Minggu ke : 3
Penyusun : Dian Usdiyana
Rini Marwati
Materi : 3. Tipe Data String dan Format Data
3.1 Tipe Data String
3.2 Format Data

URAIAN POKOK PERKULIAHAN

Tipe data string

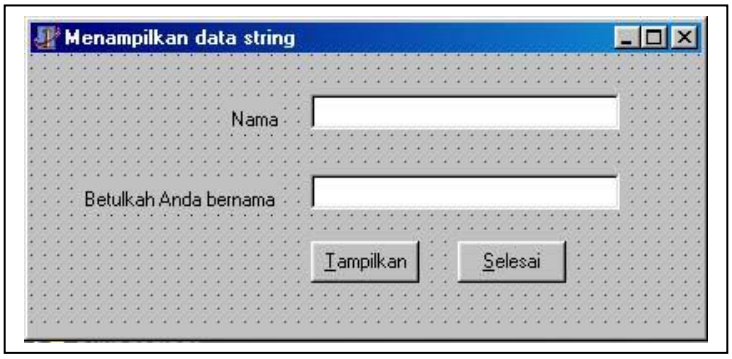
Salah satu tipe data yang sering digunakan dalam pemrograman adalah string. Tipe ini digunakan untuk menyatakan sederet karakter, baik berupa huruf, angka, tanda baca/operasi, dan spasi kosong.

Delphi memiliki tiga macam string, yaitu ShortString, WideString, dan AnsiString. Isi masing-masing string tersebut secara berurutan adalah 256, 2^{30}, dan 2^{31} karakter. Pada saat digunakan Anda dapat menyatakan tipe data sebagai String, namun pada saat dikompilasi tipe ini akan langsung dianggap sebagai AnsiString. Untuk mengubahnya menjadi tipe yang lain dapat dilakukan dengan menggunakan kompiler direktif berrikut:

1. {$H-} untuk mengaktifkn tipe data ShortString
2. {$H+} untuk mengaktifkan tipe data WideString

Berikut adalah salah satu contoh program yang menggunakan tipe data string. Pada program ini Anda diminta memasukkan nama. Kemudian, nama tersebut akan ditampilkan kembali sebagai konfirmasi.

Proyek 3.1:
Buatlah proyek untuk membaca data string, seperti pada Gambar 5.1



Gambar 3.1 Tampilan form baca string

Prosedur untuk membuatnya lakukanlah langkah berikut ini:

1. Buat form baru, misalnya dengan menggunakan menu File-New Application, kemudian ubah propertiesnya menjadi sebagai berikut:

Properties	Isi
Caption	Menampilkan data string
Name	FormString

2. Siapkan masing-masing palette label dan edit yang dibutuhkan. Ubahlah masing-masing propertiesnya menjadi seperti pada tabel 3.1

Tabel 3.1 Beberapa properties untuk program string

Komponen	Properties	Isi
Label1	Caption Name	Nama? LabelNama
Label2	Caption Name	Betulkah Anda bernama LabelTulis
Edit1	Name Text Enable	EditNama Kosongkan True
Edit2	Name Text Enable	TulisNama Kosongkan False

3. Tambahkan pula dua tombol komponen button untuk melaksanakan perintah tampilkan ulang dan mengakhirinya (Tampilkan dan Selesai). Ubahlah propertiesnya seperti pada tabel 3.2

Tabel 3.2 Beberapa properties untuk program string

Komponen	Properties	Isi
Button1	Caption Name	&Tampilkan Tampilkan
Button2	Caption Name	&Selesai Selesai

4. Klik dua kali tombol Selesai, kemudian lengkapi prosedur yang muncul menjadi seperti berikut:

```
procedure TFormString.ButtonSelesaiClick(Sender: TObject);
begin
  Application.Terminate;
end;
```

5. Klik dua kali tombol Tampilkan, kemudian lengkapi prosedur yang muncul menjadi seperti berikut:

```
procedure TFormString.ButtonBacaClick(Sender: TObject);
var
  nama: string;
begin
  nama := EditNama.Text;
  TulisNama.Text:=nama;
end;
```

Pernyataan `nama:=EditNama.Text;` dapat dibaca: Teks yang terdapat di dalam EditNama dinyatakan sebagai `nama`. `EditNama` dapat berfungsi sebagau input data, yang isinya dinyatakan bertipe string,. Sedangkan `TulisNama.Text` adalah suatu output data, yang menampilkan apapun pada `nama`, karena *enable*nya bernilai false. Prosedur tersebut dapat juga dibuat lebih singkat, seperti berikut.

```
procedure TFormString.ButtonBacaClick(Sender: TObject);
begin
  TulisNama.Text:=EditNama.Text;
end;
```

Secara lengkap, program untuk membaca dan menampilkan tipe data string di atas adalah sebagai berikut.

