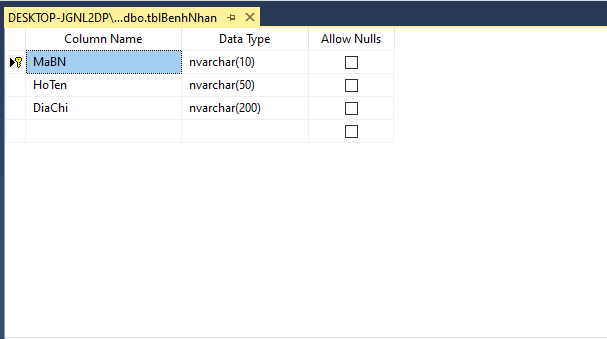
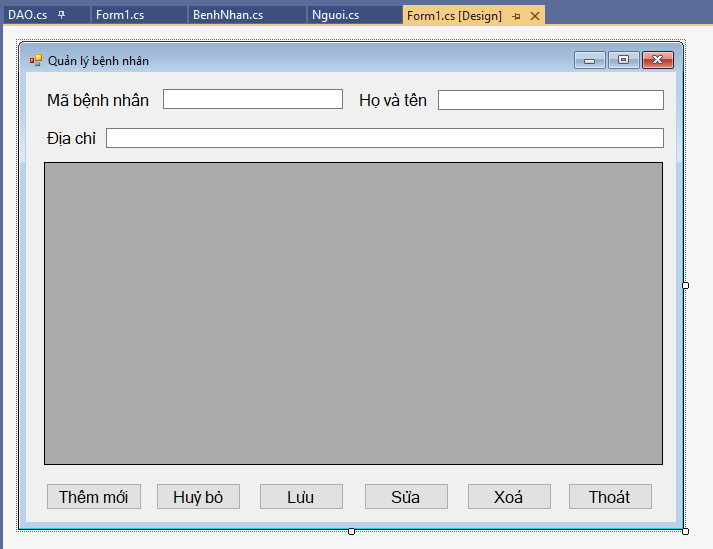
# Tạo tblBenhNhan



# Dữ liệu trong tblBenhNhan

## 

# Form QuanLyBenhNhan



# Source Code

### Class DAO

class DAO

{

public static SqlConnection con;

public static string connectionString = "Data Source=DESKTOP-JGNL2DP\\SQLEXPRESS;" +

"Initial Catalog=QuanLyBenhNhan;Integrated Security=True";

public static void setupConnection()

{

con = new SqlConnection();

con.ConnectionString = connectionString;

}

public static void openConnection()

{

if (con.State == ConnectionState.Closed)

con.Open();

}

public static void closeConnection()

{

if (con.State == ConnectionState.Open)

con.Close();

}

public static DataTable getDatatoTable(string sql)

{

SqlDataAdapter adapter = new SqlDataAdapter(sql, con);

DataTable table = new DataTable();

adapter.Fill(table);

return table;

}

public static void executeQuery(string sql)

{

SqlCommand command = new SqlCommand(sql, con);

command.ExecuteNonQuery();

}

public static bool check\_key\_exit(string sql)

{

SqlDataAdapter myadapter = new SqlDataAdapter(sql, con);

DataTable mytable = new DataTable();

myadapter.Fill(mytable);

if (mytable.Rows.Count >= 1)

return true;

else return false;

}

}

### Class Form1

public partial class Form1 : Form

{

public Form1()

{

InitializeComponent();

}

private void Form1\_Load(object sender, EventArgs e)

{

DAO.setupConnection();

try

{

DAO.openConnection();

LoadDatatoGridView();

DAO.closeConnection();

}

catch (Exception ex)

{

MessageBox.Show(ex.ToString());

}

}

private void LoadDatatoGridView()

{

string sql = "Select \* from tblBenhNhan";

