

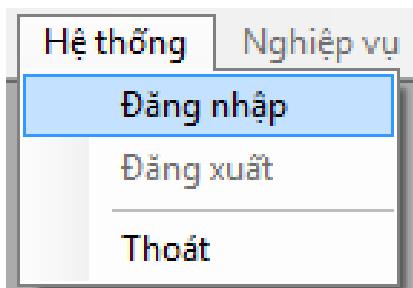
BÀI TẬP THỰC HÀNH 07

Lưu ý:

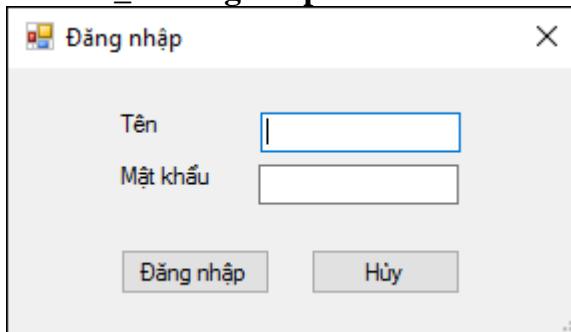
- SV sử dụng tập tin dữ liệu **LTQL_B7\qlnv.sql** cho sẵn.
- SV sử dụng **LTQL_B7\QLNhanVien.sln** (bài mẫu của buổi trước) để làm tiếp các yêu cầu trong bài tập này.

Yêu cầu:

- Bổ sung vào menu **Hệ thống** các lệnh: **Đăng nhập, Đăng xuất**



- Lệnh Đăng nhập sẽ mở **frm_htDangNhap**:



- Đặt lại thuộc tính cho các phần tử như sau:

frm_htDangNhap	StartPosition = CenterParent
	MaximizeBox = False
	MinimizeBox = False
txtTen	Modifiers = Public
txtMatKhau	Modifiers = Public
btnDangNhap	DialogResult = OK
btnHuy	DialogResult = Cancel

- Giả sử chương trình chỉ có 1 tài khoản người dùng (tên admin, mật khẩu 123)

- Tại form frmMain, ta viết hàm **HienThiMenu()**:
 - nếu chưa đăng nhập: bật Đăng nhập, tắt Đăng xuất và tắt các menu còn lại;
 - nếu đã đăng nhập: tắt Đăng nhập, bật Đăng xuất và bật các menu còn lại.

frmMain.cs

```
public bool bDangNhap; // true: đăng nhập thành công, false: chưa đăng nhập
frm_htDangNhap fDN;

// Hàm hiển thị các lệnh trên menu tương ứng với trạng thái đăng nhập
private void HienThiMenu() {
    i_htDangNhap.Enabled = !bDangNhap;
    i_htDangXuat.Enabled = bDangNhap;
    i_NghiepVu.Enabled = bDangNhap;
    // tương tự cho các menu còn lại
```

```

    }
    private void frmChinh_Load(object sender, EventArgs e)
    {
        HienThiMenu();
    }

    private void mnu_i_htDangNhap_Click(object sender, EventArgs e)
    {
        fDN=new frm_htDangNhap();
        if (fDN.ShowDialog() == DialogResult.OK)
        {
            // kiểm tra đúng tên và mật khẩu
            if( fDN.txtTen.Text.ToString()=="admin" &&
                fDN.txtMatKhau.Text.ToString()=="123")
            {
                bDangNhap = true; // đăng nhập thành công
            }
            else
            {
                bDangNhap = false; // đăng nhập thất bại hoặc chưa đăng nhập
            }
            HienThiMenu();
        }

        private void mnu_i_htDangXuat_Click(object sender, EventArgs e)
        {
            // Đóng tất cả form đang mở
            foreach (Form f in this.MdiChildren)
            {
                if (!f.IsDisposed)
                    f.Close();
            }
            // Đăng xuất và thiết lập lại menu
            bDangNhap = false;
            HienThiMenu();
        }
    }
}