List Program 3.1 Membaca data string

```
unit UnitBaca;
interface
uses
  Windows, Messages, SysUtils, Classes,
  Graphics, Controls, Forms, Dialogs,
  StdCtrls;
type
  TFormBaca = class(TForm)
    EditNama: TEdit;
    TulisNama: TEdit;
    LabelNama: TLabel;
    LabelTulis: TLabel;
    ButtonBaca: TButton;
    ButtonSelesai: TButton;
    procedure ButtonSelesaiClick(Sender: TObject);
    procedure ButtonBacaClick(Sender: TObject);
  private
    { Private declarations }
  public
    { Public declarations }
  end;
var
  FormBaca: TFormBaca;
implementation
{$R *.DFM}
procedure TFormBaca.ButtonSelesaiClick(Sender: TObject);
begin
  Application.Terminate;
end;
procedure TFormBaca.ButtonBacaClick(Sender: TObject);
var
  nama: string;
begin
  nama := EditNama.Text;
  TulisNama.Text:=nama;
end;
end.
```

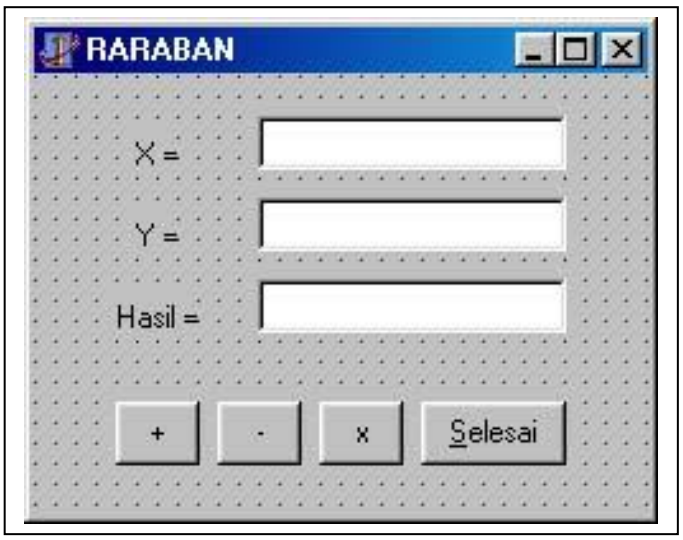
Cobalah jalankan program tersebut, maka akan muncul tampilan seperti pada Gambar 3.2.

Gambar 3.2 Tampilan program baca string

Tuliskan data nama Anda pada jendela edit nama. Kemudian tekan tombol Tampilkan, maka nama Anda akan ditulis ulang pada jendela edit di bawahnya. Jika selesai tekan tombol Selesai.

Untuk lebih memahami bagaimana tipe data ordinal, seperti yang telah dijelaskan di atas, berikut ini disajikan contoh pembuatan proyek untuk membuat kalkulator sederhana yang diberi nama raraban.

Proyek 3.2:
Buatlah kalkulator sederhana yang hanya memiliki operasi hitung tambah, kurang, dan kali dan formatlah hasilnya dengan membatasi banyaknya decimal.. Fasilitas yang terdapat di dalam kalkukator tersebut seperti terlihat pada



Gambar 3.3 Tampilan form kalkulator sederhana

Untuk membuat tampilan seperti Gambar 3.3, lakukanlah langkah-langkah berikut ini:

1. Buat form baru, misalnya dengan menggunakan menu File-New Application, kemudian ubah judulnya melalui properties caption menjadi RARABAN.
2. Tambahkan tiga label dari komponen palette, ubahlah masing-masing propertiesnya menjadi seperti pada Tabel 3.3.

Tabel 3.3 Properties label

Komponen	Properties	Isi
Labe11	Caption Name	X = inputX
Label2	Caption Name	Y = inputY
Label3	Caption Name	Hasil = Hitung

3. Tambahkan tiga edit dari komponen palette, ubahlah masing-masing propertiesnya menjadi seperti pada Tabel 3.4

Tabel 3.4 Properties edit

Komponen	Properties	Isi
Edit1	Text Name Enabled	kosongkan nilaiX True
Edit2	Text Name Enabled	kosongkan nilaiY True
Edit2	Text Name Enabled	kosongkan HasilHitung False

4. Tambahkan empat button dari komponen palette, ubahlah masing-masing propertiesnya menjadi seperti pada Tabel 3.5

Tabel 3.5 Properties button

Komponen	Properties	Isi
Button1	Caption Name	+ tambah
Button2	Caption Name	- kurang
Button3	Caption Name	x kali
Button4	Caption Name	&Selesai Selesai

5. Klik dua kali tombol +, kemudian lengkapi prosedur yang muncul menjadi seperti berikut:

```
procedure TForm1.TambahClick(Sender: TObject);
var
  x,y,z:integer;
begin
  x := StrToInt(NilaiX.Text);
  y := StrToInt(NilaiY.Text);
  z := x+y;
  HasilHitung.Text := IntToStr(z);
end;
```

