

酒店管理系统

一、登录

```
using System.Windows.Forms;
using BLL;
using Models;
1. 登录按钮
private void btnLogin_Click(object sender, EventArgs e)
{
    //1. 接收用户输入的数据
    string strName = txtName.Text;
    string strPwd = PwdEncrypt.GetMD5(txtPwd.Text);

    //2. 验证文本框是否为空
    if (string.IsNullOrEmpty(strName))
    {
        MessageBox.Show("用户名不能为空", "提示", MessageBoxButtons.OK,
        MessageBoxIcon.Information);
        txtName.Focus();
        return; //退出, 不再执行后面的代码
    }
    if (string.IsNullOrEmpty(strPwd))
    {
        MessageBox.Show("密码不能为空", "提示", MessageBoxButtons.OK,
        MessageBoxIcon.Information);
        txtPwd.Focus();
        return;
    }
    //3. 验证数据库中是否存在该用户
    User user = UserManager.GetUser(strName, strPwd);
    if (user == null)
    {
        MessageBox.Show("用户名或密码错误", "提示", MessageBoxButtons.OK,
        MessageBoxIcon.Information);
    }
    else
    {
        RoleManager.curUser = user; //将用户存为当前用户, 用于后面的操作
        this.Hide();
        MainForm frm = new MainForm();
        frm.Show();
    }
}
2. 退出系统
private void btnExit_Click(object sender, EventArgs e)
```

```

    {
        Application.Exit();
    }

```

二、主窗体

```

using System.Windows.Forms;
using BLL;

```

1. 窗体加载

```

private void MainForm_Load(object sender, EventArgs e)
{
    //当前用户如果是普通员工，则不显示与用户管理相关的功能
    if (RoleManager.curUser.Role == 0)
    {
        miUserManage.Visible = false;
        miNewUser.Visible = false;
        miEditUser.Visible = false;
    }
}

```

三、添加新用户

```

using System.Windows.Forms;
using Models;
using BLL;

```

1. 添加按钮

```

private void btnAdd_Click(object sender, EventArgs e)
{
    //1. 接收用户输入的数据
    string strName = txtName.Text;
    string strPwd = PwdEncrypt.GetMD5 ( txtPwd.Text);
    //员工（0，默认）；管理员（1）
    int role = 0;
    if (rbAdmin.Checked == true)
    {
        role = 1;
    }
    //2. 验证文本框是否为空
    if (string.IsNullOrEmpty(strName))
    {
        MessageBox.Show("用户名不能为空", "提示", MessageBoxButtons.OK,
        MessageBoxIcon.Information);
        txtName.Focus();
        return; //退出，不再执行后面的代码
    }
    if (string.IsNullOrEmpty(strPwd))
    {
        MessageBox.Show("密码不能为空", "提示", MessageBoxButtons.OK,

```

```

MessageBoxIcon.Information);
        txtPwd.Focus();
        return;
    }
    //3. 存入（数据库）
    try
    {
        User user = new User(strName, strPwd, role);
        if (UserManager.AddUser(user))
        {
            MessageBox.Show("添加成功！");
        }
        else
        {
            MessageBox.Show("添加失败！");
        }
    }
    catch (Exception)
    {
        MessageBox.Show("添加失败！！");
    }
}

```

2. 取消按钮

```

private void btnCancel_Click(object sender, EventArgs e)
{
    //1. 清空文本框
    txtName.Text = "";
    txtPwd.Text = "";
    rbEmp.Checked = true;
    //2. 关闭当前窗口
    this.Close();
}

```

四、修改/删除用户

```

using System.Windows.Forms;
using BLL;
using Models;

```

1. 窗体加载

```

private void UserManagerForm_Load(object sender, EventArgs e)
{
    DataTable dt = UserManager.GetUserInfo();
    dgvUsers.DataSource = dt;
}

