

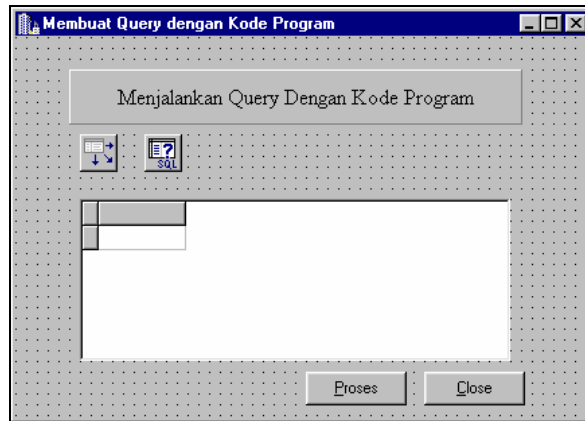
Pemrograman dengan C++ Builder

13.1. Pernyataan SQL

Ada empat buah pernyataan SQL yaitu SELECT, UPDATE, DELETE, INSERT. Perintah Select adalah perintah SQL yang sering digunakan, perintah ini akan memberikan intruksi pada engine SQL agar menyaring field-field tertentu dari tabel yang dipilih.

Berikut ini adalah contoh menggunakan kode program C++ Builder untuk mengaktifkan dan membentuk sebuah Query.

Buat Form seperti berikut:



Gambar 13.1. Desain Form Program Aplikasi

Masukan Properti berikut:

Tabel 13.1. Mengganti Properti Objek

Komponen	Properti	Nilai
Button1	Caption	Proses
Button2	Caption	Close
Query1	DatabaseName	BCBDefaultDD
	Active	False
DataSource1	DataSet	Query1
DBGrid1	DataSource	DataSource1

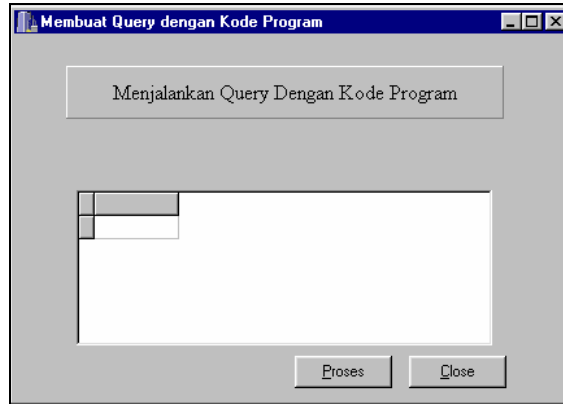
Pada tombol Proses, masukan kode berikut:

```
void __fastcall TForm1::Button1Click(TObject *Sender)
{
    Query1->Close();
    Query1->SQL->Clear();
    Query1->SQL->Add("Select * From Transaksi");
    Query1->Open();
}
```

Pada tombol Close, masukan kode berikut:

```
void __fastcall TForm1::Button2Click(TObject *Sender)
{
    Close();
}
```

Jika Anda Run Program tersebut, maka akan menghasilkan output seperti berikut:



Gambar 13.2. Tampilan Awal Program Aplikasi

Kemudian tekan tombol proses, maka akan didapatkan output seperti berikut:



Gambar 13.3. Output Program Aplikasi

13.2. Perintah Matematis

Di dalam SQL ada beberapa perintah matematis (Agregat) yang terdiri dari lima anggota, yaitu SUM, AVG, MIN, MAX dan COUNT. Empat perintah pertama digunakan untuk menghitung nilai dalam tabel sedangkan perintah terakhir digunakan untuk menghitung jumlah record.

13.2.1 Perintah SUM

Perintah SUM digunakan untuk melakukan proses penjumlahan dari sebuah field, misalnya dibawah ini akan diberikan contoh menghitung berapa total transaksi yang dilakukan oleh seorang pelanggan.

