

# **RESERVASI KAMAR HOTEL**

## **LAPORAN PEMROGRAMAN II (PBO)**

### **Dosen Pengampu:**

Indra Riksa Herlambang, S.Tr.Kom., M.Kom.



Disusun Oleh:

Bargana Kukuh Raditya      714220013

Alfian Benardo Rusli      714220048

**PROGRAM STUDI DIV TEKNIK INFORMATIKA  
UNIVERSITAS LOGISTIK & BISNIS INTERNASIONAL  
BANDUNG**

**2023**

## PEMBAHASAN

### ➤ AAS (714220013 & 714220048)

- Source code dan penjelasan

#### SOURCE CODE

```
namespace TB_714220013_714220048.model
{
    internal class M_tipe
    {
        string id_tipekamar, tipe_kamar;

        public M_tipe()
        {
        }

        public M_tipe(string id_tipekamar, string tipe_kamar)
        {
            this.ID = id_tipekamar;
            this.Tipe = tipe_kamar;
        }

        public string ID { get => id_tipekamar; set => id_tipekamar = value; }
        public string Tipe { get => tipe_kamar; set => tipe_kamar = value; }
    }
}
```

#### Penjelasan :

Source code diatas digunakan untuk manajemen dan menyimpan data pada aplikasi yang mencakup data dari database dengan atribut id\_tipekamar dan tipe\_kamar. Properti yang sudah dibuat dapat diakses dari luar kelas yang nilainya sudah diatur melalui getter dan setter.

#### SOURCE CODE

```
namespace TB_714220013_714220048.controller
{
    internal class TipeKamar
    {
        Koneksi koneksi = new Koneksi();

        // Method Insert
        public bool Insert(M_tipe tipe)
        {
            Boolean status = false;
            try
            {
                koneksi.OpenConnection();
                koneksi.ExecuteNonQuery("INSERT INTO tipe_kamar (id_tipekamar, tipe_kamar) VALUES('" + tipe.ID + "','" + tipe.Tipe + "')");
                status = true;
                MessageBox.Show("Data berhasil ditambahkan", "Informasi",
                    MessageBoxButtons.OK, MessageBoxIcon.Information);
            }
            catch { }
        }
    }
}
```

```

        koneksi.CloseConnection();
    }
    catch (Exception e)
    {
        MessageBox.Show(e.Message, "Gagal Insert", MessageBoxButtons.OK,
        MessageBoxIcon.Error);
    }
    return status;
}

// Method Update
public bool Update(M_tipe tipe, string id)
{
    Boolean status = false;
    try
    {
        koneksi.OpenConnection();
        koneksi.ExecuteNonQuery("UPDATE tipe_kamar SET tipe_kamar='" +
tipe.Tipe + "' WHERE id_tipekamar='" + id + "'");
        status = true;
        MessageBox.Show("Data Berhasil Diubah", "Informasi",
        MessageBoxButtons.OK, MessageBoxIcon.Information);
        koneksi.CloseConnection();
    }
    catch (Exception e)
    {
        MessageBox.Show(e.Message, "Gagal Update", MessageBoxButtons.OK,
        MessageBoxIcon.Error);
    }
    return status;
}

// Method Delete
public bool Delete(string id)
{
    Boolean status = false;
    try
    {
        koneksi.OpenConnection();
        koneksi.ExecuteNonQuery("DELETE FROM tipe_kamar WHERE
id_tipekamar='" + id + "'");
        status = true;
        MessageBox.Show("Data Berhasil Dihapus", "Informasi",
        MessageBoxButtons.OK, MessageBoxIcon.Information);
        koneksi.CloseConnection();
    }
    catch (Exception e)
    {
        MessageBox.Show(e.Message, "Gagal Hapus", MessageBoxButtons.OK,
        MessageBoxIcon.Error);
    }
    return status;
}
}
}

```

**Penjelasan :**

Pada Source code controller ini, terdapat 3 method yaitu sebagai berikut.

- Method Insert

Digunakan untuk menambahkan data tipe kamar ke dalam tabel database tipe\_kamar. Metode ini membuka koneksi ke database, menjalankan query SQL untuk insert data, serta memberikan pesan ke pengguna jika data diinput dengan benar, maka akan menampilkan MessageBox

	<p>informasi dengan pesan “Data Berhasil Ditambahkan”.</p> <ul style="list-style-type: none"> <li>• <b>Method Update</b> Digunakan untuk memperbarui data tipe kamar yang ada pada tabel database tipe_kamar. Metode ini membuka koneksi ke database, menjalankan query SQL untuk update data berdasarkan id_tipekamar, serta memberikan pesan ke pengguna jika data diinput dengan benar, maka akan menampilkan MessageBox informasi dengan pesan “Data Berhasil Diubah”.</li> <li>• <b>Method Delete</b> Digunakan untuk menghapus data tipe kamar yang ada pada tabel database tipe_kamar. Metode ini membuka koneksi ke database, menjalankan query SQL untuk hapus data berdasarkan id_tipekamar, serta memberikan pesan ke pengguna jika data dihapus, maka akan menampilkan MessageBox informasi dengan pesan “Data Berhasil Dihapus”.</li> </ul>
--	--

