

Nama : Asmat Baiadawi

NIM : 2021520021

Soal

1. Silahkan ubah tugas 3 dengan konsep class dan tambahkan pencarian nama barang !!

Jawaban

Outpt dari Program

Disini saya sudah menambahkan data yang nanti akan menjadi bahan percobaan.

Form1

Nama : Asmat Baidawi
NIM : 2021520021
Kelas : A

Cari

Kode Barang :
Nama Barang :
Stok Barang :

Delete Edit Tambah

Kode Barang	Nama Barang	Stok Barang
BRT000	susu	30
BRT001	sabun	40
BRT002	sampo	50
BRT003	pepsodent	25
BRT004	kopi	60
BRT005	mari	43

Selanjutnya kita melakukan picarian mesalnya kita ingin mencari barang kopi

Form1

Nama : Asmat Baidawi
NIM : 2021520021
Kelas : A

Cari


Kode Barang :
Nama Barang :
Stok Barang :

Delete Edit Tambah

Kode Barang	Nama Barang	Stok Barang
BRT000	susu	30
BRT001	sabun	40
BRT002	sampo	50
BRT003	pepsodent	25
BRT004	kopi	60
BRT005	mari	43

Maka disini akan tampil data yang kita cari

Form1



Nama : Asmat Baidawi
NIM : 2021520021
Kelas : A

Kode Barang :


Nama Barang :

Stok Barang :

	Kode Barang	Nama Barang	Stok Barang
▶	BRT004	kopi	60

Dan Ketika click tombol riset maka data sebelumnya akan kembali

Form1



Nama : Asmat Baidawi
NIM : 2021520021
Kelas : A

Kode Barang :

Nama Barang :

Stok Barang :

	Kode Barang	Nama Barang	Stok Barang
▶	BRT000	susu	30
	BRT001	sabun	40
	BRT002	sampo	50
	BRT003	pepsodent	25
	BRT004	kopi	60
	BRT005	mari	43

Penjelasan Program

1. Pertama kita akan menambahkan class Bernama DataBarang

```
7 namespace Tugas_DataGrindView
8 {
9     internal class DataBarang
10    {
11        DataTable data = new DataTable();
12
13        public DataBarang()
14        {
15            data.Columns.Add("Kode Barang");
16            data.Columns.Add("Nama Barang");
17            data.Columns.Add("Stok Barang");
18        }
19
20        public DataTable GetData()
21        {
22            return data;
23        }
24
25
26        public void TambahBarang(string kode, string nama, string stok)
27        {
28            string[] arrayData = new string[3];
29            arrayData[0] = kode;
30            arrayData[1] = nama;
31            arrayData[2] = stok;
32            data.Rows.Add(arrayData);
33
34        }
35
36        public void EditBarang(int rowIndex, string kode, string nama, string stok)
37        {
38            data.Rows[rowIndex][0] = kode;
39            data.Rows[rowIndex][1] = nama;
40            data.Rows[rowIndex][2] = stok;
41        }
42
43        public void DeleteBarang(int rowIndex)
44        {
45            if (data.Rows.Count > 0)
46            {
47                data.Rows[rowIndex].Delete();
48
49                if (data.Rows.Count == 0)
50                {
51                }
52            }
53        }
54    }
55 }
56
57
```

Class **DataBarang** yang memiliki struktur data dalam bentuk **DataTable**. Kelas ini memiliki beberapa metode:

Konstruktor **DataBarang()** digunakan untuk inisialisasi **DataTable** dengan tiga kolom: "Kode Barang", "Nama Barang", dan "Stok Barang".

Metode **GetData()** mengembalikan **DataTable** yang berisi data barang.

Metode **TambahBarang()** digunakan untuk menambahkan data barang baru ke **DataTable**.

Metode **EditBarang()** memungkinkan pengeditan data barang yang ada di **DataTable** berdasarkan indeks baris.

Metode **DeleteBarang()** menghapus data barang dari **DataTable** berdasarkan indeks baris.

2. **DataTable data = new DataTable();** Ini adalah deklarasi dan inisialisasi objek data sebagai instance dari **DataTable**. **DataTable** digunakan untuk menyimpan dan mengelola data dalam bentuk tabel, seperti data barang.

DataBarang toko = new DataBarang(); Ini adalah deklarasi dan inisialisasi objek toko sebagai instance dari kelas **DataBarang**. **DataBarang** adalah kelas yang Anda buat sendiri, dan objek toko digunakan untuk mengelola operasi terkait data barang, seperti menambah, mengedit, atau menghapus data barang.

```
12 public partial class Form1 : Form
13 {
14     DataTable data = new DataTable();
15
16     DataBarang toko = new DataBarang();
17 }
```

3. Kode ini mengatur tampilan **dataGridView1** dengan data dari objek **toko** saat form aplikasi dimuat yang ada di class **DataBarang**.

```
23 private void Form1_Load(object sender, EventArgs e)
24 {
25     dataGridView1.DataSource = toko.GetData();
26
27 }
```

4. Kode ini digunakan untuk menambahkan data barang baru ke objek toko dengan mengambil nilai yang ada di `textKode`, `textNama`, dan `textStok`, yang mungkin merupakan input dari pengguna.