DataTable table = DAO.getDatatoTable(sql);

dataGridViewBenhNhan.DataSource = table;

dataGridViewBenhNhan.Columns[0].HeaderText = "Mã bệnh nhân";

dataGridViewBenhNhan.Columns[1].HeaderText = "Họ và tên";

dataGridViewBenhNhan.Columns[2].HeaderText = "Địa chỉ";

dataGridViewBenhNhan.Columns[0].Width = 80;

dataGridViewBenhNhan.Columns[1].Width = 140;

dataGridViewBenhNhan.Columns[2].Width = 100;

dataGridViewBenhNhan.AllowUserToAddRows = false;

dataGridViewBenhNhan.EditMode = DataGridViewEditMode.EditProgrammatically;

}

private void dataGridViewBenhNhan\_Click(object sender, EventArgs e)

{

DAO.openConnection();

txtMaBN.Text = dataGridViewBenhNhan.CurrentRow.Cells["MaBN"].Value.ToString();

txtHoTen.Text = dataGridViewBenhNhan.CurrentRow.Cells["HoTen"].Value.ToString();

txtDiaChi.Text = dataGridViewBenhNhan.CurrentRow.Cells["DiaChi"].Value.ToString();

txtMaBN.Enabled = false;

}

private void btnThemMoi\_Click(object sender, EventArgs e)

{

txtMaBN.Enabled = true;

txtMaBN.Clear();

txtHoTen.Clear();

txtDiaChi.Clear();

txtMaBN.Focus();

}

private void btnHuy\_Click(object sender, EventArgs e)

{

txtMaBN.Clear();

txtHoTen.Clear();

txtDiaChi.Clear();

}

private void btnLuu\_Click(object sender, EventArgs e)

{

if (txtHoTen.Text.Trim() == "")

{

MessageBox.Show("Không được để trống Họ và tên");

txtHoTen.Focus();

return;

}

if (txtDiaChi.Text.Trim() == "")

{

MessageBox.Show("Không được để trống Địa chỉ");

txtDiaChi.Focus();

return;

}

string sql\_check\_key = "select \* from tblBenhNhan where MaBN = '"

+ txtMaBN.Text.Trim() + "'";

if (DAO.check\_key\_exit(sql\_check\_key) == true)

{

MessageBox.Show("Mã bệnh nhân đã tồn tại");

txtMaBN.Focus();

return;

}

string sql = "Insert into tblBenhNhan values('" + txtMaBN.Text + "',N'"

+ txtHoTen.Text + "',N'" + txtDiaChi.Text + "')";

//MessageBox.Show(sql);

try

{

DAO.openConnection();

DAO.executeQuery(sql);

MessageBox.Show("Cập nhật dữ liệu thành công");

LoadDatatoGridView();

DAO.closeConnection();

}

catch (Exception ex)

{

MessageBox.Show(ex.ToString());

}

}

private void btnSua\_Click(object sender, EventArgs e)

{

txtMaBN.Enabled = false;

if (txtHoTen.Text.Trim() == "")

{

MessageBox.Show("Không được để trống Họ và tên");

txtHoTen.Focus();

return;

}

if (txtDiaChi.Text.Trim() == "")

{

MessageBox.Show("Không được để trống Địa chỉ");

txtDiaChi.Focus();

return;

}

string sql = "Update tblBenhNhan set HoTen = N'" + txtHoTen.Text.Trim()

+ "', DiaChi = N'" + txtDiaChi.Text.Trim() + "' where MaBN = '"

+ txtMaBN.Text + "'";

MessageBox.Show(sql);

try

{

DAO.openConnection(); ;

DAO.executeQuery(sql);

LoadDatatoGridView();

DAO.closeConnection();

}

catch (Exception ex)

{

MessageBox.Show(ex.ToString());

}

}

private void btnXoa\_Click(object sender, EventArgs e)

{

string sql = "Delete from tblBenhNhan where MaBN = '" + txtMaBN.Text + "'";

try

{

DAO.openConnection();

DAO.executeQuery(sql);

MessageBox.Show("Xoá dữ liệu thành công");

LoadDatatoGridView();

DAO.closeConnection();

}

catch (Exception ex)

{

MessageBox.Show(ex.ToString());

}

}

private void btnThoat\_Click(object sender, EventArgs e)

{

this.Close();

}

}