

## **DISCLAIMER**

Seluruh dokumen E-Trik di dalam CD ini dapat digunakan dan disebarakan secara bebas untuk tujuan belajar bukan komersial (non-profit), dengan syarat tidak menghapus atau merubah atribut penulis dan pernyataan copyright yang disertakan dalam dokumen. Tidak diperbolehkan melakukan penulisan ulang, kecuali mendapatkan ijin tertulis terlebih dahulu dari Penerbit Maxikom.

# **E-Trik Visual C++ 6.0**

**Dasar Pemrograman**

**MUHAMMAD SADELI**



**[www.maxikom.co.id](http://www.maxikom.co.id)**


E-Trik ini merupakan e-trik Dasar pemrograman **Microsoft Visual C++ 6.0**, disini saya akan membahas dasar pemrograman Visual C++ 6.0 mulai dari pengertian, pengenalan area kerja, type data, operator, variabel, kontrol program, dan sampai pada latihan pembuatan program aplikasi yang pertama dengan Microsoft Visual C++ 6.0.

### Mengenal Microsoft Visual C++ 6.0

Visual C++ 6.0 Adalah bahasa pemrograman yang menggunakan bahasa C yang merupakan bahasa tingkat menengah dan juga merupakan lanjutan dari bahasa B. Bahasa C++ diciptakan oleh *Bjarne Stroustrup* pada tahun 1983 dan yang memberikan nama C++ adalah *Rick Mascitti* pada tahun 1983. keistimewaan dari bahasa ini dikarenakan bahasa ini mendukung pemrograman berorientasi object (OOP) yaitu bahasa pemrograman yang memperlakukan data dan procedure sebagai sebuah objek dengan identitas dan ciri - ciri yang khusus. Visual C++ 6.0 tergabung dalam satu kumpulan software visual yaitu Microsoft Visual Studio 6.0, Visual C++ 6.0 sendiri memiliki beberapa macam jenis *project* salah satunya **MFC AppWizard**, yang dapat dibangun menjadi suatu aplikasi yang sangat besar dan kompleks.

### Membuka Program Microsoft Visual C++ 6.0

Untuk membuka program Microsoft Visual C++ 6.0, Silakan Anda Ikuti Langkah - langkah dibawah ini

1. Klik Tombol **Start** , pada windows taksbar, kemudian pilih **All Programs > Microsoft Visual Studio 6.0 > Microsoft Visual C++ 6.0**, lihat **Gambar 1**

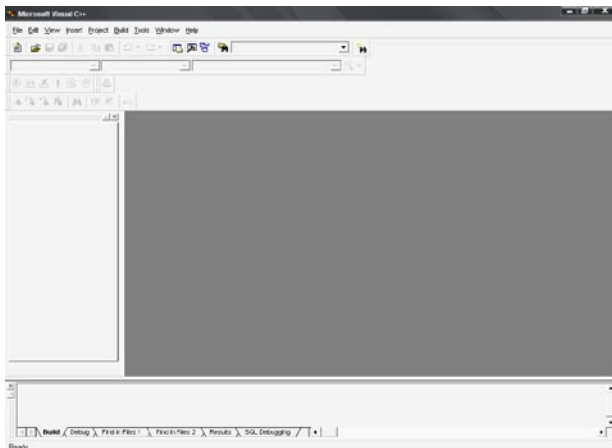
## Dasar Pemrograman Microsoft Visual C++ 6.0

---



*Gambar 1. Membuka program Microsoft Visual C++ 6.0*

2. Setelah itu tampilan awal program **Microsoft Visual C++ 6.0** yang akan keluar bisa Anda lihat pada gambar dibawah ini, lihat **Gambar 2**

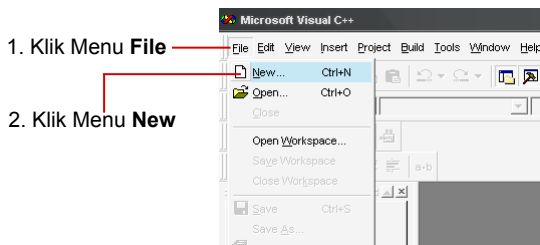


*Gambar 2. Tampilan awal program Microsoft Visual C++ 6.0*

## Membuat Project Baru

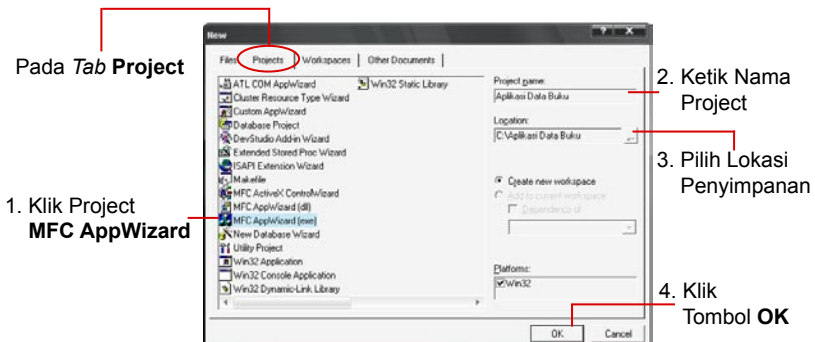
Selanjutnya Anda akan membuat *project* baru dengan nama “Aplikasi Data Buku”. Untuk itu ikuti langkah berikut ini

1. Klik menu **File** > **New** atau tekan **Ctrl+N**, lihat **Gambar 3**



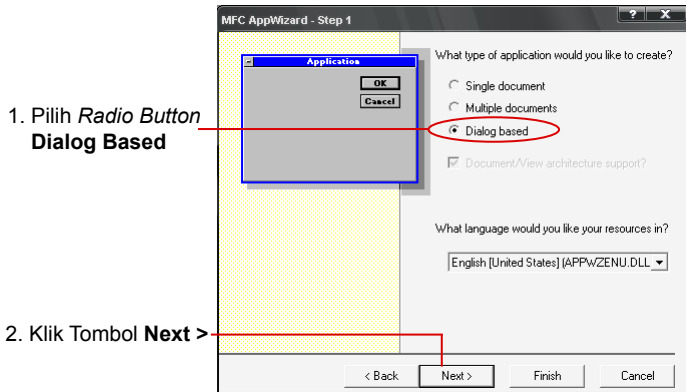
Gambar 3. Membuat project baru

2. Pada kotak dialog **New**, pada Tab project pilih jenis project **MFC AppWizard(exe)**, lalu ketikkan nama project dan tentukan lokasi penyimpanan dengan menekan tombol **...**, pada kolom Location kemudian klik **OK**, lihat **Gambar 4**



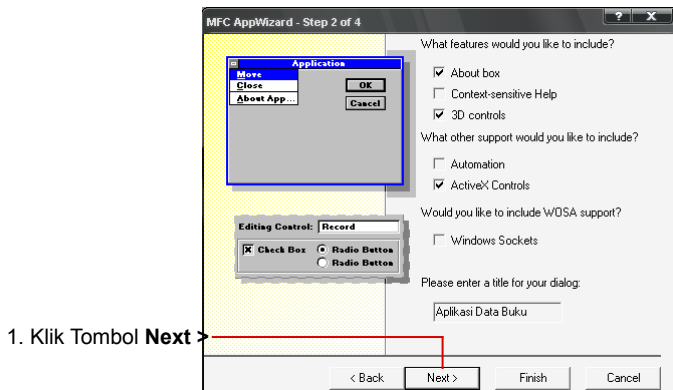
Gambar 4. Menentukan dan menyimpan jenis project

