SILVER OAKS HOTEL SYSTEM DOCUMENTATION

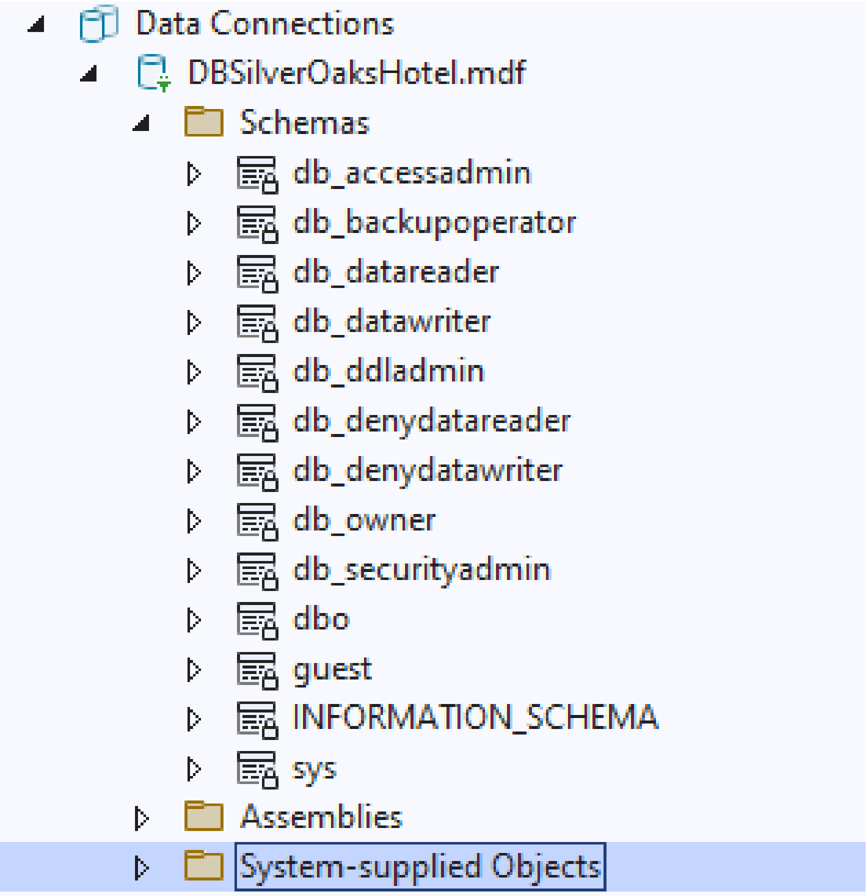
ADEQUATE SCREENSHOTS

GROUP 03

Hein du Plessis	– 41125614
Lohard j.v. Rensburg	– 37248529
Salim Sofinia	– 37284177
Coenri van Rooyen	– 37330527

