# 

**SESSION 2015-2016**

# Submitted to H.P. University in the Partial Fulfillment of Bachelor of Computer Application Degree

**UNDER THE GUIDANCE OF**: **SUBMITTED BY:**

While working on our minor project we encountered many new experiences and also several difficulties. But as each such point, we found someone is coming for our help with his full co-operation. I take the privilege in conveying heartiest gratitude to all those help unable to complete this minor project.I “sincerely and heartedly” very much thankful to our project guide, {***TEACHER NAMe},*** for constant help and valuable suggestions throughout the project. Her advice has greatly enhanced the worth this report.

# I also appreciate and comment the supportive role – played by other Friends for helping us to complete the requirement of minor project.

# After lot of days today we can say that our project is now complete and ready for implementation.

# {Name}

# {Class}

# {ROLL NO}

This is to certify that this “HOTEL MANAGEMENT SYSTEM” has been designed and developed by {NAME} student of {CLASS NAME} under the guidance of {***TEACHER NAME}***

Date: (Signature)

**STUDENT’S PROFILE**

***Team Members:***

**Course**

**Project Title HOTEL MANAGEMENT SYSTEM**

**Project Guide**

**INDEX**

|  |  |
| --- | --- |
| **Topic** | **Page No** |
| Introduction to project |  |
| Study of Existing System |  |
| System Specification Requirement |  |
| Data Flow Diagrams |  |
| Database Detail |  |
| Front End Used |  |
| Back End Used |  |
| Forms Detail |  |
| Coding |  |
| Testing and Implementation |  |
| Bibliography |  |

Hotel management system is a system that provides us to reserving rooms, checking whether the rooms are vacant are or not etc by using online browsing. This system is very useful to all especially for business people.

For Business people they don’t have sufficient time for these then they can use these type of online Hotel Management Systems. By this project we will reduce the faults in bills of their expenditure and decrease time of delay to give the bills to the customers. We can also save the bills of the customer. By this project we can also include all the taxes on the bills according to their expenditures. It has a scope to reduce the errors in making the bills. Computerized bill can be printed within fraction of seconds. Online ordering of Booking is possible by using this software.

**OBJECTIVE**

During the past several decades’ personnel function has been transformed from a relatively obscure record keeping staff to central and top level management function. There are many factors that have influenced this transformation like technological advances, professionalism, and general recognition of human beings as most important resources.

 A computer based management system is designed to handle all the primary information required to calculate monthly statements. Separate database is  maintained to handle all the details required for the correct statement calculation and generation.

* This project intends to introduce more user friendliness in the various activities such as record updation, maintenance, and searching.
* The searching of record has been made quite simple as all the details of the customer can be obtained by simply keying in the identification of that customer.
* Similarly, record maintenance and updation can also be accomplished by using the identification of the customer with all the details being automatically generated. These details are also being promptly automatically updated in the master file thus keeping the record absolutely up-to-date.
* The entire information has maintained in the database or Files and whoever wants to retrieve can’t retrieve, only authorization user can retrieve the necessary information which can be easily be accessible from the file.

* It is complete manual System.
* Due to manual process, it requires more time for completion of any work
* In this system availability of vacant room status is to checked manually through the registers.
* Each an every entry has to be search manually via register.

**SOFTWARE SPECIFICATIONS**

Operating System: Windows 7/8/10

Font end : Visual Basic.NET

Back end : MS Access

**HARDWARE SPECIFICATIONS**

Processor : X86 Compatible processor with 1.7

GHz Clock speed

RAM : 1GB or more

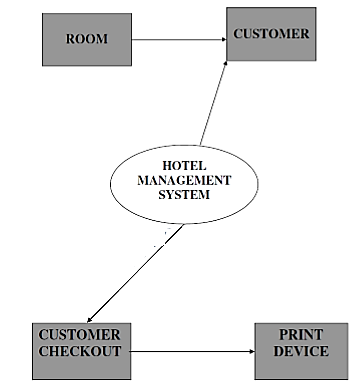
Hard disk: 20 GB or more

Monitor : VGA/SVGA

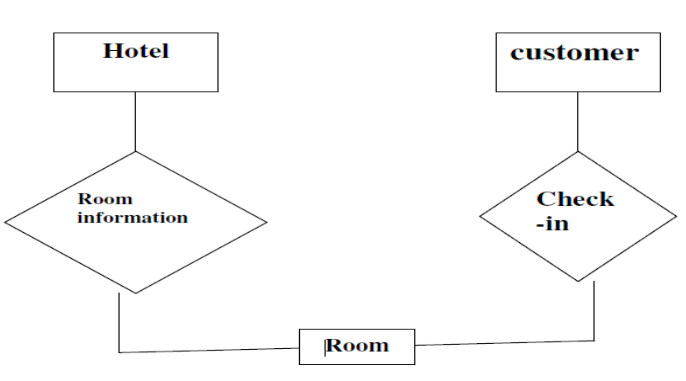
Keyboard: 104 Keys

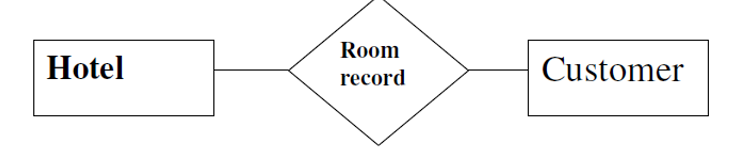
Mouse : 2 buttons/ 3 buttons

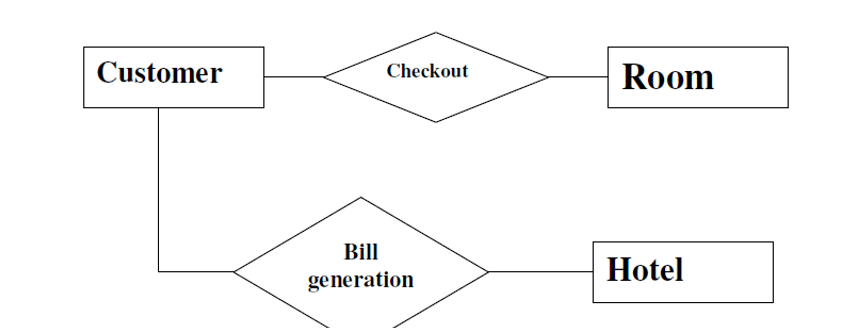
**CONTEXT DIAGRAM:-**

****

**ER DIAGRAM:-**

****

****

****

Visual Basic .NET (VB.NET) is a multi-paradigm, object-oriented programming language, implemented on the .NET Framework. Microsoft launched VB.NET in 2002 as the successor to its original Visual Basic language. Although the ".NET" portion of the name was dropped in 2005, this article uses "Visual Basic [.NET]" to refer to all Visual Basic languages releases since 2002, in order to distinguish between them and the classic Visual Basic. Along with Visual C#, it is one of the two main languages targeting the .NET framework.

Microsoft's integrated development environment (IDE) for developing in Visual Basic .NET language is Visual Studio. Most of Visual Studio editions are commercial; the only exceptions are Visual Studio Express and Visual Studio Community, which are freeware. In addition, .NET Framework SDK includes a freeware command-line compiler called vbc.exe. Mono also includes a command-line VB.NET compiler.

**SYNTAX:-**

VB.NET uses statements to specify actions. The most common statement is an expression statement, consisting of an expression to be evaluated, on a single line. As part of that evaluation, functions or subroutines may be called and variables may be assigned new values. To modify the normal sequential execution of statements, VB.NET provides several control-flow statements identified by reserved keywords. Structured programming is supported by several constructs including two conditional execution constructs (If … Then … Else … End If and Select Case ... Case ... End Select ) and three iterative execution (loop) constructs (Do … Loop, For … To, and For Each) . The For … To statement has separate initialisation and testing sections, both of which must be present. (See examples below.) The For Each statement steps through each value in a list.

In addition, in Visual Basic .NET:

There is no unified way of defining blocks of statements. Instead, certain keywords, such as "If … Then" or "Sub" are interpreted as starters of sub-blocks of code and have matching termination keywords such as "End If" or "End Sub".

Statements are terminated either with a colon (":") or with the end of line. Multiple line statements in Visual Basic .NET are enabled with “\_" at the end of each such line. The need for the underscore continuation character was largely removed in version 10 and later versions.

The equals sign ("=") is used in both assigning values to variable and in comparison.

Round brackets (parentheses) are used with arrays, both to declare them and to get a value at a given index in one of them. Visual Basic .NET uses round brackets to define the parameters of subroutines or functions.

A single quotation mark ('), placed at the beginning of a line or after any number of space or tab characters at the beginning of a line, or after other code on a line, indicates that the (remainder of the) line is a comment.

**Why I prefer Visual Basic**

Graphical User Interface:

Interface that use graphics came to be known as Graphical User Interfaces (GUI). These became very popular because the users could identify with the graphics displayed on the screen. In every day life too, we associate more with images than words. Mention the word Print to a person and it invokes the image of a printer.

The primary requirement for an interface presented to us. This is possible if the interface presented is such that it emulates real life.

Visual Programming:

Visual Programming aims at providing the user with an interface that is intuitive and easy-to-use. In developing such an interface, the programmer employs user-friendly features such as windows, menus, buttons and list boxes.

A visual programming environment provides all features that are required to develop a graphical user interface as ready-to-use components. The programmer does not have to write code to create and display commonly required user-friendly features each time around.

**Advantages of Visual Programming:**

1. Visual programming enables Visual development of graphical user interfaces which are easy to use and easy to learn.

2. A programmer need not write code to display the required component. The visual programming environment displays a list of available components. The programmer picks up the required component from this list to display it.

3. Component can be moved, resized and even deleted, if so required.

4. There is no restriction on the number of controls that can be used.

5. The interface components provided by the visual programming environment have some code built into them.

6. The programmer can create the user interface visually; he can align, move or size the components as required without having to resort to writing code.

**Disadvantages of Visual Programming:**

While visual programming makes it very simple to create complex user interfaces, it suffers from some disadvantages:

1. As the name implies, the entire process of developing an application using a visual development environment is visual. Thus the development environment in itself is highly graphical is nature and therefore requires more memory.

2. Visual development environment require computers of a highs configuration in comparison to the conventional programming tools:

> Larger capacity hard disk

> More RAM

Faster Processor

3. Primarily, visual development environment can be used only with GUI operating systems such as windows.

**Events:**

A major part of the interactions between people in everyday life is in the form of events and responses to events. An event is any action directed at the application. An event driven application operates by responding to the user events. The ability to recognize events is built into interface components. The events that an interface component recognizes are different for each type of component.

A database is a collection of information that's related to a particular subject or purpose, such as tracking customer orders or maintaining a collection. If your database isn't stored on a computer, or only parts of it are, you may be tracking information from a variety of sources that you have to coordinate and organize yourself.

Access database files

Using Microsoft Access, you can manage all your information from a single database file. Within the file, you can use:

· Tables to store your data.

· Queries to find and retrieve just the data you want.

· Forms to view, add, and update data in tables.

· Reports to analyze or print data in a specific layout.

· Data access pages to view, update, or analyze the database's data from the Internet or an intranet.

**Tables and relationship**

To store your data, create one table for each type of information that you track. To bring the data from multiple tables together in a query, form, report, or data access page, define relationships between the tables.

**Relationship in a database**

After you've set up different tables for each subject in your Microsoft Access database, you need a way of telling Microsoft Access how to bring that information back together again. The first step in this process is to define relationship between your tables. After you've done that, you can create queries, forms, and reports to display information from several tables at once.

1. One-to-many relationship
2. Many-to-many relationship
3. One-to-one relationship

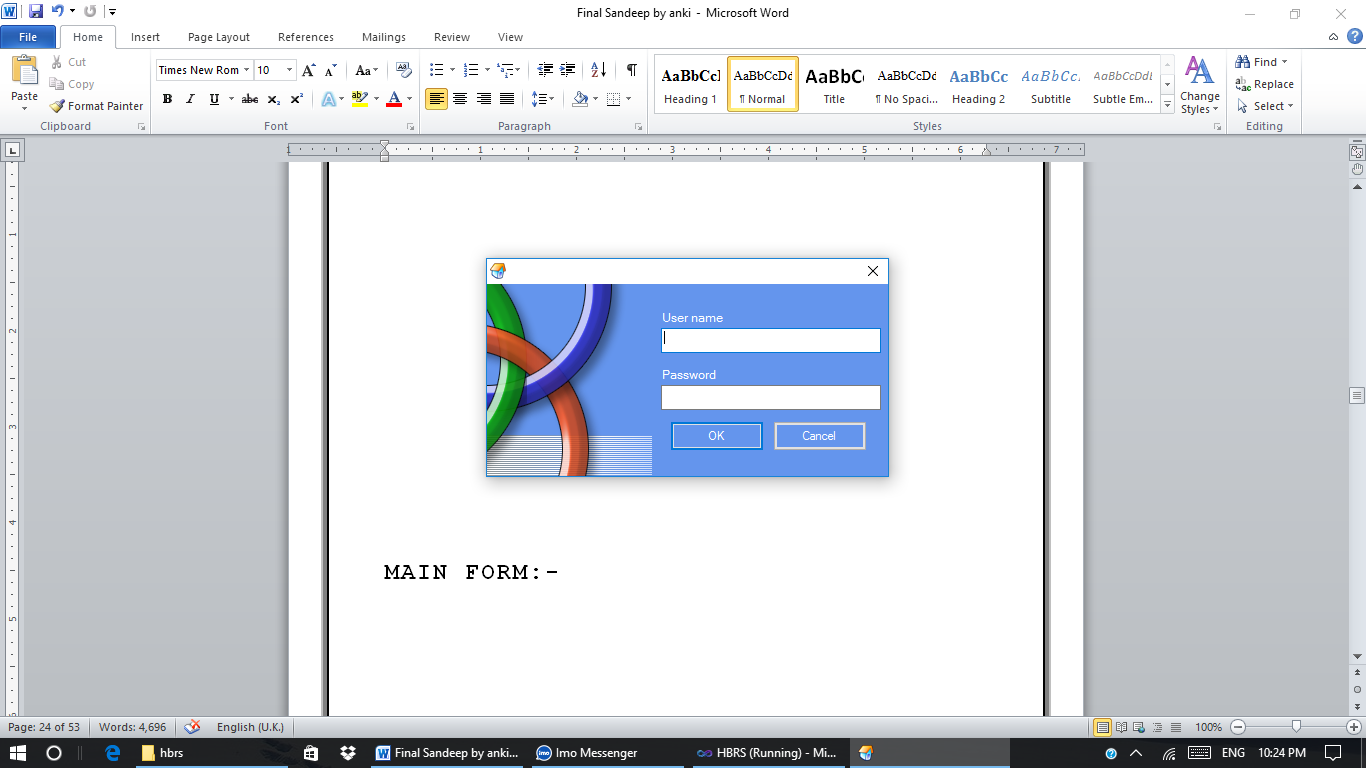
**Queries**

To find and retrieve just the data that meets conditions that you specify, including data from multiple tables, create a query. A query can also update or delete multiple records at the same time, and perform predefined or custom calculations on your data.