3. Pada kotak dialog **MFC AppWizard - Step 1**, pilih radio button **Dialog Based** kemudian klik tombol **Next >**, lihat **Gambar 5**



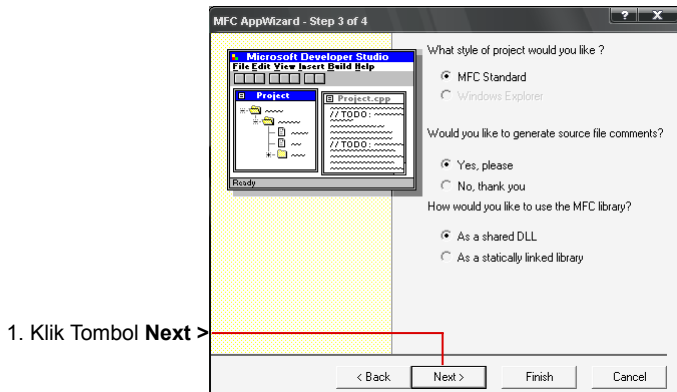
*Gambar 5. Menentukan tipe aplikasi*

4. Pada kotak dialog **MFC AppWizard - Step 2 of 4**, klik tombol **Next >**, lihat **Gambar 6**



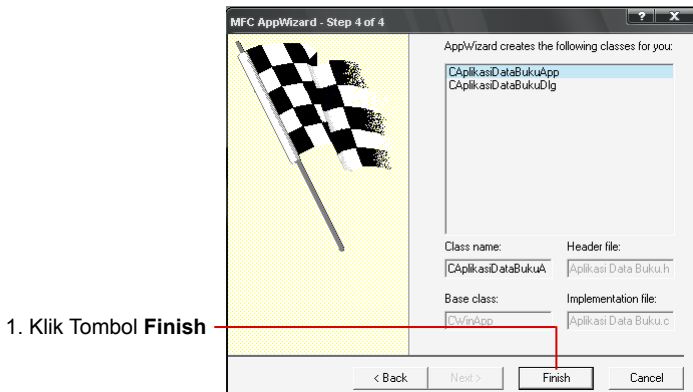
*Gambar 6. Menentukan fitur yang ingin dipakai*

5. Pada kotak dialog **MFC AppWizard - Step 3 of 4**, klik tombol **Next >**, lihat **Gambar 7**



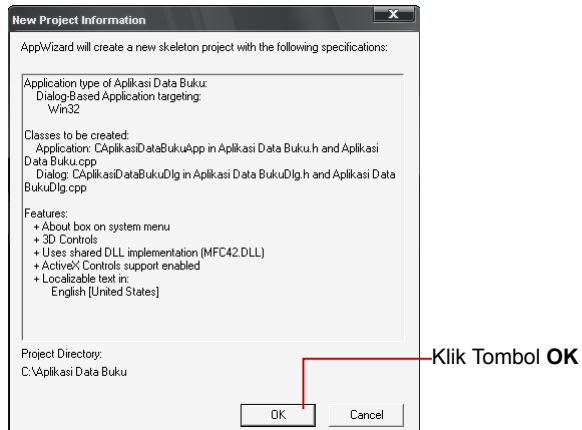
*Gambar 7. Menentukan style project*

6. Pada kotak dialog **MFC AppWizard - Step 4 of 4**, klik tombol **Finish**, lihat **Gambar 8**



*Gambar 8. Hasil pembuatan project baru dari step 1 - step 4*

7. Terakhir muncul kotak dialog **New Project Information**, klik tombol **OK**, lihat **Gambar 9**

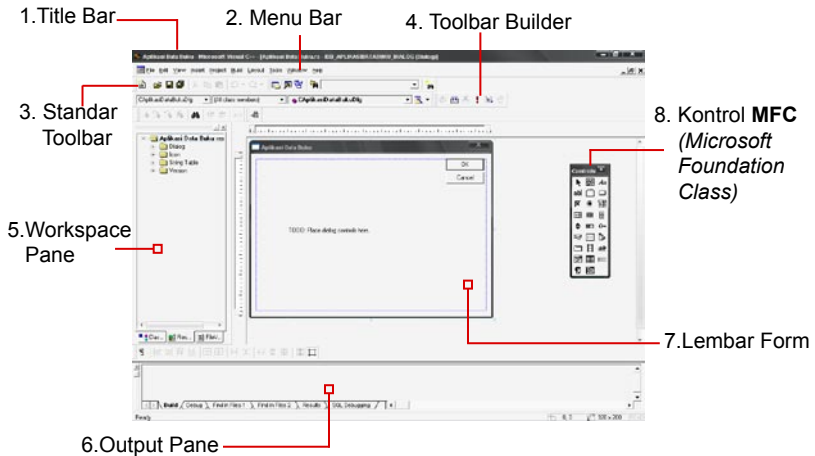


*Gambar 9. Menampilkan informasi project secara sfesifikasi*

## Mengenal Area Kerja Microsoft Visual C++ 6.0

Setelah pembuatan project baru selesai, ada baiknya Anda mengenal terlebih dahulu area kerja Microsoft Visual C++ 6.0. Agar Anda tidak kesulitan dalam memahami serta menggunakan kontrol - kontrol atau komponen - komponen yang terdapat dalam Microsoft Visual C++ 6.0. Serta Anda tidak akan mengalami kesulitan yang berarti untuk membuat suatu Aplikasi yang baru nanti, untuk itu lihat gambar dibawah ini, lihat **Gambar 10**





Gambar 10. Area kerja Microsoft Visual C++ 6.0

### 1. Title Bar

Baris judul yang akan menampilkan file *project* serta nama dokumen yang sedang aktif

### 2. Menu Bar

Baris menu yang terdiri dari 10 menu utama, masing - masing memiliki sub menu dan perintah lengkap dengan *shortcut key*.

### 3. Standar Toolbar

Baris ini mempunyai fungsi yang sama dengan Tool Standar pada umumnya seperti fungsi untuk menyimpan, mengcopy, menambah menu tampilan program, mengatur tampilan program dan masih banyak lagi.

### 4. Toolbar Builder

Merupakan toolbar yang berfungsi untuk menjalankan atau mengexecute program yang telah Anda buat.

## 5. **Workspace Pane**

Lembar yang berfungsi untuk melihat sub - sub project yang telah kita bangun,. Lembar ini memiliki 3 bagian untuk melihat sub - sub *project* yaitu *Class View*, *Resources View*, *File View*.

## 6. **Output Pane**

Lembar ini berfungsi untuk menampilkan informasi kompiler dari program yang telah Anda buat, dan menampilkan setiap kesalahan dalam pembuatan Aplikasi.

## 7. **Kontrol MFC (*Microsoft Foundation Class*)**

Toolbar yang berisi semua komponen - komponen yang memiliki banyak fungsi untuk membangun suatu aplikasi pemrograman Visual.

