

Praktikum IV

Pemrograman Visual

Pernyataan Berkondisi / Percabangan

A. Pendahuluan

Pernyataan berkondisi digunakan untuk mengambil keputusan yang melibatkan dua alternatif atau lebih. Misalkan untuk menentukan nilai yang diperoleh seseorang.


Ada dua jenis pernyataan berkondisi yang ada dalam Delphi, yaitu :

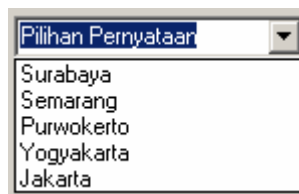
1. Percabangan menggunakan pernyataan IF
2. Percabangan menggunakan pernyataan CASE

B. Pernyataan Kondisi menggunakan Komponen ComboBox, RadioButton, RadioGroup dan CheckBox


Beberapa *Visual Component Library* (VCL) pada Borland Delphi dapat digunakan untuk menangani pernyataan berkondisi dari beberapa pilihan yang disimpan pada komponen tersebut. Komponen-konponen tersebut antara lain :

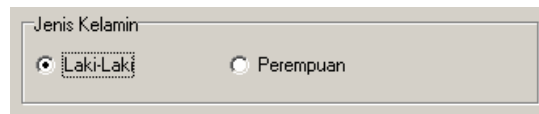
1. ComboBox

ComboBox  digunakan untuk memilih salah satu di antara sejumlah pilihan. Komponen menggabungkan sifat kotak edit dan ListBox. Daftar pilihan disimpan pada properti **items**.



2. RadioButton

RadioButton  biasa digunakan untuk menangani pilihan dan hanya satu pilihan yang bisa dipilih.

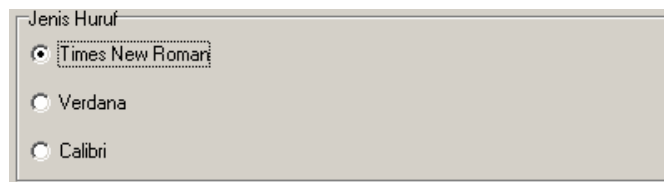


Komponen ini akan membangkitkan properti *checked=true* apabila radio button dalam keadaan terpilih.

3. RadioGroup



RadioGroup merupakan alternatif untuk membuat sejumlah tombol radio (RadioButton). Komponen ini mampu menciptakan sejumlah RadioButton dengan mudah.



Beberapa properti yang terdapat pada komponen ini adalah :

- a. Caption : judul kotak RadioGroup.
- b. Column : menentukan pengaturan jumlah kolom RadioButton. RadioButton tersusun secara horisontal.
- c. ItemIndex : menentukan urutan RadioButton pada kondisi awal yang terpilih. Index dimulai dari 0, atau -1 jika tidak ada RadioButton yang terpilih pada kondisi awal.
- d. Items : Daftar pernyataan pilihan untuk masing-masing RadioButton.

4. CheckBox




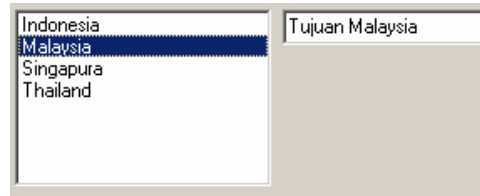
Sama halnya dengan RadioButton, CheckBox biasa digunakan untuk menangani pilihan dan hanya satu pilihan yang bisa dipilih.



Komponen ini akan membangkitkan properti *checked=true* apabila radio button dalam keadaan terpilih.

5. ListBox



ListBox  merupakan komponen yang berguna untuk melakukan pilihan dari sejumlah daftar pilihan.



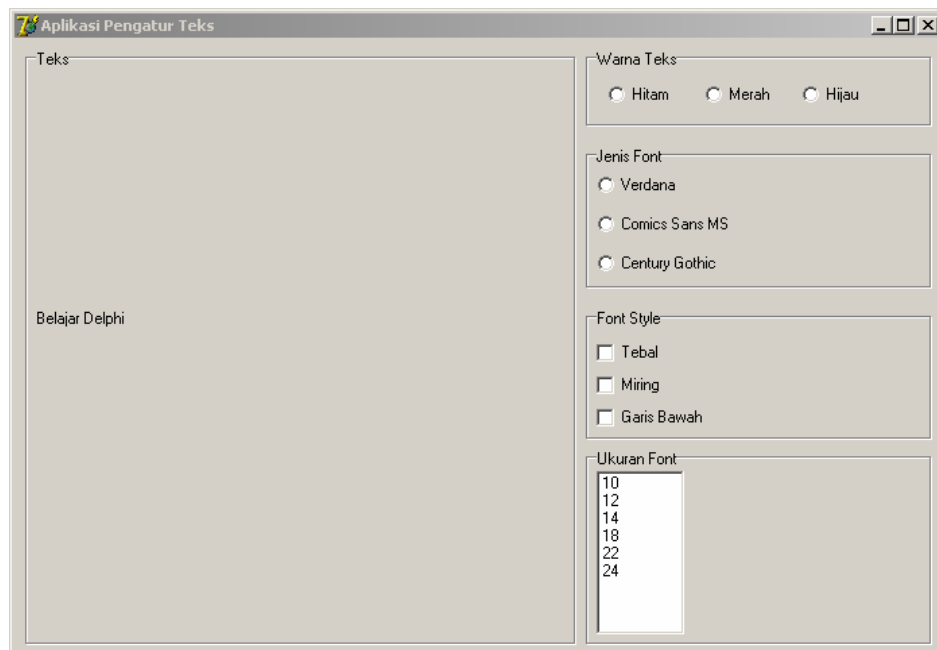
Daftar pilihan dapat ditambahkan melalui properti **items**. Untuk melakukan penanganan pada daftar pilihan, gunakan event Onclick dan masukkan kode berikut :

```
if ListBox1.ItemIndex=1 then  
    Edit1.Text:='Tujuan Malaysia';
```