6. Klik dua kali tombol -, kemudian lengkapi prosedur yang muncul menjadi seperti berikut:

```
procedure TForm1.KurangClick(Sender: TObject);
var
  x,y,z:integer;
begin
  x := StrToInt(NilaiX.Text);
  y := StrToInt(NilaiY.Text);
  z := x-y;
  HasilHitung.Text := IntToStr(z);
end;
```

7. Klik dua kali tombol x, kemudian lengkapi prosedur yang muncul menjadi seperti berikut:

```
procedure TForm1.KaliClick(Sender: TObject);
var
  x,y,z:integer;
begin
  x := StrToInt(NilaiX.Text);
  y := StrToInt(NilaiY.Text);
  z := x*y;
  HasilHitung.Text := IntToStr(z);
end;
```

8. Terakhir klik dua kali tombol Selesai, Lengkapi prosedur yang muncul dengan perintah `Close` atau `Application.Terminate` seperti pada contoh pada bagian 1.

Pada langkah 5 sampai 7 Anda memasukkan deklarasi variabel-variabel dengan bilangan bertipe integer, `var x,y,z:integer;`. Selain itu ditambahkan pula pernyataan `x := StrToInt(NilaiX.Text);`, yang dapat dipahami bahwan Text yang terdapat di dalam nilaiX diubah tipenya dari string menjadi integer dan selanjutnya dinyatakan sebagai x. Untuk y juga dapat diartikan sama dengan x.

Setelah proses memberi nilai kepada masing-masing variabel selesai, kemudian dilakukan proses perhitungan. Untuk operasi tambah dan kurang menggunakan operator + dan -, sedangkan untuk operasi kali menggunakan operator *.

Pernyataan `HasilHitung.Text := IntToStr(z);` dapat diartikan nilai z diubah dari data bertipe integer menjadi string, yang selanjutnya dinyatakan sebagai Text di dalam edit HasilHitung. Pernyataan ini menyebabkan hasil perhitungan ditampilkan pada jendela edit. Secara lengkap, proyek raraban ini terdapat pada List Program 3.2.

List Program 3.2 Aplikasi tipe data integer pada proyek raraban

```
unit UnitRaraban;
interface
uses
  Windows, Messages, SysUtils, Classes, Graphics,
  Controls, Forms, Dialogs,
  StdCtrls;
type
  TForm1 = class(TForm)
    inputX: TLabel;
    inputY: TLabel;
    Hitung: TLabel;
```

```
    nilaiX: TEdit;
    nilaiY: TEdit;
    HasilHitung: TEdit;
    Tambah: TButton;
    Kurang: TButton;
    Kali: TButton;
    Selesai: TButton;
    procedure TambahClick(Sender: TObject);
    procedure KurangClick(Sender: TObject);
    procedure KaliClick(Sender: TObject);
    procedure SelesaiClick(Sender: TObject);
  private
    { Private declarations }
  public
    { Public declarations }
  end;
var
  Form1: TForm1;
implementation
{$R *.DFM}

procedure TForm1.TambahClick(Sender: TObject);
var
  x,y,z:integer;
begin
  x := StrToInt(NilaiX.Text);
  y := StrToInt(NilaiY.Text);
  z := x+y;
  HasilHitung.Text := IntToStr(z);
end;


procedure TForm1.KurangClick(Sender: TObject);
var
  x,y,z:integer;
begin
  x := StrToInt(NilaiX.Text);
  y := StrToInt(NilaiY.Text);
  z := x-y;
  HasilHitung.Text := IntToStr(z);
end;

procedure TForm1.KaliClick(Sender: TObject);
var
  x,y,z:integer;
begin
  x := StrToInt(NilaiX.Text);
  y := StrToInt(NilaiY.Text);
  z := x*y;
  HasilHitung.Text := IntToStr(z);
end;

procedure TForm1.SelesaiClick(Sender: TObject);
begin
  Application.Terminate;
end;
end.
```

Jalankan proyek yang telah Anda buat tersebut, masukkan angka 45 untuk nilai x dan 32 untuk nilai y, kemudian tekan tombol-tombol operasi yang tersedia, maka hasilnya akan segera tampak seperti pada Gambar 3.2.

TUGAS

1. Jalankan program tersebut dengan memasukkan angka-angka, apakah yang terjadi?
2. Buatlah program (input dan output) data mahasiswa:
   Nama:
   NIM:
   Prodi:
   Jendela edit input ditempatkan di sebelah kiri, sedangkan jendela edit output ditempatkan di sebelah kanan.
3. Tambahkan operasi aritmetika lainnya pada proyek kalkulator, seperti pembagian, akar, logaritma, dan sebagainya