# Type Data, Variabel Dan Operator

## 1. Type Data

Merupakan suatu bentuk penggolongan jenis data berdasarkan kategori data, ukuran, dan kegunaan data dalam media bahasa pemrograman khususnya bahasa pemrograman Visual C++ 6.0. Beberapa type data dapat Anda lihat Pada tabel di bawah ini, lihat **Tabel 1**

*Tabel 1. Type data*

Type Data	Kategori	Ukuran	Jangkauan	Digit
Char	Integral	1 Byte	- 128 hingga + 127	-
Integer	Integral	4 Byte	- 32768 hingga + 32767	-
Short	Integral	2 Byte		
_Int <i>n</i>	Integral	Varies		
Long	Integral	4 Byte	- 2.147.438.648 hingga 2.147.438.647	-
Float	Floating	4 Byte	- 3,4E-38 hingga 3,4E38	6 - 7
Double	Floating	8 Byte	1.7E-308 hingga 1.7E308	15 - 16
Long Double	Floating	10 Byte	3.4E-4932 hingga 1.1E4932	19

## 2. Variabel

Merupakan simbol yang terdapat dalam suatu besaran yang dapat memberitahukan suatu lokasi yang tersimpan didalam memori komputer, informasi yang tersimpan dalam lokasi inilah yang disebut dengan nilai Variabel. Beberapa variabel dapat Anda lihat pada tabel di bawah ini, lihat **Tabel 2**

*Tabel 2. Tabel Variabel*

Variabel	Penulisan Simbol
Integer	Int
Floating Point	Float
Double Precision	Double
Charakter	Char
Unsigned Integer	Unsigned Int
Unsigned Karakter	Unsigned Char
Long Integer	Long Int
Unsigned Long Integer	Unsigned Long Int

### 3. Operator

Merupakan simbol yang sering berperan banyak dalam pembuatan kode pemrograman, yang memiliki fungsi untuk mengontrol data dalam pembuatan aplikasi pemrograman. Sebagai contoh operator aritmatika dapat Anda lihat pada tabel di bawah ini, lihat **Tabel 3**.

*Tabel 3. Tabel Operator Aritmatika*

Operator	Deskripsi	Contoh
+	Penjumlahan	$x + y$
-	Pengurangan	$x - y$
*	Perkalian	$x * y$
/	Pembagian	$x / y$
%	Sisa Pembagian Integer	$x \% y$
-	Negrasi	$-x$

## Kontrol Program

Merupakan fungsi - fungsi yang dapat melengkapi dalam pembuatan kode pemrograman, beberapa diantaranya seperti fungsi pencabangan dan fungsi perulangan. Dibawah ini akan dijelaskan beberapa fungsi dari kontrol program.

## 1. Fungsi Pencabangan *if*

Merupakan suatu kontrol pencabangan yang mempunyai satu pencabangan atau satu blok perintah, tergantung pada nilai yang akan diuji biasanya terdiri dari satu nilai atau satu syarat.

Contoh Penulisan :

```
if (Kondisi)
{
    Perintah ;
}
```

Contoh Kode pemrograman :

1	if(strlen(m_NAMA)==0)
2	{
3	MessageBox("Biodata Anda belum Lengkap");
4	}

### Keterangan Kode

Baris 1 - 4 Merupakan contoh kode pemrograman fungsi pencabangan *if*, yang apabila pada kolom biodata Nama kosong, maka akan ditampilkan pesan yang berisi informasi bahwa biodata Anda belum lengkap.

## 2. Fungsi Pencabangan *if else*

Merupakan suatu kontrol pencabangan yang dapat menjalankan satu blok perintah yang memiliki dua nilai atau dua syarat yang akan diuji.

Contoh Penulisan :

```
if (kondisi)
    Pernyataan 1 ;
else
    Pernyataan 2 ;
```

Contoh Kode pemrograman :

```
1 if(strlen(m_NAMA)==0)
2     MessageBox("Biodata Anda belum Lengkap");
3 else
4     MessageBox("Biodata Anda Lengkap");
```

**Keterangan Kode**

Baris 1 - 4 Merupakan contoh kode pemrograman fungsi percabangan *if else*, yang apabila pada kolom biodata Nama kosong maka akan ditampilkan pesan yang berisi informasi bahwa biodata Anda belum lengkap dan apabila kolom biodata Nama terisi maka akan tampil pesan yang berisi informasi bahwa biodata Anda sudah lengkap.

### 3. Fungsi Perulangan *For*

Merupakan suatu kontrol perulangan yang berfungsi untuk melakukan perulangan (*iterasi*), dari suatu blok program secara berulang - ulang yang ditentukan oleh nilai awal dan nilai akhir.

Contoh Penulisan :

```
for (inisialisasi ; kondisi ; perubahan)
{
    pernyataan;
}
```

Contoh Kode pemrograman :

```
1 for(int x=1 ; x<=1000 ; x++)
2 {
3     cout<<x<<endl;
4 }
```

**Keterangan Kode**

Baris 1 - 4 Merupakan contoh kode pemrograman fungsi perulangan *For*, yang akan mencetak angka 1 sampai 1000

#### 4. Fungsi Perulangan *While*

Merupakan suatu kontrol perulangan yang berfungsi untuk melakukan perulangan yang memiliki suatu syarat tertentu, dan akan terus dijalankan selama syarat tersebut terpenuhi dan begitupun sebaliknya jika kondisi tidak terpenuhi maka pernyataan tidak akan dieksekusi.

Contoh Penulisan :

```
While (Kondisi)
{
    Pernyataan;
}
```

Contoh Kode pemrograman :

```
1 While (bil<=100)
2     {
3         cout<<bil<<" ";
4         ++bil;
5     }
```

##### Keterangan Kode

Baris 1 - 5 Merupakan contoh kode pemrograman fungsi perulangan *While*, yang akan mencetak angka 1 sampai 100

#### 5. Fungsi Perulangan *Do While*

Merupakan suatu kontrol perulangan yang melakukan perulangan terlebih dahulu dan bentuk pengujian dilakukan belakangan.

Contoh Penulisan :

```
Do
{
    Pernyataan;
}
While (Kondisi);
```

Contoh Kode pemrograman :

```
1 do
2     {
3         cout<<bil<<" ";
4         bil+=2;
5     }
6     while(bil<=100)
```

#### Keterangan Kode

Baris 1 - 6 Merupakan contoh kode pemrograman fungsi pengulangan *do while*, yang akan mencetak bilangan genap 1 - 100

## Membuat Aplikasi

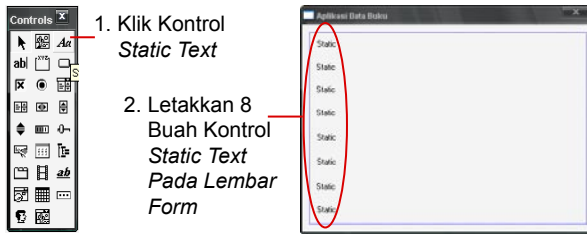
Sekarang Anda akan mencoba membuat aplikasi yang pertama yaitu Aplikasi Data Buku yang sangat sederhana. Aplikasi yang akan Anda buat ini berfungsi untuk menghitung total pembelian buku oleh pelanggan. Untuk itu ikuti langkah - langkah dibawah ini