#### SOURCE CODE

```
namespace TB_714220013_714220048.view
{
    public partial class FormTipeKamar : Form
    {
        Koneksi koneksi = new Koneksi();
        M_tipe m_tipe = new M_tipe();
        string id_tipekamar;

        public void Tampil()
        {
            string query = "SELECT * FROM tipe_kamar";

            // Query
            DataTipeKamar.DataSource = koneksi.ShowData(query);

            // Mengubah nama kolom tabel
            DataTipeKamar.Columns[0].HeaderText = "ID Tipe Kamar";
            DataTipeKamar.Columns[1].HeaderText = "Tipe Kamar";
        }

        public void resetForm()
        {
            cbTipe.SelectedIndex = -1;
            tbCari.Text = "";
        }

        public FormTipeKamar()
        {
            InitializeComponent();
        }

        private void FormTipeKamar_Load(object sender, EventArgs e)
        {
            Tampil();
        }

        private void btnSimpan_Click(object sender, EventArgs e)
        {
            if (cbTipe.SelectedIndex == -1)

```

```

        {
            MessageBox.Show("Harap pilih tipe kamar dengan benar!",
"Peringatan",
                MessageBoxButtons.OK, MessageBoxIcon.Error);
        }
        else
        {
            TipeKamar tipekamar = new TipeKamar();
            m_tipe.Tipe = cbTipe.Text;

            tipekamar.Insert(m_tipe);

            resetForm();
            Tampil();
        }
    }

    private void btnUbah_Click(object sender, EventArgs e)
    {
        if (cbTipe.SelectedIndex == -1)
        {
            MessageBox.Show("Ubah data dengan benar!", "Peringatan!",
                MessageBoxButtons.OK, MessageBoxIcon.Error);
        }
        else
        {
            TipeKamar tipekamar = new TipeKamar();
            m_tipe.Tipe = cbTipe.Text;

            tipekamar.Update(m_tipe, id_tipekamar);

            resetForm();
            Tampil();
        }
    }

    private void btnHapus_Click(object sender, EventArgs e)
    {
        DialogResult pesan = MessageBox.Show(
            "Apakah yakin akan menghapus data ini?",
            "Perhatian",
            MessageBoxButtons.YesNo,
            MessageBoxIcon.Question);

        if (pesan == DialogResult.Yes)
        {
            TipeKamar tipekamar = new TipeKamar();
            tipekamar.Delete(id_tipekamar);
            resetForm();
            Tampil();
        }
    }

    private void btnRefresh_Click(object sender, EventArgs e)
    {
        resetForm();
        Tampil();
    }

    private void DataTipeKamar_CellClick(object sender,
DataGridViewCellEventArgs e)
    {

```

```

        id_tipekamar =
DataTipeKamar.Rows[e.RowIndex].Cells[0].Value.ToString();
        cbTipe.Text =
DataTipeKamar.Rows[e.RowIndex].Cells[1].Value.ToString();
    }

    private void tbCari_TextChanged(object sender, EventArgs e)
    {
        // Query Search Data
        DataTipeKamar.DataSource = koneksi.ShowData("SELECT * FROM
tipe_kamar WHERE id_tipekamar LIKE '%" + tbCari.Text + "%' OR tipe_kamar LIKE
 '%" + tbCari.Text + "%'");
    }
}