```

2. 在表格中点击某个用户时

```

private void dgvUsers_CellContentClick(object sender, DataGridViewCellEventArgs e)

```

```

{
    txtUserName.Text = dgvUsers.SelectedRows[0].Cells[0].Value.ToString();
    txtPwd.Text = dgvUsers.SelectedRows[0].Cells[1].Value.ToString();
    txtRole.Text = dgvUsers.SelectedRows[0].Cells[2].Value.ToString();
}

```

3. 用户输入验证

```

private bool checkdata()
{
    if (txtUserName.Text == "")
    {
        MessageBox.Show("用户名不能为空");
        txtUserName.Focus();
        return false;
    }
    if (txtPwd.Text == "")
    {
        MessageBox.Show("密码不能为空");
        txtPwd.Focus();
        return false;
    }
    if (txtRole.Text == "")
    {
        MessageBox.Show("权限不能为空");
        txtRole.Focus();
        return false;
    }

    return true;
}

```

4. 修改按钮

```

private void btnChange_Click(object sender, EventArgs e)
{
    User user = new User();
    user.Name = txtUserName.Text;
    user.Pwd = PwdEncrypt.GetMD5 ( txtPwd.Text);
    user.Role = int.Parse(txtRole.Text);
    if (checkdata())
    {
        if (UserManager.UpdateUser(user))
        {
            MessageBox.Show("用户信息修改成功");
            DataTable dt = UserManager.GetUserInfo();
            dgvUsers.DataSource = dt;
        }
    }
}

```

```

        else
        {
            MessageBox.Show("用户信息修改失败");
        }
    }
}

```

5. 删除按钮

```

private void btnDel_Click(object sender, EventArgs e)
{
    if (MessageBox.Show("确定要删除用户数据吗？", "提示", MessageBoxButtons.OKCancel,
        MessageBoxIcon.Information) == DialogResult.OK)
    {
        User user = new User();
        user.Name = txtUserName.Text;
        user.Pwd = txtPwd.Text;
        user.Role = int.Parse(txtRole.Text);
        if (UserManager.DeleteUser(user))
        {
            MessageBox.Show("用户删除成功");
            DataTable dt = UserManager.GetUserInfo();
            dgvUsers.DataSource = dt;
        }
        else
        {
            MessageBox.Show("用户删除失败");
        }
    }
}

```

五、入住登记

```

using System.Windows.Forms;
using BLL;
using Models;

```

1. 窗体加载

```

private void CheckInRoomForm_Load(object sender, EventArgs e)
{
    txtOper.Text = RoleManager.curUser.Name;
    DataTable dt = RoomManager.GetRoomInfo();
    if (dt.Rows.Count != 0)
    {
        dgvRoomList.DataSource = dt;
    }
    else
    {
        MessageBox.Show("已没有空房间！");
    }
}

```

```

        return;
    }
}

```

2、选择某个房间

```

private void dgvRoomList_CellContentClick(object sender, DataGridViewCellEventArgs e)
{
    if (dgvRoomList.SelectedRows.Count == 1)
    {
        this.txtRoomId.Text = Convert.ToString(dgvRoomList.SelectedRows[0].Cells["房间号"].Value.ToString());
        this.txtPrice.Text = Convert.ToString(dgvRoomList.SelectedRows[0].Cells["价格"].Value.ToString());
    }
}

```

3、验证用户输入

```

private bool CheckData()
{
    if (txtRoomId.Text == "")
    {
        MessageBox.Show("房间号不能为空");
        txtRoomId.Focus();
        return false;
    }
    if (txtPrice.Text == "")
    {
        MessageBox.Show("价格不能为空");
        txtPrice.Focus();
        return false;
    }
    if (txtForegift.Text == "")
    {
        MessageBox.Show("押金不能为空");
        txtForegift.Focus();
        return false;
    }
    if (txtClientName.Text == "")
    {
        MessageBox.Show("住客姓名不能为空");
        txtClientName.Focus();
        return false;
    }
    if (txtPhone.Text == "")
    {
        MessageBox.Show("电话不能为空");
    }
}