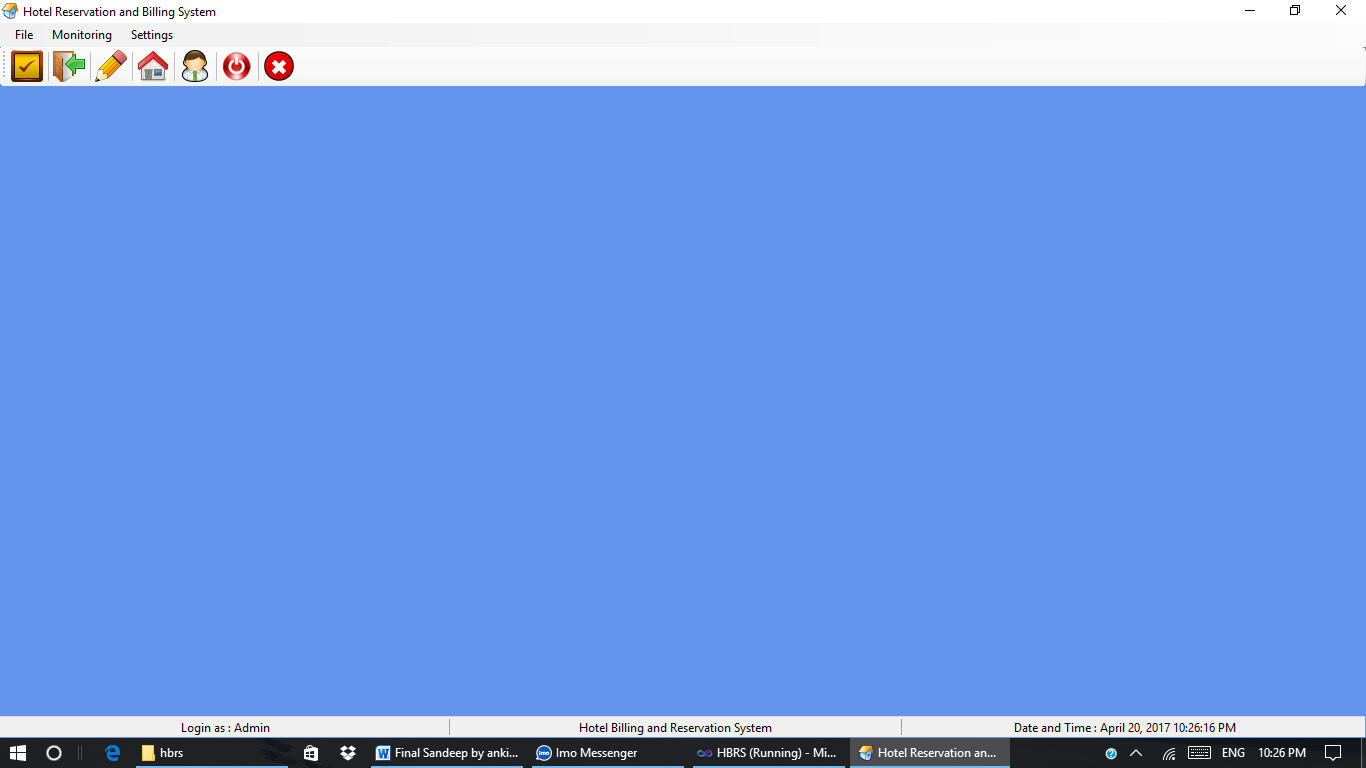
About designing a database Good database design ensures that your database is easy to maintain. You store data in tables and each table contains data about only one subject, such as customers. Therefore, you update a particular piece of data, such as an address, in just one place and that change automatically appears throughout the database.

A well-designed database usually contains different types of queries that show the information you need. A query might show a subset of data, such as all customers in Delhi, or combinations of data from different tables, such as order information combined with customer information.