```

**Penjelasan :**

- Koneksi digunakan untuk mengelola koneksi ke database.
- M\_tipe digunakan untuk merepresentasikan data atau model tipe kamar yang telah dibuat.
- Method Tampil digunakan untuk menampilkan data pada tabel database tipe\_kamar ke dalam DataGridView dengan nama DataTipeKamar. Kemudian nama kolom tabel dapat diatur menggunakan HeaderText.
- Method resetForm digunakan untuk me reset form yang sudah diisi.
- FormTipeKamar Load digunakan untuk menampilkan data yang sudah dibuat pada method Tampil.
- btnSimpan\_Click digunakan ketika tombol Simpan ditekan. Jika pengguna telah memilih tipe kamar dari ComboBox (cbTipe), data tipe kamar akan disimpan ke dalam database menggunakan method Insert pada objek TipeKamar. Sebaliknya, jika pengguna tidak memilih tipe kamar, akan muncul pesan peringatan. Setelah operasi simpan, form akan direset dan data tipe kamar diperbarui pada tampilan.
- btnUbah\_Click digunakan ketika tombol Ubah ditekan. Jika pengguna telah memilih tipe kamar dari ComboBox (cbTipe), data tipe kamar akan diubah dalam database menggunakan metode Update pada objek TipeKamar. Sebaliknya, jika pengguna tidak memilih tipe kamar, akan muncul pesan peringatan. Setelah operasi ubah, form akan direset dan data tipe kamar diperbarui pada tampilan.
- btnHapus\_Click digunakan ketika tombol Hapus ditekan. Akan menampilkan dialog konfirmasi untuk memastikan apakah pengguna benar-benar ingin menghapus data. Jika pengguna memilih "Yes", data tipe kamar dihapus dari database menggunakan metode Delete pada objek TipeKamar. Setelah itu, form direset dan data tipe kamar diperbarui pada tampilan.
- btnRefresh\_Click digunakan ketika tombol Refresh ditekan. Ketika tombol tersebut ditekan, form direset ke kondisi awal menggunakan metode resetForm, dan kemudian data tipe kamar ditampilkan kembali pada form dengan memanggil metode Tampil. Tujuannya yaitu untuk memperbarui tampilan form dengan data terbaru dari database setelah melakukan operasi tambah, ubah, atau hapus.
- DataTipeKamar\_CellClick dijalankan ketika baris di dalam DataGridView di klik maka akan menampilkan data kembali pada form pengisian. CellClick memungkinkan pengguna untuk mengidentifikasi dan mengubah data tipe kamar yang terkait dengan sel yang diklik.

	<ul style="list-style-type: none"> <li>• tbCari_TextChanged digunakan ketika mengetikkan sesuatu pada textbox cari maka data yang terkait akan tampil pada DataGridView.</li> </ul>
--	---

#### SOURCE CODE

```
namespace TB_714220013_714220048.model
{
    internal class M_kamar
    {
        string id_tipekamar, kamar, harga, status;

        public M_kamar()
        {
        }

        public M_kamar(string kamar, string harga)
        {
            this.ID = id_tipekamar;
            this.Kamar = kamar;
            this.Harga = harga;
            this.Status = status;
        }

        public string ID { get => id_tipekamar; set => id_tipekamar = value; }
        public string Kamar { get => kamar; set => kamar = value; }
        public string Harga { get => harga; set => harga = value; }
        public string Status { get => status; set => status = value; }
    }
}
```

#### Penjelasan :

- internal kata kunci untuk menunjukkan bahwa kelas ini hanya dapat diakses oleh kode dalam assembly yang sama
- string id\_tipekamar, kamar, harga, status merupakan property/atribut yang akan digunakan untuk mempresntasikan data terkait kamar
- konstruktor dan property ini digunakan untuk menginisialisasi dan mengakses nilai-nilai terkait kamar dalam aplikasi

#### SOURCE CODE

```
namespace TB_714220013_714220048.controller
{
    internal class Kamar
    {
        Koneksi koneksi = new Koneksi();

        // Method Insert
        public bool Insert(M_kamar kamar)
        {
            Boolean status = false;
            try
            {
                koneksi.OpenConnection();
                koneksi.ExecuteNonQuery("INSERT INTO kamar(id_tipekamar, kamar,
harga, status) VALUES(' + kamar.ID + "', ' + kamar.Kamar + "', ' +
kamar.Harga + "', ' + kamar.Status + "')");
            }
            catch { }
        }
    }
}
```

```

        status = true;
        MessageBox.Show("Data berhasil ditambahkan", "Informasi",
            MessageBoxButtons.OK, MessageBoxIcon.Information);
        koneksi.CloseConnection();
    }
    catch (Exception e)
    {
        MessageBox.Show(e.Message, "Gagal Insert", MessageBoxButtons.OK,
            MessageBoxIcon.Error);
    }
    finally
    {
        koneksi.CloseConnection();
    }
    return status;
}

// Method Update
public bool Update(M_kamar kamar, string id)
{
    Boolean status = false;
    try
    {
        koneksi.OpenConnection();
        koneksi.ExecuteNonQuery("UPDATE kamar SET kamar='" + kamar.Kamar +
            "','" + "harga='" + kamar.Harga + "' WHERE kamar='" + id + "'");
        status = true;
        MessageBox.Show("Data Berhasil Diubah", "Informasi",
            MessageBoxButtons.OK, MessageBoxIcon.Information);
        koneksi.CloseConnection();
    }
    catch (Exception e)
    {
        MessageBox.Show(e.Message, "Gagal Update", MessageBoxButtons.OK,
            MessageBoxIcon.Error);
    }
    return status;
}

// Method Delete
public bool Delete(string id)
{
    Boolean status = false;
    try
    {
        koneksi.OpenConnection();
        koneksi.ExecuteNonQuery("DELETE FROM kamar WHERE id_kamar='" + id +
            "'");
        status = true;
        MessageBox.Show("Data Berhasil Dihapus", "Informasi",
            MessageBoxButtons.OK, MessageBoxIcon.Information);
        koneksi.CloseConnection();
    }
    catch (Exception e)
    {
        MessageBox.Show(e.Message, "Gagal Hapus", MessageBoxButtons.OK,
            MessageBoxIcon.Error);
    }
    return status;
}
}

