* El nombre de usuario para ingresar es la cedula
* Para que funcione el cliente ocupa una foto, que se hace desde registra cliente
* En la bd no se pueden insertar vacios, debe se “” comillas, xq al devolver se cae
* Al instalar la bd ocupo, crear un medico en las personas, registrarlo en usuario, crear una persona con foto y hacerle la anamnesis.
* Asi para editar el tipo de una columna: alter table persona alter column empresa type numeric using empresa::numeric
* Para que la columna se le pueda cambiar el ancho debe ser not set en autosizemode
* SELECT \* FROM persona WHERE nombre LIKE LOWER('k%') or nombre LIKE UPPER('y%')
* Para poner todas las columnas en mayusculas UPDATE
* persona
* SET
* nombre = UPPER(nombre)

Para el label de division.. /\*label3.AutoSize = false;

label3.Height = 2;

label3.BorderStyle = BorderStyle.Fixed3D;

label3.Width = 1366;

* \*/
* En caso de editar un paciente….
* //editar un paciente
* //private void btnEditarPaciente\_Click(object sender, EventArgs e)
* //{
* // if (dgClientes.SelectedCells.Count > 0)
* // {
* // DataGridViewRow selectedRow = dgClientes.Rows[dgClientes.SelectedCells[0].RowIndex];
* // string ced = Convert.ToString(selectedRow.Cells["Cedula"].Value);
* // //////////////////////////////
* // List<ClienteEntity> lista = ClienteIT.selectClienteAnamnesis(ced);
* // //informacion personal
* // string nombre = lista.ElementAt(0).nombre.ToString();
* // string apellido1 = lista.ElementAt(0).ape1.ToString();
* // string apellido2 = lista.ElementAt(0).ape2.ToString();
* // string cedula = lista.ElementAt(0).cedula.ToString();
* // DateTime fecha = lista.ElementAt(0).fecha;
* // Char sexo = lista.ElementAt(0).sexo;
* // string estado\_Civil = lista.ElementAt(0).estado\_civil.ToString();
* // string grupo = lista.ElementAt(0).grupo\_sanguineo.ToString();
* // string profesion = lista.ElementAt(0).profesion.ToString(); ;
* // int telefono = Convert.ToInt32(lista.ElementAt(0).telefono\_fijo);
* // int movil = Convert.ToInt32(lista.ElementAt(0).telefono\_movil);
* // string email = lista.ElementAt(0).email.ToString();
* // string direccion = lista.ElementAt(0).direccion.ToString();
* // //anamnesis
* // char tabaquismo = Convert.ToChar(lista.ElementAt(0).tabaquismo);
* // char ingesta = Convert.ToChar(lista.ElementAt(0).ingesta\_medicamentos);
* // char alcoholismo = Convert.ToChar(lista.ElementAt(0).alcoholismo);
* // char rehabilitacion = Convert.ToChar(lista.ElementAt(0).rehabilitacion);
* // char diabetes = Convert.ToChar(lista.ElementAt(0).diabetes);
* // string diabetes\_trat = Convert.ToString(lista.ElementAt(0).diabetes\_trat);
* // char hipertension = Convert.ToChar(lista.ElementAt(0).hipertension);
* // string hipertension\_trat = Convert.ToString(lista.ElementAt(0).hipertension\_trat);
* // char dolor\_cabeza = Convert.ToChar(lista.ElementAt(0).dolor\_Cabeza);
* // char epilepsia = Convert.ToChar(lista.ElementAt(0).epilepsia);
* // char vertigo = Convert.ToChar(lista.ElementAt(0).vertigo);
* // char depresion = Convert.ToChar(lista.ElementAt(0).depre);
* // char falta\_aire = Convert.ToChar(lista.ElementAt(0).falta\_aire);
* // char enf\_ojos\_oidos = Convert.ToChar(lista.ElementAt(0).enf\_ojos\_oidos);
* // string observaciones = Convert.ToString(lista.ElementAt(0).observaciones);
* // this.Hide();
* // //se le envia toda la informacion al form para llenarlo
* // frm\_ExpedienteMG frm = new frm\_ExpedienteMG(nombre, apellido1, apellido2, cedula, fecha, sexo, estado\_Civil, grupo,
* // profesion, telefono, movil, email, direccion, tabaquismo, ingesta, alcoholismo, rehabilitacion, diabetes, diabetes\_trat,
* // hipertension, hipertension\_trat, dolor\_cabeza, epilepsia, vertigo, depresion, falta\_aire, enf\_ojos\_oidos, observaciones, true);
* // //se llama el siguiente form con toda la informacion del paciente
* // //al editar le quito una tab
* // frm.ShowDialog();
* // }
* //}

Para castear:

SELECT \* from persona where CAST(empresa AS VARCHAR(100)) LIKE '2%'