```

```

        txtPhone.Focus();
        return false;
    }
    if (txtCertId.Text == "")
    {
        MessageBox.Show("证件号不能为空");
        txtCertId.Focus();
        return false;
    }
    if (txtAddress.Text == "")
    {
        MessageBox.Show("地址不能为空");
        txtAddress.Focus();
        return false;
    }
    if (txtPersonNum.Text == "")
    {
        MessageBox.Show("住宿人数不能为空");
        txtPersonNum.Focus();
        return false;
    }
    else if (Convert.ToInt32(txtPersonNum.Text) >
Convert.ToInt32(dgvRoomList.SelectedRows[0].Cells["可入住人数"].Value.ToString()))
    {
        MessageBox.Show("入住人数超过可容纳人数，请更换房间");
        txtRoomId.Text = "";
        txtPrice.Text = "";
        txtPersonNum.Text = "";
        txtRoomId.Focus();
        return false;
    }
    DateTime inTime = DateTime.Parse(dtpInTime.Text.ToString());
    DateTime outTime = DateTime.Parse(dtpOutTime.Text.ToString());
    if (inTime.CompareTo(DateTime.Today) < 0)
    {
        MessageBox.Show("入住时间不能小于当前时间");
        dtpInTime.Focus();
        return false;
    }
    else if (outTime.CompareTo(DateTime.Today) < 0)
    {
        MessageBox.Show("离开时间不能小于当前时间");
        dtpOutTime.Focus();
        return false;
    }

```

```

    }
    else if (outTime < inTime)
    {
        MessageBox.Show("离开时间不能小于入住时间");
        dtpOutTime.Focus();
        return false;
    }
    return true;
}

```

4、保存按钮

```

private void btnSave_Click(object sender, EventArgs e)
{
    string strTimeNow = string.Format("{0:T}", DateTime.Now);
    string inTime = dtpInTime.Text.ToString() + strTimeNow;
    string outTime = dtpOutTime.Text.ToString() + strTimeNow;
    bool sex = true;
    if (rbMale.Checked)
    {
        sex = false;
    }
    CheckInRoom registerInfo = new CheckInRoom();
    registerInfo.RoomId = txtRoomId.Text;
    registerInfo.Price = Convert.ToDouble(txtPrice.Text);
    registerInfo.Foregift = Convert.ToDouble(txtForegift.Text);
    registerInfo.InTime = Convert.ToDateTime(inTime);
    registerInfo.OutTime = Convert.ToDateTime(outTime);
    registerInfo.ClientName = txtClientName.Text;
    registerInfo.Sex = sex;
    registerInfo.Phone = txtPhone.Text;
    registerInfo.CertType = cmbCertType.SelectedItem.ToString();
    registerInfo.CertId = txtCertId.Text;
    registerInfo.PersonNum = Convert.ToInt32(txtPersonNum.Text);
    registerInfo.Oper = txtOper.Text;
    registerInfo.Address = txtAddress.Text;
    registerInfo.DelMark = 0;
    if (CheckData())
    {
        if (RoomManager.InsertRoomInfo(registerInfo))
        {
            MessageBox.Show("入住登记成功！");
        }
        else
        {
            MessageBox.Show("入住登记失败！");
        }
    }
}

```



```

    }
}
}

```

5、重置按钮

```

private void btnReset_Click(object sender, EventArgs e)
{
    txtRoomId.Text = "";
    txtPrice.Text = "";
    txtForegift.Text = "";
    dtpInTime.Text = "";
    dtpOutTime.Text = "";
    rbMale.Checked = true;
    txtClientName.Text = "";
    txtPhone.Text = "";
    cmbCertType.SelectedIndex = 0;
    txtCertId.Text = "";
    txtAddress.Text = "";
    txtPersonNum.Text = "";
}

