Text = dr(2)

TextBox5.Text = dr(3)

TextBox6.Text = dr(5)

End If

con.Close()

End Sub

Protected Sub ListBox1\_SelectedIndexChanged(ByVal sender As Object, ByVal e As System.EventArgs) Handles ListBox1.SelectedIndexChanged

con.Open()

Dim co As New SqlCommand("select \* from appli where appno=" & ListBox1.Text & "", con)

dr = co.ExecuteReader

If dr.Read = True Then

Label2.Text = dr(2)

If dr(35) = "-" Then

Label3.Text = "Waiting for approval...."

ElseIfdr(35) = "NO" Then

Label3.Text = "Your leave not sanctioned"

ElseIfdr(35) = "YES" Then

Label3.Text = "Your leave sancationed from HOD"

If (dr(36)) = "-" Then

Label4.Text = "Waiting for approval..."

ElseIfdr(36) = "NO" Then

Label4.Text = "Your leave request rejected..."

ElseIfdr(36) = "YES" Then

Label4.Text = "Leave Approved from Principal"

End If

End If

End If

con.Close()

End Sub

End Class

**HOD Login**

Imports System.Data.SqlClient

Partial Class Default7

Inherits System.Web.UI.Page

Private con As New SqlConnection("server=DESKTOP\SQLEXPRESS8;uid=sa;pwd=.;database=Leave")

Private drAsSqlDataReader

Protected Sub Page\_Init(ByVal sender As Object, ByVal e As System.EventArgs) Handles Me.Init

LinkButton4.PostBackUrl = "default3.aspx"

TextBox2.TextMode = TextBoxMode.Password

Label1.Text = ""

LinkButton4.Visible = False

End Sub

Protected Sub Page\_Load(ByVal sender As Object, ByVal e As System.EventArgs) Handles Me.Load

con.Open()

Dim co As New SqlCommand("select \* from regis where facultid='" & Label1.Text & "'", con)

dr = co.ExecuteReader

If dr.Read = True Then

TextBox3.Text = dr(1)

TextBox4.Text = dr(2)

TextBox5.Text = dr(3)

TextBox6.Text = dr(5)

End If

con.Close()

End Sub

Protected Sub LinkButton1\_Click(ByVal sender As Object, ByVal e As System.EventArgs) Handles LinkButton1.Click

con.Open()

Dim co1 As New SqlCommand("select \* from regis where uname='" & TextBox1.Text & "' and pwd='" & TextBox2.Text & "' and designation='H.O.D'", con)

dr = co1.ExecuteReader

If dr.Read = True Then

Label1.Text = "Login Success"

Session("uname") = TextBox1.Text

LinkButton4.Visible = True

Else

Label1.Text = "Login Failed"

LinkButton4.Visible = False

End If

con.Close()

End Sub

Protected Sub LinkButton2\_Click(ByVal sender As Object, ByVal e As System.EventArgs) Handles LinkButton2.Click

TextBox1.Text = ""

TextBox2.Text = ""

Label1.Text = ""

TextBox1.Focus()

End Sub

End Class

**Principal Login**

Imports System.Data.SqlClient

Partial Class Default8

Inherits System.Web.UI.Page

Private con As New SqlConnection("server=DESKTOP\SQLEXPRESS8;uid=sa;pwd=.;database=Leave")

Private drAsSqlDataReader

Protected Sub LinkButton1\_Click(ByVal sender As Object, ByVal e As System.EventArgs) Handles LinkButton1.Click

con.Open()

Dim co1 As New SqlCommand("select \* from admin where uname='" & TextBox1.Text & "' and pwd='" & TextBox2.Text & "'", con)

dr = co1.ExecuteReader

If dr.Read = True Then

Label1.Text = "Login Success"

LinkButton4.Visible = True

Else

Label1.Text = "Login Failed"

LinkButton4.Visible = False

End If

con.Close()

End Sub

Protected Sub Page\_Init(ByVal sender As Object, ByVal e As System.EventArgs) Handles Me.Init

LinkButton4.PostBackUrl = "default4.aspx"

TextBox2.TextMode = TextBoxMode.Password

Label1.Text = ""

LinkButton4.Visible = False

End Sub

Protected Sub Page\_Load(ByVal sender As Object, ByVal e As System.EventArgs) Handles Me.Load

End Sub

Protected Sub LinkButton2\_Click(ByVal sender As Object, ByVal e As System.EventArgs) Handles LinkButton2.Click

TextBox1.Text = ""

TextBox2.Text = ""

Label1.Text = ""

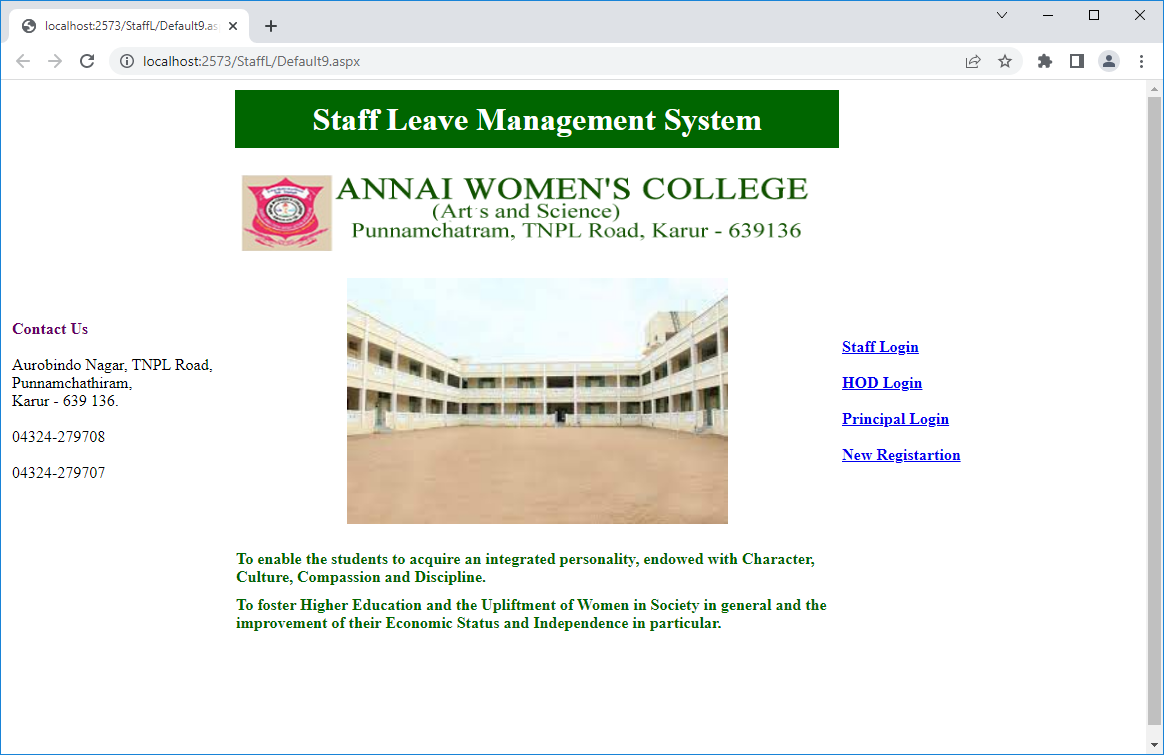
TextBox1.Focus()