TABLE OF CONTENTS

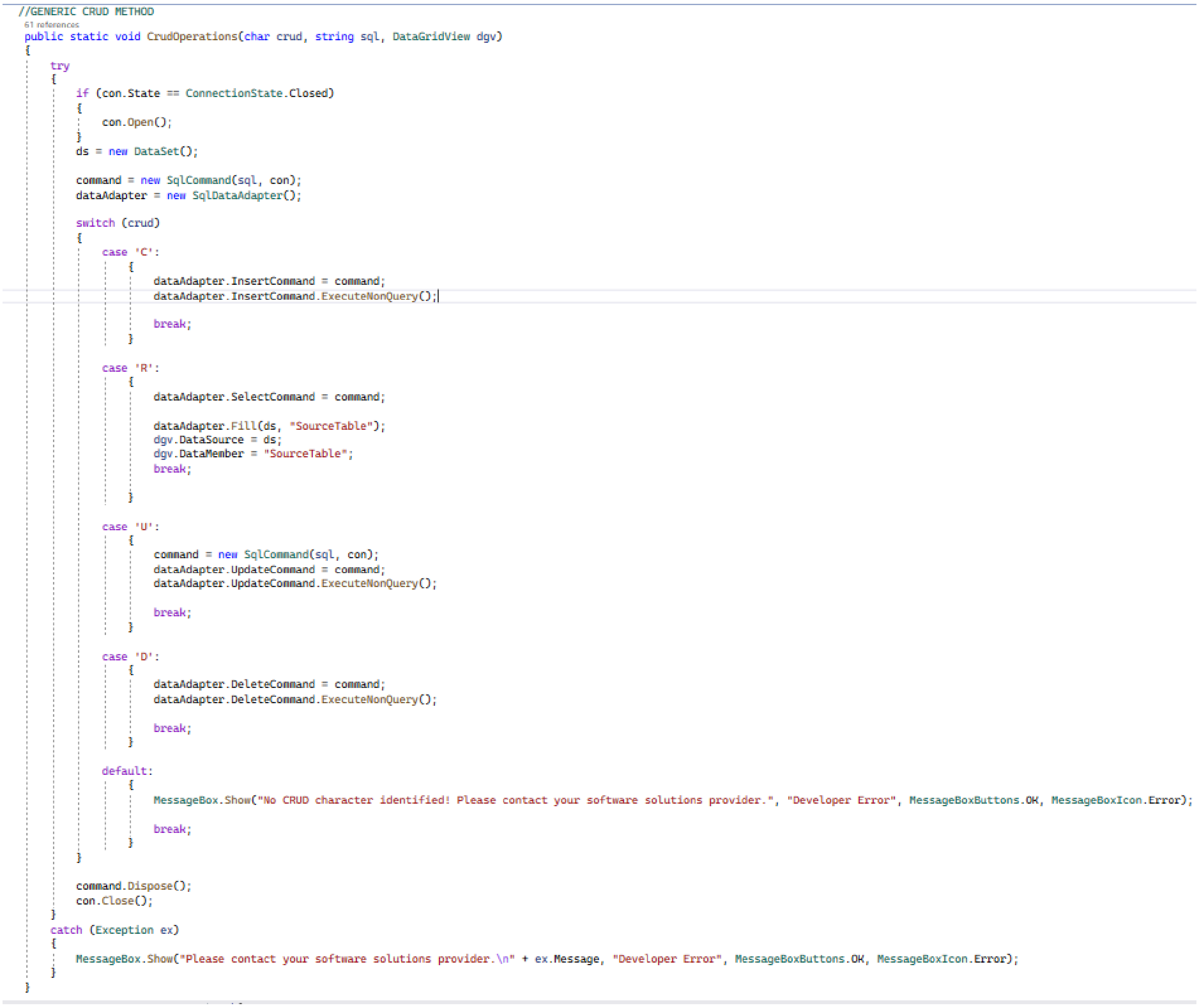
DATABASE SCHEMA



- Data Connections
  - DBSilverOaksHotel.mdf
    - Schemas
      - db_accessadmin
      - db_backupoperator
      - db_datareader
      - db_datawriter
      - db_ddladmin
      - db_denydatareader
      - db_denydatawriter
      - db_owner
      - db_securityadmin
      - dbo
      - guest
      - INFORMATION_SCHEMA
      - sys
    - Assemblies
    - System-supplied Objects