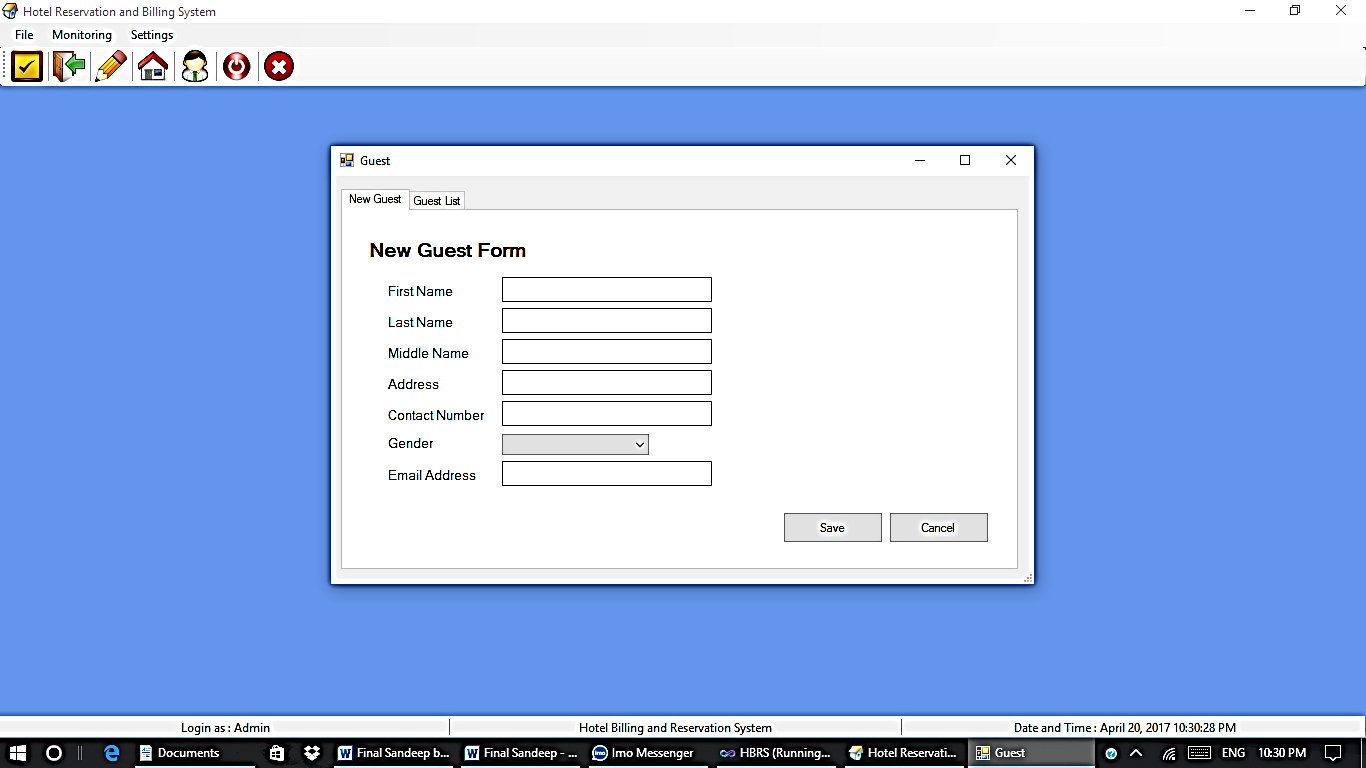
**LOGIN FORM:-**



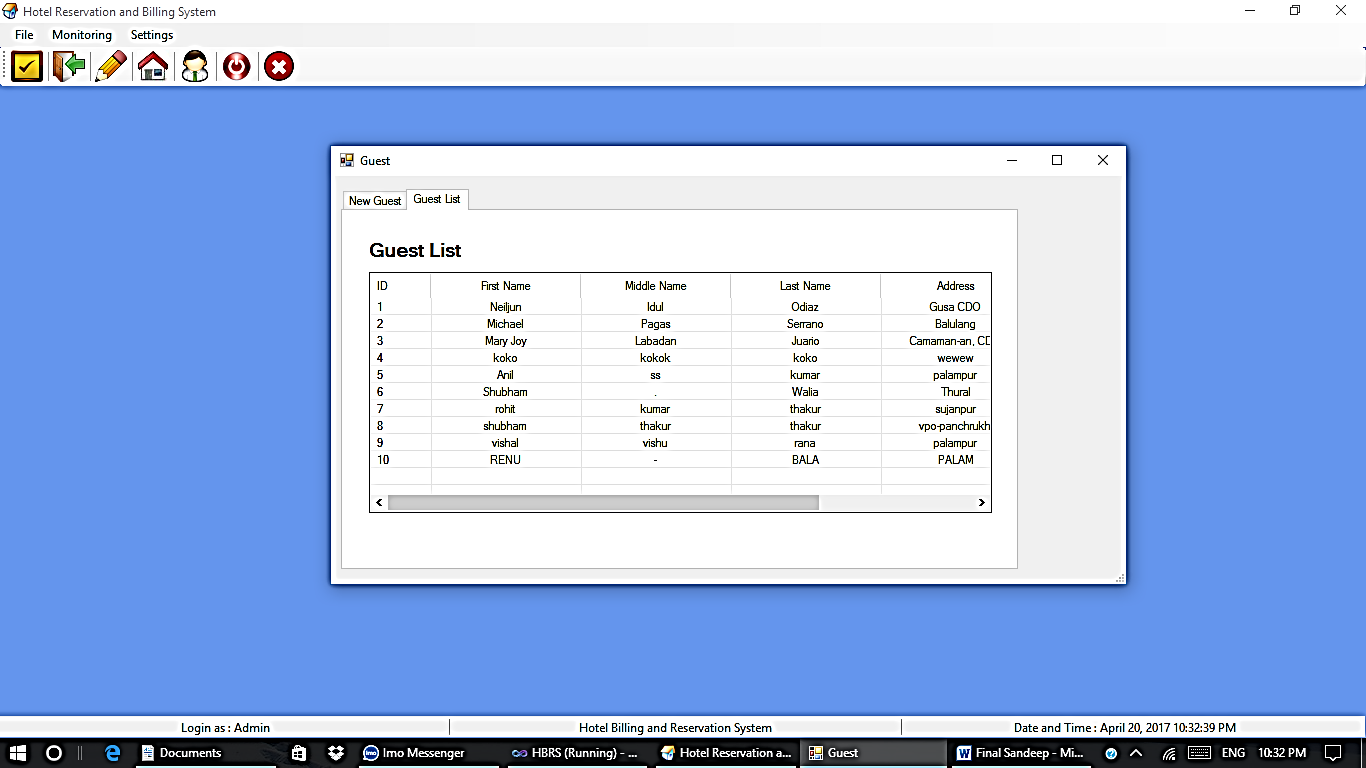
**MAIN FORM:-**



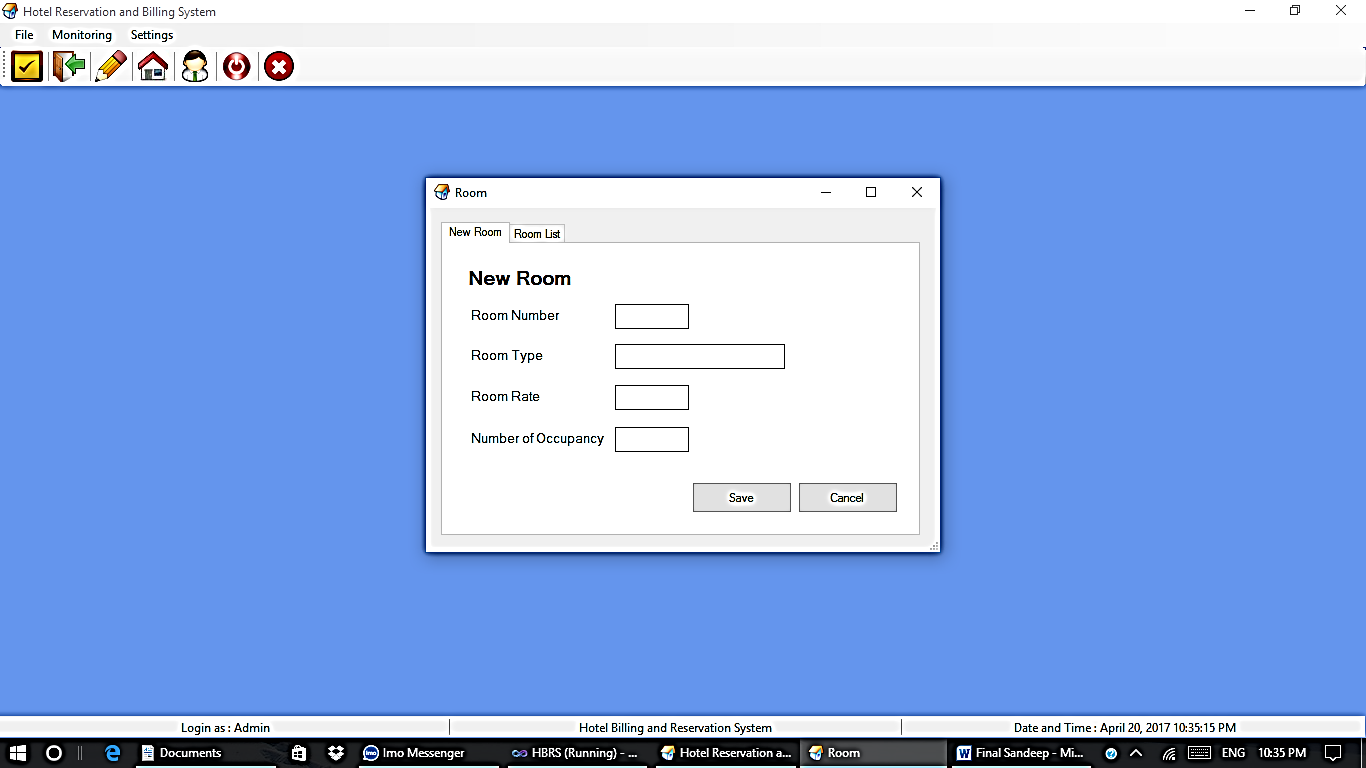
**NEW GUEST FORM:-**



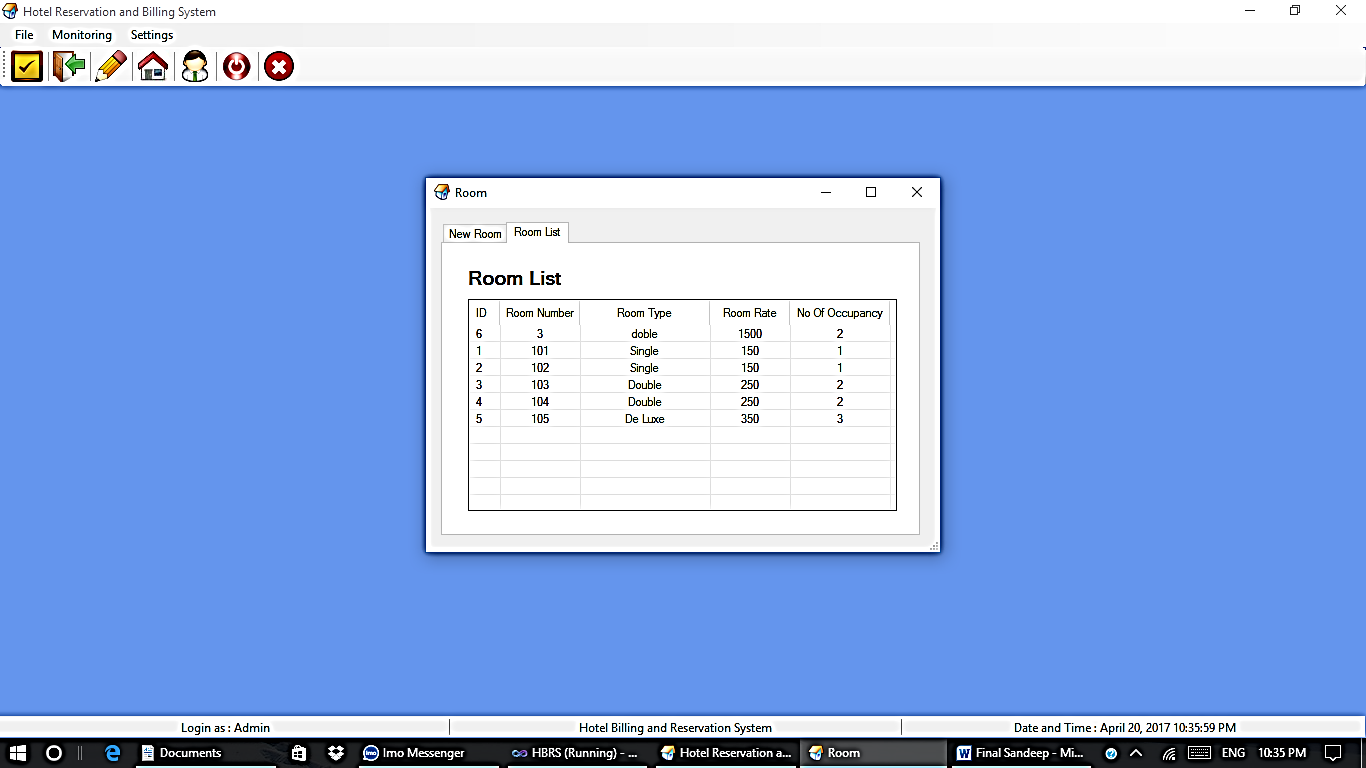
**GUEST LIST FORM:-**



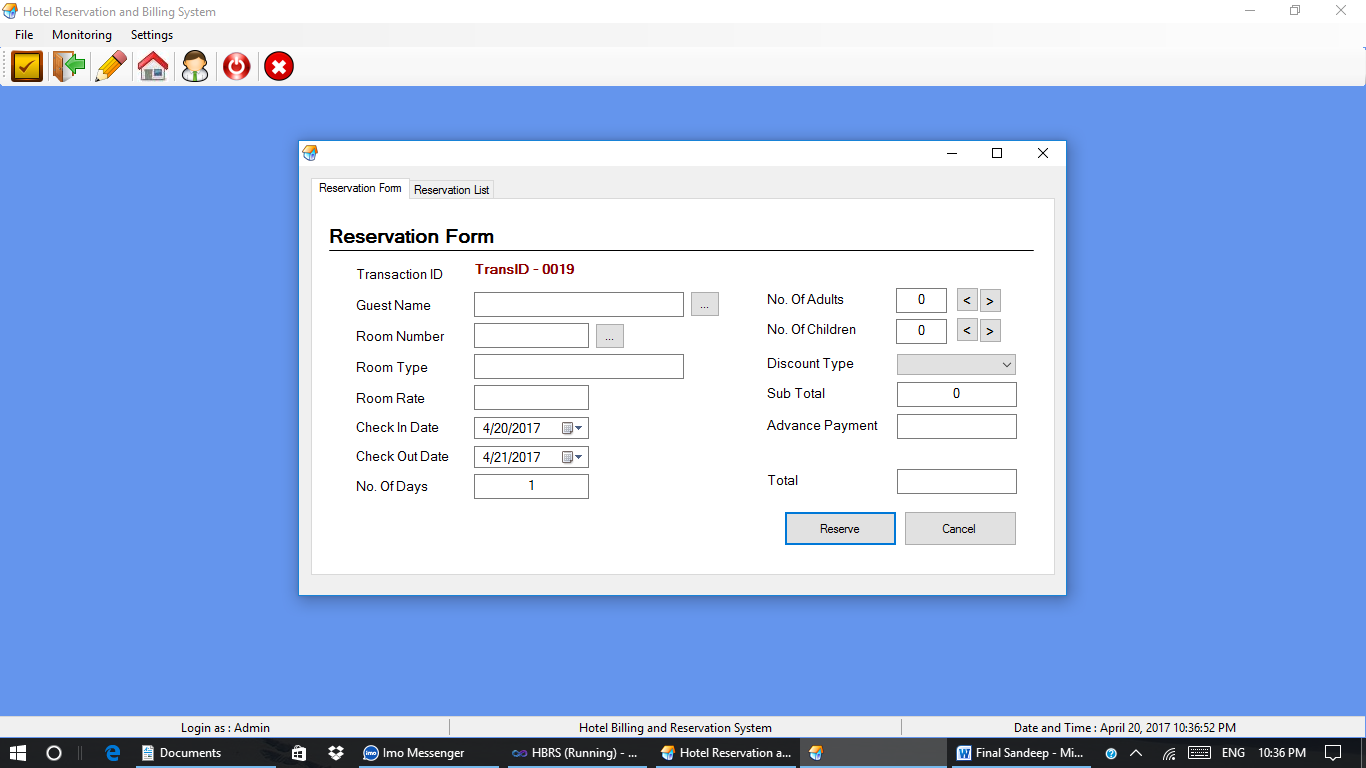
**NEW ROOM FORM:-**



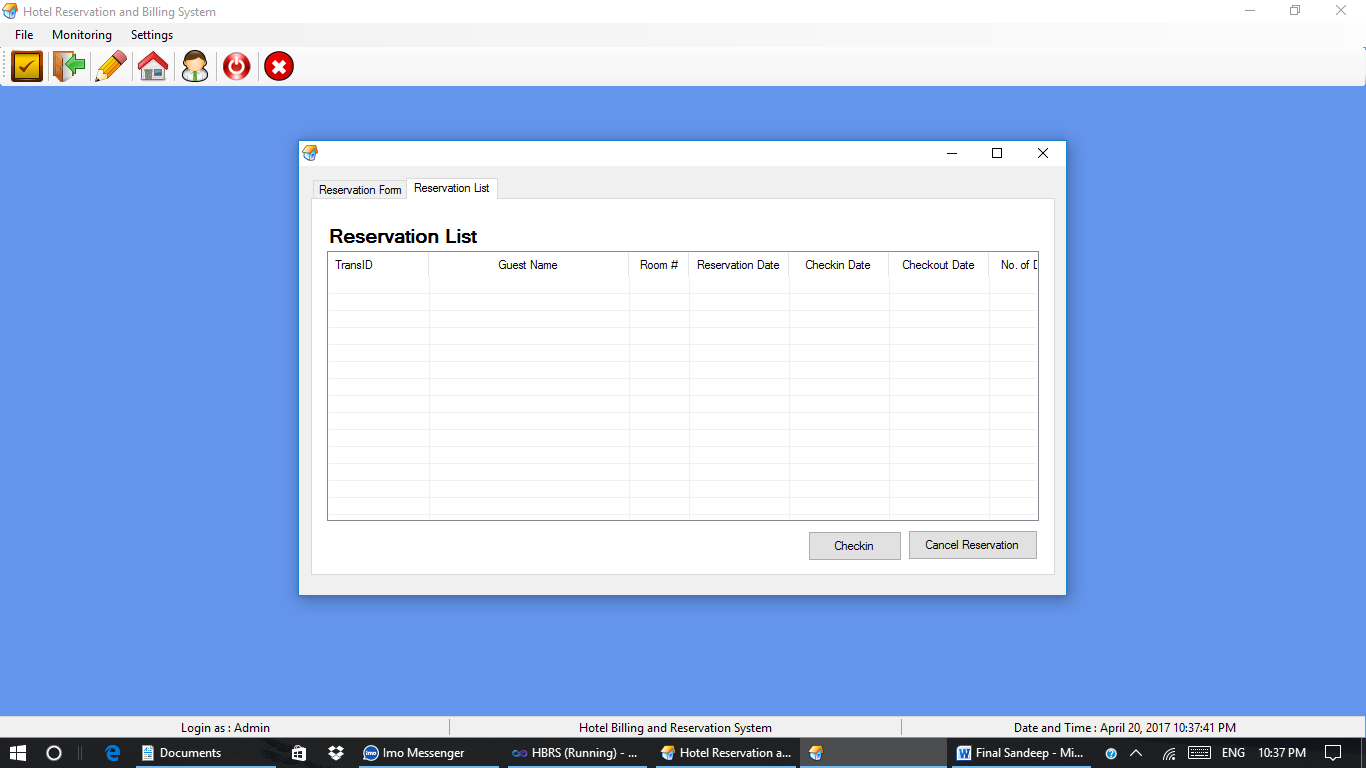
**ROOM LIST FORM:-**



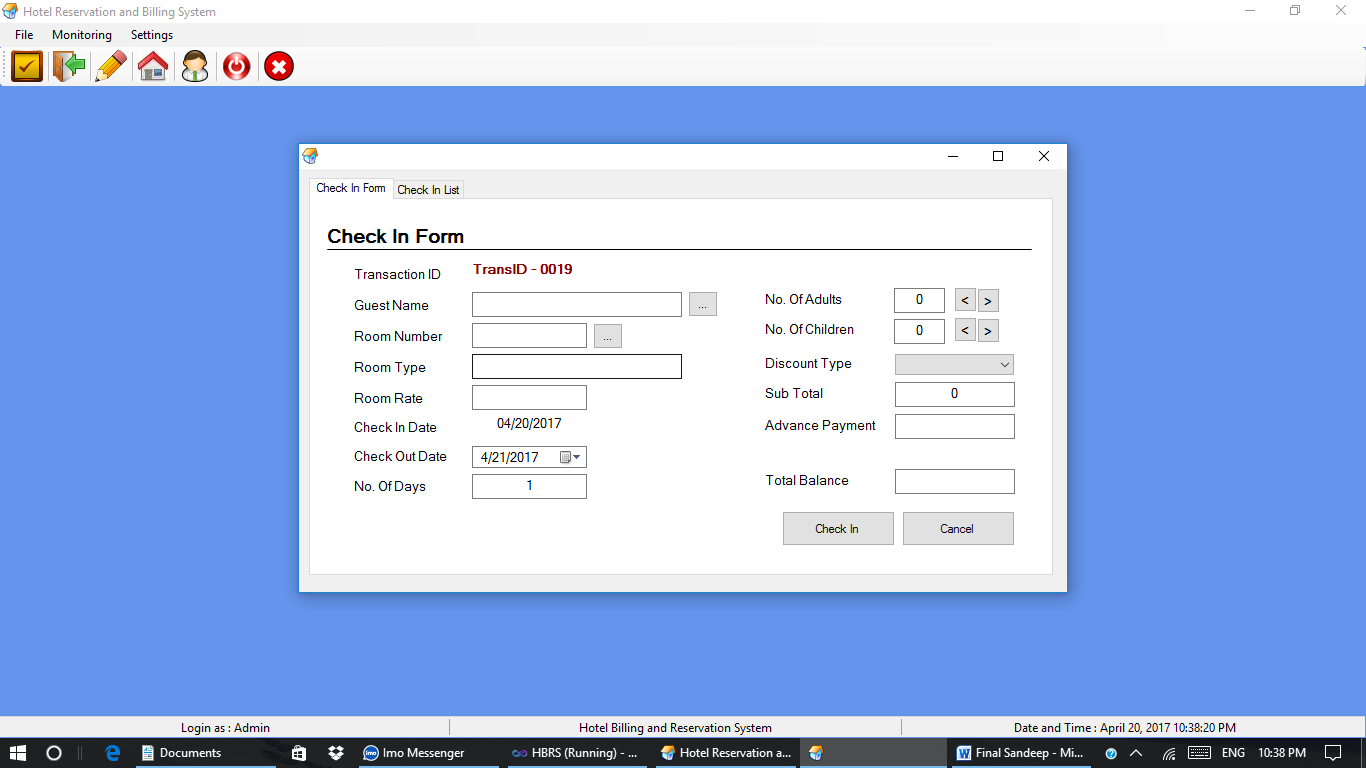
**RESERVATION FORM:-**



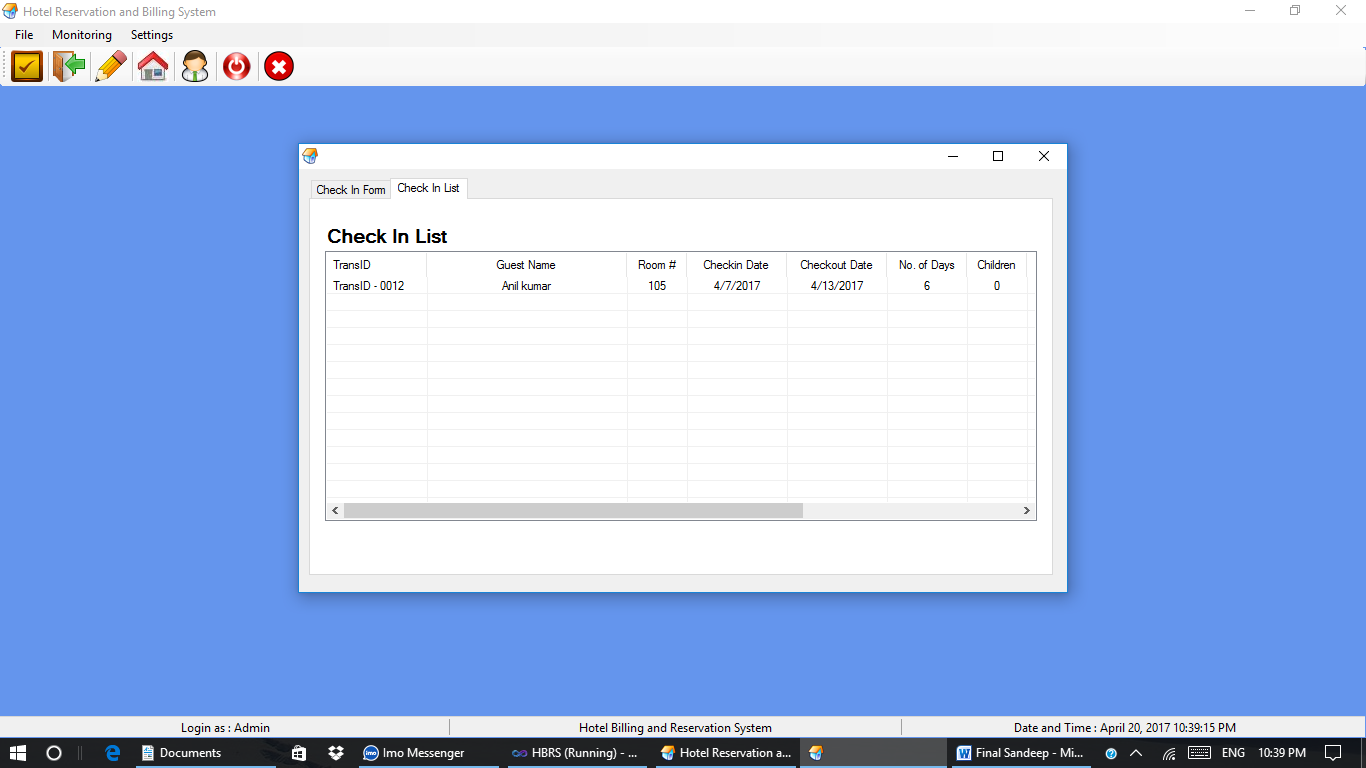
**RESERVATION LIST FORM:-**



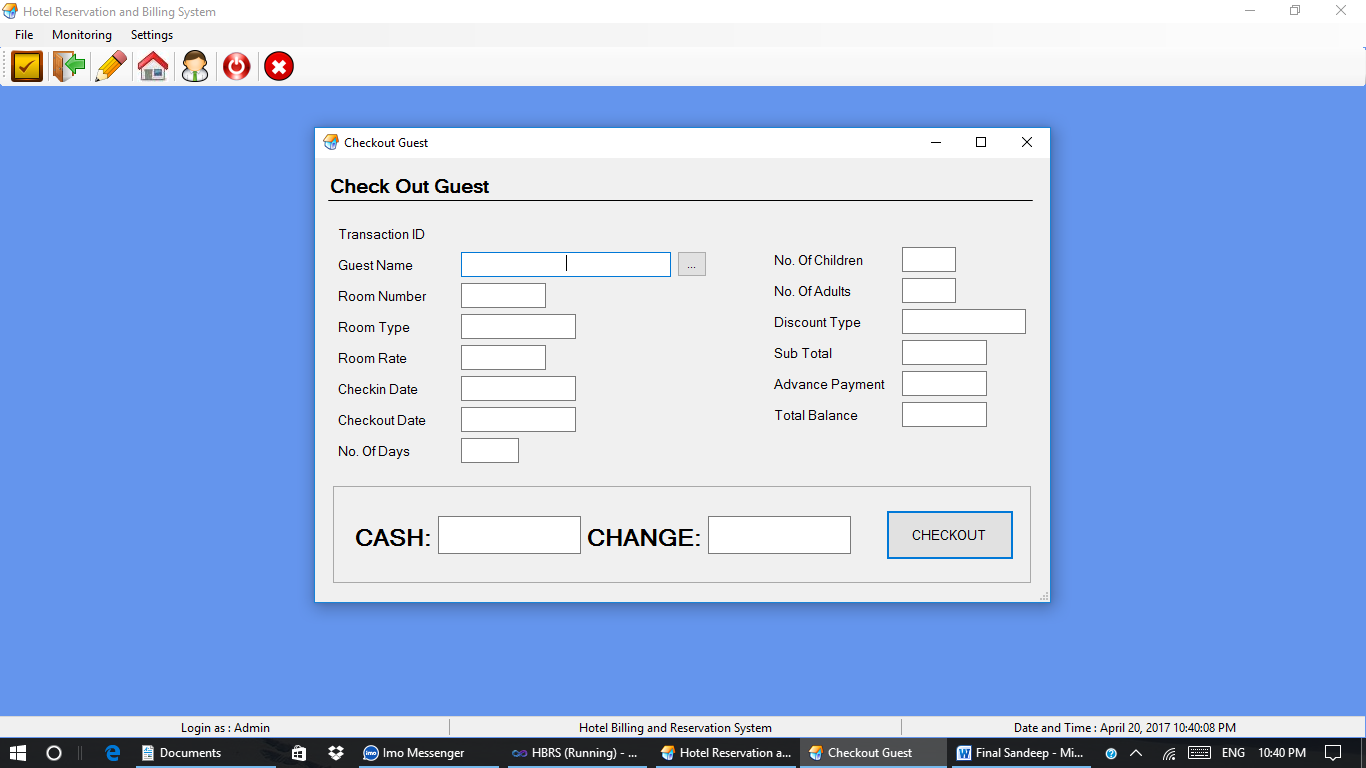
**CHECK IN FORM:-**



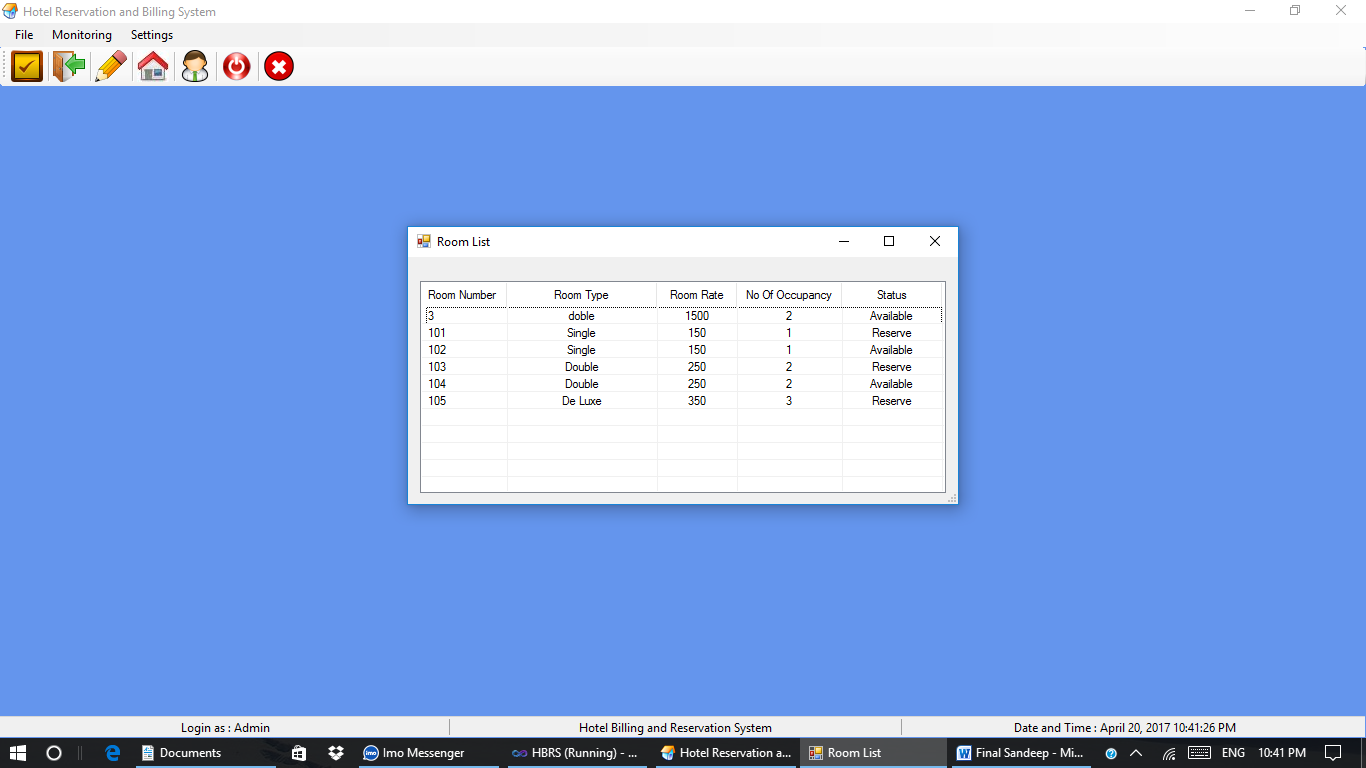
**CHECK IN LIST FORM:-**



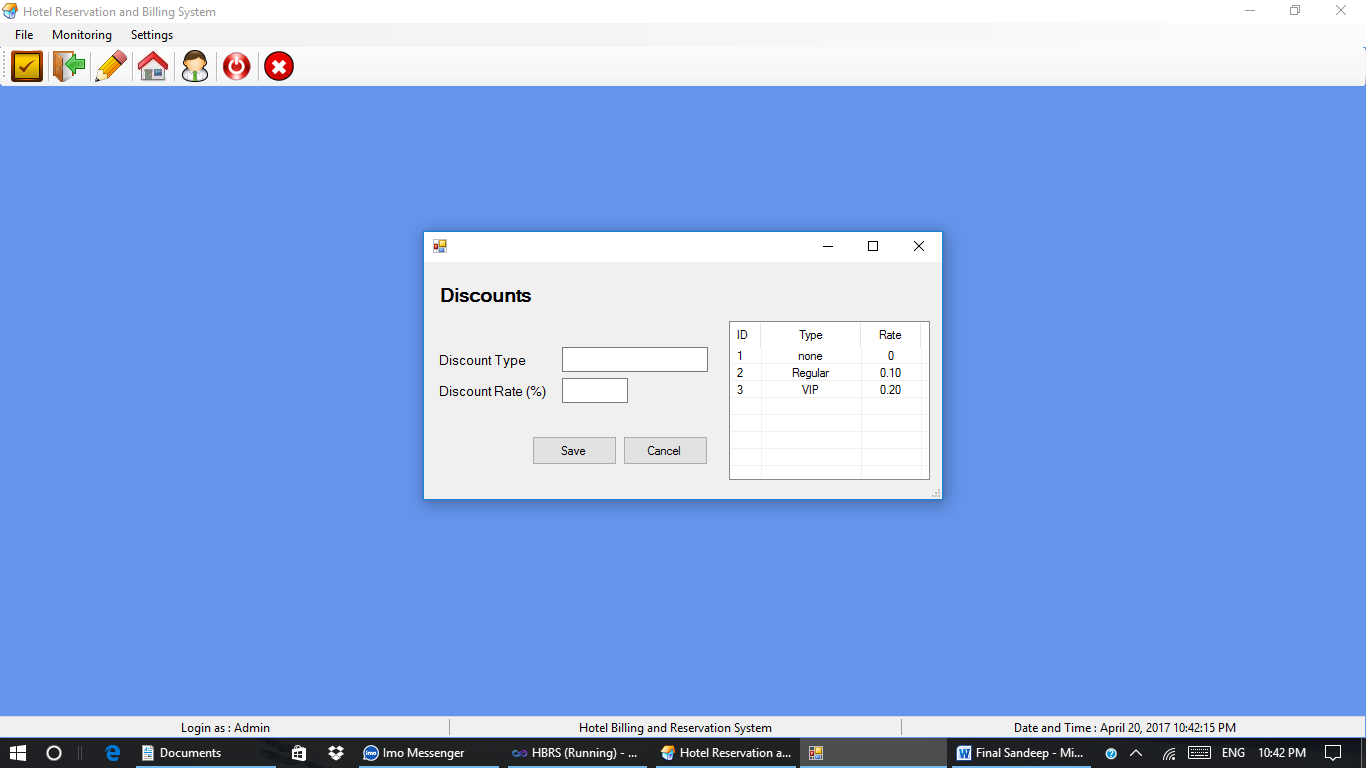
**CHECK OUT GUEST FORM:-**



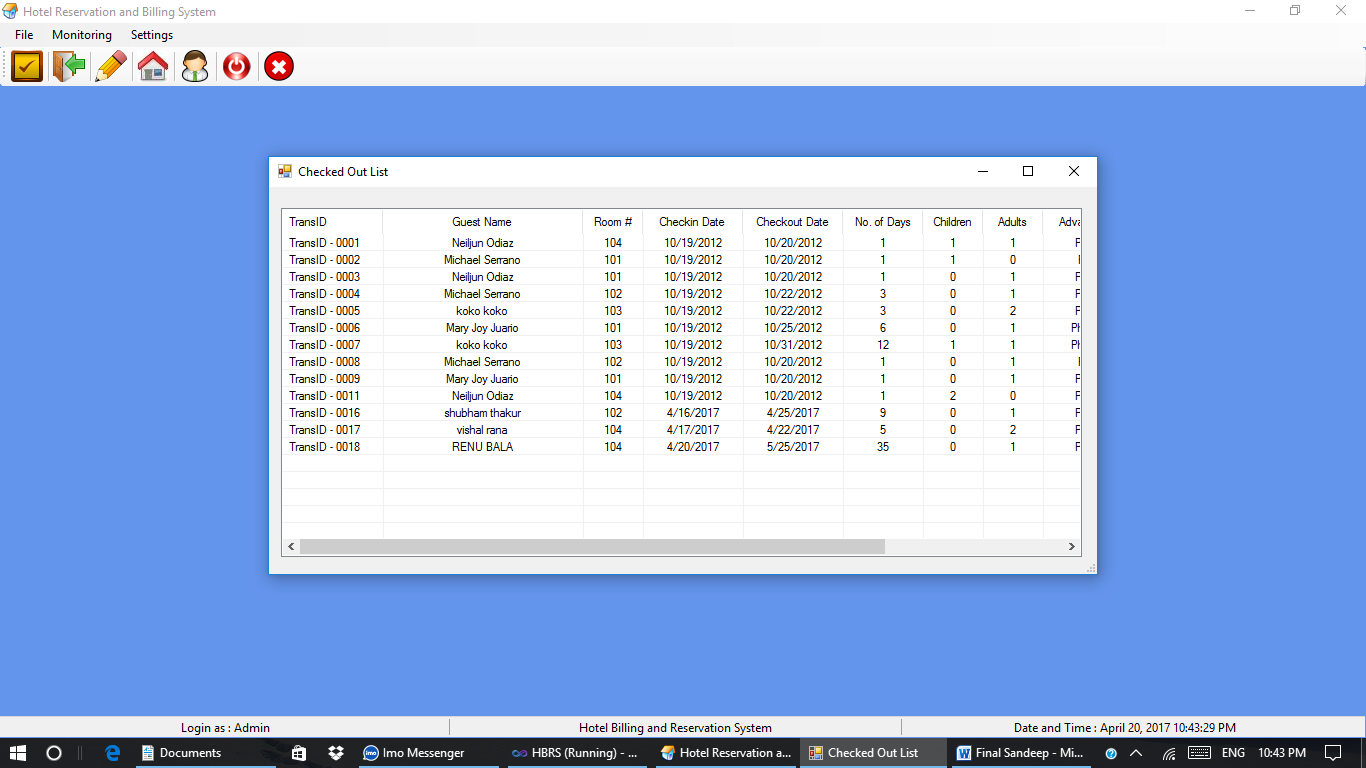
**ROOM LIST FORM:-**



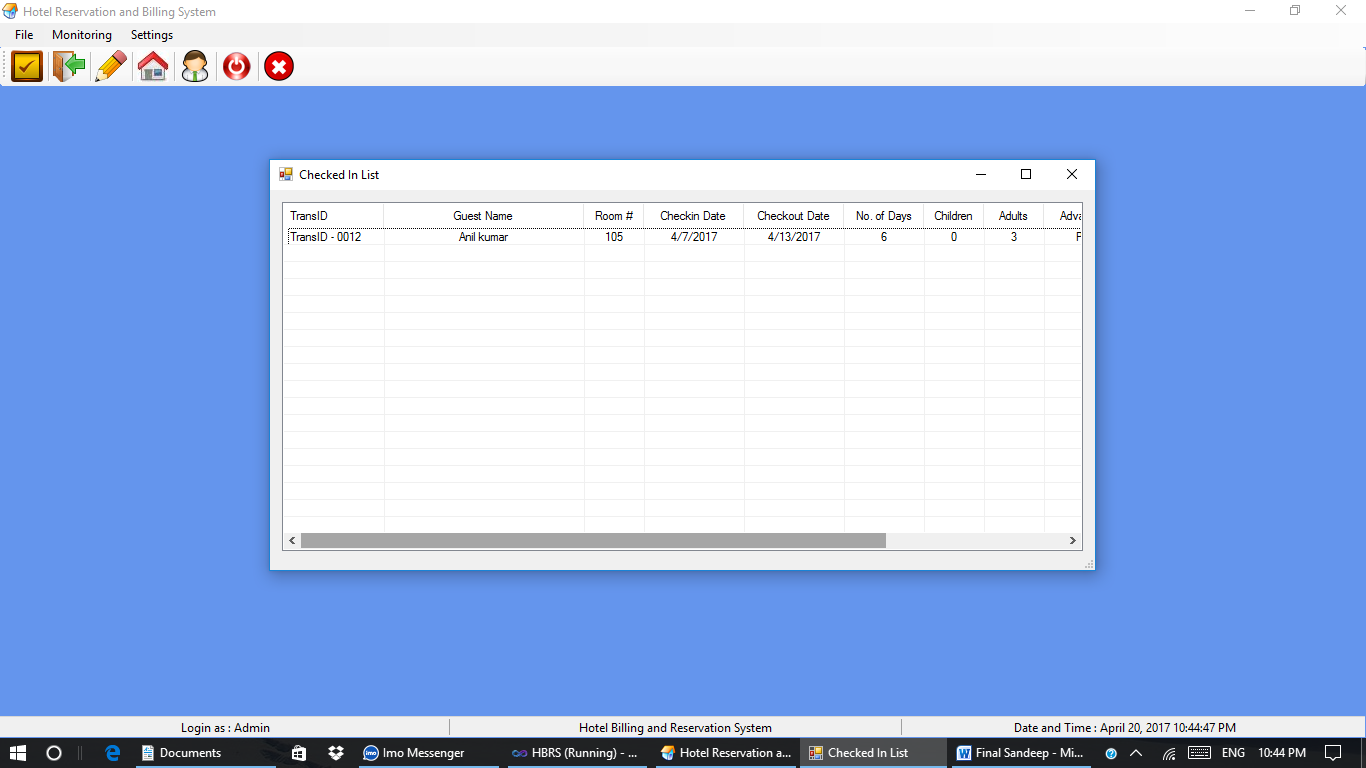
**ADD/UPDATE DISCOUNT FORM:-**



**CHECKED OUT LIST FORM:-**



**CHECKED IN LIST FORM:-**



**CODING**

**Login form:-**

Imports System.Data.OleDb

Public Class frmLogin

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

If Trim(UsernameTextBox.Text) = "" Or Trim(PasswordTextBox.Text) = "" Then

MsgBox("Please Enter Both Fields!", vbInformation, "Note")

Else

con.Open()

Dim sql = "SELECT \* FROM tblUser WHERE username = '" & SafeSqlLiteral(UsernameTextBox.Text, 2) & "' AND password = '" & SafeSqlLiteral(PasswordTextBox.Text, 2) & "'"

Dim cmd = New OleDbCommand(sql, con)