```



<b>Penjelasan :</b>	<ul style="list-style-type: none"> <li>• public bool insert merupakan berisi informasi mengenai kamar yang akan ditambahkan ke dalam database.</li> <li>• Boolean status variable berikut digunakan untuk menentukan apakah oprasi insert berhasil atau tidak</li> <li>• Try digunakan untuk mengeksekusi oprasi yang berkemungkinan menimbulkan masalah</li> <li>• OpenConnection untuk membuka koneksi ke database</li> <li>• ExecuteQuery untuk menjalankan query SQL untuk memasukan data baru ke dalam tabel</li> <li>• Massagaebox.show menampilkan pesan informasi</li> <li>• Closeconnection menutup koneksi ke database</li> <li>• Finnaly selalu dijalankan, baik terjadi kesalahan atau tidak</li> </ul>
---------------------	---

#### SOURCE CODE

```

namespace TB_714220013_714220048.view
{
    public partial class FormDataKamar : Form
    {
        Koneksi koneksi = new Koneksi();
        M_kamar m_kamar = new M_kamar();
        string id_kamar;

        public void Tampil()
        {
            string query = "SELECT k.id_kamar, k.id_tipekamar, t.tipe_kamar,
k.kamar, k.harga FROM kamar k JOIN tipe_kamar t ON (k.id_tipekamar =
t.id_tipekamar)";
            DataKamar.DataSource = koneksi.ShowData(query);

            DataKamar.Columns[0].HeaderText = "ID";
            DataKamar.Columns[1].HeaderText = "ID Tipe Kamar";
            DataKamar.Columns[2].HeaderText = "Tipe Kamar";
            DataKamar.Columns[3].HeaderText = "Kamar";
            DataKamar.Columns[4].HeaderText = "Harga";
        }

        //ambil data tipe kamar dari table (tipe_kamar) untuk mengisi data pada
        combobox ID_tipeKamar
        public void GetDataKamar()
        {
            koneksi.OpenConnection();
            MySqlDataReader reader = koneksi.reader("SELECT * FROM tipe_kamar");
            while (reader.Read())
            {
                cbID.Items.Add(reader.GetString("id_tipekamar"));
            }
            reader.Close();
            koneksi.CloseConnection();
        }

        public void GetTipeKamar()
        {
            koneksi.OpenConnection();

```

```

        MySqlDataReader reader = koneksi.reader("SELECT tipe_kamar FROM
tipe_kamar WHERE id_tipekamar = '" + cbID.Text + "'");
        if (reader.HasRows)
        {
            while (reader.Read())
            {
                tbTipe.Text = reader.GetString(0);
            }
        }
        reader.Close();
        koneksi.CloseConnection();
    }

    public FormDataKamar()
    {
        InitializeComponent();
        tbTipe.ReadOnly = true;
        tbTipe.Enabled = false;
    }

    private void textBox2_TextChanged(object sender, EventArgs e)
    {
    }

    private void DataKamar_CellContentClick(object sender,
DataGridViewCellEventArgs e)
    {
    }

    private void DataKamar_CellClick(object sender,
DataGridViewCellEventArgs e)
    {
        id_kamar = DataKamar.Rows[e.RowIndex].Cells[0].Value.ToString();
        cbID.Text = DataKamar.Rows[e.RowIndex].Cells[1].Value.ToString();
        tbTipe.Text = DataKamar.Rows[e.RowIndex].Cells[2].Value.ToString();
        tbKamar.Text = DataKamar.Rows[e.RowIndex].Cells[3].Value.ToString();
        tbHarga.Text = DataKamar.Rows[e.RowIndex].Cells[4].Value.ToString();
    }

    private void tbCari_TextChanged(object sender, EventArgs e)
    {
        DataKamar.DataSource = koneksi.ShowData("SELECT k.id_kamar,
k.id_tipekamar, t.tipe_kamar, k.kamar, k.harga " +
"FROM kamar k JOIN tipe_kamar t ON k.id_tipekamar = t.id_tipekamar "
+
"WHERE k.kamar LIKE '%" + tbCari.Text + "%' " +
"OR t.tipe_kamar LIKE '%" + tbCari.Text + "%'");
    }

    private void cbID_SelectedIndexChanged(object sender, EventArgs e)
    {
        GetTipeKamar();
    }

    public void resetForm()
    {
        cbID.SelectedIndex = -1;
        tbTipe.Text = "";
        tbKamar.Text = "";
        tbHarga.Text = "";
        tbCari.Text = "";
    }

