

KONEKSI DATABASE

Konsep MVC

Model-View-Controller atau MVC adalah sebuah metode untuk membuat sebuah aplikasi dengan memisahkan data dari tampilan dan cara bagaimana memprosesnya. Dalam implementasinya kebanyakan kerangka kerja dalam aplikasi web adalah berbasis arsitektur MVC.

- **Model**

Model adalah bagian kode program yang menangani query atau database. Jadi isi dari model merupakan bagian (fungsi-fungsi) yang berhubungan langsung dengan database untuk memanipulasi data seperti memasukkan data, pembaruan data, hapus data, dan lain-lain, namun tidak dapat berhubungan langsung dengan bagian view.

- **View**

View adalah bagian kode program ataupun komponen yang mengatur tampilan

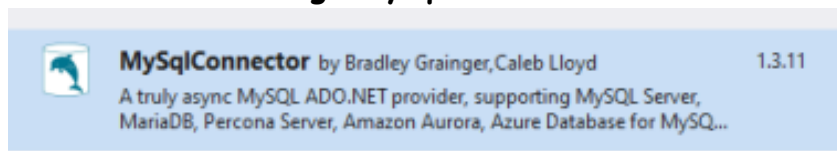
- **Controller**

Komponen yang digunakan untuk mengontrol semua alur yang diharapkan dari sebuah program. Oleh karena itu, semua logika-logika yang diperlukan berada di komponen Controller.

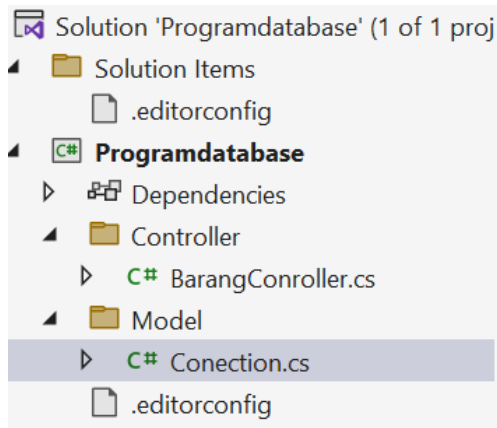
Membuat Koneksi

Membuat koneksi database pada C# sangat perlu dilakukan untuk mengambil data dari database. Aplikasi yang dibangun dengan bahasa C# sangat identik dengan database karena aplikasi C# digunakan untuk CRUD (Create, Read, Update, Delete).