Dim dr As OleDbDataReader = cmd.ExecuteReader

Try

If dr.Read = False Then

MsgBox("Login Failed!", vbCritical, "Note")

Else

MsgBox("Login Successful!", vbInformation, "Note")

frmMain.status.Items(0).Text = "Login as : " & Trim(UsernameTextBox.Text)

Dim datenow As Date = Now

frmMain.status.Items(2).Text = "Date and Time : " & datenow.ToString("MMMM dd, yyyy") & " " & TimeOfDay

con.Close()

Me.Hide()

frmMain.ShowDialog()

End If

Catch ex As Exception

MsgBox(ex.Message)

End Try

con.Close()

End If

End Sub

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

Me.Close()

End

End Sub

End Class

**Main form:-**

Public Class frmMain

Private Sub frmMain\_FormClosing(ByVal sender As Object, ByVal e As System.Windows.Forms.FormClosingEventArgs) Handles Me.FormClosing

Dim exit\_app As String = MsgBox("Exit application?", vbQuestion + vbYesNo, "Exit")

If exit\_app = vbNo Then

e.Cancel = True

Else

End

End If

End Sub

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

frmCheckin.ShowDialog()

End Sub

Private Sub ToolStripButton13\_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles ToolStripButton13.Click

Dim exit\_app As String = MsgBox("Exit application?", vbQuestion + vbYesNo, "Exit")

If exit\_app = vbYes Then

End

End If

End Sub

Private Sub ToolStripButton12\_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles ToolStripButton12.Click

