

Mohammad Rizky Ramdhani

222410103042

Praktik PBO CRUD

Link Github : https://github.com/mohammadrizkyramdhani/projek_pbo

1. Class CRUD

Pada kode program tersebut berfungsi untuk mengoknesikan database yang telah dibuat di PgAdmin dengan main program formnya di c#. CRUD dijadikan class khusus karena agar mudah untuk di panggil nantinya pada main programnya. Disini harus menginstall NuGet Npgsql agar dapat terhubung dengan database PgAdmin.

```
1  using System;
2  using System.Data;
3  using System.Windows.Forms;
4  using Npgsql;
5
6  namespace projek_kyyme
7  {
8      23 references
9      internal class CRUD
10     {
11         1 reference
12         private static string getConnectionString()
13         {
14             string host = "Host=localhost;";
15             string port = "Port=5432;";
16             string db = "Database=crud;";
17             string user = "Username= postgres;";
18             string pass = "Password=mafia0890;";
19
20             string conString = string.Format("{0}{1}{2}{3}{4}", host, port, db, user, pass);
21
22             return conString;
23         }
24
25         public static NpgsqlConnection con = new NpgsqlConnection(getConnectionString());
26         public static NpgsqlCommand cmd = default(NpgsqlCommand);
27         public static string sql = string.Empty;
28
29         2 references
30         public static DataTable PerformCRUD(NpgsqlCommand com)
31         {
32             NpgsqlDataAdapter da = default(NpgsqlDataAdapter);
33             DataTable dt = new DataTable();
34         }
35     }
36 }
```

```
31
32     try
33     {
34         da = new NpgsqlDataAdapter();
35         da.SelectCommand = com;
36         con.Open();
37         da.Fill(dt);
38         con.Close();
39
40         return dt;
41     }
42     catch (Exception ex)
43     {
44         MessageBox.Show("An error occured: " + ex.Message, "Perform CRUD Operation Failed : Rizky Ramdhani",
45             MessageBoxButtons.OK, MessageBoxIcon.Error);
46         dt = null;
47     }
48
49     return dt;
50 }
51
52 }
53
54 }
55
56 }
```

2. Insert

Pada methodode ini berfungsi untuk menjalankan tombol button pada design interface yang telah dibuat. Methodode ini berada di main program dan sudah terkoneksi dengan database, sehingga ketika kita memasukkan sebuah data , maka data tersebut akan tampil di database juga.

```
0 references
private void insertButton_Click(object sender, EventArgs e)
{
    if (string.IsNullOrEmpty(namaDepanTextBox.Text.Trim()) || string.IsNullOrEmpty(namaBelakangTextBox.Text.Trim()))
    {
        MessageBox.Show("Tolong masukkan data nama depan dan nama belakang.", "Insert Data : Rizky Ramdhani",
            MessageBoxButtons.OK, MessageBoxIcon.Exclamation);
        return;
    }

    CRUD.sql = "INSERT INTO dataplayer(namadepan, namabelakang, jeniskelamin) VALUES (@namaDepan, @namaBelakang, @jenisKelamin)";
    execute(CRUD.sql, "Insert");

    MessageBox.Show("Jejak Anda telah disimpan.", "Insert Data : Rizky Ramdhani",
        MessageBoxButtons.OK, MessageBoxIcon.Information);

    loadData("");
    resetMe();
}
```