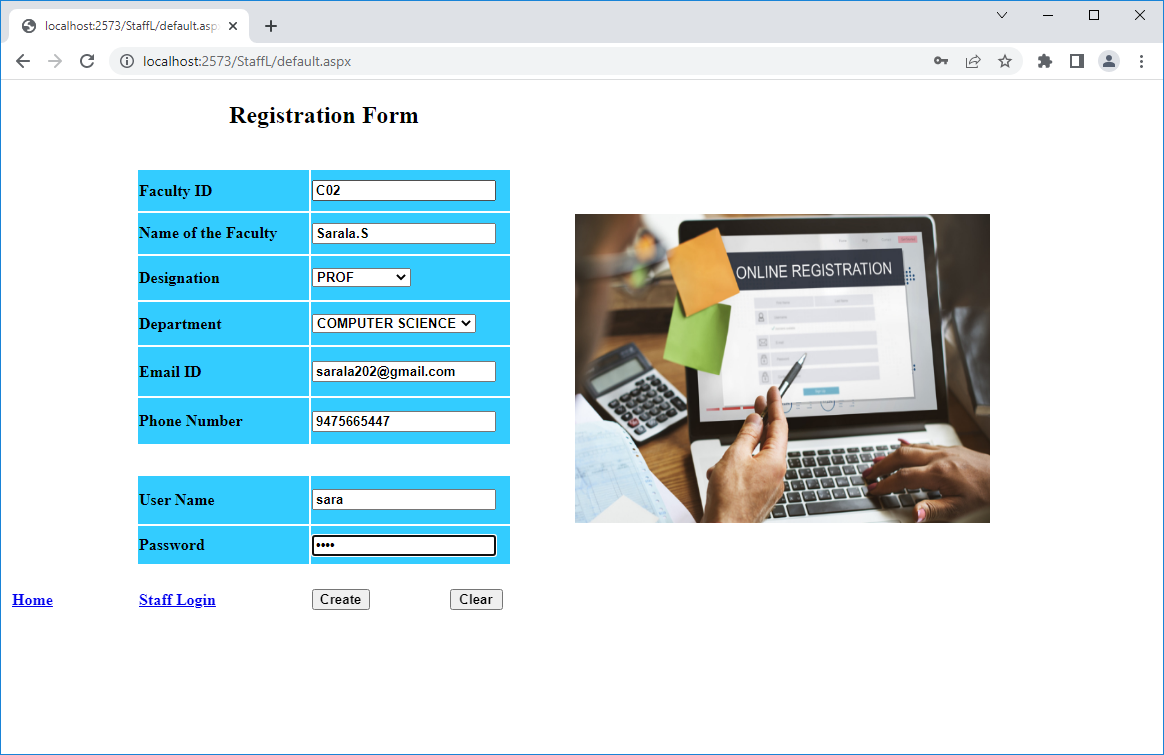
End Sub

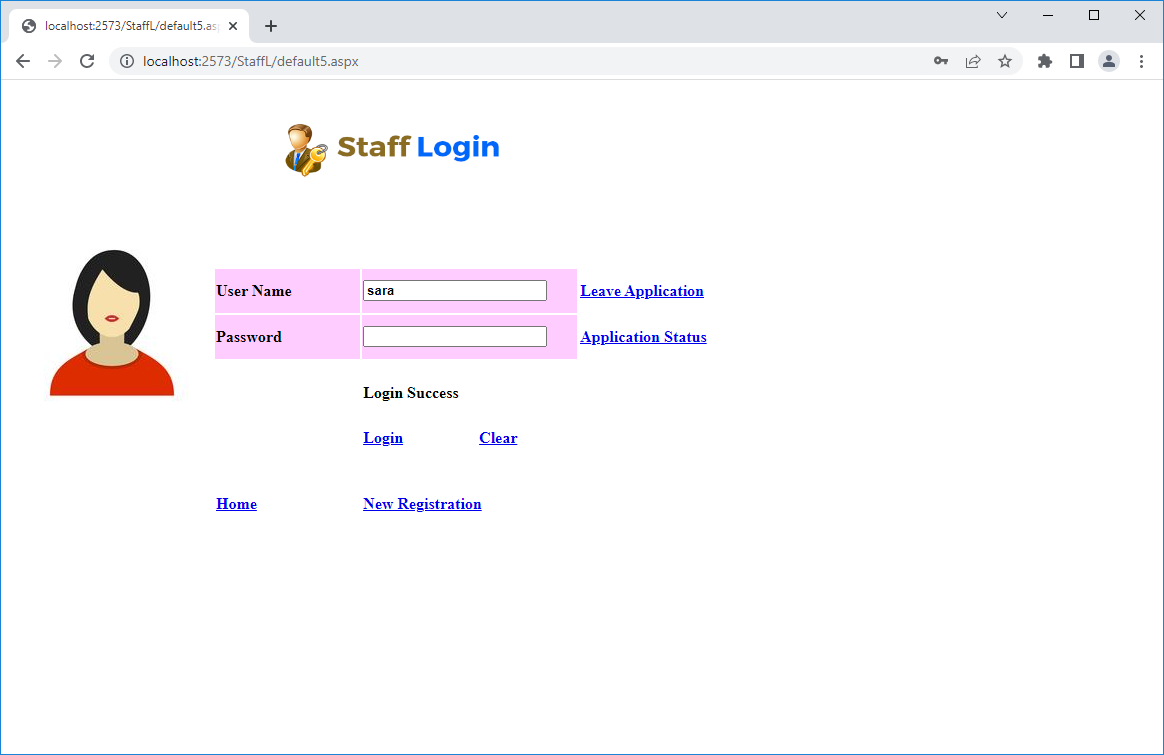
End Class

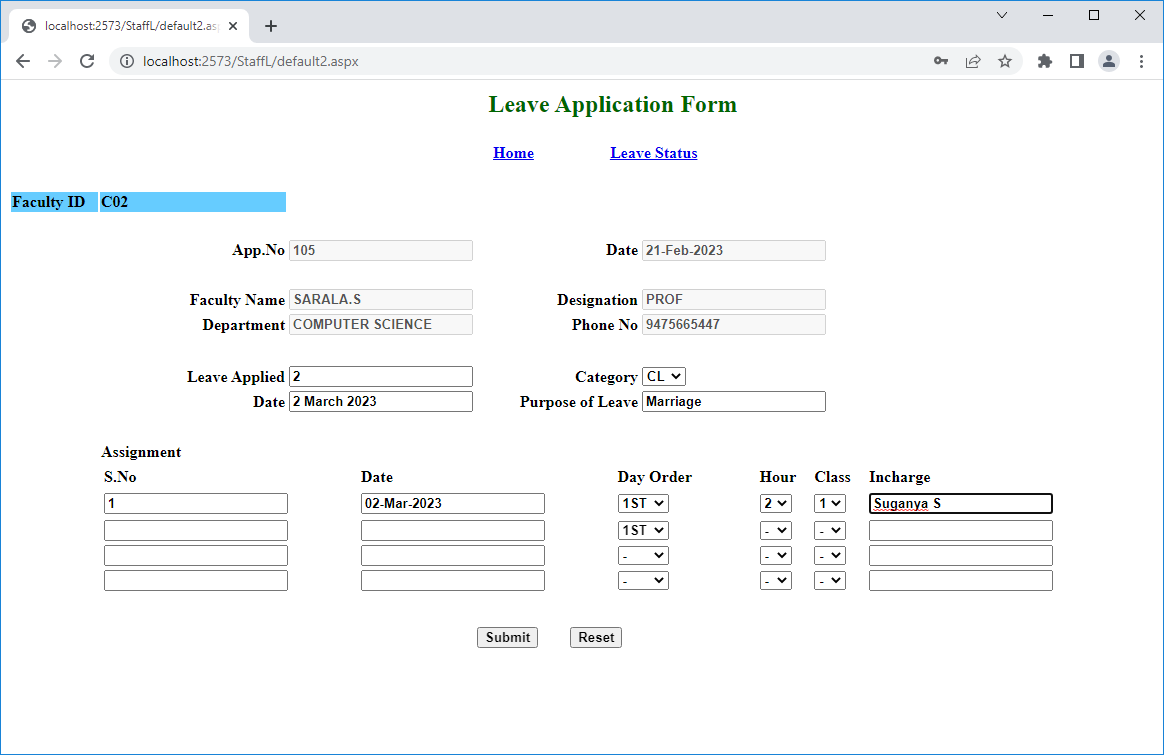
**FORM DESIGN**

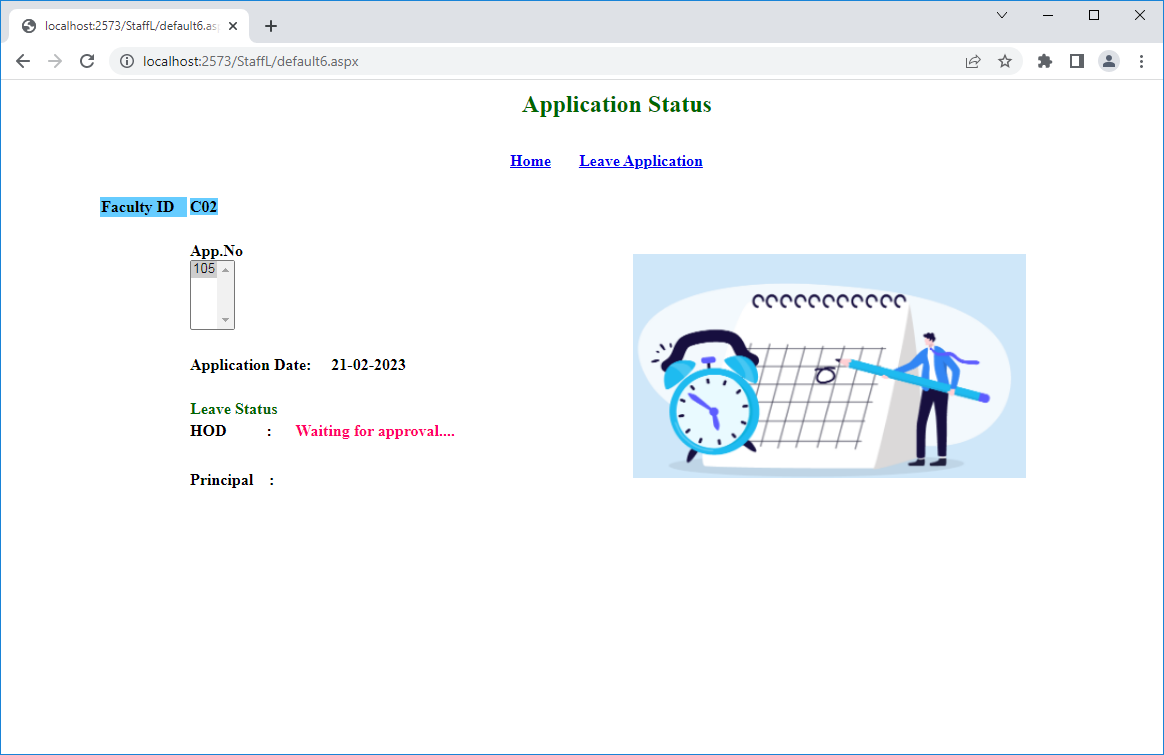
**7. FORM DESIGN**

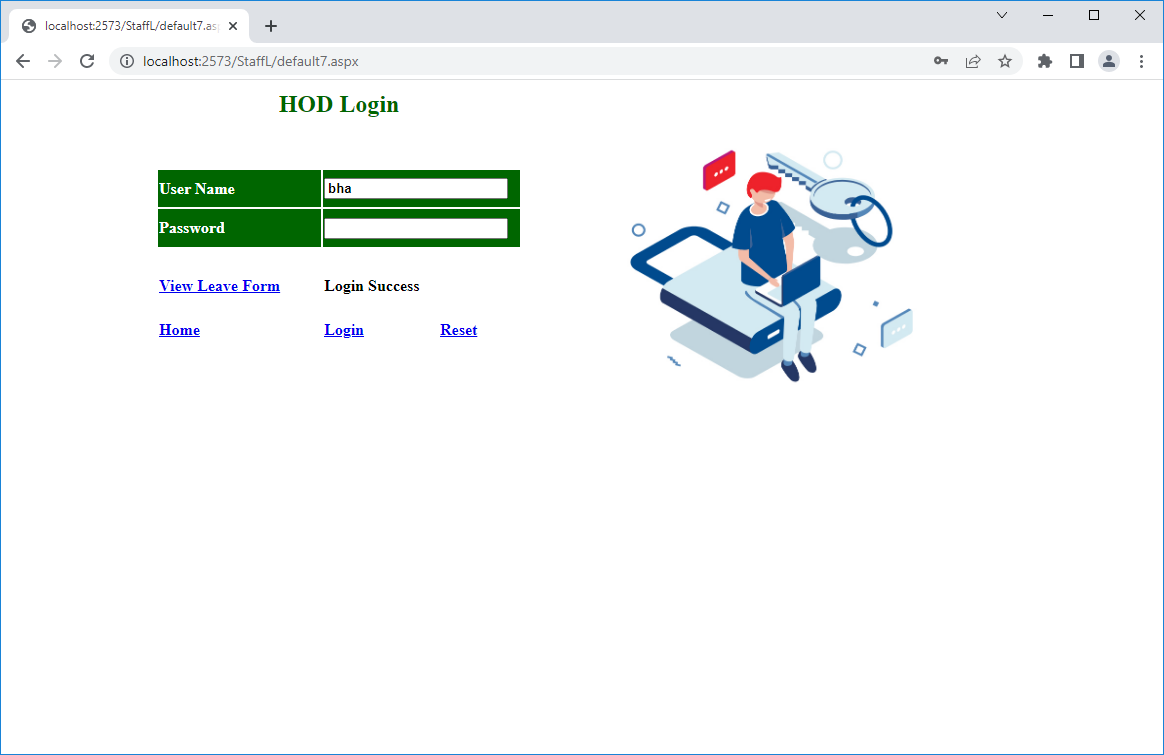


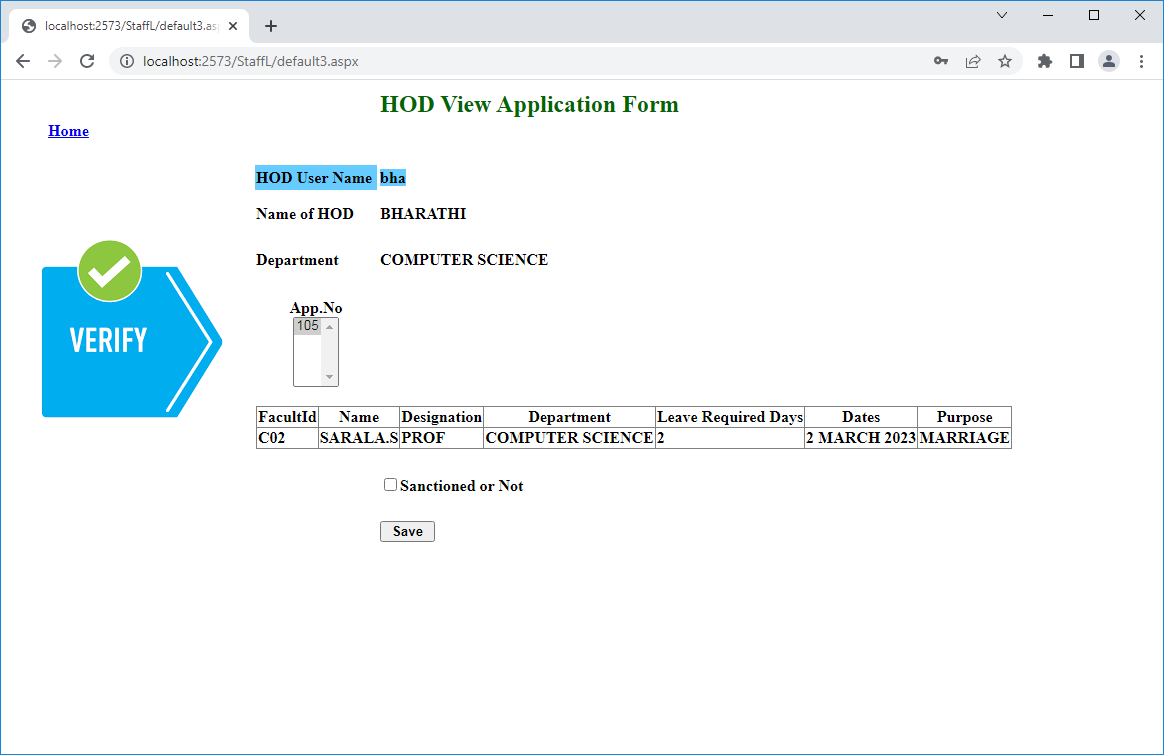


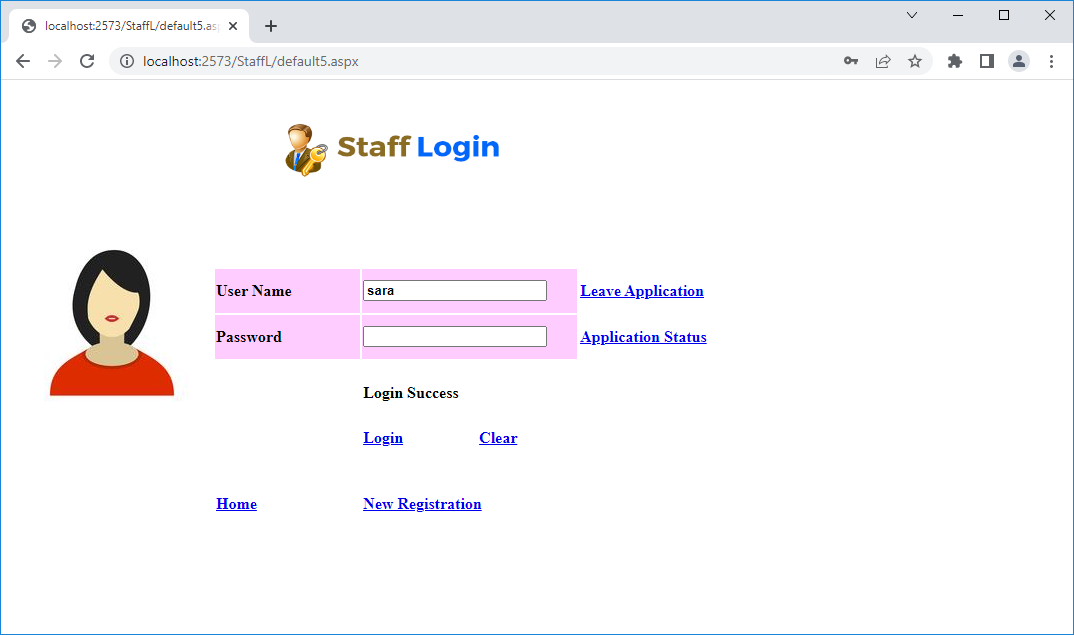


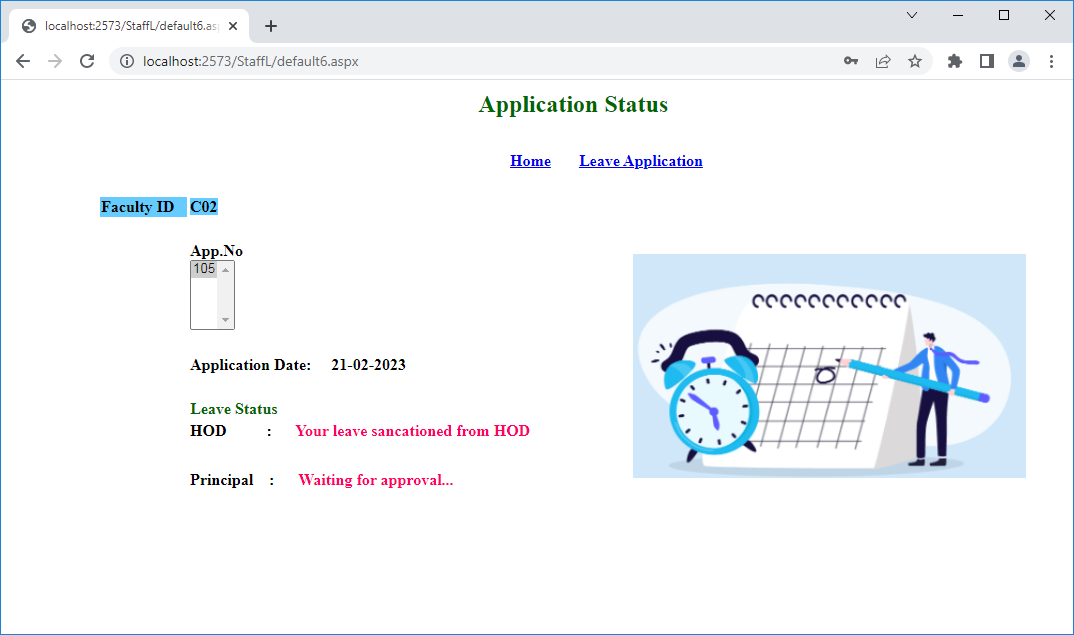


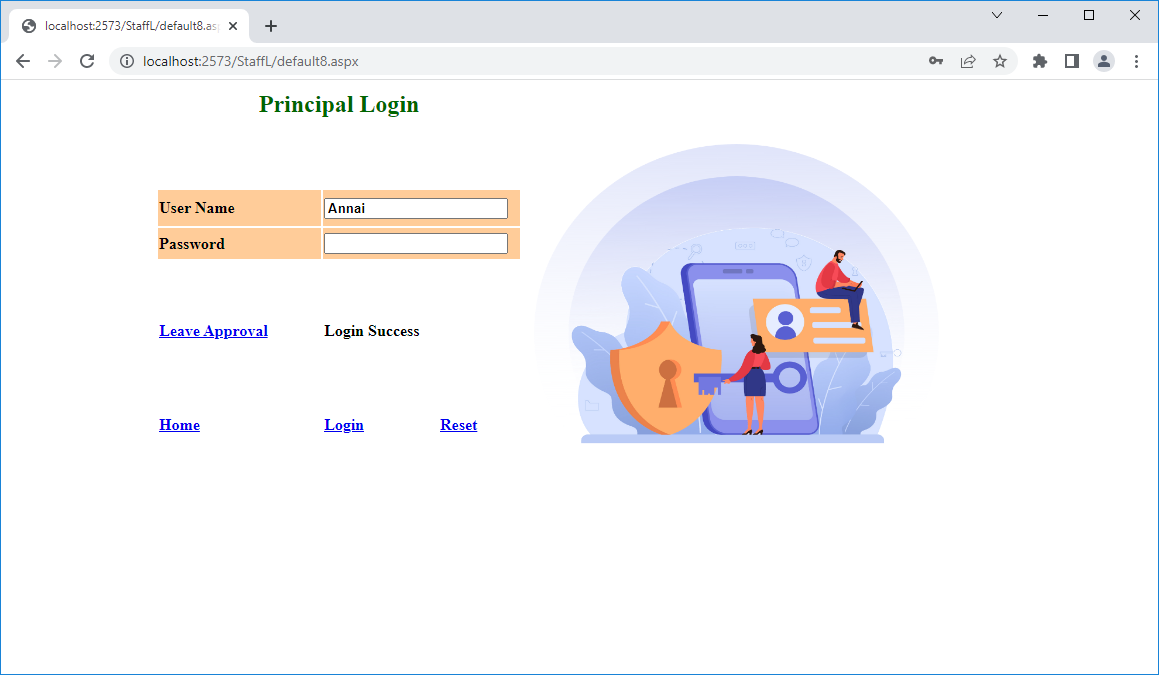


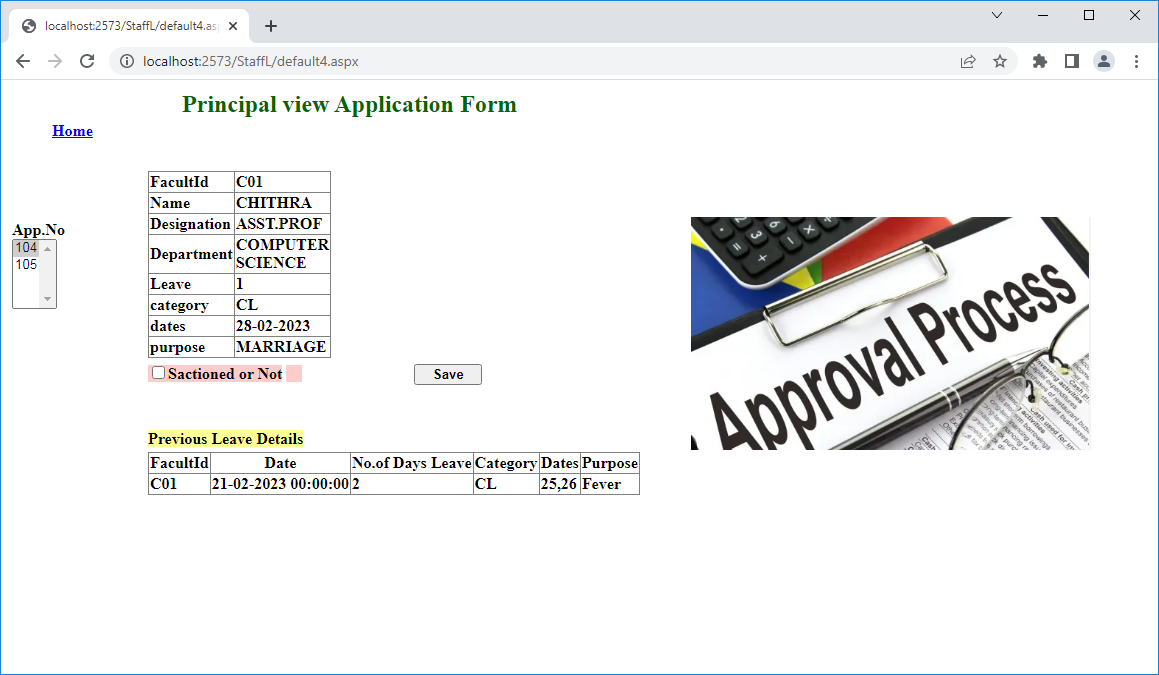


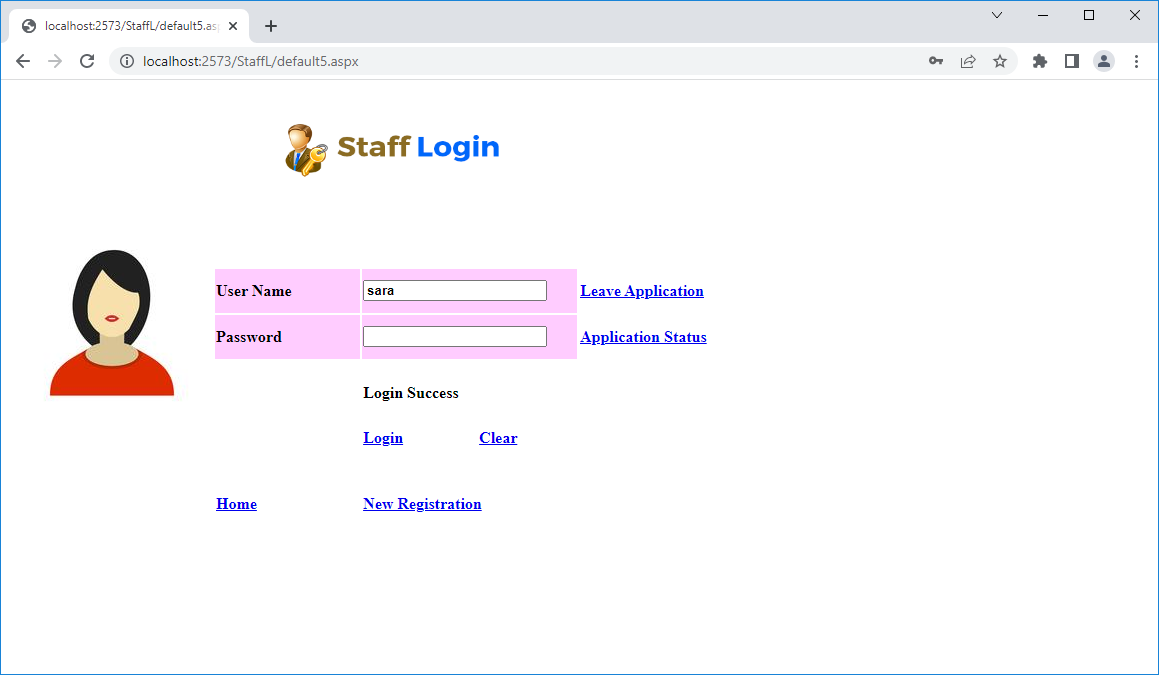


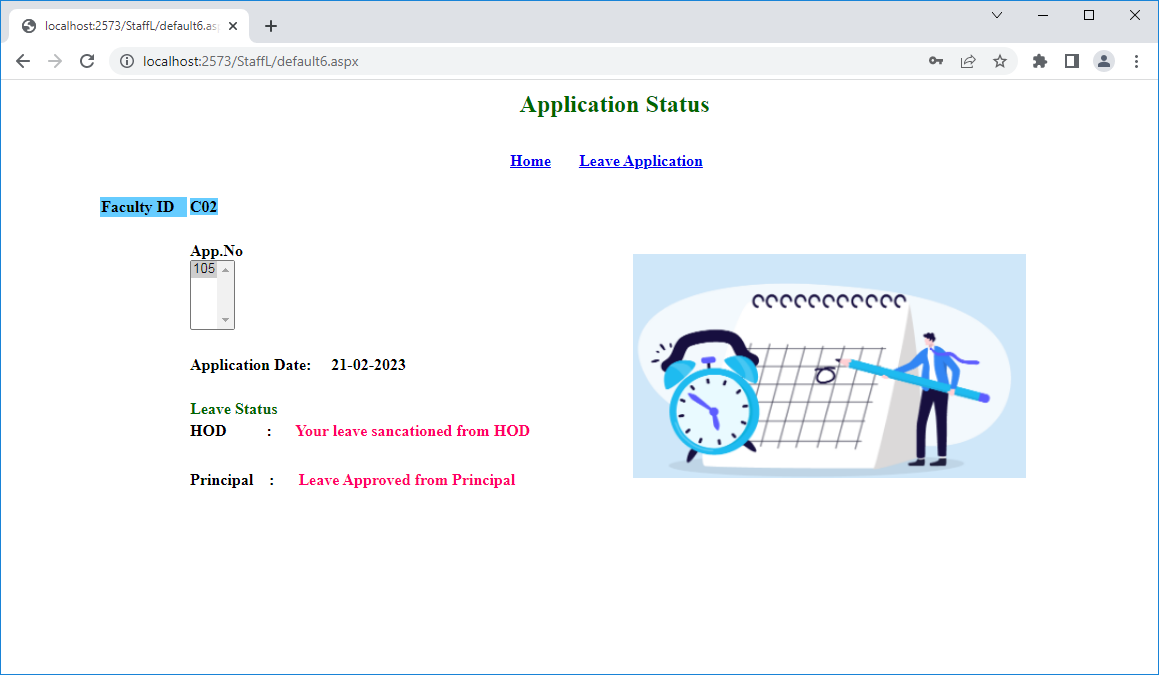