Dim out\_app As String = MsgBox("Logout from application?", vbQuestion + vbYesNo, "Logout")

If out\_app = vbYes Then

con.Close()

Me.Hide()

frmLogin.Show()

End If

End Sub

Private Sub ToolStripButton10\_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles ToolStripButton10.Click

frmGuest.ShowDialog()

End Sub

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

frmRoom.ShowDialog()

End Sub

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

frmReserve.ShowDialog()

End Sub

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

frmCheckin.ShowDialog()

End Sub

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

frmReserve.ShowDialog()

End Sub

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

Dim out\_app As String = MsgBox("Logout from application?", vbQuestion + vbYesNo, "Logout")

If out\_app = vbYes Then

Me.Hide()

frmLogin.Show()

End If

End Sub

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

Me.Close()

End Sub

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

frmCheckout.ShowDialog()

End Sub

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

End Sub

Private Sub FileToolStripMenuItem1\_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles FileToolStripMenuItem1.Click

End Sub

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

frmCheckinListMonitor.ShowDialog()

End Sub

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

frmRoomListMonitor.ShowDialog()

End Sub

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

frmReserveListMonitor.ShowDialog()

End Sub

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

frmCheckOutList.ShowDialog()

End Sub

Private Sub Timer1\_Tick(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Timer1.Tick

Dim datenow As Date = Now

status.Items(2).Text = "Date and Time : " & datenow.ToString("MMMM dd, yyyy") & " " & TimeOfDay

End Sub

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

frmDiscount.ShowDialog()

End Sub

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

frmRoom.ShowDialog()

frmRoom.TabPage2.Select()

End Sub

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

frmGuest.ShowDialog()

End Sub

End Class

**New Guest form:-**

Imports System.Data.OleDb

Public Class frmGuest

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

con.Open()

Dim fname As String = Trim(txtFName.Text)

Dim mname As String = Trim(txtMName.Text)

Dim lname As String = Trim(txtLName.Text)

Dim add As String = Trim(txtAddress.Text)

Dim num As String = Trim(txtNumber.Text)

Dim email As String = Trim(txtEmail.Text)

Dim stat As String = "Active"

Dim remark As String = "Available"

If fname = Nothing Or mname = Nothing Or lname = Nothing Or add = Nothing Or num = Nothing Then

MsgBox("Please Fill All Fields", vbInformation, "Note")

Else

Dim add\_guest As New OleDbCommand("INSERT INTO tblGuest(GuestFName,GuestMName,GuestLName,GuestAddress,GuestContactNumber,GuestGender,GuestEmail,Status,Remarks) values ('" &

fname & "','" &

mname & "','" &

lname & "','" &

add & "','" &

num & "','" &

cboGender.Text & "','" &

email & "','" &

stat & "','" &

remark & "')", con)

add\_guest.ExecuteNonQuery()

add\_guest.Dispose()

MsgBox("Guest Added!", vbInformation, "Note")

txtFName.Clear()

txtMName.Clear()

txtLName.Clear()

txtAddress.Clear()

txtNumber.Clear()

txtEmail.Clear()

End If

con.Close()

display\_guest()

End Sub

Private Sub frmGuest\_Load(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles MyBase.Load

display\_guest()

TabControl1.SelectTab(0)

End Sub

Private Sub display\_guest()

con.Open()

Dim Dt As New DataTable("tblGuest")

Dim rs As OleDbDataAdapter

rs = New OleDbDataAdapter("Select \* from tblGuest", con)

rs.Fill(Dt)

Dim indx As Integer

lvGuest.Items.Clear()

For indx = 0 To Dt.Rows.Count - 1

Dim lv As New ListViewItem

lv.Text = Dt.Rows(indx).Item("ID")

lv.SubItems.Add(Dt.Rows(indx).Item("GuestFName"))

lv.SubItems.Add(Dt.Rows(indx).Item("GuestMName"))

lv.SubItems.Add(Dt.Rows(indx).Item("GuestLName"))

lv.SubItems.Add(Dt.Rows(indx).Item("GuestAddress"))

lv.SubItems.Add(Dt.Rows(indx).Item("GuestContactNumber"))

lv.SubItems.Add(Dt.Rows(indx).Item("Status"))

lvGuest.Items.Add(lv)

Next

rs.Dispose()

con.Close()

End Sub

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

txtFName.Clear()

txtMName.Clear()

txtLName.Clear()

txtAddress.Clear()

txtNumber.Clear()

txtEmail.Clear()

End Sub

End Class

**Guest list form:-**

Imports System.Data.OleDb

Public Class frmSelectGuest

Private Sub frmSelectGuest\_Load(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles MyBase.Load

display\_guest()

End Sub

Private Sub display\_guest()

con.Open()

Dim Dt As New DataTable("tblGuest")

Dim rs As OleDbDataAdapter

rs = New OleDbDataAdapter("Select \* from tblGuest WHERE Remarks = 'Available' ", con)

rs.Fill(Dt)

Dim indx As Integer

lvGuest.Items.Clear()

For indx = 0 To Dt.Rows.Count - 1

Dim lv As New ListViewItem

lv.Text = Dt.Rows(indx).Item("ID")

lv.SubItems.Add(Dt.Rows(indx).Item("GuestFName"))

lv.SubItems.Add(Dt.Rows(indx).Item("GuestMName"))

lv.SubItems.Add(Dt.Rows(indx).Item("GuestLName"))

lvGuest.Items.Add(lv)

Next

rs.Dispose()

con.Close()

End Sub

Private Sub lvGuest\_DoubleClick(ByVal sender As Object, ByVal e As System.EventArgs) Handles lvGuest.DoubleClick

frmCheckin.lblGuestID.Text = lvGuest.SelectedItems(0).Text

frmCheckin.txtGuestName.Text = lvGuest.SelectedItems(0).SubItems(1).Text & " " & lvGuest.SelectedItems(0).SubItems(2).Text & " " & lvGuest.SelectedItems(0).SubItems(3).Text

frmReserve.lblGuestID.Text = lvGuest.SelectedItems(0).Text

frmReserve.txtGuestName.Text = lvGuest.SelectedItems(0).SubItems(1).Text & " " & lvGuest.SelectedItems(0).SubItems(2).Text & " " & lvGuest.SelectedItems(0).SubItems(3).Text

Me.Close()

End Sub

End Class

**Reservation form:-**

Imports System.Data.OleDb

Public Class frmReserve

Dim trans\_id As String

Dim id, guest\_id, room\_num As Integer

Private Sub dtCheckOutDate\_ValueChanged(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles dtCheckOutDate.ValueChanged

Dim T As TimeSpan = dtCheckOutDate.Value - dtCheckInDate.Value

If T.Days < 1 Then

dtCheckOutDate.Text = dtCheckInDate.Value.Date.AddDays(1D)

txtDaysNumber.Text = "1"

Else

txtDaysNumber.Text = T.Days

End If

lblTotal.Text = Val(txtRoomRate.Text) \* Val(txtDaysNumber.Text)

txtSubTotal.Text = Val(txtRoomRate.Text) \* Val(txtDaysNumber.Text)

lblDateNow.Text = Now.Date

End Sub

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

frmSelectGuest.ShowDialog()

End Sub

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

frmSelectRoom.ShowDialog()

End Sub

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

Me.Close()

End Sub

Private Sub frmReserve\_FormClosing(ByVal sender As Object, ByVal e As System.Windows.Forms.FormClosingEventArgs) Handles Me.FormClosing

Dim a As String = MsgBox("Cancel Transaction?", vbQuestion + vbYesNo, "Cancel")

If a = vbNo Then

e.Cancel = True

Else

clear\_text()

End If

End Sub

Private Sub clear\_text()

txtGuestName.Clear()

txtRoomNumber.Clear()

txtRoomType.Clear()

txtRoomRate.Clear()

txtChildren.Text = "0"

txtAdults.Text = "0"

cboDiscount.Refresh()

txtAdvance.Clear()

txtSubTotal.Clear()

txtTotal.Clear()

lblDiscountID.Text = ""

lblDiscountRate.Text = ""

lblGuestID.Text = ""

lblAdvancePayment.Text = ""

lblNoOfOccupancy.Text = "0"

Dim time As DateTime = DateTime.Now

Dim format As String = "MM/d/yyyy"

dtCheckInDate.Text = time.ToString(format)

dtCheckOutDate.Text = Now.AddDays(1D)

End Sub

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

clear\_text()

Dim time As DateTime = DateTime.Now

Dim format As String = "MM/d/yyyy"

dtCheckInDate.Text = time.ToString(format)

dtCheckOutDate.Text = dtCheckInDate.Value.Date.AddDays(1D)

lblDateNow.Text = Now.Date

transID()

pop\_discount()

display\_reserve()

End Sub

Public Sub transID()

con.Open()

Dim dt As New DataTable("tblTransaction")

rs = New OleDbDataAdapter("SELECT \* FROM tblTransaction ORDER BY TransID DESC", con)

rs.Fill(dt)

If dt.Rows.Count = 0 Then

txtTransID.Text = "TransID - 0001"

Else

Dim value As Integer = Val(dt.Rows(0).Item("TransID"))

value = value + 1

txtTransID.Text = "TransID - " & value.ToString("0000")

End If

rs.Dispose()

con.Close()

End Sub

Private Sub pop\_discount()

Dim dt As New DataTable

rs = New OleDbDataAdapter("SELECT \* FROM tblDiscount", con)

rs.Fill(dt)

cboDiscount.Items.Clear()

Dim i As Integer

For i = 0 To dt.Rows.Count - 1

cboDiscount.Items.Add(dt.Rows(i).Item("DiscountType"))

Next

rs.Dispose()

End Sub

Private Sub display\_reserve()

Dim Dt As New DataTable("tblGuest")

Dim rs As OleDbDataAdapter

rs = New OleDbDataAdapter("Select \* from tblTransaction, tblGuest, tblDiscount, tblRoom WHERE tblTransaction.GuestID = tblGuest.ID AND tblTransaction.DiscountID = tblDiscount.ID AND tblTransaction.RoomNum = tblRoom.RoomNumber AND tblTransaction.Remarks = 'Reserve'", con)

rs.Fill(Dt)

Dim indx As Integer

lvlreserve.Items.Clear()

For indx = 0 To Dt.Rows.Count - 1

Dim lv As New ListViewItem

Dim getdate As TimeSpan

Dim days, subtotal, total, rate As Integer

Dim discount As Double

Dim value As Integer = Val(Dt.Rows(indx).Item("TransID"))

lv.Text = "TransID - " & value.ToString("0000")

lv.SubItems.Add(Dt.Rows(indx).Item("GuestFName") & " " & Dt.Rows(indx).Item("GuestLName"))

lv.SubItems.Add(Dt.Rows(indx).Item("RoomNum"))

rate = Dt.Rows(indx).Item("RoomRate")

lv.SubItems.Add(Dt.Rows(indx).Item("CheckInDate"))

lv.SubItems.Add(Dt.Rows(indx).Item("CheckOutDate"))

dtIn.Value = Dt.Rows(indx).Item("CheckOutDate")

dtOut.Value = Dt.Rows(indx).Item("CheckInDate")

getdate = dtIn.Value - dtOut.Value

days = getdate.Days

lv.SubItems.Add(days)

lv.SubItems.Add(Dt.Rows(indx).Item("NoOfChild"))

lv.SubItems.Add(Dt.Rows(indx).Item("NoOfAdult"))

lv.SubItems.Add(Dt.Rows(indx).Item("AdvancePayment"))

lv.SubItems.Add(Dt.Rows(indx).Item("DiscountType"))

discount = Val(Dt.Rows(indx).Item("DiscountRate"))

subtotal = (days \* rate) - ((days \* rate) \* discount)

total = Val(subtotal) - Val(Dt.Rows(indx).Item("AdvancePayment"))

lv.SubItems.Add(Val(total))

lvlreserve.Items.Add(lv)

Next

rs.Dispose()

End Sub

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

Dim children As Integer = Val(txtChildren.Text)

Dim adult As Integer = Val(txtAdults.Text)

Dim advance As Integer = Val(txtAdvance.Text)

Dim discount As Integer = Val(lblDiscountID.Text)

Dim reserve As String = "0"

Dim remarks As String = "Reserve"

Dim stat As String = "Active"

If lblGuestID.Text = "GuestID" Or lblGuestID.Text = Nothing Or txtRoomNumber.Text = Nothing Or Val(children + adult) = Nothing Or advance = Nothing Or discount = Nothing Then

MsgBox("Please Fill All Fields", vbInformation, "Note")

Else

Dim a As String = MsgBox("Confirm Reservation Transaction?", vbQuestion + vbYesNo, "Reservation")

If a = vbYes Then

con.Open()

Dim checkin As New OleDbCommand("INSERT INTO tblTransaction(GuestID,RoomNum,CheckInDate,CheckOutDate,ReservationDate,NoOfChild,NoOfAdult,AdvancePayment,DiscountID,Remarks,Status) values ('" &

lblGuestID.Text & "','" &

txtRoomNumber.Text & "','" &

dtCheckInDate.Text & "','" &

dtCheckOutDate.Text & "','" &

lblDateNow.Text & "','" &

txtChildren.Text & "','" &

txtAdults.Text & "','" &

txtAdvance.Text & "','" &

lblDiscountID.Text & "','" &

remarks & "','" &

stat & "')", con)

checkin.ExecuteNonQuery()

Dim update\_guest As New OleDbCommand("UPDATE tblGuest SET Remarks = 'Reserve' WHERE ID = " & lblGuestID.Text & "", con)

update\_guest.ExecuteNonQuery()

Dim update\_room As New OleDbCommand("UPDATE tblRoom SET Status = 'Reserve' WHERE RoomNumber = " & txtRoomNumber.Text & "", con)

update\_room.ExecuteNonQuery()

MsgBox("Guest Successfully Reserved!", vbInformation, "Reservation")

clear\_text()

con.Close()

transID()

display\_reserve()

End If

End If

End Sub

Private Sub dtCheckInDate\_ValueChanged(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles dtCheckInDate.ValueChanged

Dim t As Date = dtCheckInDate.Value

If t.Date < Now.Date Then

dtCheckInDate.Value = Now.Date

Else

dtCheckOutDate.Value = dtCheckInDate.Value.Date.AddDays(1D)

End If

End Sub

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

Dim tao As Integer

tao = Val(txtAdults.Text) + Val(txtChildren.Text)

If tao = Val(lblNoOfOccupancy.Text) Then

Else

txtAdults.Text = Val(txtAdults.Text) + 1

End If

End Sub

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

Dim tao As Integer

tao = Val(txtAdults.Text) + Val(txtChildren.Text)

If tao = Val(lblNoOfOccupancy.Text) Then

Else

txtChildren.Text = Val(txtChildren.Text) + 1

End If

End Sub

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

If Val(txtAdults.Text) = 0 Then

txtAdults.Text = Val(txtAdults.Text)

Else

txtAdults.Text = Val(txtAdults.Text) - 1

End If

End Sub

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

If Val(txtChildren.Text) = 0 Then

txtChildren.Text = Val(txtChildren.Text)

Else

txtChildren.Text = Val(txtChildren.Text) - 1

End If

End Sub

Private Sub cboDiscount\_TextChanged(ByVal sender As Object, ByVal e As System.EventArgs) Handles cboDiscount.TextChanged

con.Open()

Dim dt As New DataTable

rs = New OleDbDataAdapter("SELECT \* FROM tblDiscount WHERE DiscountType = '" & cboDiscount.Text & "'", con)

rs.Fill(dt)

lblDiscountID.Text = dt.Rows(0).Item("ID")

lblDiscountRate.Text = dt.Rows(0).Item("DiscountRate")

'lblTotal.Text = Val(txtSubTotal.Text) - (Val(txtSubTotal.Text) \* Val(lblDiscountRate.Text))

txtSubTotal.Text = Val(lblTotal.Text) - (Val(lblTotal.Text) \* Val(lblDiscountRate.Text))

lblAdvancePayment.Text = "Advance payment must be atleast Php " & (Val(txtSubTotal.Text) \* 0.5)

rs.Dispose()

con.Close()

End Sub

Private Sub txtAdvance\_KeyPress(ByVal sender As Object, ByVal e As System.Windows.Forms.KeyPressEventArgs) Handles txtAdvance.KeyPress