3. Update

Pada methodode update berfungsi untuk memperbarui data yang sudah di masukkan ke tabel dan menggantinya sesuai apa yang kita inginkan. Fungsi update akan berjalan jika terdapat minimal satu data yang di masukkan.

```
1 reference
private void updateButton_Click(object sender, EventArgs e)
{
    if (dataGridView1.Rows.Count == 0)
    {
        return;
    }

    if (string.IsNullOrEmpty(this.id))
    {
        MessageBox.Show("Tolong pilih item dari list.", "Update Data : Rizky Ramdhani",
            MessageBoxButtons.OK, MessageBoxIcon.Exclamation);
        return;
    }

    if (string.IsNullOrEmpty(namaDepanTextBox.Text.Trim()) || string.IsNullOrEmpty(namaBelakangTextBox.Text.Trim()))
    {
        MessageBox.Show("Tolong masukkan data nama depan dan nama belakang.", "Insert Data : Rizky Ramdhani",
            MessageBoxButtons.OK, MessageBoxIcon.Exclamation);
        return;
    }

    CRUD.sql = "UPDATE dataplayer SET namadepan = @namaDepan, namabelakang = @namaBelakang, jeniskelamin = @jenisKelamin WHERE idplay";
    execute(CRUD.sql, "Update");

    MessageBox.Show("Jejak Anda telah di ubah.", "Update Data : Rizky Ramdhani",
        MessageBoxButtons.OK, MessageBoxIcon.Information);

    loadData("");
    resetMe();
}
```

4. Delete

Berbeda dengan fungsi Update dan Insert, Fungsi Delete yaitu untuk menghapus data yang tidak diperlukan dari tabel.

```
1 reference
private void deleteButton_Click(object sender, EventArgs e)
{
    if (dataGridView1.Rows.Count == 0)
    {
        return;
    }

    if (string.IsNullOrEmpty(this.id))
    {
        MessageBox.Show("Tolong pilih item dari list.", "Delete Data : Rizky Ramdhani",
            MessageBoxButtons.OK, MessageBoxIcon.Exclamation);
        return;
    }

    if (MessageBox.Show("Apakah Anda yakin ingin menghapusnya?", "Delete Data : Rizky Ramdhani",
        MessageBoxButtons.YesNo, MessageBoxIcon.Question, MessageBoxDefaultButton.Button1) == DialogResult.Yes)

    CRUD.sql = "DELETE FROM dataplayer WHERE idplayer = @id::integer";

    execute(CRUD.sql, "Update");

    MessageBox.Show("Jejak Anda telah dihapus.", "Delete Data : Rizky Ramdhani",
        MessageBoxButtons.OK, MessageBoxIcon.Information);

    loadData("");
    resetMe();
}
```

5. Search

Method search ini sebagai fitur tambahan yang dapat memudahkan pengguna untuk mencari data yang berada pada table sesuai keyword yang dimasukkan.

```
1 reference
private void searchButton_Click(object sender, EventArgs e)
{
    if (!string.IsNullOrEmpty(keywordTextBox.Text.Trim()))
    {
        loadData("");
    }
    else
    {
        loadData(keywordTextBox.Text.Trim());
    }

    resetMe();
}
```

Output

Kode program yang sudah kita buat akan ditampilkan dan menjadi sebuah jendela aplikasi yang sudah siap dioperasikan.

The screenshot shows a Windows application window titled "Form1". Inside the window, there is a section titled "Data Pemain Honkai" in a blue box. Below this title, there is a search bar and a "Cari" button. Further down, there are three input fields: "Nama Depan", "Nama Belakang", and "Jenis Kelamin" (which is a dropdown menu currently showing "Laki-Laki"). Below these fields are three buttons: "Insert", "Update ()", and "Delete ()". At the bottom of the form is a table with the following headers: "ID", "Nama Depan", "Nama Belakang", and "Nama Lengkap". The table body is currently empty and has a grey background. At the very bottom of the window, there is a status bar that says "Jumlah kolom(s): 0".

ID	Nama Depan	Nama Belakang	Nama Lengkap
----	------------	---------------	--------------

Jika kita lihat pada Interface tersebut , secara sederhana kita dapat memasukkan inputan berupa “Nama Depan”, “Nama Belakang”, dan memilih Jenis Kelamin (laki-laki atau perempuan). Setelah menginputkan data tersebut, akan muncul kolom baru yang bernama “Nama Lengkap” untuk menggabungkan keduanya.