```

六、退房登记

```

using System.Windows.Forms;
using BLL;
using Models;
using System.IO;

```

1、窗体初始化

```

private DataTable dt;
public CheckOutRoomForm()
{
    dt = RoomManager.GetUseRoomInfo();
    InitializeComponent();
}

```

2、窗体加载

```

private void CheckOutRoomForm_Load(object sender, EventArgs e)
{
    if (dt.Rows.Count > 0)
    {
        for (int i = 0; i <= dt.Rows.Count - 1; i++)
        {
            cmbRoomId.Items.Add(dt.Rows[i]["房间号"].ToString());
        }
    }
}

```

3、选择退房房间

```

private void cmbRoomId_SelectedIndexChanged(object sender, EventArgs e)

```

```

{
    for (int i = 0; i < dt.Rows.Count; i++)
    {
        if (cmbRoomId.SelectedItem.ToString() == dt.Rows[i]["房间号"].ToString())
        {
            txtPrice.Text = dt.Rows[i]["价格"].ToString();
            txtForegift.Text = dt.Rows[i]["押金"].ToString();
            txtInTime.Text = dt.Rows[i]["入住时间"].ToString();
            txtClientName.Text = dt.Rows[i]["住客姓名"].ToString();
            string outTime = dtpOutTime.Text + string.Format("{0:T}", DateTime.Now);
            TimeSpan ts1 = new TimeSpan(DateTime.Parse(outTime).Ticks);
            TimeSpan ts2 = new TimeSpan(DateTime.Parse(txtInTime.Text).Ticks);
            TimeSpan ts = ts1.Subtract(ts2);
            int dayCount = ts.Days;
            int hourCount = ts.Hours;
            if (dayCount == 0 & hourCount < 24)
            {
                dayCount = 1;
            }
            double consumTotal = double.Parse(txtPrice.Text.ToString()) * dayCount;
            txtTotal.Text = consumTotal.ToString();
            txtAccount.Text = Convert.ToString(consumTotal - double.Parse(txtForegift.Text));
            return;
        }
    }
}

```

4、验证用户输入

```

private bool checkdata()
{
    DateTime inTime =
Convert.ToDateTime(DateTime.Parse(txtInTime.Text.ToString()).ToString("yyyy-MM-dd"));
    DateTime outTime = DateTime.Parse(dtpOutTime.Text.ToString());
    if (outTime < inTime)
    {
        MessageBox.Show("结账日期不能小于入住日期");
        dtpOutTime.Focus();
        return false;
    }
    return true;
}

```

5、保存按钮

```

private void btnSave_Click(object sender, EventArgs e)
{
    if (checkdata() == false)

```

```

    {
        return;
    }

    string inTime = txtInTime.Text.ToString();
    string strTimeNow = string.Format("{0:T}", DateTime.Now);
    string outTime = dtpOutTime.Text.ToString() + strTimeNow;
    int inId =
int.Parse(RoomManager.GetInID(cmbRoomId.SelectedItem.ToString()).Rows[0][0].ToString());
    CheckOutRoom checkOutRoom = new CheckOutRoom();
    checkOutRoom.OutId = DateTime.Now.ToString("yyyyMMddHHmmss");
    checkOutRoom.InId = inId;
    checkOutRoom.RoomId = cmbRoomId.SelectedItem.ToString();
    checkOutRoom.Price = double.Parse(txtPrice.Text.ToString());
    checkOutRoom.Foregift = double.Parse(txtForegift.Text.ToString());
    checkOutRoom.Total = double.Parse(txtTotal.Text.ToString());
    checkOutRoom.Account = double.Parse(txtAccount.Text.ToString());
    checkOutRoom.InTime = DateTime.Parse(inTime);
    checkOutRoom.OutTime = DateTime.Parse(outTime);
    checkOutRoom.ClientName = txtClientName.Text.ToString();
    checkOutRoom.Oper = RoleManager.curUser.Name;
    checkOutRoom.Note = txtNote.Text.ToString();
    if (RoomManager.InsertCheckOutRoomInfo(checkOutRoom))
    {
        MessageBox.Show("退房信息保存成功");
        return;
    }
    else
    {
        MessageBox.Show("退房信息保存失败");
        return;
    }
}