**SYSTEM TESTING**

**8. SYSTEM TESTING**

Testing is the process of executing a program with the intent of finding any errors. A good test of course has the high probability of finding a yet undiscovered error. A successful testing is the one that uncovers a yet undiscovered error.

A test is vital to the success of the system. System test makes a logical assumption that if all parts of the system are correct, then goal will be successfully achieved. The candidate system is subjected to a variety of tests online like responsiveness, its value, stress and security. A series of tests are performed before the system is ready for user acceptance testing.

**Types of Testing**

The different types of testing are

* Verification Testing
* Validation Testing
* Unit Testing
* Integration Testing
* Output Testing

**8.1 Verification Testing**

The verification testing is used in all forms. Verification is the process of evaluating a system or component to determine whether the product of a given phase satisfies the conditions imposed at start of that phase. Verification takes care of activities to focus on question” Are we building the product right”.

**8.2 Validation Testing**

The validation testing is used in all forms. At the culmination of the black box testing, software is completely assembled as a package. Interfacing errors have been uncovered and correct and final series of test, i.e., validation test begins.

Validation test van is defined with a simple definition that validation succeeds when the software functions in a manner that can be reasonably accepted by the customer.

**8.3Integration testing**

Integration Testing is used all forms. Sample data is input and check all the forms. Data can be lost across an interface, one module can have an adverse effect on the other sub-functions, when combined may not produce the desired functions. Integrated testing is the systematic testing to uncover the errors within the interface. This testing is done with simple data and the developed system has run successfully with this simple data. The need for integrated system is to find the overall system performance.

**8.4 Unit Testing:**

Unit testing focuses verification efforts even in the smallest unit of software design in each module. This is also known as “Module Testing”. The modules of the system are tested separately. This testing is carried out in the programming style itself. In this testing each module is focused to work satisfactorily as regard to expected output from the module. There are some validation checks for the fields.

**8.5Output Testing**

After performing validation testing, the next step is output testing of the proposed system. Since the system cannot be useful if it does not produce the required output. Asking the user about the format in which the system is required tests the output displayed or generated by the system under consideration. Here the output format is considered in two ways. One is on screen format and another one is printed format. The output format on the screen is found to be corrected as the format was designed in the system phase according to the user needs. As for the hard copy the output comes according to the specification requested by the user. Here the output testing does not result in any correction in the system.