Buat Form seperti berikut:

No_faktur	Kode_plg	Kode_produk	Nama_barang	Har
1	111	B111	TINTA HP	\$280
2	112	B112	LASER 4L	2,500
3	113	B113	HARDDISK QUANTUM 4	\$500
4	111	B345	EPSON LX 300	1,250
5	111	B543	MONITOR LG 440SI	\$99
6	112	B765	PROCESSOR PII-450	1,250

Gambar 13.4. Desain Form Perintah SUM

2. Masukan Properti berikut

Tabel 13.2. Mengganti Properti Objek

Komponen	Properti	Nilai
Button1	Caption	Proses
Button2	Caption	Close
Query1	DatabaseName Active	BCBDefaultDD False
DataSource1	DataSet	Table1
Table1	DataBaseName TableName Active	BCBDefaultDD transaksi.DB True
DBGrid1	DataSource	DataSource1
DataSource2	DataSet	Query1

3. Pada Tombol Proses masukan kode berikut

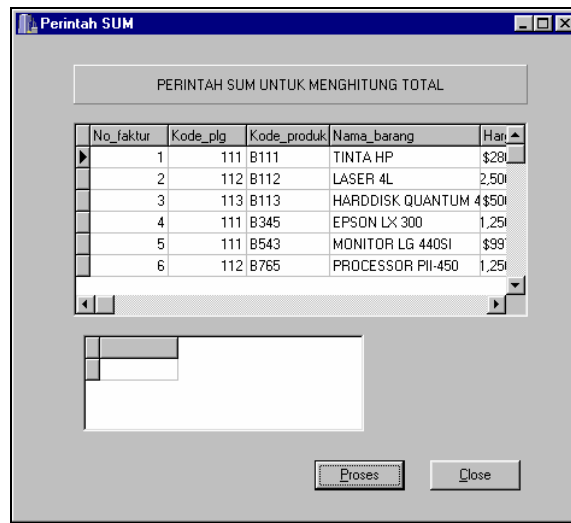
```
void __fastcall TForm1::Button1Click(TObject *Sender)
{
    Query1->Close();
    Query1->SQL->Clear();
    Query1->SQL->Add("Select Kode_plg, SUM(Harga) as Total From Transaksi Group By Kode_plg");
    Query1->Open();
}
```

4. Pada tombol Close, masukan kode berikut:

```
void __fastcall TForm1::Button2Click(TObject *Sender)
{
    Close();
}
```

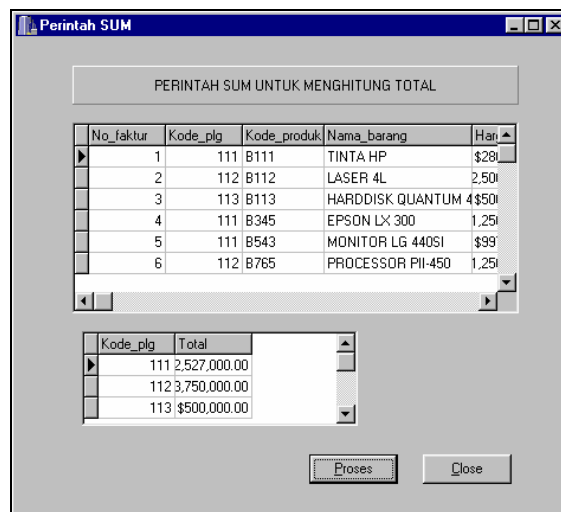
Pemrograman dengan C++ Builder

5. Jika Anda Run, maka akan ditampilkan output seperti berikut:



Gambar 13.5. Tampilan Awal Program Aplikasi

Jika Anda menekan tombol Proses, maka hasil penghitungan dengan menggunakan SUM akan ditampilkan dibawahnya.



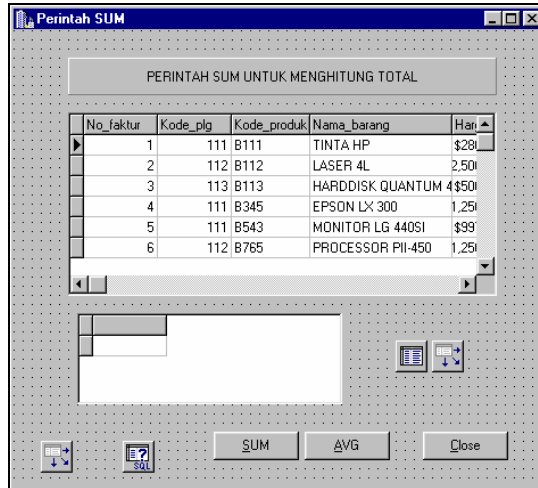
Gambar 13.6. Output Perintah SUM

13.2.2 Perintah AVG

Perintah AVG digunakan untuk mencari berapa nilai rata-rata dari sekelompok data yang ada dalam tabel.