```

6、重置按钮

```

private void btnReset_Click(object sender, EventArgs e)
{
    cmbRoomId.Text = "";
    txtClientName.Text = "";
    txtInTime.Text = "";
    txtPrice.Text = "";
    txtForegift.Text = "";
    txtTotal.Text = "";
    txtAccount.Text = "";
    txtNote.Text = "";
}

```

7、导出按钮

```
private void btnPrint_Click(object sender, EventArgs e)
{
    string path = "D:\\发票单\\" + DateTime.Now.ToString("yyyyMMddHHmmss") + ".txt";
    StreamWriter w = new StreamWriter(path);
    w.WriteLine("—专家公寓—");
    w.WriteLine("住客姓名: " + txtClientName.Text);
    w.WriteLine("住宿费: " + txtTotal.Text);
    w.WriteLine("开票日期: " + DateTime.Now.ToString("yyyy-MM-dd"));
    w.Close();
    MessageBox.Show("导出成功");
}
```

七、住客信息查询与修改

```
using System.Windows.Forms;
using BLL;
using Models;
```

1、窗体加载

```
private void ClientForm_Load(object sender, EventArgs e)
{
    DataTable dt = ClientManager.GetClientInfo();
    dgvClient.DataSource = dt;
}
```

2、格式化表格数据

```
private void dgvClient_CellFormatting(object sender, DataGridViewCellFormattingEventArgs e)
{
    if (e.ColumnIndex == 6)
    {
        if (e.Value.ToString() == "1")
        {
            e.Value = "女";
        }
        else
        {
            e.Value = "男";
        }
    }
    if (e.ColumnIndex == 13)
    {
        if (e.Value.ToString() == "0")
        {
            e.Value = "否";
        }
        else
        {

```

```

        e.Value = "是";
    }
}

```

3、查询按钮

```

private void btnQuery_Click(object sender, EventArgs e)
{
    string clientName = txtQueryName.Text;
    DataTable dt = ClientManager.GetAllClient(clientName);
    dgvClient.DataSource = dt;
}

```

4、点击表格中数据

```

private void dgvClient_CellContentClick(object sender, DataGridViewCellEventArgs e)
{
    var rows = dgvClient.SelectedRows;
    if (rows.Count == 0)
    {
        txtClientName.Text = "";
        rbMale.Checked = false;
        rbFemale.Checked = false;
        txtPhone.Text = "";
        cmbCertType.SelectedItem = "";
        txtCertId.Text = "";
        txtAddress.Text = "";
        txtPersonNum.Text = "";
        btnSave.Enabled = false;
        return;
    }
    btnSave.Enabled = true;
    var row = rows[0];
    txtClientName.Text = row.Cells[5].Value.ToString();
    if (row.Cells[6].Value.ToString() == "0")
    {
        rbMale.Checked = true;
    }
    else
    {
        rbFemale.Checked = true;
    }
    txtPhone.Text = row.Cells[7].Value.ToString();
    cmbCertType.SelectedItem = row.Cells[8].Value;
    txtCertId.Text = row.Cells[9].Value.ToString();
    txtAddress.Text = row.Cells[10].Value.ToString();
    txtPersonNum.Text = row.Cells[11].Value.ToString();
}