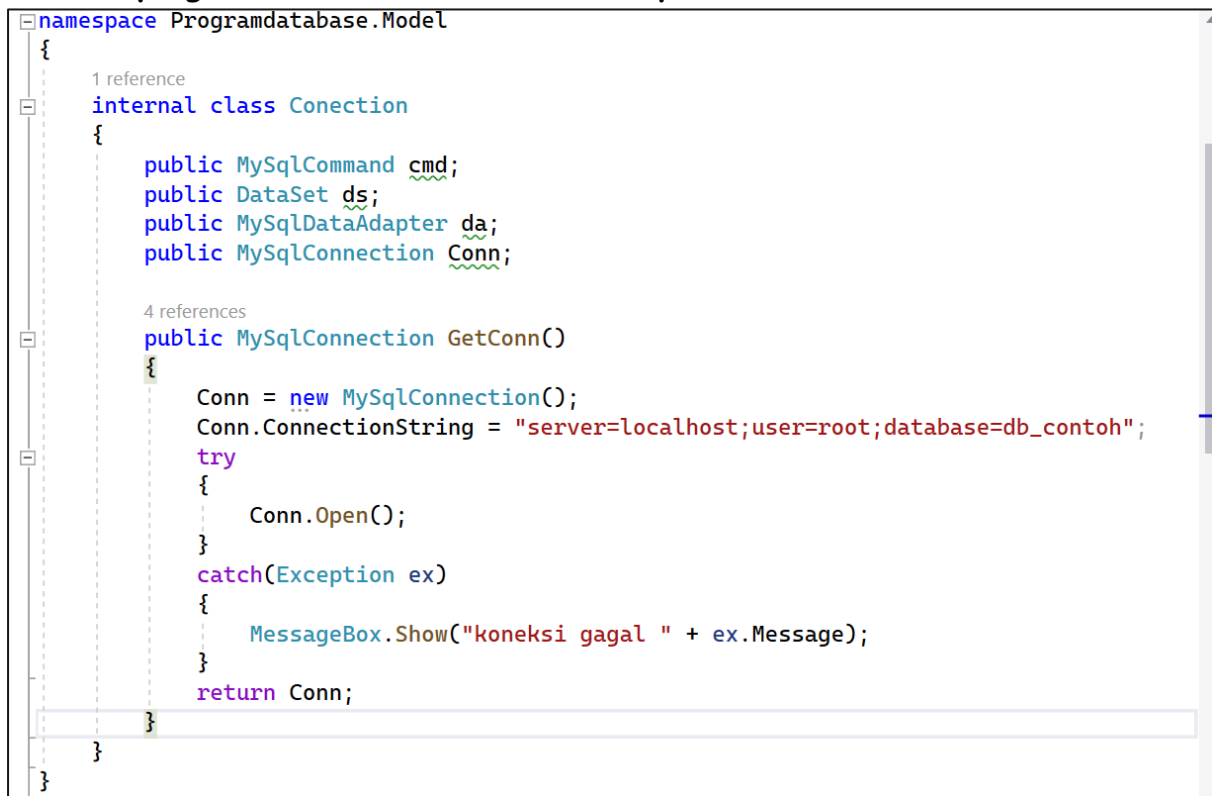
1. Instal NuGet Package mysqlconnector



2. Buat Class dengan nama Conection



3. Buat program dalam class Conection seperti berikut



CRUD (CREATE, READ, UPDATE, DELETE)

1. READ

Yang pertama kita pelajari bagaimana cara menampilkan data dari database kedalam program. Caranya, ikuti Langkah sebelumnya sampai tahap 6. Kemudian sediakan form seperti berikut:

Data Barang

Add Update Delete

Search... Search

Print Next

Kemudian buat class dengan nama **BararangController**, lalu buat fungsi **tampilBarang**

```
using System.Data;

namespace Programdatabase.Controller
{
    9 references
    internal class BararangController : Model.Connection
    {
        1 reference
        public DataTable tampilBarang()
        {
            DataTable data = new DataTable();
            try
            {
                string tampil = "SELECT * FROM barang";
                da = new MySqlConnection.MySqlDataAdapter(tampil, GetConn());
                da.Fill(data);
            }
            catch (Exception ex)
            {
                MessageBox.Show(ex.Message);
            }
            return data;
        }
    }
}
```

Kemudian panggil fungsi tersebut pada form dan masukan datanya ke dalam datagridview

```
using MySqlConnection;
using Programdatabase.Controller;
using System.Data;

namespace Programdatabase
{
    18 references
    public partial class Form1 : Form
    {
        private BarangController brgController;
        8 references
        public Form1()
        {
            brgController = new BarangController();
            InitializeComponent();
        }
        1 reference
        private void Form1_Load(object sender, EventArgs e)
        {
            segarkan();
        }
        1 reference
        private void segarkan()
        {
            dataGridView1.DataSource = brgController.tampilBarang();
            dataGridView1.AutoSizeColumnsMode = DataGridViewAutoSizeColumnsMode.Fill;
        }
    }
}
```

2. CREATE

Cara menambahkan data ke dalam database seperti berikut. Buat form baru seperti berikut:



The image shows a Windows application window titled "Form2". Inside the window, the word "CREATE" is displayed in large, bold, black letters at the top center. Below this, there are four labels with corresponding text boxes: "Id barang :", "Nama barang :", "Harga barang :", and "Stok barang :". Each label is followed by a white rectangular text box. At the bottom of the form, there are two buttons: a green button labeled "Save" on the left and a red button labeled "Cancel" on the right. The window has standard Windows window controls (minimize, maximize, close) in the top right corner.

Lalu panggil form ini di button add yang ada di form1

```
private void button1_Click(object sender, EventArgs e)
{
    Form2 mn = new Form2();
    mn.Show();
    this.Hide();
}
```

Kemudian pada class **BarangController**, buat fungsi/void dengan nama **tambahBarang**

```
1 reference
public void tambahBarang(string idbarang, string namabarang, string hargabarang, string stok)
{
    string tambah = "insert into barang values(" + "@id_barang,@nama_barang,@harga_barang,@stok)";
    try
    {
        cmd = new MySqlConnection.MySqlCommand(tambah, GetConn());
        cmd.Parameters.Add("@id_barang", MySqlConnection.MySqlDbType.VarChar).Value = idbarang;
        cmd.Parameters.Add("@nama_barang", MySqlConnection.MySqlDbType.VarChar).Value = namabarang;
        cmd.Parameters.Add("@harga_barang", MySqlConnection.MySqlDbType.VarChar).Value = hargabarang;
        cmd.Parameters.Add("@stok", MySqlConnection.MySqlDbType.VarChar).Value = stok;
        cmd.ExecuteNonQuery();
    }
    catch (Exception ex)
    {
        MessageBox.Show("tambah data gagal " + ex.Message);
    }
}
```

Kemudian panggil fungsi tambahBarang (di tombol save - form2)

```
1 reference
private void button1_Click(object sender, EventArgs e)
{
    brgController = new BarangController();
    brgController.tambahBarang(textBox1.Text, textBox2.Text, textBox3.Text, textBox4.Text);
    this.Controls.Clear();
    this.InitializeComponent();
    textBox1.Focus();
    MessageBox.Show("data disimpan");
    Form1 lk = new Form1();


    lk.Show();
    this.Hide();
}
```

Tombol cancel

```
private void button2_Click(object sender, EventArgs e)
{
    MessageBox.Show("Keluar");
    Form1 cl = new Form1();
    cl.Show();
    this.Hide();
}
```

3. UPDATE

Cara menambahkan data ke dalam database seperti berikut. Buat form baru seperti berikut:



The image shows a Windows form titled "Form3". The form has a light gray background. At the top center, the word "UPDATE" is displayed in a large, bold, black font. Below this, there are four text boxes arranged vertically, each preceded by a label: "Id barang", "Nama barang", "Harga barang", and "Stok barang". At the bottom left of the form is a green button labeled "Save". At the bottom right is a red button labeled "Cancel". The form has a standard Windows window border with minimize, maximize, and close buttons in the top right corner.