```

```

    }

    private void btnSimpan_Click(object sender, EventArgs e)
    {
        if (cbID.SelectedIndex == -1 || tbKamar.Text == "" || tbHarga.Text
== "")
        {
            MessageBox.Show("Data tidak boleh kosong", "Peringatan",
            MessageBoxButtons.OK, MessageBoxIcon.Warning);
        }
        else
        {
            Kamar kamar = new Kamar();
            m_kamar.ID = cbID.Text;
            m_kamar.Kamar = tbKamar.Text;
            m_kamar.Harga = tbHarga.Text;

            kamar.Insert(m_kamar);

            resetForm();
            Tampil();
        }
    }

    private void btnUbah_Click(object sender, EventArgs e)
    {
        if (cbID.SelectedIndex == -1 || tbKamar.Text == "" || tbHarga.Text
== "")
        {
            MessageBox.Show("Data tidak boleh kosong", "Peringatan",
            MessageBoxButtons.OK, MessageBoxIcon.Warning);
        }
        else
        {
            Kamar kamar = new Kamar();
            m_kamar.Kamar = tbKamar.Text;
            m_kamar.Harga = tbHarga.Text;

            kamar.Update(m_kamar, id_kamar);
            resetForm();
            Tampil();
        }
    }

    private void btnHapus_Click(object sender, EventArgs e)
    {
        DialogResult pesan = MessageBox.Show("Apakah yakin akan menghapus
data ini?",
        "Perhatian", MessageBoxButtons.YesNo, MessageBoxIcon.Question);
        if (pesan == DialogResult.Yes)
        {
            Kamar kamar = new Kamar();
            kamar.Delete(id_kamar);
            resetForm();
            Tampil();
        }
    }

    private void btnRefresh_Click(object sender, EventArgs e)
    {
        resetForm();
        Tampil();
    }

```

```

private void FormDataKamar_Load(object sender, EventArgs e)
{
    Tampil();
    GetDataKamar();
}

private void FormatDigit(TextBox textBox)
{
    if (!string.IsNullOrEmpty(textBox.Text) &&
textBox.Text.All(char.IsDigit))
    {
        string number = textBox.Text.Replace(".", "");

        number = string.Format("{0:#,###0}", double.Parse(number));

        textBox.Text = number;
        textBox.SelectionStart = textBox.Text.Length;
    }
}

private string FormatDigitAngka(int number)
{
    return string.Format("Rp {0:N0}", number);
}

private void DataKamar_CellFormatting_1(object sender,
DataGridViewCellFormattingEventArgs e)
{
    if (e.ColumnIndex == 4 && e.Value != null)
    {
        if (int.TryParse(e.Value.ToString(), out int number))
        {
            e.Value = FormatDigitAngka(number);
            e.FormattingApplied = true;
        }
    }
}
}

```

**Penjelasan :**

- Mode Tampil untuk menampilkan data kamar dari database ke DataGridView, mengatur nama kolom pada DataGridView
- GetDataKamar Mengambil data tipe kamar dari tabel tipe\_kamar untuk mengisi ComboBox pada form.
- GetTipeKamar mendapatkan dan menampilkan nama tipe kamar berdasarkan ID yang dipilih
- tbCari mencari data kamar berdasarkan nama atau tipe kamar saat teks pada textbox
- btnSimpan untuk menyimpan
- btnUbah untuk mengubah
- btHapus untuk menghapus
- btnRefresh untuk mengosongkan form menampilkan kembali data kamar
- from data kamar menampilkan data kamar
- format digit mengatur format digit pada textbox
- format digit angka format digit untuk angka menjadi string

	<ul style="list-style-type: none"> <li>cell formatting untuk mengatur format angka ke format mata uang rupiah pada kolom harga</li> </ul>
--	---

#### SOURCE CODE

```
namespace TB_714220013_714220048.model
{
    class M_reservasi
    {
        string id_customer, nama, email, nohp, tipe_kamar, checkin, checkout,
total;

        public M_reservasi()
        {
        }

        public M_reservasi(string id_customer, string nama, string email, string
nohp, string tipe_kamar, string checkin, string checkout, string total)
        {
            this.ID = id_customer;
            this>Nama = nama;
            this.Email = email;
            this.Nohp = nohp;
            this.Tipe = tipe_kamar;
            this.Checkin = checkin;
            this.Checkout = checkout;
            this.Total = total;
        }

        public string ID { get => id_customer; set => id_customer = value; }
        public string Nama { get => nama; set => nama = value; }
        public string Email { get => email; set => email = value; }
        public string Nohp { get => nohp; set => nohp = value; }
        public string Tipe { get => tipe_kamar; set => tipe_kamar = value; }
        public string Checkin { get => checkin; set => checkin = value; }
        public string Checkout { get => checkout; set => checkout = value; }
        public string Total { get => total; set => total = value; }
    }
}
```