### A. Mendesain Tampilan Program

Pertama - tama Anda harus mendesain tampilan program yang akan Anda buat, ingin seperti apa tampilanya nanti sewaktu dijalankan. Ikuti langkah dibawah ini

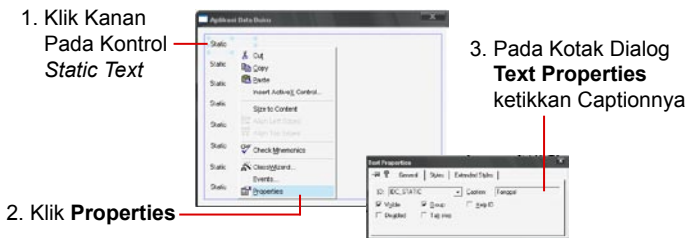
1. Pada toolbar kontrol **MFC** Anda klik kontrol *Static Text* **Aa**, kemudian Anda letakkan pada lembar **Form** dengan mengklik lembar Form, kemudian tahan dan drag sesuai ukuran, letakkan pada lembar Form sebanyak 8 buah, lihat **Gambar 11**





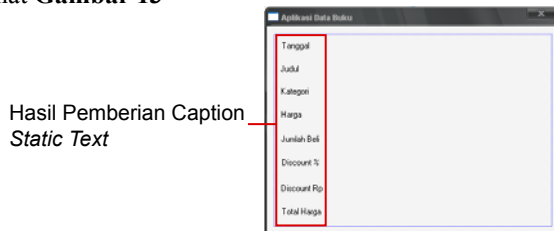
Gambar 11. Meletakkan kontrol MFC static text

2. Kemudian beri caption pada 8 buah kontrol MFC *Static Text* yang melekat pada lembar Form. Caranya klik kanan pada kontrol MFC *Static Text* kemudian pilih **Properties**, maka akan muncul kotak dialog **Text Properties**. Kemudian pada kolom caption kotak dialog **Text Properties** Anda ketikkan nama kontrol *Static Text* tersebut, lihat **Gambar 12**




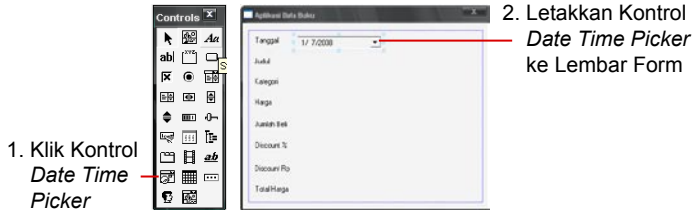
Gambar 12. Memberikan Caption Pada Kontrol *Static Text*

3. Hasil akhir dari pemberian Caption ke 8 buah kontrol MFC *Static Text* yang melekat pada lembar Form dapat Anda lihat seperti gambar dibawah ini, lihat **Gambar 13**




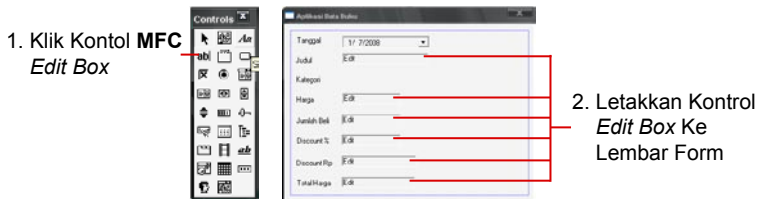
Gambar 13. Hasil pemberian caption pada kontrol MFC *Static Text*

4. Pada toolbar kontrol **MFC** Anda klik kontrol *Date Time Picker* , kemudian Anda letakkan pada lembar **Form** dengan mengklik lembar Form kemudian tahan dan drag sesuai ukuran, lihat **Gambar 14**




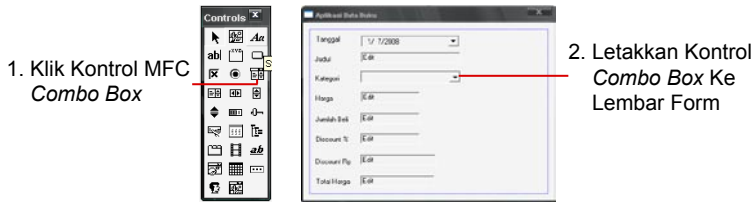
*Gambar 14. Meletakkan kontrol MFC Date Time Picker*

5. Pada toolbar kontrol **MFC** Anda klik kontrol *Edit Box* , kemudian Anda letakkan pada lembar **Form** dengan mengklik lembar Form, kemudian tahan dan drag sesuai ukuran, aturlah ukuran serta posisi masing masing kontrol **MFC Edit Box** pada lembar Form sebanyak 6 buah, lihat **Gambar 15**



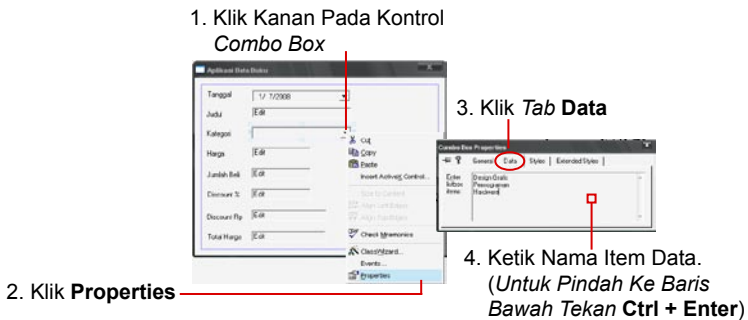
*Gambar 15. Meletakkan kontrol MFC Edit Box*

6. Pada toolbar kontrol **MFC** Anda klik kontrol *Combo Box* , kemudian Anda letakkan pada lembar **Form** dengan mengklik lembar Form, kemudian tahan dan drag sesuai ukuran, lihat **Gambar 16**



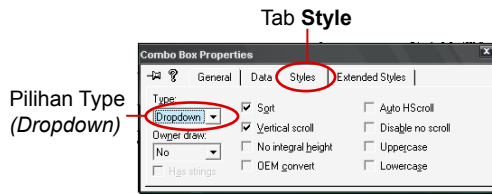
Gambar 16. Meletakkan kontrol MFC Combo Box

7. Kemudian beri item data pada pilihan *combo box*. Caranya klik kanan pada kontrol *combo box* kemudian pilih **Properties**, maka akan muncul kotak dialog **Combo Box Properties**. Kemudian Anda klik *Tab Data* isilah item data pada kolom **Enter listbox items**. Dan untuk jumlah pilihan item yang lebih dari satu anda tekan **Ctrl + Enter** pada tombol keyboard Anda, lalu kursor akan pindah ke baris baru pada kolom Enter listbox items, isilah nama item selanjutnya, lihat **Gambar 17**





Gambar 17. Membuat item data pada Combo Box

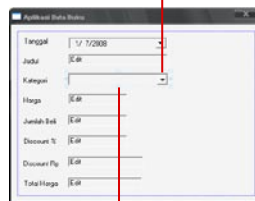
8. Pastikan kolom pilihan *Type* pada Tab *Style* bernama *Dropdown*, lihat **Gambar 18**