Lalu panggil form ini di button update yang ada di form1

```
1 reference
private void button2_Click(object sender, EventArgs e)
{
    Form3 upd = new Form3();
    upd.Show();
    this.Hide();

    upd.textBox1.Text = this.dataGridView1.CurrentRow.Cells[0].Value.ToString();
    upd.textBox2.Text = this.dataGridView1.CurrentRow.Cells[1].Value.ToString();
    upd.textBox3.Text = this.dataGridView1.CurrentRow.Cells[2].Value.ToString();
    upd.textBox4.Text = this.dataGridView1.CurrentRow.Cells[3].Value.ToString();
}
1 reference
```

Kemudian pada class **BarangController**, buat fungsi/void dengan nama **updatebarang**

```
1 reference
public void updatebarang(string idbarang, string namabarang, string hargabarang, string stok)
{
    string update = "update barang set " + "nama_barang=@nama_barang,harga_barang=@harga_barang,stok=@stok " +
        "where id_barang=" + idbarang;

    try
    {
        cmd = new MySqlConnection.MySqlCommand(update, GetConn());
        cmd.Parameters.Add("@id_barang", MySqlConnection.MySqlDbType.VarChar).Value = idbarang;
        cmd.Parameters.Add("@nama_barang", MySqlConnection.MySqlDbType.VarChar).Value = namabarang;
        cmd.Parameters.Add("@harga_barang", MySqlConnection.MySqlDbType.VarChar).Value = hargabarang;
        cmd.Parameters.Add("@stok", MySqlConnection.MySqlDbType.VarChar).Value = stok;
        cmd.ExecuteNonQuery();
    }
    catch (Exception ex)
    {
        MessageBox.Show("update data gagal " + ex.Message);
    }
}
```

Kemudian pada panggil fungsi updatebarang (di tombol save - form update)

```
1 reference
private void button1_Click(object sender, EventArgs e)
{
    brgController = new BarangController();
    brgController.updatebarang(textBox1.Text, textBox2.Text, textBox3.Text, textBox4.Text);
    this.Controls.Clear();
    this.InitializeComponent();
    textBox1.Focus();
    MessageBox.Show("data disimpan");
    Form1 upd1 = new Form1();

    upd1.Show();
    this.Hide();
}
```

Tombol cancel

```
private void button2_Click(object sender, EventArgs e)
{
    Form1 upd2 = new Form1();
    upd2.Show();
    this.Hide();
}
```


4. DELETE

1 reference

```
public void hapusbarang(string id_barang)
{
    string hapus = "delete from barang where id_barang=" + id_barang;

    try
    {
        cmd = new MySqlConnection.MySqlCommand(hapus, GetConn());
        cmd.Parameters.Add("@id_barang", MySqlConnection.MySqlDbType.VarChar).Value = id_barang;
        cmd.ExecuteNonQuery();
    }
    catch (Exception ex)
    {
        MessageBox.Show("delete data gagal" + ex.Message);
    }
}
```

```
private void button3_Click(object sender, EventArgs e)
{
    brgController = new BarangController();
    //textBox1.Text = this.dataGridView1.CurrentRow.Cells[0].Value.ToString();

    //segarikan();
    string selectedValue = dataGridView1.CurrentRow.Cells[0].Value.ToString();
    Application.Restart();
}
```

PRINT

1 reference

```
private void printDocument1_PrintPage(object sender, System.Drawing.Printing.PrintPageEventArgs e)
{
    Bitmap btm = new Bitmap(this.dataGridView1.Width, this.dataGridView1.Height);
    dataGridView1.DrawToBitmap(btm, new Rectangle(0, 0, this.dataGridView1.Width, this.dataGridView1.Height));
    e.Graphics.DrawImage(btm, 170, 120);
    e.Graphics.DrawString(label1.Text, new Font("Consolas", 23, FontStyle.Bold), Brushes.Black, new Point(310, 50));
}
```

printPreviewDialog1.Document = printDocument1;

SEARCH

```
private void button4_Click(object sender, EventArgs e)
{
    string koneksi = "server=localhost;user=root;database=db_contoh";
    MySqlConnection kon = new MySqlConnection(koneksi);
    kon.Open();
    string cari = "select * from barang where concat(id_barang,nama_barang,harga_barang,stok)like '%" + textBox1.Text + "%'";
    MySqlDataAdapter da = new MySqlDataAdapter(cari, kon);
    DataTable tabel = new DataTable();
    da.Fill(tabel);
    dataGridView1.DataSource = tabel;
}
```