Dengan menggunakan metod `toko.TambahBarang(..)`, Anda memasukkan data baru ke dalam objek toko yang kemudian dapat disimpan atau ditampilkan dalam `dataGridView1`.

```
57 |  
58 |  
59 |         toko.TambahBarang(textKode.Text, textNama.Text, textStok.Text);
```

5. **`int rowIndex = dataGridView1.CurrentRow.Index;`** Baris ini mengambil indeks (posisi) dari baris yang saat ini dipilih dalam `dataGridView1` dan menyimpannya dalam variabel `rowIndex`. Dengan demikian, Anda tahu baris mana yang akan diedit.

`toko.EditBarang(rowIndex, textKode.Text, textNama.Text, textStok.Text);`; Kode ini memanggil metode `EditBarang` pada objek toko dengan parameter `rowIndex` dan nilai dari `textKode`, `textNama`, dan `textStok`. Ini memungkinkan Anda untuk mengganti data di baris yang dipilih dengan nilai yang baru, sehingga data barang dapat diperbarui.

```
93 |  
94 |  
95 |         int rowIndex = dataGridView1.CurrentRow.Index;  
96 |         toko.EditBarang(rowIndex ,textKode.Text, textNama.Text, textStok.Text);
```

6. **`int rowIndex = dataGridView1.CurrentRow.Index;`**; Kode ini mengambil indeks (posisi) dari baris yang saat ini dipilih dalam `dataGridView1` dan menyimpannya dalam variabel `rowIndex`. Dengan ini, Anda tahu baris mana yang akan dihapus.

`toko.DeleteBarang(rowIndex);`; Kode ini memanggil metode `DeleteBarang` pada objek toko dengan parameter `rowIndex`. Hal ini memungkinkan Anda untuk menghapus data pada baris yang dipilih dari objek toko, sehingga data barang dapat dihapus dari daftar barang.

```
122 | private void button3_Click(object sender, EventArgs e)  
123 | {  
124 |     int rowIndex = dataGridView1.CurrentRow.Index;  
125 |     toko.DeleteBarang(rowIndex);  
126 | }  
127 |
```

7. `private bool mencari = false;`: Variabel boolean yang melacak apakah aplikasi dalam mode pencarian atau tidak.

`string.IsNullOrEmpty(namaBarang)`: Memeriksa apakah input `namaBarang` kosong atau hanya berisi spasi.

```
145     private bool mencari = false;
146
147     private void button4_Click(object sender, EventArgs e)
148     {
149         if (!mencari)
150         {
151             // Ambil nama barang dari textBox1
152             string namaBarang = textcari.Text;
153
154             if (string.IsNullOrEmpty(namaBarang))
155             {
156                 MessageBox.Show("Silakan masukkan nama barang untuk melakukan pencarian.", "Info", MessageBoxButtons.OK, MessageBoxIcon.Information);
157                 return;
158             }
159
160             // Lakukan pencarian berdasarkan nama barang
161             DataRow[] cariBaris = originalData.Select("[Nama Barang] = '" + namaBarang + "'");
162
163             if (cariBaris.Length > 0)
164             {
165                 // Buat tabel baru untuk menampilkan hasil pencarian
166                 DataTable searchResult = cariBaris.CopyToDataTable();
167
168                 // Tampilkan hasil pencarian di dataGridView1
169                 dataGridView1.DataSource = searchResult;
170
171                 // Ubah teks tombol menjadi "Kembalikan"
172                 button4.Text = "Riset";
173                 mencari = true;
174             }
175             else
176             {
177                 MessageBox.Show("Data tidak ditemukan", "Info", MessageBoxButtons.OK, MessageBoxIcon.Information);
178             }
179         }
180         else
181         {
182             // Mengembalikan data ke keadaan asli
183             dataGridView1.DataSource = originalData;
184
185             // Ubah teks tombol menjadi "Cari" kembali
186             button4.Text = "Cari";
187             mencari = false;
188         }
189     }
190
191 }
192
193 }
194
```

Saat tombol "Cari" (button4) diklik, kode ini akan memeriksa apakah dalam mode "mencari" (`mencari = true`) atau tidak.

Jika tidak dalam mode mencari, kode akan:

- Mengambil input dari `textBox1` (`textcari`) sebagai nama barang yang dicari.

- Memeriksa apakah input kosong dan menampilkan pesan jika iya.

- Melakukan pencarian dalam data (`originalData`) berdasarkan nama barang.

Jika data ditemukan, akan membuat tabel baru (searchResult) untuk menampilkan hasil pencarian di dataGridView1.

Mengubah teks tombol menjadi "Riset" dan memasuki mode "mencari."

Jika sudah dalam mode mencari, kode akan:

Mengembalikan tampilan dataGridView1 ke data asli (originalData).

Mengubah teks tombol kembali menjadi "Cari" dan keluar dari mode "mencari."