Gambar 18. Menentukan pilihan Type pada Tab Style Combo Box

9. Untuk mengeluarkan lembar gulung pada saat tanda , pada combo box di klik, Anda harus melebarkan ukuran *dropdown* combo box pada lembar Form. Caranya, tekan **Ctrl** pada keyboard kemudian tahan lalu klik tanda , pada kontrol combo box. Setelah itu lepaskan lalu tarik garis blok pada combo box dengan menggunakan mouse aturlah ukuran sesuai dengan kebutuhan yang diperlukan, lihat **Gambar 19**

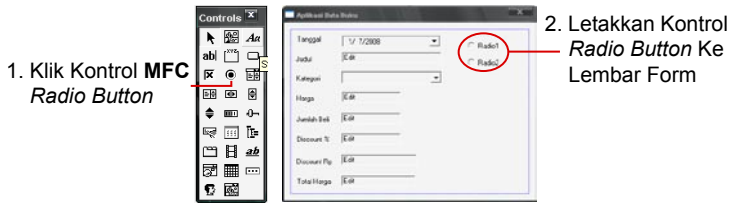
1. Tekan **Ctrl** kemudian  
Klik tanda panah ke  
bawah Pada *Combo Box*



2. Tarik Garis Blok *Combo Box* kemudian  
aturlah sesuai dengan kebutuhan

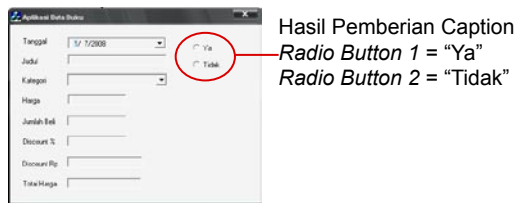
Gambar 19. Mengatur ukuran lembar dropdown

10. Pada toolbar kontrol **MFC** Anda klik kontrol *Radio Button*, kemudian Anda letakkan pada lembar **Form** dengan mengklik lembar Form, kemudian tahan dan drag sesuai ukuran, letakkan sebanyak 2 buah pada lembar Form, lihat **Gambar 20**




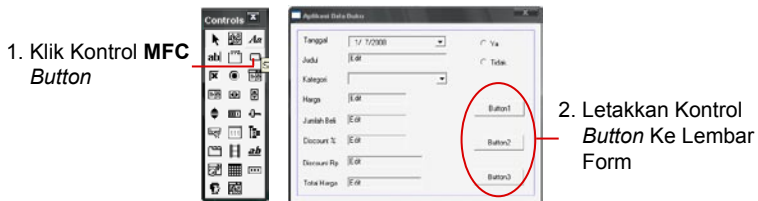
Gambar 20. Meletakkan Kontrol MFC Radio Button

11. Kemudian beri caption pada 2 buah kontrol MFC *Radio Button* yang melekat pada lembar Form. Caranya klik kanan pada kontrol MFC *Radio Button* kemudian pilih **Properties**, maka akan muncul kotak dialog **Text Properties**. Kemudian pada kolom caption kotak dialog **Text Properties** Anda ketikkan nama kontrol *Radio Button* tersebut, lihat **Gambar 21**



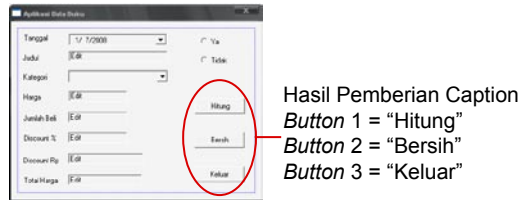
Gambar 21. Hasil pemberian caption pada control MFC Radio Button

12. Pada toolbar kontrol **MFC** Anda klik kontrol *Button* , kemudian Anda letakkan pada lembar **Form** dengan mengklik lembar Form, kemudian tahan dan drag sesuai ukuran, letakkan sebanyak 3 buah pada lembar Form, lihat **Gambar 22**




Gambar 22. Meletakkan kontrol MFC Button

13. Kemudian beri caption pada 2 buah kontrol MFC *Button* yang melekat pada lembar Form. Caranya klik kanan pada kontrol MFC *Button* kemudian pilih **Properties**, maka akan muncul kotak dialog **Text Properties**. Kemudian pada kolom Caption kotak dialog **Text Properties** Anda ketikkan nama kontrol *Button* tersebut, lihat **Gambar 23**



Gambar 23. Hasil pemberian caption pada kontrol MFC Button

14. Pada toolbar kontrol MFC Anda klik kontrol *Group Box* , kemudian Anda letakkan pada lembar **Form** dengan mengklik lembar Form, kemudian tahan dan drag sesuai ukuran, letakkan sebanyak 2 buah pada lembar Form, lihat **Gambar 24**



Gambar 24. Meletakkan kontrol MFC Group Box

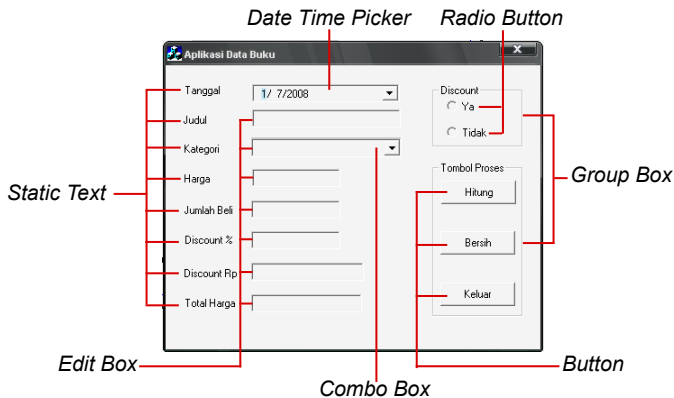
15. Kemudian beri caption pada 2 buah kontrol MFC *Group Box* yang melekat pada lembar Form. Caranya klik kanan pada kontrol MFC *Group Box* kemudian pilih **Properties**, maka akan muncul kotak dialog **Text Properties**. Kemudian pada kolom Caption kotak dialog **Text Properties** Anda ketikkan nama kontrol *Group Box* tersebut, lihat **Gambar 25**



Hasil Pemberian Caption  
Group Box 1 = "Discount"  
Group Box 2 = "Tombol Proses"

Gambar 25. Hasil pemberian caption pada kontrol MFC Group Box

16. Sampai disini Anda telah selesai mendesain Form program Anda, hasil akhir desain Form program Anda dapat dilihat pada gambar dibawah ini, lihat **Gambar 26**



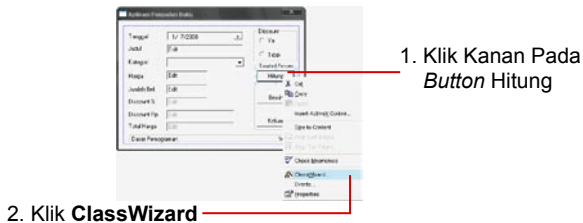
Gambar 26. Tampilan akhir desain program

## B. Membuat Kode Atau Listing Program

Setelah Anda selesai dengan mendesain tampilan **Form**, selanjutnya anda akan disibukkan lagi dengan pemberian kode pada masing - masing kontrol yang terempel pada lembar Form sesuai dengan perintah yang diinginkan, untuk pemberian kode Ada beberapa kontrol yang harus diberikan nama variabel member terlebih dahulu, agar kode yang kita buat dapat dikenal oleh

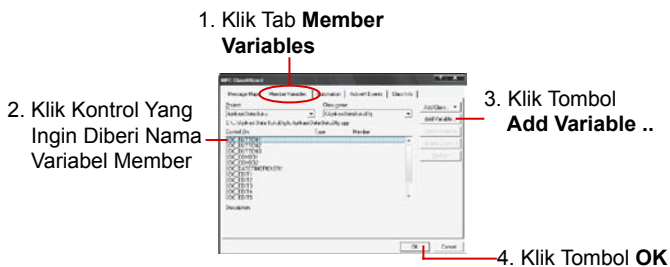
bahasa pemrograman Visual C++ 6.0, untuk itu ikuti langkah - langkah berikut ini.