```

```
}
```

5、验证用户输入

```
private bool checkdata()
{
    if (txtClientName.Text == "")
    {
        MessageBox.Show("住客姓名不能为空");
        txtClientName.Focus();
        return false;
    }
    if (txtPhone.Text == "")
    {
        MessageBox.Show("电话号不能为空");
        txtPhone.Focus();
        return false;
    }
    if (txtCertId.Text == "")
    {
        MessageBox.Show("证件号不能为空");
        txtCertId.Focus();
        return false;
    }
    if (txtAddress.Text == "")
    {
        MessageBox.Show("地址不能为空");
        txtAddress.Focus();
        return false;
    }
    DataTable dt = RoomManager.GetRoomInfo(dgvClient.SelectedRows[0].Cells[0].Value.ToString());
    string personNum = dt.Rows[0][4].ToString();
    if (txtPersonNum.Text == "")
    {
        MessageBox.Show("住宿人数不能为空");
        txtPersonNum.Focus();
        return false;
    }
    else if (int.Parse(txtPersonNum.Text) > int.Parse(personNum))
    {
        MessageBox.Show("入住人数大于可容纳人数，请重新输入！");
        return false;
    }
    return true;
}
```

6、保存按钮

```

private void btnSave_Click(object sender, EventArgs e)
{
    bool sex = default(bool);
    if (rbMale.Checked)
    {
        sex = false;
    }
    else
    {
        sex = true;
    }
    CheckInRoom registerInfo = new CheckInRoom();
    registerInfo.RoomId = dgvClient.SelectedRows[0].Cells[0].Value.ToString();
    registerInfo.Price = double.Parse(dgvClient.SelectedRows[0].Cells[1].Value.ToString());
    registerInfo.Foregift = double.Parse(dgvClient.SelectedRows[0].Cells[2].Value.ToString());
    registerInfo.InTime = DateTime.Parse(dgvClient.SelectedRows[0].Cells[3].Value.ToString());
    registerInfo.OutTime = DateTime.Parse(dgvClient.SelectedRows[0].Cells[4].Value.ToString());
    registerInfo.ClientName = txtClientName.Text.ToString();
    registerInfo.Sex = sex;
    registerInfo.Phone = txtPhone.Text.ToString();
    registerInfo.CertType = cmbCertType.SelectedItem.ToString();
    registerInfo.CertId = txtCertId.Text.ToString();
    registerInfo.PersonNum = int.Parse(txtPersonNum.Text.ToString());
    registerInfo.Address = txtAddress.Text.ToString();
    registerInfo.Oper = dgvClient.SelectedRows[0].Cells[12].Value.ToString();
    registerInfo.DelMark = int.Parse(dgvClient.SelectedRows[0].Cells[13].Value.ToString());
    registerInfo.InId = int.Parse(dgvClient.SelectedRows[0].Cells[14].Value.ToString());
    if (checkdata())
    {
        if (ClientManager.UpdateRegister(registerInfo))
        {
            MessageBox.Show("住客信息修改成功");
            DataTable dt = ClientManager.GetClientInfo();
            dgvClient.DataSource = dt;
        }
        else
        {
            MessageBox.Show("住客信息修改失败");
        }
    }
}
}

```

八、客房信息查询

```

using System.Windows.Forms;
using BLL;

```

1、窗体加载

```
private void RoomForm_Load(object sender, EventArgs e)
{
    DataTable dt = RoomManager.GetRoomUseInfo();
    dgvRoom.DataSource = dt;
}
```

2、选中某个房间信息时

```
private void dgvRoom_SelectionChanged(object sender, EventArgs e)
{
    if (dgvRoom.SelectedRows.Count != 0)
    {
        string roomId = dgvRoom.SelectedRows[0].Cells[0].Value.ToString();
        DataTable dt = ClientManager.GetClient(roomId);
        dgvClient.DataSource = dt;
    }
}
```

3、住客信息表格数据格式化

```
private void dgvClient_CellFormatting(object sender, DataGridViewCellFormattingEventArgs e)
{
    if (e.ColumnIndex == 6)
    {
        if (e.Value.ToString() == "1")
        {
            e.Value = "女";
        }
        else
        {
            e.Value = "男";
        }
    }
}
```