<b>Penjelasan :</b>	<ul style="list-style-type: none"> <li>• M_reservasi berfungsi untuk representasi data untuk informasi Reservasi</li> </ul>
---------------------	---

#### SOURCE CODE

```

namespace TB_714220013_714220048.controller
{
    internal class TipeKamar
    {
        Koneksi koneksi = new Koneksi();

        // Method Insert
        public bool Insert(M_tipe tipe)
        {
            Boolean status = false;
            try
            {
                koneksi.OpenConnection();
                koneksi.ExecuteNonQuery("INSERT INTO tipe_kamar (id_tipekamar,
tipe_kamar) VALUES('" + tipe.ID + "','" + tipe.Tipe + "')");
                status = true;
                MessageBox.Show("Data berhasil ditambahkan", "Informasi",
                    MessageBoxButtons.OK, MessageBoxIcon.Information);
                koneksi.CloseConnection();
            }
            catch (Exception e)
            {
                MessageBox.Show(e.Message, "Gagal Insert", MessageBoxButtons.OK,
                MessageBoxIcon.Error);
            }
            return status;
        }

        // Method Update
        public bool Update(M_tipe tipe, string id)
        {

```

```

        Boolean status = false;
        try
        {
            koneksi.OpenConnection();
            koneksi.ExecuteNonQuery("UPDATE tipe_kamar SET tipe_kamar='" +
tipe.Tipe + "' WHERE id_tipekamar='" + id + "'");
            status = true;
            MessageBox.Show("Data Berhasil Diubah", "Informasi",
                MessageBoxButtons.OK, MessageBoxIcon.Information);
            koneksi.CloseConnection();
        }
        catch (Exception e)
        {
            MessageBox.Show(e.Message, "Gagal Update", MessageBoxButtons.OK,
MessageBoxIcon.Error);
        }
        return status;
    }

    // Method Delete
    public bool Delete(string id)
    {
        Boolean status = false;
        try
        {
            koneksi.OpenConnection();
            koneksi.ExecuteNonQuery("DELETE FROM tipe_kamar WHERE
id_tipekamar='" + id + "'");
            status = true;
            MessageBox.Show("Data Berhasil Dihapus", "Informasi",
                MessageBoxButtons.OK, MessageBoxIcon.Information);
            koneksi.CloseConnection();
        }
        catch (Exception e)
        {
            MessageBox.Show(e.Message, "Gagal Hapus", MessageBoxButtons.OK,
MessageBoxIcon.Error);
        }
        return status;
    }
}

```

**Penjelasan :**

- Insert(M\_tipe tipe) untuk menambahkan data tipe kamar ke database ,menerima objek m\_tipe sebagai parameter, mengambil status kenerhasilan operasi
- Update(M\_tipe tipe, string id) untuk memperbarui data tipe kamar berdasarkan ID, menerima objek m\_tipe dan ID sebagai parameter, mengembalikan status keberhasilan operasi
- Delete(string id) untuk menghapus data tipe kamar berdasarkan ID, menerima ID sebagai parameter, mengembalikan status keberhasilan operasi
- Kelas ini untuk melakukan CRUD

**SOURCE CODE**

```

namespace TB_714220013_714220048.view
{

```

```

public partial class FormTipeKamar : Form
{
    Koneksi koneksi = new Koneksi();
    M_tipe m_tipe = new M_tipe();
    string id_tipekamar;

    public void Tampil()
    {
        string query = "SELECT * FROM tipe_kamar";

        // Query
        DataTipeKamar.DataSource = koneksi.ShowData(query);

        // Mengubah nama kolom tabel
        DataTipeKamar.Columns[0].HeaderText = "ID Tipe Kamar";
        DataTipeKamar.Columns[1].HeaderText = "Tipe Kamar";
    }

    public void resetForm()
    {
        cbTipe.SelectedIndex = -1;
        tbCari.Text = "";
    }

    public FormTipeKamar()
    {
        InitializeComponent();
    }

    private void FormTipeKamar_Load(object sender, EventArgs e)
    {
        Tampil();
    }

    private void btnSimpan_Click(object sender, EventArgs e)
    {
        if (cbTipe.SelectedIndex == -1)
        {
            MessageBox.Show("Harap pilih tipe kamar dengan benar!",
"Peringatan",
                MessageBoxButtons.OK, MessageBoxIcon.Error);
        }
        else
        {
            TipeKamar tipekamar = new TipeKamar();
            m_tipe.Tipe = cbTipe.Text;

            tipekamar.Insert(m_tipe);

            resetForm();
            Tampil();
        }
    }

    private void btnUbah_Click(object sender, EventArgs e)
    {
        if (cbTipe.SelectedIndex == -1)
        {
            MessageBox.Show("Ubah data dengan benar!", "Peringatan!",
                MessageBoxButtons.OK, MessageBoxIcon.Error);
        }
        else
        {