1. Klik kanan pada kontrol yang ingin diberi nama variabel member kemudian klik **ClassWizard**. Sebagai contoh saya akan memberikan nama variabel member pada kontrol *Button* dengan nama Caption “Hitung”, lihat **Gambar 27**



*Gambar 27. Membuka kotak dialog MFC AppWizard*

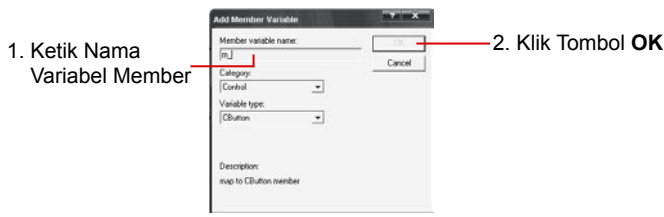
2. Pada kotak dialog **MFC AppWizard** klik Tab **Member Variables**, kemudian klik nama kontrol yang ingin diberi nama variabel kemudian klik tombol **Add Variable**, lihat **Gambar 28**



*Gambar 28. Membuka kotak dialog Add member Variable*

3. Pada kotak dialog **Add member Variable** ketikkan nama variabel yang diinginkan pada kolom *Member Variable Name* kemudian klik **OK**, lihat **Gambar 29**





Gambar 39. Memberikan nama variabel member pada kontrol

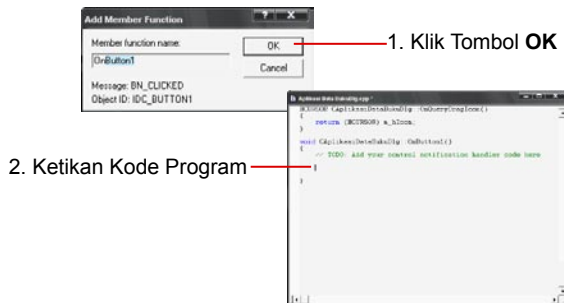
4. Sampai disini pemberian nama variabel member untuk kontrol *Button* dengan nama caption “Hitung” telah selesai. Silakan Anda berikan nama variabel member pada kontrol yang lain yang dianggap perlu dengan mengikuti langkah - langkah seperti diatas, dan dibawah ini sebagai contoh akan saya tampilkan berupa tabel yang akan memberikan informasi mengenai kontrol - kontrol apa saja yang perlu diberikan nama variabel member, lihat **Tabel 4**

Tabel 4. Tabel Nama variabel member

ID Kontrol	Type Kontrol	Variabel	Caption	Type
IDC_BUTTON1	BUTTON	m_Hitung	Hitung	CButton
IDC_BUTTON2	BUTTON	m_Bersih	Bersih	CButton
IDC_BUTTON3	BUTTON	m_Keluar	Keluar	CButton
IDC_COMBO1	COMBO BOX	m_Ktg	-	CString
IDC_DATE TIMEPICKER1	DATE TIME PICKER	m_Tgl	-	CTime
IDC_EDIT1	EDIT BOX	m_Judul	-	CString
IDC_EDIT2	EDIT BOX	m_Harga	-	CString
IDC_EDIT3	EDIT BOX	m_Jumlah	-	CString

ID Kontrol	Type Kontrol	Variabel	Caption	Type
IDC_EDIT4	EDIT BOX	m_Dprsn	-	CString
IDC_EDIT5	EDIT BOX	m_Dhrg	-	CString
IDC_EDIT6	EDIT BOX	m_Total	-	CString

- Sebagai contoh Anda akan memberikan kode pada tombol button dengan caption “Hitung” *double* klik pada kontrol *Button* Hitung. Kemudian klik tombol **OK** pada kotak dialog **Add Member Function**, maka jendela editor kode pemrograman Visual C++ akan terbuka, lihat **Gambar 40**



Gambar 40. Menampilkan jendela editor kode pemrograman

- Pada Function **OnButton1**, ketikkan kode program dibawah kalimat teks “*// TODO: Add your control notification handler code here*” seperti dibawah ini

```

1  UpdateData(true);
2      if (m_Judul == "")
3      {
4  MessageBox("Input Judul Buku !", "Maxikom");
5      CEdit *code;
6      code=(CEdit*) GetDlgItem(IDC_EDIT1);
7      code->SetFocus();
8  UpdateData(false);
9      goto lanjut;
10     }
11 UpdateData(true);
12     if (m_Ktg == "")
13     {
14 MessageBox("Input Kategori Buku !", "Maxikom");
15     CEdit *code;
16     code=(CEdit*) GetDlgItem(IDC_COMBO1);
17     code->SetFocus();
18 UpdateData(false);
19 goto lanjut;
20     }
21 UpdateData(true);
22     if (m_Harga ==0)
23     {
24 MessageBox("Input Harga Buku !", "Maxikom");
25     CEdit *code;
26     code=(CEdit*) GetDlgItem(IDC_EDIT2);
27     code->SetFocus();
28 UpdateData(false);
29     goto lanjut;
30     }
31 UpdateData(true);
32     if (m_Jumlah ==0)
33     {
34 MessageBox("Input Jumlah Pembelian Buku
35 !", "Maxikom");
36     CEdit *code;
37     code=(CEdit*) GetDlgItem(IDC_EDIT3);
38     code->SetFocus();
39 UpdateData(false);
40     goto lanjut;
41     }

```

```

41         UpdateData(true);
42         if (m_Dprsn ==0)
43         {
44 CButton* pChk1=(CButton*)GetDlgItem(IDC_RADIO2);
45 pChk1->SetCheck(true);
46 UpdateData(false);
47         }
48 UpdateData(true)
49 double jumlah, harga, potongan, total1, total2,
total3;
50     jumlah = (m_Jumlah);
51     harga = (m_Harga);
52     potongan = (m_Dprsn);
53     total1 = jumlah * harga * potongan / 100;
54     total2 = jumlah * harga;
55     total3 = total2 - total1;
56     m_Dhrg.Format("Rp %.0f", total1);
57     m_Total.Format("Rp %.0f", total3);
58 lanjut:
59     UpdateData(false);
60 }

```

#### Keterangan Kode

- Baris 1 - 10 Merupakan fungsi pencabangan *if* yang apabila pada kolom judul tidak terisi text atau kosong, maka pesan akan tampil yang memerintahkan untuk mengisi kolom tersebut dan kursor akan fokus ke kolom judul dan apabila kolom tersebut terisi maka perintah pencabangan *goto* untuk ke baris *lanjut* akan dieksekusi.
- Baris 11 - 20 Merupakan fungsi pencabangan *if* yang apabila pada kolom kategori combo box tidak terisi pilihan atau kosong, maka pesan akan tampil yang memerintahkan untuk mengisi kolom pilihan tersebut dan kursor akan fokus ke kolom kategori combo box dan apabila kolom tersebut terisi maka perintah pencabangan *goto* untuk ke baris *lanjut* akan dieksekusi..