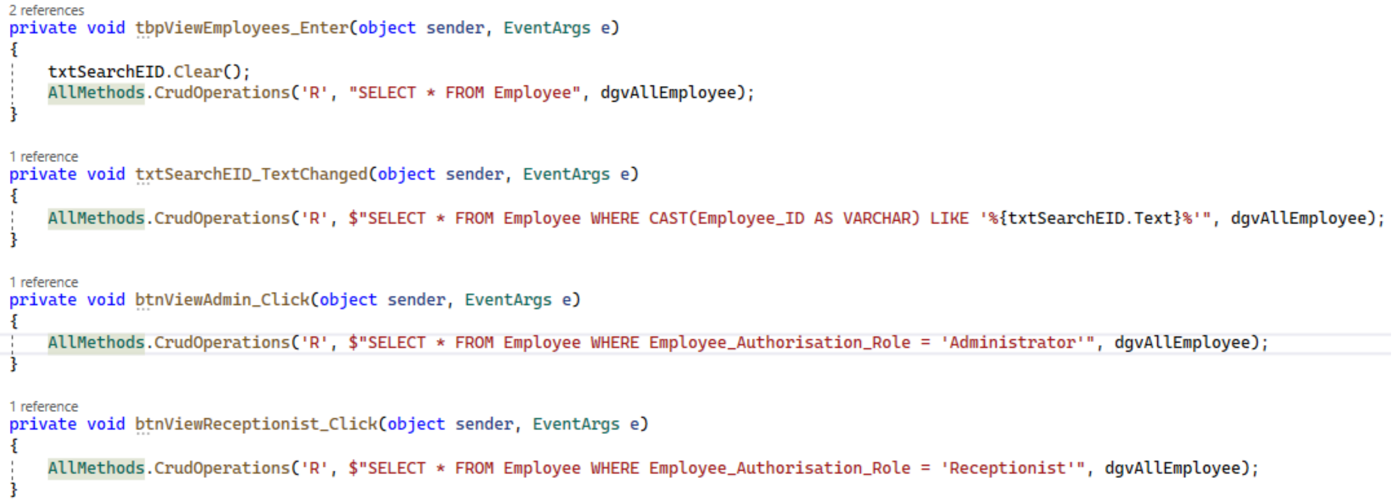
EFFICIENT REUSE OF CODE

The CrudOperations Method:



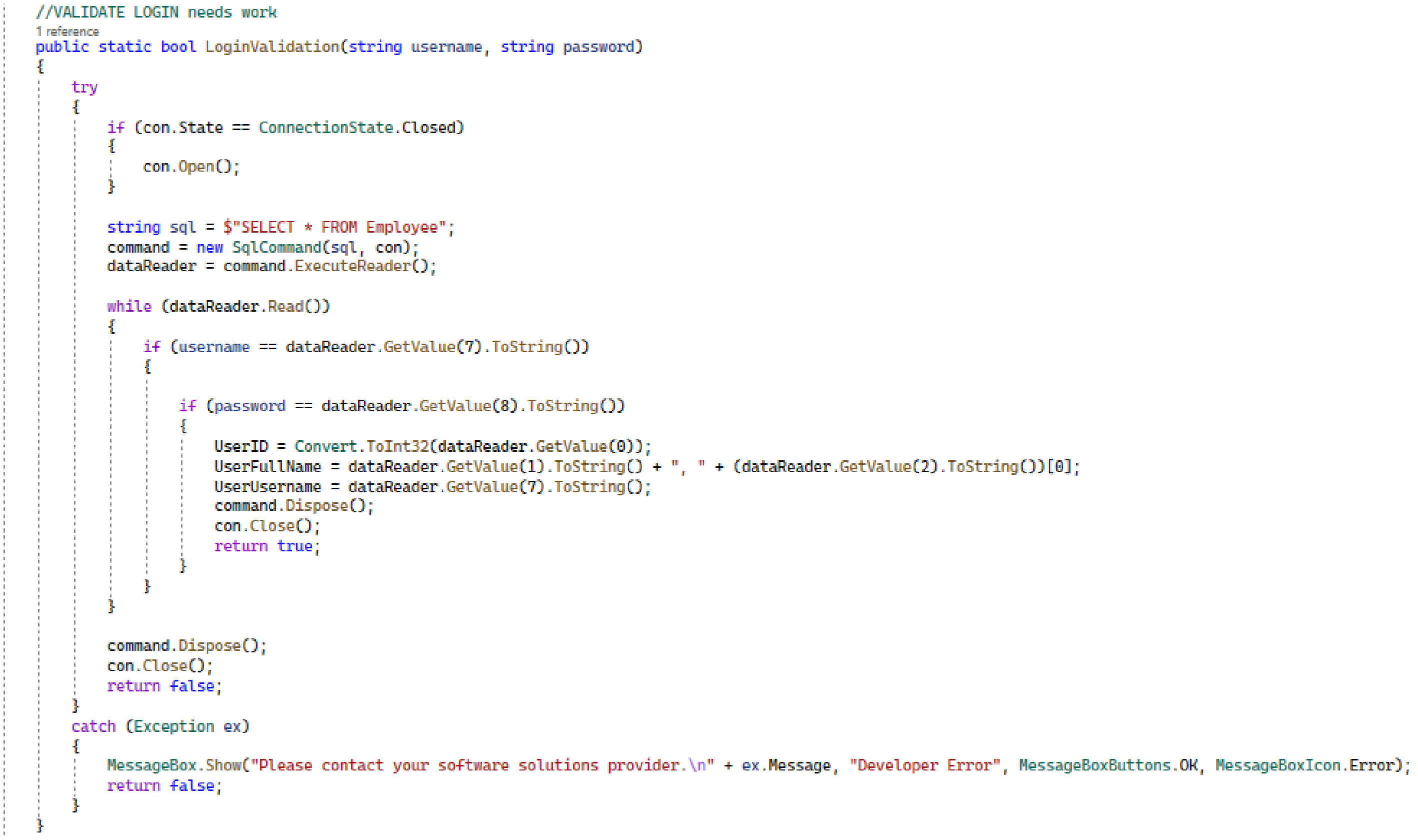
```
//GENERIC CRUD METHOD
61 references
public static void CrudOperations(char crud, string sql, DataGridView dgv)
{
    try
    {
        if (con.State == ConnectionState.Closed)
        {
            con.Open();
        }
        ds = new DataSet();

        command = new SqlCommand(sql, con);
        dataAdapter = new SqlDataAdapter();

        switch (crud)
        {
            case 'C':
                {
                    dataAdapter.InsertCommand = command;
                    dataAdapter.InsertCommand.ExecuteNonQuery();

                    break;
                }

            case 'R':
                {
                    dataAdapter.SelectCommand = command;

                    dataAdapter.Fill(ds, "SourceTable");
                    dgv.DataSource = ds;
                    dgv.DataMember = "SourceTable";
                    break;

                }

            case 'U':
                {
                    command = new SqlCommand(sql, con);
                    dataAdapter.UpdateCommand = command;
                    dataAdapter.UpdateCommand.ExecuteNonQuery();

                    break;
                }

            case 'D':
                {
                    dataAdapter.DeleteCommand = command;
                    dataAdapter.DeleteCommand.ExecuteNonQuery();

                    break;
                }

            default:
                {
                    MessageBox.Show("No CRUD character identified! Please contact your software solutions provider.", "Developer Error", MessageBoxButtons.OK, MessageBoxIcon.Error);

                    break;
                }
        }

        command.Dispose();
        con.Close();
    }
    catch (Exception ex)
    {
        MessageBox.Show("Please contact your software solutions provider.\n" + ex.Message, "Developer Error", MessageBoxButtons.OK, MessageBoxIcon.Error);
    }
}
```