Berikut adalah contoh untuk mencari berapa rata-rata transaksi yang dilakukan oleh seseorang yang mempunyai kode pelanggan tertentu.

Dari Form diatas, tambahkan satu tombol lagi

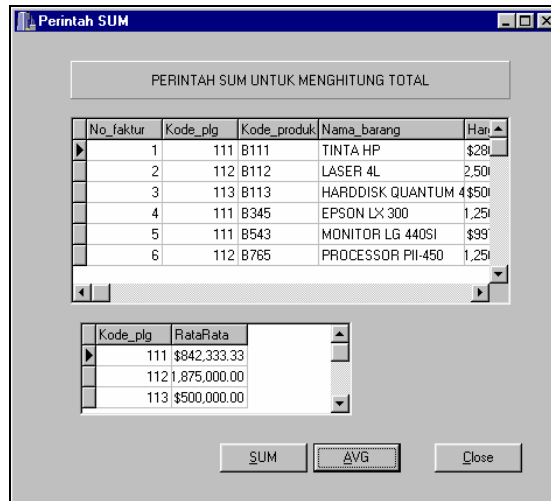


Gambar 13.7. Desain Form Perintah AVG

Pada tombol AVG masukan kode berikut:

```
void __fastcall TForm1::Button3Click(TObject *Sender)
{
    Query1->Close();
    Query1->SQL->Clear();
    Query1->SQL->Add("Select Kode_plg, AVG(Harga) as RataRata From
    Transaksi Group By Kode_plg");
    Query1->Open();
}
```

Jika Anda RUN, kemudian anda tekan tombol AVG maka akan ditampilkan rata-rata transaksi yang dilakukan oleh seseorang



Gambar 13.8. Output Perintah AVG

13.2.3 Perintah MIN

Perintah MIN digunakan untuk mencari nilai minimal dari sekelompok data yang ada dalam tabel.

Berikut adalah contoh untuk mencari berapa minimum transaksi yang dilakukan oleh seseorang yang mempunyai kode pelanggan tertentu.

Pemrograman dengan C++ Builder

Dari Form diatas, tambahkan satu tombol lagi.

The screenshot shows a Windows-style application window titled "Perintah SUM". Inside, there's a header "PERINTAH SUM UNTUK MENGHITUNG TOTAL". Below it is a table with 5 columns: "No_faktur", "Kode_plg", "Kode_produk", "Nama_barang", and "Harga". The table contains 6 rows of data. Below the table is a scroll bar. At the bottom of the form are four buttons: "SUM", "AVG", "MIN", and "Close".

No_faktur	Kode_plg	Kode_produk	Nama_barang	Harga
1	111	B111	TINTA HP	\$280
2	112	B112	LASER 4L	2,500
3	113	B113	HARDDISK QUANTUM 4	\$500
4	111	B345	EPSON LX 300	1,250
5	111	B543	MONITOR LG 440SI	\$99
6	112	B765	PROCESSOR PII-450	1,250

Gambar 13.9. Desain Form Perintah MIN

Masukan kode berikut pada tombol MIN

```
void __fastcall TForm1::Button4Click(TObject *Sender)
{
    Query1->Close();
    Query1->SQL->Clear();
    Query1->SQL->Add("Select Kode_plg, MIN(Harga) as RataRata From
Transaksi Group By Kode_plg");
    Query1->Open();
}
```

Jika Anda RUN, kemudian anda tekan tombol MIN maka akan ditampilkan minimal transaksi yang dilakukan oleh seseorang.

The screenshot shows the same "Perintah SUM" form, but now with an additional table below the main transaction table. This new table has two columns: "Kode_plg" and "Minimum". It contains three rows of data, showing the minimum price for each product code. The buttons at the bottom now include "SUM", "AVG", "MIN" (which is highlighted), "MAX", "COUNT", and "Close".