If (e.KeyChar < "0" OrElse e.KeyChar > "9") \_

AndAlso e.KeyChar <> ControlChars.Back AndAlso e.KeyChar <> "." Then

'cancel keys

e.Handled = True

End If

End Sub

Private Sub txtAdvance\_TextChanged(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles txtAdvance.TextChanged

txtTotal.Text = Val(lblTotal.Text) - Val(txtAdvance.Text)

End Sub

Private Sub txtRoomRate\_TextChanged(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles txtRoomRate.TextChanged

lblTotal.Text = Val(txtRoomRate.Text) \* Val(txtDaysNumber.Text)

txtSubTotal.Text = Val(txtRoomRate.Text) \* Val(txtDaysNumber.Text)

End Sub

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

trans\_id = lvlreserve.SelectedItems(0).Text

con.Open()

Dim dt As New DataTable("tblTransaction")

rs = New OleDbDataAdapter("SELECT \* FROM tblTransaction", con)

rs.Fill(dt)

Dim indx As Integer

For indx = 0 To dt.Rows.Count - 1

If trans\_id = "TransID - " & Val(dt.Rows(indx).Item("TransID")).ToString("0000") Then

guest\_id = dt.Rows(0).Item("GuestID")

room\_num = dt.Rows(0).Item("RoomNum")

id = dt.Rows(indx).Item("TransID")

Exit For

End If

Next

rs.Dispose()

con.Close()

End Sub

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

Dim check\_in As String = MsgBox("Checkin Guest?", vbQuestion + vbYesNo, "Checkin")

If check\_in = vbYes Then

con.Open()

Dim update\_trans As New OleDbCommand("UPDATE tblTransaction SET Remarks = 'Checkin' WHERE TransID = " & id & "", con)

update\_trans.ExecuteNonQuery()

Dim update\_guest As New OleDbCommand("UPDATE tblGuest SET Remarks = 'Checkin' WHERE ID = " & guest\_id & "", con)

update\_guest.ExecuteNonQuery()

Dim update\_room As New OleDbCommand("UPDATE tblRoom SET Status = 'Occupied' WHERE RoomNumber = " & room\_num & "", con)

update\_room.ExecuteNonQuery()

con.Close()

display\_reserve()

MsgBox("Guest Checkedin!", vbInformation, "Checkin")

End If

End Sub

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

Dim check\_in As String = MsgBox("Cancel Reservation?", vbQuestion + vbYesNo, "Cancel")

If check\_in = vbYes Then

con.Open()

Dim update\_trans As New OleDbCommand("UPDATE tblTransaction SET Remarks = 'Cancelled' WHERE TransID = " & id & "", con)

update\_trans.ExecuteNonQuery()

Dim update\_guest As New OleDbCommand("UPDATE tblGuest SET Remarks = 'Available' WHERE ID = " & guest\_id & "", con)

update\_guest.ExecuteNonQuery()

Dim update\_room As New OleDbCommand("UPDATE tblRoom SET Status = 'Available' WHERE RoomNumber = " & room\_num & "", con)

update\_room.ExecuteNonQuery()

con.Close()

display\_reserve()

MsgBox("Reservation Cancelled!", vbInformation, "Cancel")

End If

End Sub

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

End Sub

End Class

**Reservation list form:-**

Imports System.Data.OleDb

Public Class frmReserveListMonitor

Private Sub frmReserveListMonitor\_Load(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles MyBase.Load

Dim Dt As New DataTable("tblGuest")

Dim rs As OleDbDataAdapter

rs = New OleDbDataAdapter("Select \* from tblTransaction, tblGuest, tblDiscount, tblRoom WHERE tblTransaction.GuestID = tblGuest.ID AND tblTransaction.DiscountID = tblDiscount.ID AND tblTransaction.RoomNum = tblRoom.RoomNumber AND tblTransaction.Remarks = 'Reserve'", con)

rs.Fill(Dt)

Dim indx As Integer

lvlreserve.Items.Clear()

For indx = 0 To Dt.Rows.Count - 1

Dim lv As New ListViewItem

Dim getdate As TimeSpan

Dim days, rate As Integer

Dim discount, subtotal, total, advance As Double

Dim value As Integer = Val(Dt.Rows(indx).Item("TransID"))

lv.Text = "TransID - " & value.ToString("0000")

lv.SubItems.Add(Dt.Rows(indx).Item("GuestFName") & " " & Dt.Rows(indx).Item("GuestLName"))

lv.SubItems.Add(Dt.Rows(indx).Item("RoomNum"))

rate = Dt.Rows(indx).Item("RoomRate")

lv.SubItems.Add(Dt.Rows(indx).Item("CheckInDate"))

lv.SubItems.Add(Dt.Rows(indx).Item("CheckOutDate"))

dtIn.Value = Dt.Rows(indx).Item("CheckOutDate")

dtOut.Value = Dt.Rows(indx).Item("CheckInDate")

getdate = dtIn.Value - dtOut.Value

days = getdate.Days

lv.SubItems.Add(days)

lv.SubItems.Add(Dt.Rows(indx).Item("NoOfChild"))

lv.SubItems.Add(Dt.Rows(indx).Item("NoOfAdult"))

advance = Dt.Rows(indx).Item("AdvancePayment")

lv.SubItems.Add("Php " & Val(advance).ToString("N"))

lv.SubItems.Add(Dt.Rows(indx).Item("DiscountType"))

discount = Val(Dt.Rows(indx).Item("DiscountRate"))

subtotal = (days \* rate) - ((days \* rate) \* discount)

total = Val(subtotal) - Val(Dt.Rows(indx).Item("AdvancePayment"))

lv.SubItems.Add("Php " & Val(total).ToString("N"))

lvlreserve.Items.Add(lv)

Next

rs.Dispose()

End Sub

End Class

**Check-In form:-**

Imports System.Data.OleDb

Public Class frmCheckin

Dim guestID, roomID, trans\_ID As Integer

Private Sub frmCheckin\_FormClosing(ByVal sender As Object, ByVal e As System.Windows.Forms.FormClosingEventArgs) Handles Me.FormClosing

Dim a As String = MsgBox("Cancel Transaction?", vbQuestion + vbYesNo, "Cancel")

If a = vbNo Then

e.Cancel = True

Else

clear\_text()

End If

End Sub

Private Sub frmCheckin\_Load(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles MyBase.Load

clear\_text()

Dim time As DateTime = DateTime.Now

Dim format As String = "MM/d/yyyy"

txtCheckInDate.Text = time.ToString(format)

dtCheckOutDate.Text = Now.AddDays(1D)

transID()

pop\_discount()

display\_checkin()

End Sub

Public Sub transID()

con.Open()

Dim dt As New DataTable("tblTransaction")

rs = New OleDbDataAdapter("SELECT \* FROM tblTransaction ORDER BY TransID DESC", con)

rs.Fill(dt)

If dt.Rows.Count = 0 Then

txtTransID.Text = "TransID - 0001"

Else

Dim value As Integer = Val(dt.Rows(0).Item("TransID"))

value = value + 1

txtTransID.Text = "TransID - " & value.ToString("0000")

trans\_ID = value

End If

rs.Dispose()

con.Close()

End Sub

Private Sub bttnCheckIn\_Click\_1(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles bttnCheckIn.Click

Dim children As Integer = Val(txtChildren.Text)

Dim adult As Integer = Val(txtAdults.Text)

Dim advance As Integer = Val(txtAdvance.Text)

Dim discount As Integer = Val(lblDiscountID.Text)

Dim reserve As String = "0"

Dim remarks As String = "Checkin"

Dim stat As String = "Active"

If lblGuestID.Text = "GuestID" Or lblGuestID.Text = Nothing Or txtRoomNumber.Text = Nothing Or Val(children + adult) = Nothing Or advance = Nothing Or discount = Nothing Then

MsgBox("Please Fill All Fields", vbInformation, "Note")

Else

If Val(Val(txtSubTotal.Text) \* 0.5) > Val(txtAdvance.Text) Then

MsgBox("Ops")

Exit Sub

End If

Dim a As String = MsgBox("Confirm Checkin Transaction?", vbQuestion + vbYesNo, "Check In")

If a = vbYes Then

con.Open()

Dim checkin As New OleDbCommand("INSERT INTO tblTransaction(GuestID,RoomNum,CheckInDate,CheckOutDate,NoOfChild,NoOfAdult,AdvancePayment,DiscountID,Remarks,Status) values ('" &

lblGuestID.Text & "','" &

txtRoomNumber.Text & "','" &

txtCheckInDate.Text & "','" &

dtCheckOutDate.Text & "','" &

txtChildren.Text & "','" &

txtAdults.Text & "','" &

txtAdvance.Text & "','" &

lblDiscountID.Text & "','" &

remarks & "','" &

stat & "')", con)

checkin.ExecuteNonQuery()

Dim update\_guest As New OleDbCommand("UPDATE tblGuest SET Remarks = 'Checkin' WHERE ID = " & lblGuestID.Text & "", con)

update\_guest.ExecuteNonQuery()

Dim update\_room As New OleDbCommand("UPDATE tblRoom SET Status = 'Occupied' WHERE RoomNumber = " & txtRoomNumber.Text & "", con)

update\_room.ExecuteNonQuery()

If Val(txtSubTotal.Text) < Val(txtAdvance.Text) Or Val(txtSubTotal.Text) = Val(txtAdvance.Text) Then

MsgBox("Guest Successfully Checkin! " & "Change: Php " & Val(Val(txtAdvance.Text) - Val(txtSubTotal.Text)).ToString("00.00"), vbInformation, "Check In")

Dim change As String = MsgBox("Return change to customer?", vbQuestion + vbYesNo, "Change")

If change = vbYes Then

Dim update\_trans As New OleDbCommand("UPDATE tblTransaction SET AdvancePayment = " & Val(txtSubTotal.Text) & " WHERE TransID = " & trans\_ID & "", con)

update\_trans.ExecuteNonQuery()

End If

Else

MsgBox("Guest Successfully Checkin!", vbInformation, "Check In")

End If

clear\_text()

con.Close()

transID()

display\_checkin()

End If

End If

End Sub

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

clear\_text()

End Sub

Private Sub dtCheckOutDate\_ValueChanged\_1(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles dtCheckOutDate.ValueChanged

Dim T As TimeSpan = dtCheckOutDate.Value - Now

If T.Days < 1 Then

dtCheckOutDate.Text = Now.AddDays(1D)

txtDaysNumber.Text = "1"

Else

txtDaysNumber.Text = T.Days + 1

End If

lblTotal.Text = Val(txtRoomRate.Text) \* Val(txtDaysNumber.Text)

txtSubTotal.Text = Val(txtRoomRate.Text) \* Val(txtDaysNumber.Text)

End Sub

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

frmSelectGuest.ShowDialog()

End Sub

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

frmSelectRoom.ShowDialog()

End Sub

Private Sub txtRoomRate\_TextChanged(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles txtRoomRate.TextChanged

lblTotal.Text = Val(txtRoomRate.Text) \* Val(txtDaysNumber.Text)

txtSubTotal.Text = (Val(txtRoomRate.Text) \* Val(txtDaysNumber.Text)).ToString("00.00")

End Sub

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

Dim tao As Integer

tao = Val(txtAdults.Text) + Val(txtChildren.Text)

If tao = Val(lblNoOfOccupancy.Text) Then

Else

txtAdults.Text = Val(txtAdults.Text) + 1

End If

End Sub

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

Dim tao As Integer

tao = Val(txtAdults.Text) + Val(txtChildren.Text)

If tao = Val(lblNoOfOccupancy.Text) Then

Else

txtChildren.Text = Val(txtChildren.Text) + 1

End If

End Sub

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

If Val(txtAdults.Text) = 0 Then

txtAdults.Text = Val(txtAdults.Text)

Else

txtAdults.Text = Val(txtAdults.Text) - 1

End If

End Sub

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

If Val(txtChildren.Text) = 0 Then

txtChildren.Text = Val(txtChildren.Text)

Else

txtChildren.Text = Val(txtChildren.Text) - 1

End If

End Sub

Private Sub pop\_discount()

con.Open()

Dim dt As New DataTable

rs = New OleDbDataAdapter("SELECT \* FROM tblDiscount", con)

rs.Fill(dt)

cboDiscount.Items.Clear()

Dim i As Integer

For i = 0 To dt.Rows.Count - 1

cboDiscount.Items.Add(dt.Rows(i).Item("DiscountType"))

Next

rs.Dispose()

con.Close()

End Sub

Private Sub cboDiscount\_TextChanged(ByVal sender As Object, ByVal e As System.EventArgs) Handles cboDiscount.TextChanged

con.Open()

Dim dt As New DataTable

rs = New OleDbDataAdapter("SELECT \* FROM tblDiscount WHERE DiscountType = '" & cboDiscount.Text & "'", con)

rs.Fill(dt)

lblDiscountID.Text = dt.Rows(0).Item("ID")

lblDiscountRate.Text = dt.Rows(0).Item("DiscountRate")

'lblTotal.Text = Val(txtSubTotal.Text) - (Val(txtSubTotal.Text) \* Val(lblDiscountRate.Text))

txtSubTotal.Text = (Val(lblTotal.Text) - (Val(lblTotal.Text) \* Val(lblDiscountRate.Text))).ToString("00.00")

lblAdvancePayment.Text = "Advance payment must be atleast Php " & (Val(txtSubTotal.Text) \* 0.5).ToString("00.00")

rs.Dispose()

con.Close()

End Sub

Private Sub txtAdvance\_KeyPress(ByVal sender As Object, ByVal e As System.Windows.Forms.KeyPressEventArgs) Handles txtAdvance.KeyPress

If (e.KeyChar < "0" OrElse e.KeyChar > "9") \_

AndAlso e.KeyChar <> ControlChars.Back AndAlso e.KeyChar <> "." Then

'cancel keys

e.Handled = True

End If

End Sub

Private Sub txtAdvance\_TextChanged(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles txtAdvance.TextChanged

If Val(txtSubTotal.Text) < Val(txtAdvance.Text) Or Val(txtSubTotal.Text) = Val(txtAdvance.Text) Then

txtTotal.Text = "0.00"

Else

txtTotal.Text = (Val(txtSubTotal.Text) - Val(txtAdvance.Text)).ToString("00.00")

End If

End Sub

Private Sub clear\_text()

txtGuestName.Clear()

txtRoomNumber.Clear()

txtRoomType.Clear()

txtRoomRate.Clear()

txtChildren.Text = "0"

txtAdults.Text = "0"

cboDiscount.Refresh()

txtAdvance.Clear()

txtSubTotal.Clear()

txtTotal.Clear()

lblDiscountID.Text = ""

lblDiscountRate.Text = ""

lblGuestID.Text = ""

lblAdvancePayment.Text = ""

lblNoOfOccupancy.Text = "0"

Dim time As DateTime = DateTime.Now

Dim format As String = "MM/d/yyyy"

txtCheckInDate.Text = time.ToString(format)

dtCheckOutDate.Text = Now.AddDays(1D)

End Sub

Private Sub display\_checkin()

con.Open()

Dim Dt As New DataTable("tblGuest")

Dim rs As OleDbDataAdapter

rs = New OleDbDataAdapter("Select \* from tblTransaction, tblGuest, tblDiscount, tblRoom WHERE tblTransaction.GuestID = tblGuest.ID AND tblTransaction.DiscountID = tblDiscount.ID AND tblTransaction.RoomNum = tblRoom.RoomNumber AND tblTransaction.Remarks = 'Checkin' AND tblTransaction.Status = 'Active'", con)

rs.Fill(Dt)

Dim indx As Integer

lvlcheckin.Items.Clear()