The using of the CrudOperations Method:



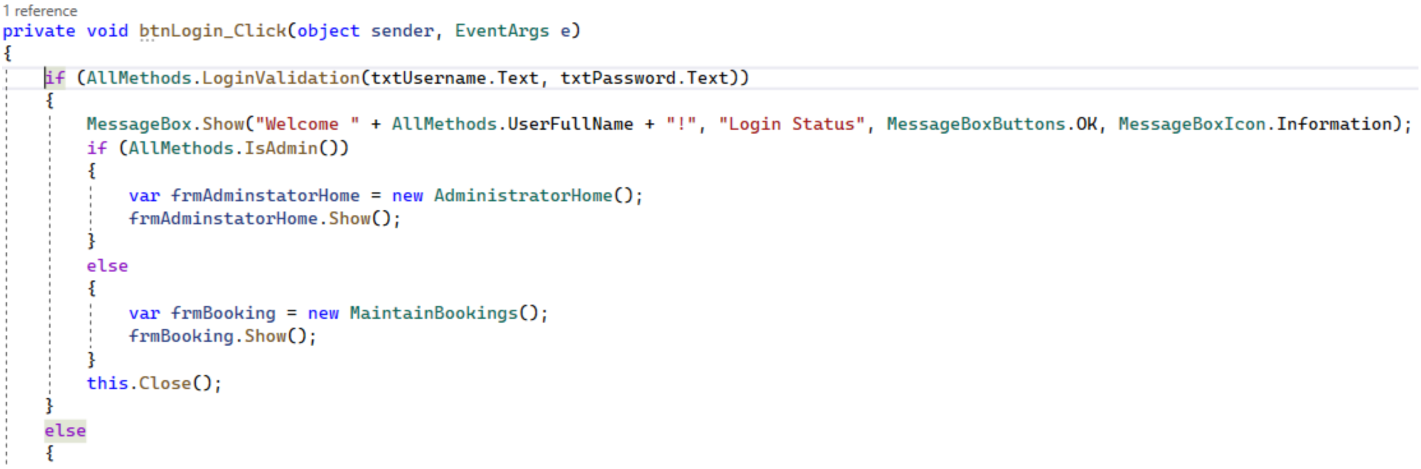
```
2 references
private void tbpViewEmployees_Enter(object sender, EventArgs e)
{
    txtSearchEID.Clear();
    AllMethods.CrudOperations('R', "SELECT * FROM Employee", dgvAllEmployee);
}

1 reference
private void txtSearchEID_TextChanged(object sender, EventArgs e)
{
    AllMethods.CrudOperations('R', $"SELECT * FROM Employee WHERE CAST(Employee_ID AS VARCHAR) LIKE '%{txtSearchEID.Text}%'", dgvAllEmployee);
}

1 reference
private void btnViewAdmin_Click(object sender, EventArgs e)
{
    AllMethods.CrudOperations('R', $"SELECT * FROM Employee WHERE Employee_Authorisation_Role = 'Administrator'", dgvAllEmployee);
}

1 reference
private void btnViewReceptionist_Click(object sender, EventArgs e)
{
    AllMethods.CrudOperations('R', $"SELECT * FROM Employee WHERE Employee_Authorisation_Role = 'Receptionist'", dgvAllEmployee);
}
```