```



```

        TipeKamar tipekamar = new TipeKamar();
        m_tipe.Tipe = cbTipe.Text;

        tipekamar.Update(m_tipe,id_tipekamar);

        resetForm();
        Tampil();
    }
}

private void btnHapus_Click(object sender, EventArgs e)
{
    DialogResult pesan = MessageBox.Show(
        "Apakah yakin akan menghapus data ini?",
        "Perhatian",
        MessageBoxButtons.YesNo,
        MessageBoxIcon.Question);

    if (pesan == DialogResult.Yes)
    {
        TipeKamar tipekamar = new TipeKamar();
        tipekamar.Delete(id_tipekamar);
        resetForm();
        Tampil();
    }
}

private void btnRefresh_Click(object sender, EventArgs e)
{
    resetForm();
    Tampil();
}

private void DataTipeKamar_CellClick(object sender,
DataGridViewCellEventArgs e)
{
    id_tipekamar =
DataTipeKamar.Rows[e.RowIndex].Cells[0].Value.ToString();
    cbTipe.Text =
DataTipeKamar.Rows[e.RowIndex].Cells[1].Value.ToString();
}

private void tbCari_TextChanged(object sender, EventArgs e)
{
    // Query Search Data
    DataTipeKamar.DataSource = koneksi.ShowData("SELECT * FROM
tipe_kamar WHERE id_tipekamar LIKE '%" + tbCari.Text + "%' OR tipe_kamar LIKE
 '%" + tbCari.Text + "%'");
}
}

```

**Penjelasan :**

- Tampil() untuk menampilkan data tipe kamar dari database ke DataGridView, dan mengubah nama kolom tabel ke DataGridView
- Reset untuk mengosongkan form ke kondisi awal
- FormTipeKamar\_Load untuk menjalankan metode Tampil saat form di-load
- Btn\_simpan untuk menyimpan data
- Btn\_Ubah untuk mengubah data

	<ul style="list-style-type: none"> <li>• Btn_Hapus untuk menghapus data</li> <li>• Btn_Refreshe untuk mengosongkan form dan menampilkan kembali data tipe kamar dari database</li> <li>• tb_cari untuk menangani perubahan teks pada TextBox pencari untuk melakukan pencarian data tipe kamar berdasarkan ID atau nama</li> </ul>
--	--

#### SOURCE CODE

```

namespace TB_714220013_714220048.controller
{
    internal class CekLogin
    {
        Koneksi koneksi = new Koneksi();

        public bool cek_Login(string username, string password)
        {
            try
            {
                koneksi.OpenConnection();
                MySqlDataReader reader = koneksi.reader("SELECT * FROM t_user
WHERE username='" +
                username + "' AND password='" + password + "'");
                if (reader.Read())
                {
                    koneksi.CloseConnection();
                    return true;
                }
            }
            else
            {
                koneksi.CloseConnection();
                return false;
            }
        }
    }
}

```

```

    }
    }
    catch (Exception e)
    {
        MessageBox.Show(e.Message, "Gagal Login", MessageBoxButtons.OK,
        MessageBoxIcon.Error);
    }
    return false;
}
}
}

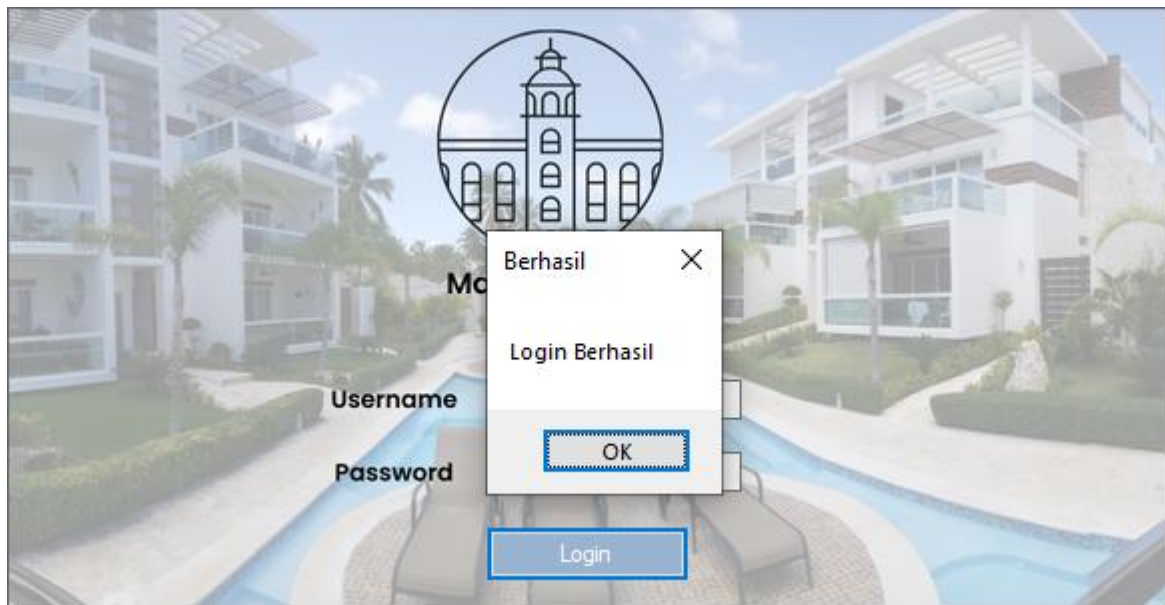
```