For indx = 0 To Dt.Rows.Count - 1

Dim lv As New ListViewItem

Dim getdate As TimeSpan

Dim days, subtotal, total, rate As Integer

Dim discount As Double

Dim value As Integer = Val(Dt.Rows(indx).Item("TransID"))

lv.Text = "TransID - " & value.ToString("0000")

lv.SubItems.Add(Dt.Rows(indx).Item("GuestFName") & " " & Dt.Rows(indx).Item("GuestLName"))

lv.SubItems.Add(Dt.Rows(indx).Item("RoomNum"))

rate = Dt.Rows(indx).Item("RoomRate")

lv.SubItems.Add(Dt.Rows(indx).Item("CheckInDate"))

lv.SubItems.Add(Dt.Rows(indx).Item("CheckOutDate"))

dtIn.Value = Dt.Rows(indx).Item("CheckOutDate")

dtOut.Value = Dt.Rows(indx).Item("CheckInDate")

getdate = dtIn.Value - dtOut.Value

days = getdate.Days

lv.SubItems.Add(days)

lv.SubItems.Add(Dt.Rows(indx).Item("NoOfChild"))

lv.SubItems.Add(Dt.Rows(indx).Item("NoOfAdult"))

lv.SubItems.Add(Dt.Rows(indx).Item("AdvancePayment").ToString("00.00"))

lv.SubItems.Add(Dt.Rows(indx).Item("DiscountType"))

discount = Val(Dt.Rows(indx).Item("DiscountRate"))

subtotal = (days \* rate) - ((days \* rate) \* discount)

total = (Val(subtotal) - Val(Dt.Rows(indx).Item("AdvancePayment"))).ToString("00.00")

If Val(subtotal) > Val(Dt.Rows(indx).Item("AdvancePayment")) Then

lv.SubItems.Add(Val(total))

Else

lv.SubItems.Add("0")

End If

lvlcheckin.Items.Add(lv)

Next

rs.Dispose()

con.Close()

End Sub

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

End Sub

End Class

**Check-In List form:-**

Imports System.Data.OleDb

Public Class frmCheckinList

Private Sub frmCheckinList\_Load(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles MyBase.Load

con.Open()

Dim Dt As New DataTable("tblTransaction")

rs = New OleDbDataAdapter("Select \* from tblTransaction, tblGuest, tblDiscount, tblRoom WHERE tblTransaction.GuestID = tblGuest.ID AND tblTransaction.DiscountID = tblDiscount.ID AND tblTransaction.RoomNum = tblRoom.RoomNumber AND tblTransaction.Remarks = 'Checkin' AND tblTransaction.Status = 'Active'", con)

rs.Fill(Dt)

Dim indx As Integer

lvlcheckin.Items.Clear()

For indx = 0 To Dt.Rows.Count - 1

Dim lv As New ListViewItem

Dim getdate As TimeSpan

Dim days, subtotal, total, rate As Integer

Dim discount As Double

Dim value As Integer = Val(Dt.Rows(indx).Item("TransID"))

lv.Text = "TransID - " & value.ToString("0000")

lv.SubItems.Add(Dt.Rows(indx).Item("GuestFName") & " " & Dt.Rows(indx).Item("GuestLName"))

lv.SubItems.Add(Dt.Rows(indx).Item("RoomNum"))

rate = Dt.Rows(indx).Item("RoomRate")

lv.SubItems.Add(Dt.Rows(indx).Item("CheckInDate"))

lv.SubItems.Add(Dt.Rows(indx).Item("CheckOutDate"))

dtIn.Value = Dt.Rows(indx).Item("CheckOutDate")

dtOut.Value = Dt.Rows(indx).Item("CheckInDate")

getdate = dtIn.Value - dtOut.Value

days = getdate.Days

lv.SubItems.Add(days)

lv.SubItems.Add(Dt.Rows(indx).Item("NoOfChild"))

lv.SubItems.Add(Dt.Rows(indx).Item("NoOfAdult"))

lv.SubItems.Add(Dt.Rows(indx).Item("AdvancePayment"))

lv.SubItems.Add(Dt.Rows(indx).Item("DiscountType"))

discount = Val(Dt.Rows(indx).Item("DiscountRate"))

subtotal = (days \* rate) - ((days \* rate) \* discount)

total = Val(subtotal) - Val(Dt.Rows(indx).Item("AdvancePayment"))

If Val(subtotal) > Val(Dt.Rows(indx).Item("AdvancePayment")) Then

lv.SubItems.Add(Val(total))

Else

lv.SubItems.Add("0")

End If

lvlcheckin.Items.Add(lv)

Next

rs.Dispose()

con.Close()

End Sub

Private Sub lvlcheckin\_DoubleClick(ByVal sender As Object, ByVal e As System.EventArgs) Handles lvlcheckin.DoubleClick

dtIn.Value = lvlcheckin.SelectedItems(0).SubItems(3).Text

dtOut.Value = Now.Date

con.Open()

Dim dt As New DataTable("tblTransaction")

rs = New OleDbDataAdapter("Select \* from tblTransaction, tblRoom, tblDiscount WHERE tblTransaction.RoomNum = tblRoom.RoomNumber AND tblTransaction.DiscountID = tblDiscount.ID", con)

rs.Fill(dt)

Dim indx As Integer

For indx = 0 To dt.Rows.Count - 1

Dim value As Double = Val(dt.Rows(indx).Item("TransID"))

If lvlcheckin.SelectedItems(0).Text = "TransID - " & value.ToString("0000") Then

frmCheckout.txtTransID.Text = "TransID - " & value.ToString("0000")

frmCheckout.lblTransID.Text = dt.Rows(indx).Item("TransID")

frmCheckout.lblGuestID.Text = dt.Rows(indx).Item("GuestID")

Dim time As DateTime = DateTime.Now

Dim format As String = "MM/d/yyyy"

Dim getdate As TimeSpan

If dtIn.Value = dtOut.Value Then

frmCheckout.txtCheckout.Text = Now.Date.AddDays(1D)

frmCheckout.txtDays.Text = "1"

Dim subtotal As Double = Val(frmCheckout.txtDays.Text) \* Val(dt.Rows(indx).Item("RoomRate"))

subtotal = Val(subtotal) - Val((dt.Rows(indx).Item("DiscountRate")) \* Val(subtotal))

If Val(subtotal) > Val(dt.Rows(indx).Item("AdvancePayment")) Then

Dim total As Double = Val(subtotal) - (dt.Rows(indx).Item("AdvancePayment"))

frmCheckout.txtSubTotal.Text = Val(subtotal).ToString("N")

frmCheckout.txtTotal.Text = Val(total).ToString("N")

frmCheckout.txtCash.Text = Val(total).ToString("N")

frmCheckout.txtChange.Text = "0.00"

Else

Dim total As Double = (dt.Rows(indx).Item("AdvancePayment")) - Val(subtotal)

Dim change As Double = Val(total) - (dt.Rows(indx).Item("AdvancePayment"))

frmCheckout.txtSubTotal.Text = Val(subtotal).ToString("N")

frmCheckout.txtTotal.Text = "0.00"

frmCheckout.txtCash.Text = "0.00"

frmCheckout.txtChange.Text = Val(total).ToString("N")

End If

Else

frmCheckout.txtCheckout.Text = Now.Date

getdate = (dtOut.Value) - (dtIn.Value)

frmCheckout.txtDays.Text = getdate.Days

Dim subtotal As Double = Val(getdate.Days) \* Val(dt.Rows(indx).Item("RoomRate"))

subtotal = Val(subtotal) - Val((dt.Rows(indx).Item("DiscountRate")) \* Val(subtotal))

If Val(subtotal) > Val(dt.Rows(indx).Item("AdvancePayment")) Then

Dim total As Double = Val(subtotal) - (dt.Rows(indx).Item("AdvancePayment"))

frmCheckout.txtSubTotal.Text = Val(subtotal).ToString("N")

frmCheckout.txtTotal.Text = Val(total).ToString("N")

frmCheckout.txtCash.Text = Val(total).ToString("N")

frmCheckout.txtChange.Text = "0.00"

Else

Dim total As Double = (dt.Rows(indx).Item("AdvancePayment")) - Val(subtotal)

Dim change As Double = Val(total) - (dt.Rows(indx).Item("AdvancePayment"))

frmCheckout.txtSubTotal.Text = Val(subtotal).ToString("N")

frmCheckout.txtTotal.Text = "0.00"

frmCheckout.txtCash.Text = "0.00"

frmCheckout.txtChange.Text = Val(total).ToString("N")

End If

End If

Exit For

End If

Next

rs.Dispose()

con.Close()

'frmCheckout.txtTransID.Text = lvlcheckin.SelectedItems(0).Text

frmCheckout.txtGuestName.Text = lvlcheckin.SelectedItems(0).SubItems(1).Text

frmCheckout.txtRoomNumber.Text = lvlcheckin.SelectedItems(0).SubItems(2).Text

frmCheckout.txtCheckin.Text = lvlcheckin.SelectedItems(0).SubItems(3).Text

'frmCheckout.txtCheckout.Text = lvlcheckin.SelectedItems(0).SubItems(4).Text

'frmCheckout.txtDays.Text = lvlcheckin.SelectedItems(0).SubItems(5).Text

frmCheckout.txtChildren.Text = lvlcheckin.SelectedItems(0).SubItems(6).Text

frmCheckout.txtAdult.Text = lvlcheckin.SelectedItems(0).SubItems(7).Text

frmCheckout.txtAdvance.Text = Val(lvlcheckin.SelectedItems(0).SubItems(8).Text).ToString("N")

frmCheckout.txtDiscountType.Text = lvlcheckin.SelectedItems(0).SubItems(9).Text

Me.Close()

End Sub

End Class

**Checked-In List monitor:-**

Imports System.Data.OleDb

Public Class frmCheckinListMonitor

Private Sub frmCheckinListMonitor\_Load(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles MyBase.Load

con.Open()

Dim Dt As New DataTable("tblTransaction")

rs = New OleDbDataAdapter("Select \* from tblTransaction, tblGuest, tblDiscount, tblRoom WHERE tblTransaction.GuestID = tblGuest.ID AND tblTransaction.DiscountID = tblDiscount.ID AND tblTransaction.RoomNum = tblRoom.RoomNumber AND tblTransaction.Remarks = 'Checkin' AND tblTransaction.Status = 'Active'", con)

rs.Fill(Dt)

Dim indx As Integer

lvlcheckin.Items.Clear()

For indx = 0 To Dt.Rows.Count - 1

Dim lv As New ListViewItem

Dim getdate As TimeSpan

Dim days, rate As Integer

Dim subtotal, total, advance As Double

Dim discount As Double

Dim value As Integer = Val(Dt.Rows(indx).Item("TransID"))

lv.Text = "TransID - " & value.ToString("0000")

lv.SubItems.Add(Dt.Rows(indx).Item("GuestFName") & " " & Dt.Rows(indx).Item("GuestLName"))

lv.SubItems.Add(Dt.Rows(indx).Item("RoomNum"))

rate = Dt.Rows(indx).Item("RoomRate")

lv.SubItems.Add(Dt.Rows(indx).Item("CheckInDate"))

lv.SubItems.Add(Dt.Rows(indx).Item("CheckOutDate"))

dtIn.Value = Dt.Rows(indx).Item("CheckOutDate")

dtOut.Value = Dt.Rows(indx).Item("CheckInDate")

getdate = dtIn.Value - dtOut.Value

days = getdate.Days

lv.SubItems.Add(days)

lv.SubItems.Add(Dt.Rows(indx).Item("NoOfChild"))

lv.SubItems.Add(Dt.Rows(indx).Item("NoOfAdult"))

advance = Dt.Rows(indx).Item("AdvancePayment")

lv.SubItems.Add("Php " & (advance).ToString("N"))

lv.SubItems.Add(Dt.Rows(indx).Item("DiscountType"))

discount = Val(Dt.Rows(indx).Item("DiscountRate"))

subtotal = (days \* rate) - ((days \* rate) \* discount)

total = (Val(subtotal) - Val(Dt.Rows(indx).Item("AdvancePayment"))).ToString("N")

If Val(subtotal) > Val(Dt.Rows(indx).Item("AdvancePayment")) Then

lv.SubItems.Add("Php " & Val(total).ToString("N"))

Else

lv.SubItems.Add("Php 0.00")

End If

lvlcheckin.Items.Add(lv)

Next

rs.Dispose()

con.Close()

End Sub

End Class

**Checked-Out form:-**

Imports System.Data.OleDb

Public Class frmCheckout

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

frmCheckinList.ShowDialog()

End Sub

Private Sub txtRoomNumber\_TextChanged(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles txtRoomNumber.TextChanged

If txtRoomNumber.Text = Nothing Then

Else

con.Open()

Dim dt As New DataTable("tblRoom")

rs = New OleDbDataAdapter("SELECT \* FROM tblRoom WHERE RoomNumber = " & txtRoomNumber.Text & "", con)

rs.Fill(dt)

txtRoomType.Text = dt.Rows(0).Item("RoomType")

txtRoomRate.Text = Val(dt.Rows(0).Item("RoomRate")).ToString("N")

rs.Dispose()

con.Close()

End If

End Sub

Private Sub txtCash\_KeyPress(ByVal sender As Object, ByVal e As System.Windows.Forms.KeyPressEventArgs) Handles txtCash.KeyPress

If (e.KeyChar < "0" OrElse e.KeyChar > "9") \_

AndAlso e.KeyChar <> ControlChars.Back AndAlso e.KeyChar <> "." Then

'cancel keys

e.Handled = True

End If

End Sub

Private Sub txtCash\_TextChanged(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles txtCash.TextChanged

If Val(txtCash.Text) < Val(txtTotal.Text) Then

txtChange.Text = "0.00"

Else

txtChange.Text = (Val(txtCash.Text) - Val(txtTotal.Text)).ToString("N")

End If

End Sub

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

If txtTransID.Text = Nothing Then

MsgBox("Please select transaction to checkout!", vbExclamation, "Note")

Else

If Val(txtCash.Text) < Val(txtTotal.Text) Then

MsgBox("Insufficient cash!", vbExclamation, "Note")

Else

Dim out As String = MsgBox("Confirm Checkout", vbQuestion + vbYesNo, "Checkout")

If out = vbYes Then

con.Open()

Dim update\_trans As New OleDbCommand("UPDATE tblTransaction SET Remarks = 'Checkout' WHERE TransID = " & lblTransID.Text & "", con)

update\_trans.ExecuteNonQuery()

Dim update\_guest As New OleDbCommand("UPDATE tblGuest SET Remarks = 'Available' WHERE ID = " & lblGuestID.Text & "", con)

update\_guest.ExecuteNonQuery()

Dim update\_room As New OleDbCommand("UPDATE tblRoom SET Status = 'Available' WHERE RoomNumber = " & txtRoomNumber.Text & "", con)

update\_room.ExecuteNonQuery()

MsgBox("Guest Checked out!", vbInformation, "Checkout")

con.Close()

clear\_text()

End If

End If

End If

End Sub

Public Sub clear\_text()

txtTransID.Clear()

txtGuestName.Clear()

txtRoomNumber.Clear()

txtRoomRate.Clear()

txtRoomType.Clear()

txtCheckin.Clear()

txtCheckout.Clear()

txtChildren.Clear()

txtAdult.Clear()

txtAdvance.Clear()

txtDiscountType.Clear()

txtTotal.Clear()

txtSubTotal.Clear()

txtDays.Clear()

txtCash.Clear()

txtChange.Clear()

End Sub

Private Sub lblTransID\_TextChanged(ByVal sender As Object, ByVal e As System.EventArgs) Handles lblTransID.TextChanged

End Sub

Private Sub frmCheckout\_FormClosing(ByVal sender As Object, ByVal e As System.Windows.Forms.FormClosingEventArgs) Handles Me.FormClosing

clear\_text()

End Sub

Private Sub frmCheckout\_Load(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles MyBase.Load

txtRoomNumber.Clear()

dtOut.Value = Date.Now

End Sub

Private Sub txtAdvance\_TextChanged(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles txtAdvance.TextChanged

End Sub

Private Sub txtTotal\_TextChanged(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles txtTotal.TextChanged

End Sub

Private Sub txtDiscountType\_TextChanged(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles txtDiscountType.TextChanged

End Sub

End Class

**Checked-out List form:-**

Imports System.Data.OleDb

Public Class frmCheckOutList

Private Sub frmCheckOutList\_Load(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles MyBase.Load

con.Open()

Dim Dt As New DataTable("tblTransaction")

rs = New OleDbDataAdapter("Select \* from tblTransaction, tblGuest, tblDiscount, tblRoom WHERE tblTransaction.GuestID = tblGuest.ID AND tblTransaction.DiscountID = tblDiscount.ID AND tblTransaction.RoomNum = tblRoom.RoomNumber AND tblTransaction.Remarks = 'Checkout' AND tblTransaction.Status = 'Active'", con)

rs.Fill(Dt)

Dim indx As Integer

lvlcheckin.Items.Clear()