The LoginValidation Method:



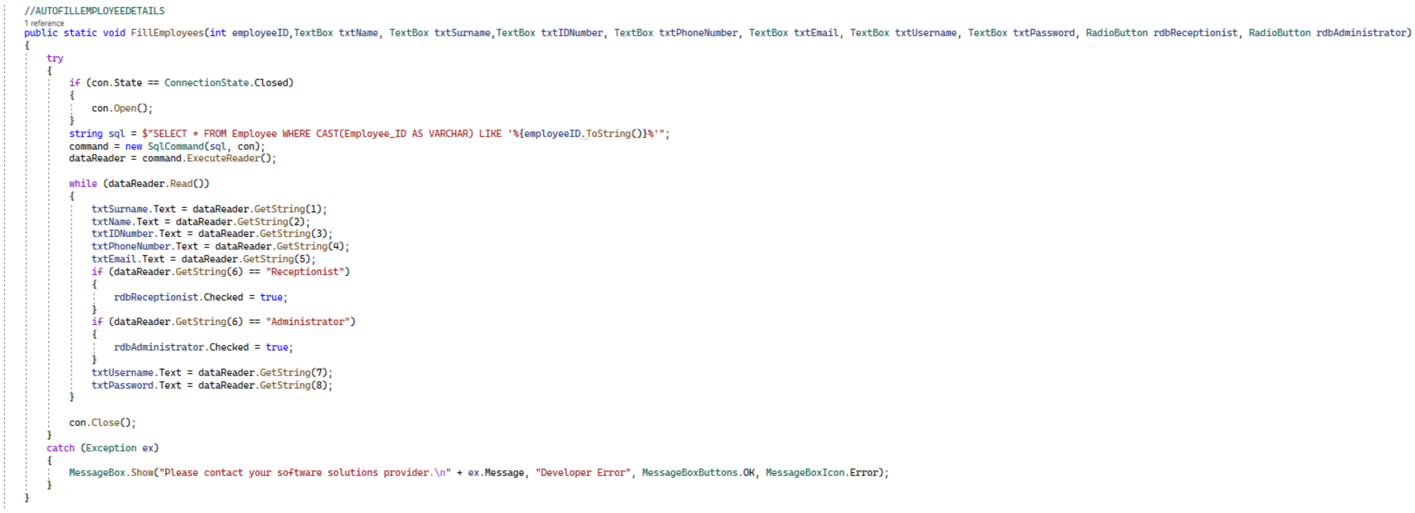
```
//VALIDATE LOGIN needs work
1 reference
public static bool LoginValidation(string username, string password)
{
    try
    {
        if (con.State == ConnectionState.Closed)
        {
            con.Open();
        }

        string sql = $"SELECT * FROM Employee";
        command = new SqlCommand(sql, con);
        dataReader = command.ExecuteReader();

        while (dataReader.Read())
        {
            if (username == dataReader.GetValue(7).ToString())
            {

                if (password == dataReader.GetValue(8).ToString())
                {
                    UserID = Convert.ToInt32(dataReader.GetValue(0));
                    UserFullName = dataReader.GetValue(1).ToString() + ", " + (dataReader.GetValue(2).ToString())[0];
                    UserUsername = dataReader.GetValue(7).ToString();
                    command.Dispose();
                    con.Close();
                    return true;
                }
            }
        }

        command.Dispose();
        con.Close();
        return false;
    }
    catch (Exception ex)
    {
        MessageBox.Show("Please contact your software solutions provider.\n" + ex.Message, "Developer Error", MessageBoxButtons.OK, MessageBoxIcon.Error);
        return false;
    }
}
```

