CREACION DE BASE DE DATOS

```sql
--procedimineto listar
go
create proc pb_listar
as
select * from clientes
go
-- procedimiento de insertado
go
create proc pb_insertar
@cod_cliente int,
@nombre varchar(30),
@correo varchar (30),
@telefono int,
@fecha date
as
insert into clientes values(@cod_cliente, @nombre, @correo, @telefono, @fecha)
go
-- procedimiento de eliminado
go
create proc pb_elimina
@cod_cliente int
as
delete from clientes where cod_cliente = @cod_cliente
go
-- procedimiento de modificar
go
create proc pb_modificar
@cod_cliente int,
@nombre varchar(30),
@correo varchar (30),
@telefono int,
@fecha date
as
update clientes set nombre = @nombre,correo = @correo,telefono = @telefono,fecha = @fecha
where cod_cliente = @cod_cliente
go
```

CREACION DEL PROYECTO

BIBLIOTECA DE CLASES

```vbnet
Imports System.Data.SqlClient
Public Class Class1
    Dim conexion As New SqlConnection("Data Source=VYMPEL;Initial Catalog=BD_PRUEBA;Integrated Security=True")

    Public Function listar() As DataTable
        Dim da As New SqlDataAdapter("pb_listar", conexion)
        Dim tbl As New DataTable
        da.Fill(tbl)
        Return tbl
    End Function

    Public Function insertar(cod As Integer, nombre As String, correo As String, telefono As Integer, fecha As Date)
        Dim da As New SqlCommand("pb_insertar", conexion)
        da.CommandType = CommandType.StoredProcedure
        da.Parameters.AddWithValue("@cod_cliente", cod)
        da.Parameters.AddWithValue("@nombre", nombre)
        da.Parameters.AddWithValue("@correo", correo)
        da.Parameters.AddWithValue("@telefono", telefono)
        da.Parameters.AddWithValue("@fecha", fecha)
        conexion.Open()
        Dim resp As Integer
        Try
            resp = da.ExecuteNonQuery
            conexion.Close()
        Catch ex As Exception
            MsgBox("Error al registro")
        End Try
        Return resp
    End Function

    Public Function eliminar(cod As Integer)
        Dim elim As New SqlCommand("pb_elimina", conexion)
        elim.CommandType = CommandType.StoredProcedure
```

```vbnet
        elim.Parameters.AddWithValue("@cod_cliente", cod)
        conexion.Open()
        Dim elim1 As Integer
        Try
            elim1 = elim.ExecuteNonQuery()
            conexion.Close()
        Catch ex As Exception
            MsgBox("Error al registro")
        End Try
        Return elim1
    End Function

    Public Function modificar(cod1 As Integer, nombre1 As String, correo1 As String, telefono1 As Integer, fecha1 As Date)
        Dim act As New SqlCommand("pb_modificar", conexion)
        act.CommandType = CommandType.StoredProcedure
        act.Parameters.AddWithValue("@cod_cliente", cod1)
        act.Parameters.AddWithValue("@nombre", nombre1)
        act.Parameters.AddWithValue("@correo", correo1)
        act.Parameters.AddWithValue("@telefono", telefono1)
        act.Parameters.AddWithValue("@fecha", fecha1)
        conexion.Open()
        Dim resp1 As Integer
        Try
            resp1 = act.ExecuteNonQuery()
            conexion.Close()
        Catch ex As Exception
            MsgBox("Error al registro")
        End Try
        Return resp1
    End Function
End Class


PRINCIPAL

Imports ClassLibrary1
Public Class Form1
    Dim obj As New Class1

    Private Sub Btn_inserta_Click(sender As Object, e As EventArgs) Handles Btn_inserta.Click
        Try
            obj.insertar(Txt_cod.Text, Txt_nombre.Text, txt_corre.Text, Txt_telefono.Text, (CDate(DateTimePicker1.Text)))
            MsgBox("Registro completo")
            DataGridView1.DataSource = obj.listar()
            Txt_cod.Text = ""
            Txt_nombre.Text = ""
            txt_corre.Text = ""
            Txt_telefono.Text = ""
            DateTimePicker1.Value = DateTime.Now
        Catch ex As Exception
            MsgBox("error al registro")
        End Try
    End Sub

    Private Sub Btn_modifica_Click(sender As Object, e As EventArgs) Handles Btn_modifica.Click
        Try
            obj.modificar(Txt_cod.Text, Txt_nombre.Text, txt_corre.Text, Txt_telefono.Text, (CDate(DateTimePicker1.Text)))
            MsgBox("se modifico correctamente")
            DataGridView1.DataSource = obj.listar()
            Txt_cod.Text = ""
            Txt_nombre.Text = ""
            txt_corre.Text = ""
            Txt_telefono.Text = ""
            DateTimePicker1.Value = DateTime.Now
        Catch ex As Exception
        End Try
    End Sub

    Private Sub Btn_elimina_Click(sender As Object, e As EventArgs) Handles Btn_elimina.Click
        Try
            obj.eliminar(Txt_cod.Text)
            DataGridView1.DataSource = obj.listar()
            MsgBox("se elimino el registro ")
            Txt_cod.Text = ""
            Txt_nombre.Text = ""
            txt_corre.Text = ""
```

```vbnet
            Txt_telefono.Text = ""
            DateTimePicker1.Value = DateTime.Now
        Catch ex As Exception
            MsgBox("404 Erro al eliminar")
        End Try
    End Sub

    Private Sub Btn_Muestra_Click(sender As Object, e As EventArgs) Handles Btn_Muestra.Click
        DataGridView1.DataSource = obj.listar()
    End Sub

    Private Sub DataGridView1_CellContentClick(sender As Object, e As DataGridViewCellEventArgs) Handles
DataGridView1.CellContentClick
        Txt_cod.Text = DataGridView1.Item(0, e.RowIndex).Value
        Txt_nombre.Text = DataGridView1.Item(1, e.RowIndex).Value
        txt_corre.Text = DataGridView1.Item(2, e.RowIndex).Value
        Txt_telefono.Text = DataGridView1.Item(3, e.RowIndex).Value
        DateTimePicker1.Text = DataGridView1.Item(4, e.RowIndex).Value
    End Sub

    Private Sub Form1_Load(sender As Object, e As EventArgs) Handles MyBase.Load
        DataGridView1.DataSource = obj.listar()
    End Sub
End Class
```