**Penjelasan :**

- CekLogin untuk memverifikasi login, dan memanggil metode cek\_login dan menyediakan username dan password

• **Output Program**





File Data Master Data Reservasi



MAJESTIC PEARL HOTEL

File
Data Master
Data Reservasi

Table Data Kamar

ID	ID Tipe Kamar	Tipe Kamar	Kamar	Harga
17	4	Family Room (6 Orang)	B200	Rp 400.000
19	5	VIP Room (2 Orang)	A001	Rp 400.000
20	1	Single Room (1 hingga 2 ...	A001	Rp 200.000
21	8	Triple Room (1 hingga 6 O...	B012	Rp 500.000
22	8	Triple Room (1 hingga 6 O...	C005	Rp 400.000
23	1	Single Room (1 hingga 2 ...	E002	Rp 300.000
24	5	VIP Room (2 Orang)	S007	Rp 600.000
25	4	Family Room (6 Orang)	F066	Rp 500.000

Form Input Data Kamar

ID Tipe Kamar

Tipe Kamar

Kamar

Harga
Rp.

Form Pencarian

Cari Data

Tombol Action

Simpan

Ubah

Hapus

File
Data Master
Data Reservasi

Table Data Tipe Kamar

ID Tipe Kamar	Tipe Kamar
1	Single Room (1 hingga 2 Orang)
2	Twin Room (1 hingga 4 Orang)
4	Family Room (6 Orang)
5	VIP Room (2 Orang)
8	Triple Room (1 hingga 6 Orang)

Form Input Tipe Kamar

Tipe Kamar

Form Pencarian

Cari Data

Tombol Action

Simpan

Ubah

Hapus

File
Data Master
Data Reservasi

Table Data Reservasi

	ID Customer	Nama	Email	No HP	Tipe Kamar	Kamar	Harga	Check-in	Check-out	Total
▶	1	Bargana K...	bargana@g...	082938475...	Family Roo...	B200	Rp 400.000	23/01/2024	24/01/2024	Rp 400.000
	2	Alfian Bena...	alfian@gma...	029384892...	Family Roo...	B200	Rp 400.000	10/01/2024	30/01/2024	Rp 0
	3	qinthar	qinthar@g...	0987654321	VIP Room (...)	A001	Rp 400.000	02/01/2024	04/01/2024	Rp 0
	4	gilang	gilang@gm...	0987654321	VIP Room (...)	A001	Rp 400.000	11/01/2024	13/01/2024	Rp 0
	5	RUTH	ruth@gmal...	1092883774	VIP Room (...)	A001	Rp 400.000	01/01/2024	05/01/2024	Rp 0
	6	adzka	adzka@gm...	0987654321	VIP Room (...)	A001	Rp 400.000	10/01/2024	30/01/2024	Rp 0
*										

Form Input Reservasi

Nama

Email

No HP

Kamar  Harga Rp.

Tipe Kamar

Check IN

Check OUT

Total Rp.

Form Pencarian

Cari Data

Tombol Action

## PENUTUP

Aplikasi form tersebut dibuat untuk memenuhi asesmen Pemrograman II (PBO). Pada Aplikasi ini terdapat validasi input, pilihan kamar, dan pemilihan berapa lama akan menyewa kamar hotel. Validasi melibatkan nama, email, noHp, Check IN, Check Out dan kamar. Kamar dapat dipilih dengan opsi no kamar yang telah disesuaikan dan jika sudah memilih akan muncul dengan tipe kamar dan harga yang sudah di pilih. Setelah pengisian dilakukan, pengguna dapat menyimpan atau membatalkan(reset).