The using of the LoginValidation Method:



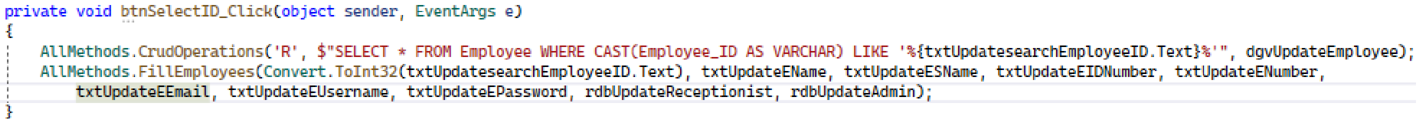
```
1 reference
private void btnLogin_Click(object sender, EventArgs e)
{
    if (AllMethods.LoginValidation(txtUsername.Text, txtPassword.Text))
    {
        MessageBox.Show("Welcome " + AllMethods.UserFullName + "!", "Login Status", MessageBoxButtons.OK, MessageBoxIcon.Information);
        if (AllMethods.IsAdmin())
        {
            var frmAdminstatorHome = new AdministratorHome();
            frmAdminstatorHome.Show();
        }
        else
        {
            var frmBooking = new MaintainBookings();
            frmBooking.Show();
        }
        this.Close();
    }
    else
    {
```

The FillEmployees Method:



```
//AUTOFILLEMPLOYEEDETAILS
1 reference
public static void FillEmployees(int employeeID,TextBox txtName, TextBox txtSurname,TextBox txtIDNumber, TextBox txtPhoneNumber, TextBox txtEmail, TextBox txtUsername, TextBox txtPassword, RadioButton rdbReceptionist, RadioButton rdbAdministrator)
{
    try
    {
        if (con.State == ConnectionState.Closed)
        {
            con.Open();
        }
        string sql = $"SELECT * FROM Employee WHERE CAST(Employee_ID AS VARCHAR) LIKE '%{employeeID.ToString()}%'";
        command = new SqlCommand(sql, con);
        dataReader = command.ExecuteReader();

        while (dataReader.Read())
        {
            txtSurname.Text = dataReader.GetString(1);
            txtName.Text = dataReader.GetString(2);
            txtIDNumber.Text = dataReader.GetString(3);
            txtPhoneNumber.Text = dataReader.GetString(4);
            txtEmail.Text = dataReader.GetString(5);
            if (dataReader.GetString(6) == "Receptionist")
            {
                rdbReceptionist.Checked = true;
            }
            if (dataReader.GetString(6) == "Administrator")
            {
                rdbAdministrator.Checked = true;
            }
            txtUsername.Text = dataReader.GetString(7);
            txtPassword.Text = dataReader.GetString(8);
        }

        con.Close();
    }
    catch (Exception ex)
    {
        MessageBox.Show("Please contact your software solutions provider.\n" + ex.Message, "Developer Error", MessageBoxButtons.OK, MessageBoxIcon.Error);
    }
}
```