Taking various kinds of data plays a vital role in system testing. After preparing the test data, system under study is tested using the test data. While testing, errors are again uncovered and corrected by using the above steps and corrections are also noted for future use. The system has been verified and validated by running test data and live data.

**CONCLUSION**

**9. CONCLUSION**

The “**STAFF LEAVE MANAGEMENT SYSTEM**” provides efficient services to approve staff leave.It concludes Staff Registration, Leave Application, Leave Status and Leave Approval.

It is easy to operate and saves the time by eliminating all the manual works. This analysis is used to staff can easily apply their leave through online and view their application status.

**FUTURE ENHANCEMENT**

**10. FUTURE ENHANCEMENT**

Staff Leave Management System is the highly development for leave application apply and approve. This refers to adding, modifying redevelopment or support changes in the specification. If it is necessary to keep up with changing user needs anoperational environment.

The project may be complete and satisfy with all necessary enhancements. The new ideas can be taken into and code is capable of being altered, as to so enhance the project.

**BIBLIOGRAPHY**

**11. BIBLIOGRAPHY**

* **Beginners SQL Server 2008 for Developers - Edition : 1st Edition**
* **Author :** Robin Dewson**, Publisher :** Kinetic
* **Professional Microsoft SQL server 2008 Administration - Edition : 2nd Edition**
* **Author :** [**Ketan Patel**](https://www.google.co.in/search?sa=X&biw=1280&bih=899&q=Ketan+Patel&stick=H4sIAAAAAAAAAOPgE-LVT9c3NEw2M8gyNizKUOIBcTOSi9KLzLLStWSyk630k_Lzs_XLizJLSlLz4svzi7KtEktLMvKLFrFye6eWJOYpBCSWpObsYGXcxc7EwQgAUrolzlMAAAA&ved=2ahUKEwj0lYHg9eX3AhVCSWwGHf77BScQmxMoAXoECEwQAw)**,** [**Brian Knight**](https://www.google.co.in/search?sa=X&biw=1280&bih=899&q=Brian+Knight&stick=H4sIAAAAAAAAAOPgE-LVT9c3NEw2M8gyNizKUIJw05MzKvMMLM21ZLKTrfST8vOz9cuLMktKUvPiy_OLsq0SS0sy8osWsfI4FWUm5il452WmZ5TsYGXcxc7EwQgA8hMayVUAAAA&ved=2ahUKEwj0lYHg9eX3AhVCSWwGHf77BScQmxMoAnoECEwQBA)**,** [**Wayne Snyder**](https://www.google.co.in/search?sa=X&biw=1280&bih=899&q=Wayne+Snyder&stick=H4sIAAAAAAAAAOPgE-LVT9c3NEw2M8gyNizKUIJwk8rTyk3KLY20ZLKTrfST8vOz9cuLMktKUvPiy_OLsq0SS0sy8osWsfKEJ1bmpSoE51WmpBbtYGXcxc7EwQgAvzLe-VUAAAA&ved=2ahUKEwj0lYHg9eX3AhVCSWwGHf77BScQmxMoA3oECEwQBQ)**,** [**Ross LoForte**](https://www.google.co.in/search?sa=X&biw=1280&bih=899&q=Ross+LoForte&stick=H4sIAAAAAAAAAOPgE-LVT9c3NEw2M8gyNizKUIJyzY3jkywzKrRkspOt9JPy87P1y4syS0pS8-LL84uyrRJLSzLyixax8gTlFxcr-OS75ReVpO5gZdzFzsTBCAAHtrBlVQAAAA&ved=2ahUKEwj0lYHg9eX3AhVCSWwGHf77BScQmxMoBHoECEwQBg)

**Publisher :Paperback**

* **Programming ASP .NET Core- Edition : 2ndEdition**

**- Author :**Dino Esposito, **Publisher : Microsoft Press**

* **ASP.NET Core 5 for Beginners: Kick Start Your ASP .NET – Edition : 1st Edition**
* **Author :** [**Andreas Helland**](https://www.google.com/search?client=firefox-b-d&sa=X&q=Andreas+Helland&stick=H4sIAAAAAAAAAOPgE-LVT9c3NMzJSTMozjJLUoJwCwsNLTMKzQy0ZLKTrfST8vOz9cuLMktKUvPiy_OLsq0SS0sy8osWsfI75qUUpSYWK3ik5uQk5qXsYGXcxc7EwQgA3Uh3g1gAAAA&ved=2ahUKEwim-Iee_eX3AhXlGKYKHYzdBSkQmxMoAHoECGsQAg)**,** [**Jeffrey Chilberto**](https://www.google.com/search?client=firefox-b-d&sa=X&q=Jeffrey+Chilberto&stick=H4sIAAAAAAAAAOPgE-LVT9c3NMzJSTMozjJLUoJwM6osk9Kyciy0ZLKTrfST8vOz9cuLMktKUvPiy_OLsq0SS0sy8osWsQp6paalFaVWKjhnZOYkpRaV5O9gZdzFzsTBCAAvWQk4WgAAAA&ved=2ahUKEwim-Iee_eX3AhXlGKYKHYzdBSkQmxMoAXoECGsQAw)**,** [**Vincent Maverick Durano**](https://www.google.com/search?client=firefox-b-d&sa=X&q=Vincent+Maverick+Durano&stick=H4sIAAAAAAAAAOPgE-LVT9c3NMzJSTMozjJLUoJw0wyq0vIMDSq0ZLKTrfST8vOz9cuLMktKUvPiy_OLsq0SS0sy8osWsYqHZeYlp-aVKPgmlqUWZSZnK7iUFiXm5e9gZdzFzsTBCAAt06F8YAAAAA&ved=2ahUKEwim-Iee_eX3AhXlGKYKHYzdBSkQmxMoAnoECGsQBA)**Publisher: Paperback**
* **Beginning ASP .NET – Edition : 2nd Edition**
* **Author : Richard Blairetall Publisher: Wrox Press**