For indx = 0 To Dt.Rows.Count - 1

Dim lv As New ListViewItem

Dim getdate As TimeSpan

Dim days, rate As Integer

Dim subtotal, total, advance As Double

Dim discount As Double

Dim value As Integer = Val(Dt.Rows(indx).Item("TransID"))

lv.Text = "TransID - " & value.ToString("0000")

lv.SubItems.Add(Dt.Rows(indx).Item("GuestFName") & " " & Dt.Rows(indx).Item("GuestLName"))

lv.SubItems.Add(Dt.Rows(indx).Item("RoomNum"))

rate = Dt.Rows(indx).Item("RoomRate")

lv.SubItems.Add(Dt.Rows(indx).Item("CheckInDate"))

lv.SubItems.Add(Dt.Rows(indx).Item("CheckOutDate"))

dtIn.Value = Dt.Rows(indx).Item("CheckOutDate")

dtOut.Value = Dt.Rows(indx).Item("CheckInDate")

getdate = dtIn.Value - dtOut.Value

days = getdate.Days

lv.SubItems.Add(days)

lv.SubItems.Add(Dt.Rows(indx).Item("NoOfChild"))

lv.SubItems.Add(Dt.Rows(indx).Item("NoOfAdult"))

advance = Dt.Rows(indx).Item("AdvancePayment")

lv.SubItems.Add("Php " & (advance).ToString("N"))

lv.SubItems.Add(Dt.Rows(indx).Item("DiscountType"))

discount = Val(Dt.Rows(indx).Item("DiscountRate"))

subtotal = (days \* rate) - ((days \* rate) \* discount)

total = (Val(subtotal) - Val(Dt.Rows(indx).Item("AdvancePayment"))).ToString("N")

If Val(subtotal) > Val(Dt.Rows(indx).Item("AdvancePayment")) Then

lv.SubItems.Add("Php " & Val(total).ToString("N"))

Else

lv.SubItems.Add("Php 0.00")

End If

lvlcheckin.Items.Add(lv)

Next

rs.Dispose()

con.Close()

End Sub

End Class

**Discount form:-**

Imports System.Data.OleDb

Public Class frmDiscount

Dim update\_discount As Boolean = False

Dim id As Integer

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

clear\_txt()

update\_discount = False

End Sub

Private Sub frmDiscount\_Load(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles MyBase.Load

display\_discount()

clear\_txt()

End Sub

Private Sub display\_discount()

con.Open()

Dim dt As New DataTable("tblDiscount")

rs = New OleDbDataAdapter("SELECT \* FROM tblDiscount", con)

rs.Fill(dt)

lvlDiscount.Items.Clear()

Dim indx As Integer

For indx = 0 To dt.Rows.Count - 1

Dim lv As New ListViewItem

lv.Text = dt.Rows(indx).Item("ID")

lv.SubItems.Add(dt.Rows(indx).Item("DiscountType"))

lv.SubItems.Add(dt.Rows(indx).Item("DiscountRate"))

lvlDiscount.Items.Add(lv)

Next

rs.Dispose()

con.Close()

End Sub

Private Sub TextBox1\_KeyPress(ByVal sender As Object, ByVal e As System.Windows.Forms.KeyPressEventArgs) Handles txtRate.KeyPress

If (e.KeyChar < "0" OrElse e.KeyChar > "9") AndAlso e.KeyChar <> ControlChars.Back AndAlso e.KeyChar <> "." Then

'cancel keys

e.Handled = True

End If

End Sub

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

If Trim(txtType.Text) = Nothing Or Trim(txtRate.Text) Then

MsgBox("Fill All Fields!", vbInformation, "Note")

Exit Sub

End If

Dim a As String = MsgBox("Save Discount Info?", vbQuestion + vbYesNo, "Save")

If a = vbYes Then

If update\_discount = False Then

con.Open()

Dim save As New OleDbCommand("INSERT INTO tblDiscount(DiscountType,DiscountRate,Status) values ('" & Trim(txtType.Text) & "','" & Trim(txtRate.Text) & "','Active')", con)

save.ExecuteNonQuery()

con.Close()

Else

con.Open()

Dim update As New OleDbCommand("UPDATE tblDiscount SET DiscountType = '" & Trim(txtType.Text) & "',DiscountRate = '" & Val(Trim(txtRate.Text)) / 100 & "' WHERE ID = '" & id & "'", con)

update.ExecuteNonQuery()

con.Close()

End If

MsgBox("Data saved successfully!", vbInformation, "Discount")

display\_discount()

clear\_txt()

End If

End Sub

Private Sub clear\_txt()

txtType.Clear()

txtRate.Clear()

update\_discount = False

End Sub

Private Sub lvlDiscount\_DoubleClick(ByVal sender As Object, ByVal e As System.EventArgs) Handles lvlDiscount.DoubleClick

Dim a As String = MsgBox("Update selected Item?", vbQuestion + vbYesNo, "Update Discount")

If a = vbYes Then

id = lvlDiscount.SelectedItems(0).Text

txtType.Text = lvlDiscount.SelectedItems(0).SubItems(1).Text

txtRate.Text = Val(lvlDiscount.SelectedItems(0).SubItems(2).Text) \* 100

update\_discount = True

bttnSave.Text = "&Update"

End If

End Sub

End Class

**Room Form:-**

Imports System.Data.OleDb

Public Class frmRoom

Dim id As Integer

Private Sub frmRoom\_Load(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles MyBase.Load

TabControl1.SelectTab(0)

display\_room()

End Sub

Private Sub display\_room()

con.Open()

Dim Dt As New DataTable("tblRoom")

Dim rs As OleDbDataAdapter

rs = New OleDbDataAdapter("Select \* from tblRoom", con)

rs.Fill(Dt)

Dim indx As Integer

lvRoom.Items.Clear()

For indx = 0 To Dt.Rows.Count - 1

Dim lv As New ListViewItem

lv.Text = Dt.Rows(indx).Item("ID")

lv.SubItems.Add(Dt.Rows(indx).Item("RoomNumber"))

lv.SubItems.Add(Dt.Rows(indx).Item("RoomType"))

lv.SubItems.Add(Dt.Rows(indx).Item("RoomRate"))

lv.SubItems.Add(Dt.Rows(indx).Item("NoOfOccupancy"))

lvRoom.Items.Add(lv)

Next

rs.Dispose()

con.Close()

End Sub

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

txtID.Clear()

txtRoomType.Clear()

txtRoomRate.Clear()

txtNoOfOccupancy.Clear()

bttnSave.Text = "&Save"

End Sub

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

con.Open()

Dim num As String = Trim(txtID.Text)

Dim type As String = Trim(txtRoomType.Text)

Dim rate As String = Trim(txtRoomRate.Text)

Dim occupancy As String = Trim(txtNoOfOccupancy.Text)

Dim stat As String = "Available"

If type = Nothing Or rate = Nothing Or occupancy = Nothing Then

MsgBox("Please Fill All Fields", vbInformation, "Note")

Else

If bttnSave.Text = "&Save" Then

Dim sql = "SELECT \* FROM tblRoom WHERE RoomNumber = " & SafeSqlLiteral(num, 2) & ""

Dim cmd = New OleDbCommand(sql, con)

Dim dr As OleDbDataReader = cmd.ExecuteReader

Try

If dr.Read = False Then

Dim add\_room As New OleDbCommand("INSERT INTO tblRoom(RoomNumber,RoomType,RoomRate,NoOfOccupancy,Status) values ('" &

SafeSqlLiteral(num, 2) & "','" &

SafeSqlLiteral(type, 2) & "','" &

SafeSqlLiteral(rate, 2) & "','" &

SafeSqlLiteral(occupancy, 2) & "','" &

stat & "')", con)

add\_room.ExecuteNonQuery()

add\_room.Dispose()

MsgBox("Room Added!", vbInformation, "Note")

txtID.Clear()

txtRoomType.Clear()

txtRoomRate.Clear()

txtNoOfOccupancy.Clear()

Else

MsgBox("Room Number Existed!", vbExclamation, "Note")

End If

Catch ex As Exception

MsgBox(ex.Message)

End Try

Else

Dim update\_room As New OleDbCommand("UPDATE tblRoom SET RoomNumber= '" & SafeSqlLiteral(num, 2) & "',RoomType = '" & SafeSqlLiteral(type, 2) & "',RoomRate = '" & SafeSqlLiteral(rate, 2) & "',NoOfOccupancy = '" & SafeSqlLiteral(occupancy, 2) & "' WHERE ID = " & id & "", con)

update\_room.ExecuteNonQuery()

update\_room.Dispose()

MsgBox("Room Saved!", vbInformation, "Note")

bttnSave.Text = "&Save"

txtID.Clear()

txtRoomType.Clear()

txtRoomRate.Clear()

txtNoOfOccupancy.Clear()

End If

End If

con.Close()

display\_room()

End Sub

Private Sub lvRoom\_DoubleClick(ByVal sender As Object, ByVal e As System.EventArgs) Handles lvRoom.DoubleClick

Dim a As String = MsgBox("Update selected Item?", vbQuestion + vbYesNo, "Update Room")

If a = vbYes Then

id = lvRoom.SelectedItems(0).Text

txtID.Text = lvRoom.SelectedItems(0).SubItems(1).Text

txtRoomType.Text = lvRoom.SelectedItems(0).SubItems(2).Text

txtRoomRate.Text = lvRoom.SelectedItems(0).SubItems(3).Text

txtNoOfOccupancy.Text = lvRoom.SelectedItems(0).SubItems(4).Text

TabControl1.SelectTab(0)

bttnSave.Text = "&Update"

End If

End Sub

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

End Sub

End Class

**Room List form:-**

Imports System.Data.OleDb

Public Class frmSelectRoom

Private Sub frmSelectRoom\_Load(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles MyBase.Load

display\_room()

End Sub

Private Sub display\_room()

con.Open()

Dim Dt As New DataTable("tblRoom")

Dim rs As OleDbDataAdapter

rs = New OleDbDataAdapter("Select \* from tblRoom WHERE Status = 'Available'", con)

rs.Fill(Dt)

Dim indx As Integer

lvRoom.Items.Clear()

For indx = 0 To Dt.Rows.Count - 1

Dim lv As New ListViewItem

lv.Text = Dt.Rows(indx).Item("RoomNumber")

lv.SubItems.Add(Dt.Rows(indx).Item("RoomType"))

lv.SubItems.Add(Dt.Rows(indx).Item("RoomRate"))

lv.SubItems.Add(Dt.Rows(indx).Item("NoOfOccupancy"))

lvRoom.Items.Add(lv)

Next

rs.Dispose()

con.Close()

End Sub

Private Sub lvRoom\_DoubleClick(ByVal sender As Object, ByVal e As System.EventArgs) Handles lvRoom.DoubleClick

frmCheckin.txtRoomNumber.Text = lvRoom.SelectedItems(0).Text

frmCheckin.txtRoomType.Text = lvRoom.SelectedItems(0).SubItems(1).Text

frmCheckin.txtRoomRate.Text = lvRoom.SelectedItems(0).SubItems(2).Text

frmCheckin.lblNoOfOccupancy.Text = lvRoom.SelectedItems(0).SubItems(3).Text

frmReserve.txtRoomNumber.Text = lvRoom.SelectedItems(0).Text

frmReserve.txtRoomType.Text = lvRoom.SelectedItems(0).SubItems(1).Text

frmReserve.txtRoomRate.Text = lvRoom.SelectedItems(0).SubItems(2).Text

frmReserve.lblNoOfOccupancy.Text = lvRoom.SelectedItems(0).SubItems(3).Text

Me.Close()

End Sub

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

End Sub

End Class

**Room List Monitor:-**

Imports System.Data.OleDb

Public Class frmRoomListMonitor

Private Sub frmRoomListMonitor\_Load(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles MyBase.Load

con.Open()

Dim Dt As New DataTable("tblRoom")

Dim rs As OleDbDataAdapter

rs = New OleDbDataAdapter("Select \* from tblRoom", con)

rs.Fill(Dt)

Dim indx As Integer

lvRoom.Items.Clear()

For indx = 0 To Dt.Rows.Count - 1

Dim lv As New ListViewItem

lv.Text = Dt.Rows(indx).Item("RoomNumber")

lv.SubItems.Add(Dt.Rows(indx).Item("RoomType"))

lv.SubItems.Add(Dt.Rows(indx).Item("RoomRate"))

lv.SubItems.Add(Dt.Rows(indx).Item("NoOfOccupancy"))

lv.SubItems.Add(Dt.Rows(indx).Item("Status"))

lvRoom.Items.Add(lv)

Next

rs.Dispose()

con.Close()

End Sub

End Class

**Module.vb File:-**

Imports System.Data.OleDb

Imports System.Text.RegularExpressions

Module Module1

Public con As New OleDbConnection("Provider=Microsoft.Jet.OLEDB.4.0;Data Source=E:\Project\hbrs\HBRS\bin\Debug\database.mdb;Persist Security Info=True")

Public rs As New OleDbDataAdapter

Function SafeSqlLiteral(ByVal strValue, ByVal intLevel) As String

'\*\*\* Written by user CWA, CoolWebAwards.com Forums. 2 February 2010

If Not IsDbNull(strValue) Then

If intLevel > 0 Then

strValue = replace(strValue, "'", "''") strValue = replace(strValue, "--", "")

strValue = replace(strValue, "[", "[[]")

strValue = replace(strValue, "%", "[%]")

End If

If intLevel > 1 Then

Dim myArray As Array

myArray = Split("xp\_ ;update ;insert ;select ;drop ;alter ;create ;rename ;delete ;replace ", ";")

Dim i, i2, intLenghtLeft As Integer

For i = LBound(myArray) To UBound(myArray)

Dim rx As New Regex(myArray(i), RegexOptions.Compiled Or RegexOptions.IgnoreCase)

Dim matches As MatchCollection = rx.Matches(strValue)

i2 = 0

For Each match As Match In matches

Dim groups As GroupCollection = match.Groups

intLenghtLeft = groups.Item(0).Index + len(myArray(i)) + i2

strValue = Left(strValue, intLenghtLeft - 1) & "&nbsp;" & right(strValue, len(strValue) - intLenghtLeft)

i2 += 5

Next

Next

End If

Return strValue

Else

Return strValue

End If

End Function

End Module

Testing forms the first step in determining the errors in a program, the success of testing in revealing errors in programs depends critically on the test cases. Testing can be broken into smaller activities, due to this incremental testing is generally performed, in which components and subsystems of the system are tested separately before integrating them. The basic purpose of the testing is to detect the errors that may be present in the program, there are two basic approaches to testing: Functional and Structural.

Functional Testing: In the functional testing the structure of the program is not considered, test cases are decided on the basis of the requirements or specifications of the program or module and the internals of the module or the program are not considered for selection of the test cases. Due to this nature, it is often called Black box testing

Structural Testing: In Structural testing test cases are generated based on the actual code of the program or module to be tested, it deals with the internal structure of the program responsible for actually implementing that function, this approach is sometimes called glass box testing

System Testing

1. The raw data is fed in the database.
2. The Module is working in a good condition.
3. Functional testing and Structural testing was done.

In the implementation phase Programs are written, files and data bases are created, Education and training of user takes place, system is tested with optional data. Then changeover takes place in the old system is replaced by new one. The implementation of a system may take place when: -

1. The target date of chance over is due.
2. The system has been proved to the satisfaction of system analyst and other implementation activities have been completed.
3. The user manager is satisfied with the results of testing, staff training and reference manuals.
4. Operation manager is satisfied with the performance of equipments and operation staff.

There are different methods of changeover for example: -

1. Direct Changeover: In this method of changeover the old system is replaced by new system in one move.
2. Parallel Changeover: In this method of changeover, the processing of data takes place by both old and new system to cross check the results.
3. Staged Changeover: A staged changeover consists of a series of limited size changeover. In this method the new system is introduced piece by piece.

Pilot Running: In this method historical data for the whole or part of the system is run on this system and results obtained from the new system are compared with old system.

Visual Basic

1. Mastering Visual Basic.NET
2. Teach Yourself Visual Basic.NET
3. Black Book of VB.NET

Software Engineering

1. An Integrated Approach to Software Engineering (Pankaj Jalote)
2. System Analysis and Designing (Awad)

Database Management System

1. Fundamentals of Database System (Elmasri, Navathe)
2. An Introduction to Database Systems (C.J.Date)