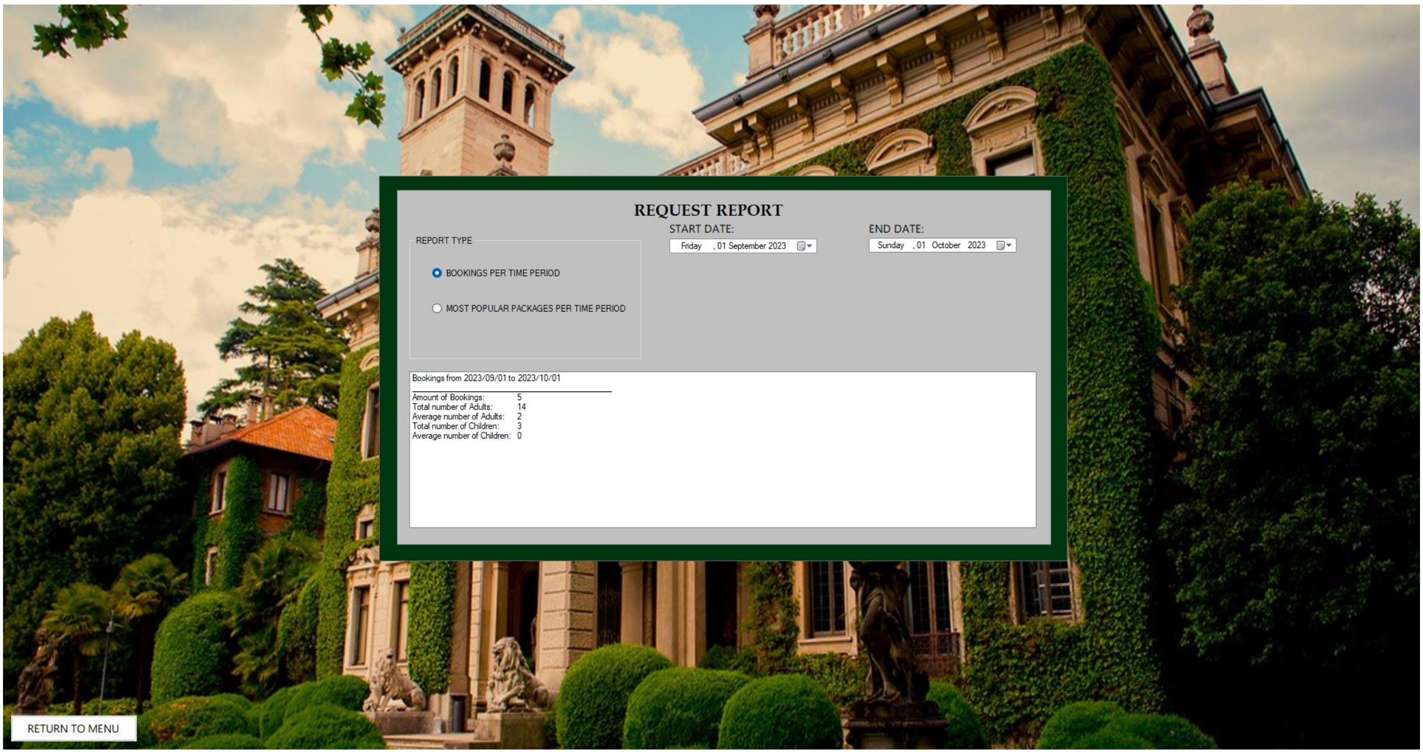
The using of the FillEmployee Method:



```
private void btnSelectID_Click(object sender, EventArgs e)
{
    AllMethods.CrudOperations('R', $"SELECT * FROM Employee WHERE CAST(Employee_ID AS VARCHAR) LIKE '%{txtUpdatesearchEmployeeID.Text}%'", dgvUpdateEmployee);
    AllMethods.FillEmployees(Convert.ToInt32(txtUpdatesearchEmployeeID.Text), txtUpdateEName, txtUpdateESName, txtUpdateEIDNumber, txtUpdateENumber,
        txtUpdateEEmail, txtUpdateEUsername, txtUpdateEPassword, rdbUpdateReceptionist, rdbUpdateAdmin);
}
```

THE 2 REPORTS GENERATED

Booking Per Time Period:



REQUEST REPORT

REPORT TYPE

● BOOKINGS PER TIME PERIOD

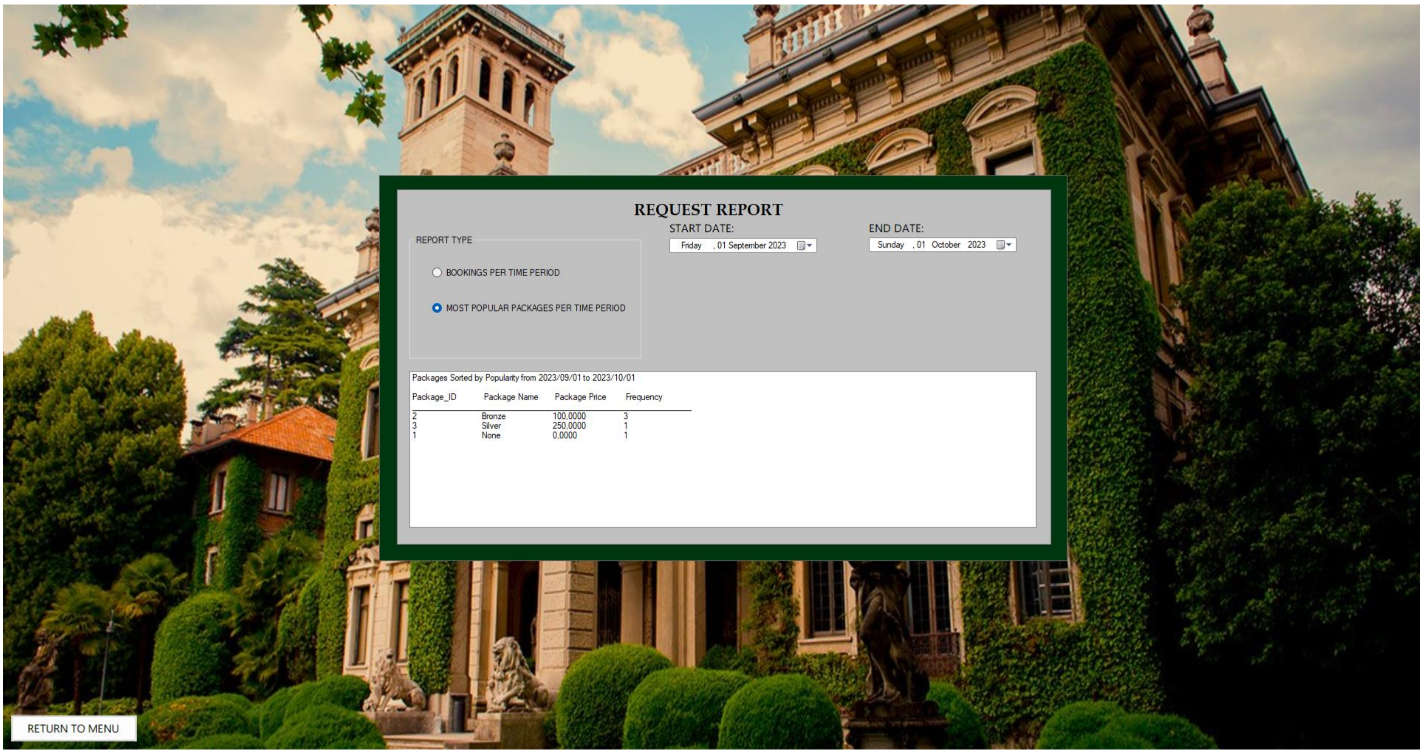
○ MOST POPULAR PACKAGES PER TIME PERIOD

START DATE: Friday , 01 September 2023

END DATE: Sunday , 01 October 2023

```
Bookings from 2023/09/01 to 2023/10/01
________________________________
Amount of Bookings:          5
Total number of Adults:      14
Average number of Adults:    2
Total number of Children:    3
Average number of Children:  0
```

Most Popular Packages Per Time Period:



REQUEST REPORT

REPORT TYPE

○ BOOKINGS PER TIME PERIOD

● MOST POPULAR PACKAGES PER TIME PERIOD

START DATE: Friday , 01 September 2023

END DATE: Sunday , 01 October 2023

```
Packages Sorted by Popularity from 2023/09/01 to 2023/10/01

Package_ID      Package Name    Package Price   Frequency
__________________________________________________________
2               Bronze          100,0000        3
3               Silver          250,0000        1
1               None            0,0000          1
```