```

3. Giả sử chương trình có nhiều tài khoản người dùng, mỗi tài khoản có quyền truy cập menu khác nhau

- Cơ sở dữ liệu có thêm bảng **nguoidung** và dữ liệu mẫu sau (có mã hóa mật khẩu bằng thuật toán MD5):

qlnv.sql

```

-- Cấu trúc bảng nguoidung
CREATE TABLE nguoidung(
    ten nvarchar(30) not null primary key,
    matkhau varchar(50),
    quyen tinyint
)
GO

-- Dữ liệu bảng nguoidung
INSERT INTO nguoidung
VALUES (N'quantri','e99a18c428cb38d5f260853678922e03',1); /*Mật khẩu abc123*/
INSERT INTO nguoidung
VALUES (N'nhanvien','a906449d5769fa7361d7ecc6aa3f6d28',2); /*Mật khẩu 123abc*/

```

- Trên form chính viết lại một số hàm: Hiển thị menu, Đăng nhập

frmMain.cs

```

using BUS;
using DTO;

namespace GUI
{
    public partial class frmMain : Form
    {

        public bool bDangNhap;
        public NguoiDung.DTO NguoiDung;
        frm_htDangNhap fDN;

        private void HienThiMenu() {
            // Hiển thị thông tin đăng nhập lên thanh trạng thái
            if (bDangNhap == true)
            {
                sttTTNguoiDung.Text = "Người dùng: " + NguoiDung.STen ;
                sttTTThoiGian.Text = "Thời điểm đăng nhập: " + DateTime.Now ;
                // Hiển thị menu theo quyền, ví dụ:
                // 1. quantri: sử dụng tất cả menu
                // 2. nhanvien: không sử dụng các menu: Danh mục, Nghiệp vụ
                int iQuyen;
                if (NguoiDung == null)
                {
                    iQuyen = 0;
                }
                else
                {
                    iQuyen = int.Parse(NguoiDung.IQuyen.ToString());
                }
                switch (iQuyen)
                {
                    case 1: // SV tự bổ sung
                        break;
                    case 2: // SV tự bổ sung
                        break;
                    default:
                        break;
                }
            }
            else
            {
                sttTTNguoiDung.Text = "Chưa đăng nhập";
                sttTTThoiGian.Text = "";
            }
        }

        private void i_htDangNhap_Click(object sender, EventArgs e)
        {
            fDN = new frm_htDangNhap();
            if (fDN.ShowDialog() == DialogResult.OK)
            {
                string sTen = fDN.txtTen.Text;
                string sMatKhau = fDN.txtMatKhau.Text;

                NguoiDung = new NguoiDung.DTO();
                NguoiDung = NguoiDung_BUS.LayNguoiDung(sTen, sMatKhau);
                if (NguoiDung != null)
                {
                    bDangNhap = true;
                }
            }
        }
    }
}

```

```

        else
        {
            bDangNhap = false;
        }
        HienThiMenu();
    }

}

```

- Lớp **NguoiDung_BUS**: Bổ sung hàm mã hóa

[NguoiDung_BUS.cs](#)

```

using DTO;
using DAO;
using System.Security.Cryptography;

namespace BUS
{
    public class NguoiDung_BUS
    {
        public static NguoiDung.DTO LayNguoiDung(string ten, string matkhau) {
            MD5 md5Hash = MD5.Create();
            string matkhauMH = NguoiDung_BUS.GetMd5Hash(md5Hash, matkhau);
            return NguoiDung.DAO.LayNguoiDung(ten, matkhauMH);
        }

        // Hàm mã hóa
        // Tham khảo tại https://msdn.microsoft.com/en-us/library/system.security.cryptography.md5.aspx
        public static string GetMd5Hash(MD5 md5Hash, string input)
        {

            // Convert the input string to a byte array and compute the hash.
            byte[] data = md5Hash.ComputeHash(Encoding.UTF8.GetBytes(input));

            // Create a new StringBuilder to collect the bytes
            // and create a string.
            StringBuilder sBuilder = new StringBuilder();

            // Loop through each byte of the hashed data
            // and format each one as a hexadecimal string.
            for (int i = 0; i < data.Length; i++)
            {
                sBuilder.Append(data[i].ToString("x2"));
            }

            // Return the hexadecimal string.
            return sBuilder.ToString();
        }
    }
}

```

- Lớp **NguoiDung.DAO**, **NguoiDung.DTO**: SV tự viết.