**Keterangan Kode**

- Baris 21 - 30 Merupakan fungsi pencabangan *if* yang apabila pada kolom Harga tidak terisi teks atau kosong, maka pesan akan tampil yang memerintahkan untuk mengisi kolom Harga tersebut dan kursor akan fokus ke kolom Harga dan apabila kolom tersebut terisi maka perintah pencabangan *goto* untuk ke baris *lanjut* akan dieksekusi.
- Baris 31 - 40 Merupakan fungsi pencabangan *if* yang apabila pada kolom Jumlah tidak terisi teks atau kosong, maka pesan akan tampil yang memerintahkan untuk mengisi kolom Jumlah tersebut dan kursor akan fokus ke kolom Jumlah dan apabila kolom tersebut terisi maka perintah pencabangan *goto* untuk ke baris *lanjut* akan dieksekusi.
- Baris 41 - 47 Merupakan fungsi pencabangan *if* yang apabila pada kolom diskon % berisi nilai “0”, maka pada pilihan radio button Diskon “Tidak” akan bernilai “True”
- Baris 48 - 49 Merupakan fungsi untuk membangun variabel dengan nama jumlah, harga, potongan, total1, total2, total3 dan memberikan nilai *double* pada variabel tersebut
- Baris 50 - 52 Merupakan fungsi untuk mendefinisikan variabel yang telah dibangun dengan kontrol yang telah diberi nama variabel member yang tertempel pada lembar Form.
- Baris 53 - 55 Merupakan fungsi Aritmatika yang akan menghitung transaksi yang terjadi dengan perantara variabel yang telah didefinisikan terlebih dahulu dengan kontrol yang sudah diberi nama variabel member yang melekat pada Form
- Baris 56 Merupakan fungsi untuk menampilkan hasil dari perhitungan variabel *Total1* ke kolom Diskon harga. dan memberikan format mata uang “Rp” pada kolom Diskon Harga.
- Baris 57 Merupakan fungsi untuk menampilkan hasil dari perhitungan variabel *Total3* ke kolom Total harga. dan memberikan format mata uang “Rp” pada kolom Total Harga.

**Keterangan Kode**

- Baris 58 - 59 Merupakan fungsi untuk mendukung pencabangan *goto* yang memerintahkan untuk ke baris “*lanjut*” dan mengakhiri perintah dari tombol button Hitung serta akhir dari perintah Function **OnButton1**.

7. Pada Function **OnButton2**, ketikkan kode program dibawah kalimat teks “*// TODO: Add your control notification handler code here*” seperti dibawah ini

```
1 UpdateData(true);
2     this->m_Jumlah =0;
3     this->m_Dhrp ="";
4     this->m_Dprsn =0;
5     this->m_Harga =0;
6     this->m_Judul ="";
7     this->m_Ktg="";
8     this->m_Total ="";
9         CEdit *code;
10         code=(CEdit*)GetDlgItem(IDC_EDIT1);
11         code->SetFocus();
12 UpdateData(false);
```

**Keterangan Kode**

- Baris 1 Merupakan fungsi untuk membaca perintah eksekusi pada tombol *Button* dengan caption Bersih
- Baris 2 Merupakan fungsi untuk mengisi kolom *Edit Box* Jumlah dengan angka “0”
- Baris 3 Merupakan fungsi untuk mengosongkan kolom *Edit Box* Discount Rp
- Baris 4 Merupakan fungsi untuk mengisi kolom *Edit Box* Discount % dengan angka “0”
- Baris 5 Merupakan fungsi untuk mengisi kolom *Edit Box* Harga dengan angka “0”

**Keterangan Kode**

- Baris 6 Merupakan fungsi untuk mengosongkan kolom kontrol *Edit Box* Judul
- Baris 7 Merupakan fungsi untuk mengosongkan kolom pilihan kategori pada *Combo Box*
- Baris 8 Merupakan fungsi untuk mengosongkan kolom kontrol *Edit Box* Total Harga
- Baris 9 - 11 Merupakan fungsi untuk menempatkan kursor pada kolom kontrol *Edit Box* Judul secara otomatis
- Baris 12 Merupakan fungsi untuk menutup perintah eksekusi pada tombol *Button* dengan caption “Bersih”- Baris 8 Merupakan fungsi untuk mengosongkan kolom kontrol *Edit Box* Total Harga
- Baris 9 - 11 Merupakan fungsi untuk menempatkan kursor pada kolom kontrol *Edit Box* Judul secara otomatis

8. Pada Function **OnButton3**, ketikkan kode program dibawah kaimat teks “*// TODO: Add your control notification handler code here*” seperti dibawah ini

```
1 CDialog::OnOK ();
```

**Keterangan Kode**

- Baris 1 Merupakan fungsi untuk mengakhiri jalannya program yang Anda buat atau keluar dari program yang Anda buat.

9. Pada Function **OnRadio1**, ketikkan kode program dibawah kalimat teks “*// TODO: Add your control notification handler code here*” seperti dibawah ini

```

1 CButton *bDSR;
2 bDSR=(CButton*) GetDlgItem(IDC_EDIT4);
3 bDSR->EnableWindow(true);
4 CEdit *code;
5 code=(CEdit*) GetDlgItem(IDC_EDIT4;
6 code->SetFocus();

```

#### Keterangan Kode

- Baris 1 - 3 Merupakan fungsi untuk mengaktifkan kolom *Edit Box* Discount % apabila pilihan Discount pada *Radio Button* dengan Caption “Ya” di pilih
- Baris 4 - 6 Merupakan fungsi untuk menempatkan kursor pada kolom *Edit Box* Discount % secara otomatis apabila pilihan Discount pada *Radio Button* dengan Caption “Ya” Dipilih

10. Pada Function **OnRadio2**, ketikkan kode program dibawah kalimat teks “ *// TODO: Add your control notification handler code here*” seperti dibawah ini

```

1 UpdateData(true);
2     CButton *bDSR;
3     bDSR=(CButton*) GetDlgItem(IDC_EDIT4);
4     bDSR->EnableWindow(false);
5     this -> m_Dprsn =0;
6 double jumlah, harga, potongan, total1, total2,
   total3;
7     jumlah = (m_Jumlah);
8     harga = (m_Harga);
9     potongan = (m_Dprsn);
10    total1 = jumlah * harga * potongan / 100;
11    total2 = jumlah * harga;
12    total3 = total2 - total1;
13    m_Dhrp.Format("Rp %.0f", total1);
14    m_Total.Format("Rp %.0f", total3);
15 UpdateData(false);

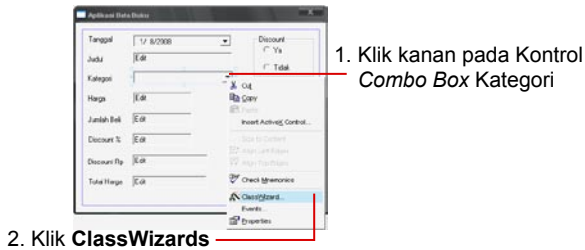
```



