

```
SELECT a.IDChucNang,KiHieu,TenChucNang,TongQuyen
FROM PhanQuyen a INNER JOIN dbo.ChucNang b ON a.IDChucNang=b.IdChucNang
INNER JOIN dbo.NhanVien c ON a.IDLoaiTaiKhoan=c.LoaiTaiKhoan
WHERE c.TaiKhoan=@TaiKhoan
END
```

Khai báo biến để sử dụng

- Khai báo biến Hashtable DsPhanQuyenTheoUser trong Cls_Main
- Viết phương thức lấy tongQuyen theo user trong BLLDangNhap

```
public DataTable LayQuyenTheoUser(ref string err,string username)
{
    return data.GetDataTable("SP_NhanVien_LayTongQuyentheoUser", CommandType.StoredProcedure, ref err, new SqlParameter("@TaiKhoan", username));
}
```

Các phương thức

Phương thức trong form main để phân quyền menu. Phương thức này được gọi trong sự kiện load của form main

```
private void ExplorerMenu(MenuStrip menu)
{
    if (menu == null)
        return;

    foreach (ToolStripMenuItem _item in menu.Items)
    {
        ExplorerItem(_item);
    }
}
/// <summary>
/// Phương thức duyệt subitem trên từng menu.
/// </summary>
/// <param name="_subitem"></param>
private void ExplorerItem(ToolStripMenuItem _subitem)
{
    if (_subitem.DropDownItems.Count > 0)
    {
        for (int j = 0; j < _subitem.DropDownItems.Count; j++)
        {
            if (_subitem.DropDownItems[j] is ToolStripMenuItem)
            {
                ExplorerItem((ToolStripMenuItem)_subitem.DropDownItems[j]);
            }
        }
    }
    if (Cls_Main.DsPhanQuyentheoUser.Contains(_subitem.Name))
    {
        if ((Convert.ToInt32(Cls_Main.DsPhanQuyentheoUser[_subitem.Name].ToString()) & 1) == 1)
        {
            _subitem.Visible = true;
        }
        else
        {
            _subitem.Visible = false;
        }
    }
}
```

```
}
}
```

Để có dữ liệu trong HashTable chứa chức năng và tổng quyền thực hiện code trong form đăng nhập. sau khi đăng nhập thành công sẽ add dữ liệu vào trong hashtable. Biến Hashtable DsPhanQuyentheoUser được khai báo toàn cục trong Cls_Main để sử dụng cho toàn chương trình.

```
if(!string.IsNullOrEmpty(txtTaiKhoan.Text))
{
    if (!string.IsNullOrEmpty(txtMatKhau.Text))
    {
        if (kiemtradangnhap(txtTaiKhoan.Text, txtMatKhau.Text))
        {
            DataTable dt= bd.LayQuyentheoUser(ref err, txtTaiKhoan.Text);
            if(dt!=null)
            {
                for (int i = 0; i < dt.Rows.Count; i++)
                {
                    if(Cls_Main.DsPhanQuyentheoUser.Contains(dt.Rows[i]["KiHieu"].ToString())
                    ==false)
                    {
                        Cls_Main.DsPhanQuyentheoUser.Add(dt.Rows[i]["KiHieu"].ToString(), Convert.ToInt32(dt.Rows[i]["TongQuyentheoUser"].ToString()));
                    }
                }
            }
            this.Close();
        }
        else
        {
            pnlerr.Visible = true;
            this.Height = panel1.Height + pnlerr.Height + 25 + label3.Height + pnllink.Height;
            err = "Đăng nhập không thành công";
            lblerr.Text = err;
        }
    }
    else
    {
        MessageBox.Show("Chưa nhập Mật khẩu");
        txtMatKhau.Focus();
    }
}
else
{
    MessageBox.Show("Chưa nhập Tài khoản");
    txtTaiKhoan.Focus();
}
```