Kode_plg	Minimum
111	\$280,000.00
112	1,250,000.00
113	\$500,000.00

Gambar 13.10. Output Perintah MIN

13.2.4. Perintah MAX

Perintah MAX digunakan untuk mencari nilai maximal dari sekelompok data yang ada dalam tabel.

Berikut adalah contoh untuk mencari berapa maximal transaksi yang dilakukan oleh seseorang yang mempunyai kode pelanggan tertentu.

Dari Form diatas, tambahkan dua tombol lagi.

The screenshot shows a Windows-style application window titled "Perintah SUM". Inside, there's a header "PERINTAH SUM UNTUK MENGHITUNG TOTAL". Below it is a table with 5 columns: "No_faktur", "Kode_plg", "Kode_produk", "Nama_barang", and "Harga". The table contains 6 rows of data. Below the table is a scrollable area. At the bottom of the form are six buttons: "SUM", "AVG", "MIN", "MAX", "COUNT", and "Close".

No_faktur	Kode_plg	Kode_produk	Nama_barang	Harga
1	111	B111	TINTA HP	\$280
2	112	B112	LASER 4L	2,500
3	113	B113	HARDDISK QUANTUM 4	\$500
4	111	B345	EPSON LX 300	1,250
5	111	B543	MONITOR LG 440SI	\$99
6	112	B765	PROCESSOR PII-450	1,250

Gambar 13.11. Desain Form Perintah MAX

Pada tombol MAX, masukan kode berikut:

```
void __fastcall TForm1::Button5Click(TObject *Sender)
{
    Query1->Close();
    Query1->SQL->Clear();
    Query1->SQL->Add("Select Kode_plg, MAX(Harga) as Maximum From
    Transaksi Group By Kode_plg");
    Query1->Open();
}
```

Jika Anda RUN kemudian Anda tekan tombol MAX, maka akan ditampilkan nilai transaksi maximal yang dilakukan oleh seorang pelanggan.

The screenshot shows the same "Perintah SUM" form, but now with an additional table below the main one. This table has two columns: "Kode_plg" and "Maximum". It displays the maximum price for each customer code from the previous table. The buttons "SUM", "AVG", "MIN", "MAX", "COUNT", and "Close" are still at the bottom.

Kode_plg	Maximum
111	1,250,000.00
112	2,500,000.00
113	\$500,000.00

Gambar 13.12. Output Perintah MAX

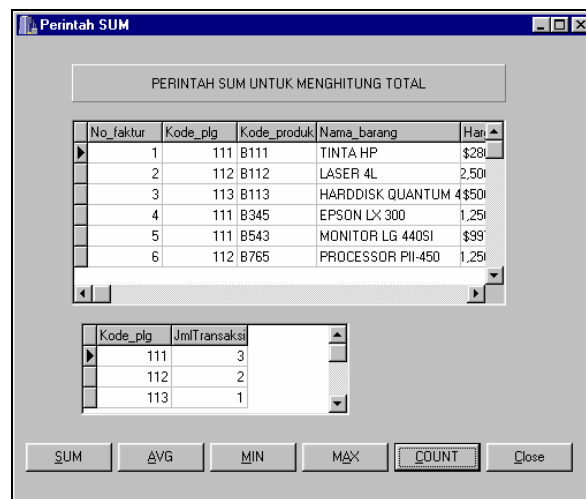
13.2.5. Perintah COUNT

Perintah COUNT digunakan untuk menghitung jumlah data dari sekelompok data yang ada dalam tabel.

Berikut adalah contoh untuk mencari berapa banyak transaksi yang dilakukan oleh seseorang yang mempunyai kode pelanggan tertentu.

Masukan Kode berikut pada tombol COUNT.

```
void __fastcall TForm1::Button6Click(TObject *Sender)
{
    Query1->Close();
    Query1->SQL->Clear();
    Query1->SQL->Add("Select Kode_plg, COUNT(Harga) as JmlTransaksi
    From Transaksi Group By Kode_plg");
    Query1->Open();
}
```



Gambar 13.13. Output Perintah COUNT