**Keterangan Kode**

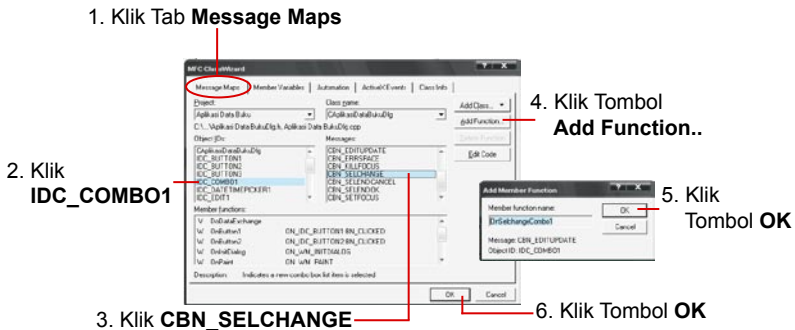
- Baris 1 Merupakan fungsi untuk membaca perintah eksekusi pada kontrol *Radio Button* dengan caption “Tidak”
- Baris 2 - 5 Merupakan fungsi untuk menonaktifkan kolom *Edit Box* Discount % apabila pilihan Discount pada *Radio Button* dengan Caption “Tidak” di pilih, dan mengisi kolom *Edit Box* Discount % dengan angka “0”
- Baris 6 Merupakan fungsi untuk membangun variabel dengan nama jumlah, harga, potongan, total1, total2, total3 dan memberikan nilai *double* pada variabel tersebut
- Baris 7 - 9 Merupakan fungsi untuk mendefinisikan variabel yang telah dibangun dengan kontrol yang telah diberi nama variabel member yang tertempel pada lembar Form.
- Baris 10 - 12 Merupakan fungsi Aritmatika yang akan menghitung transaksi yang terjadi dengan perantara variabel yang telah didefinisikan terlebih dahulu dengan kontrol yang sudah diberi nama variabel member yang melekat pada Form
- Baris 13 Merupakan fungsi untuk menampilkan hasil dari perhitungan variabel *Total1* ke kolom Diskon harga. dan memberikan format mata uang “Rp” pada kolom Diskon Harga.
- Baris 14 Merupakan fungsi untuk menampilkan hasil dari perhitungan variabel *Total3* ke kolom Total harga. dan memberikan format mata uang “Rp” pada kolom Total Harga.
- Baris 15 Merupakan fungsi untuk menutup perintah eksekusi pada kontrol *Radio Button* dengan caption “Tidak”

11. Khusus kontrol *Combo Box* Kategori, Anda harus membuat Function baru yang berfungsi apabila pada pilihan item *Combo Box* tersebut dipilih maka kode yang terdapat didalam Function tersebut akan langsung dieksekusi. oleh karena itu Anda harus membuat Function baru dengan nama **CBN\_SELCHANGE**. Caranya, klik kanan pada kontrol *Combo Box* kemudian pilih **ClassWizards** , lihat **Gambar 41**



Gambar 41. Membuka kotak dialog MFC AppWizards

12. Pada kotak dialog **MFC AppWizard** klik Tab **Message Maps**, kemudian klik nama kontrol **IDC\_COMBO1** yang ingin diberi nama Function baru. lalu pada list Messages Anda klik Nama **CBN\_SELCHANGE**. Selanjutnya Anda klik tombol **Add Function** dan klik tombol **OK** pada kotak dialog **Add Member Function**. Dan terakhir Anda klik tombol **OK** pada kotak dialog **MFC AppWizards**, lihat **Gambar 42**



Gambar 42. Membuat function baru

13. Kemudian pada Function **OnSelChangeCombo1** yang telah Anda buat tadi, ketikkan kode program dibawah kalimat teks “*// TODO: Add your control notification handler code here*” seperti dibawah ini

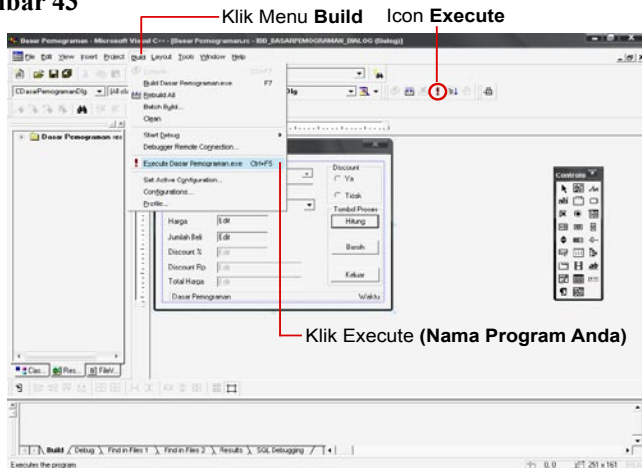
```
1 UpdateData(true);  
2 CEdit *code;  
3 code=(CEdit*)GetDlgItem(IDC_EDIT2);  
4 code->SetFocus();  
5 UpdateData(false);
```

### Keterangan Kode

- Baris 1 Merupakan fungsi untuk membaca perintah eksekusi pada kontrol *Combo Box* Kategori
- Baris 2 - 4 Merupakan fungsi untuk menempatkan kursor secara otomatis pada kolom *Edit Box* Harga
- Baris 5 Merupakan fungsi untuk menutup perintah eksekusi pada kontrol *Combo Box* Kategori

## C. Menjalankan Program

1. Sekarang Anda coba jalankan program yang telah Anda buat dengan menekan Icon **Execute** ! , atau Anda tekan **F5** pada keyboard atau bisa juga Anda klik menu **Build > Execute (Nama Program Anda)**, lihat **Gambar 43**



Gambar 43. Menjalankan program

13. Setelah program dijalankan sekarang Anda coba lakukan transaksi pembelian buku, dan coba Anda klik tombol *Button* dengan Caption “Hitung”. Dan lihat hasilnya. Sebagai Contoh Anda lihat gambar di bawah ini, lihat **Gambar 44**

The screenshot shows a Windows application window titled "Aplikasi Penjualan Buku". The window contains a form with the following fields and controls:

- Tanggal:** A date picker showing "1/17/2008".
- Judul:** A text box containing "Manipulasi Foto".
- Kategori:** A dropdown menu showing "Design Grafis".
- Harga:** A text box containing "45000".
- Jumlah Beli:** A text box containing "25".
- Discount %:** A text box containing "15".
- Discount Rp:** A text box containing "Rp 168750".
- Total Harga:** A text box containing "Rp 956250".
- Discount:** A group box containing two radio buttons: "Ya" (selected) and "Tidak" (unselected).
- Tombol Proses:** A group box containing three buttons: "Hitung", "Bersih", and "Keluar".

*Gambar 44. Tampilan program saat dijalankan*