C. Langkah Praktikum

1. Buat project aplikasi baru.
2. Simpan dengan nama unit = *praktikum4.pas* dan nama project = *praktikum4.dpr*.
3. Ubah caption form menjadi 'Aplikasi Pengatur Teks'.
4. Tambahkan komponen GroupBox  dan ubah propertinya captionnya menjadi 'Teks'
5. Tambahkan komponen Label, dan letakkan di dalam komponen GroupBox, kemudian ubah captionnya menjadi 'Belajar Delphi'.
6. Tambahkan komponen GroupBox  dan ubah properti captionnya menjadi 'Warna Teks'
7. Tambahkan 3 komponen RadioButton dan letakkan di dalam komponen GroupBox, masing-masing RadioButton diberi label = Hitam, Merah dan Hijau.
8. Tambahkan komponen RadioGroup, dan ubah properti captionnya menjadi 'Jenis Font', dan isi Itemsnya = Verdana, Comics Sans MS dan Century Gothic.

9. Tambahkan komponen GroupBox dan ubah propertinya captionnya menjadi 'Font Style'.
10. Tambahkan 3 komponen CheckBox dan letakkan di dalam komponen GroupBox, masing-masing CheckBox diberi label = Tebal, Miring dan Garis Bawah.
11. Tambahkan komponen GroupBox dan ubah propertinya captionnya menjadi 'Ukuran Font'.
12. Tambahkan komponen ListBox dan letakkan pada komponen GroupBox kemudian tambahkan pada properti itemnya ; 10,12,14,18,20,24.
13. Sehingga Tampilan Form akan terlihat seperti berikut :



14. Isikan kode untuk event OnClick pada masing-masing RadioButton.

```

procedure TForm1.RadioButton1Click(Sender: TObject);
begin
    if RadioButton1.Checked=True then
        begin
            Label1.Font.Color:=clBlack;
            RadioButton2.Checked:=False;
            RadioButton3.Checked:=False;
        end;
    end;

procedure TForm1.RadioButton2Click(Sender: TObject);
begin
    if RadioButton2.Checked=True then
        begin
            Label1.Font.Color:=clRed;
            RadioButton1.Checked:=False;
            RadioButton3.Checked:=False;
        end;
    end;

procedure TForm1.RadioButton3Click(Sender: TObject);
begin
    if RadioButton3.Checked=True then
        begin
            Label1.Font.Color:=clGreen;
            RadioButton1.Checked:=False;
            RadioButton2.Checked:=False;
        end;
    end;

```

15. Isikan kode untuk event OnClick pada RadioGroup.

```

procedure TForm1.RadioGroup1Click(Sender: TObject);
begin
    if RadioGroup1.ItemIndex = 0 then
        Label1.Font.Name:='Verdana';
    if RadioGroup1.ItemIndex = 1 then
        Label1.Font.Name:='Comis Sans MS';
    if RadioGroup1.ItemIndex = 2 then
        Label1.Font.Name:='Century Gothic';
    end;

```

16. Isikan kode untuk event OnClick pada masing-masing komponen CheckBox.

```

procedure TForm1.CheckBox1Click(Sender: TObject);
begin
    if CheckBox1.Checked=True then
        begin
            Label1.Font.Style:=[fsBold];
            CheckBox2.Checked:=False;
            CheckBox3.Checked:=False;
        end;
    end;
procedure TForm1.CheckBox2Click(Sender: TObject);
begin
    if CheckBox2.Checked=True then
        begin
            Label1.Font.Style:=[fsItalic];
            CheckBox1.Checked:=False;
            CheckBox3.Checked:=False;
        end;
    end;
procedure TForm1.CheckBox3Click(Sender: TObject);
begin
    if CheckBox3.Checked=True then
        begin
            Label1.Font.Style:=[fsUnderline];
            CheckBox2.Checked:=False;
            CheckBox3.Checked:=False;
        end;
    end;
end;

```

17. Isikan Kode untuk event OnClick pada ListBox.

```

procedure TForm1.ListBox1Click(Sender: TObject);
begin
    case ListBox1.ItemIndex of
        0 : Label1.Font.Size:=10;
        1 : Label1.Font.Size:=12;
        2 : Label1.Font.Size:=14;
        3 : Label1.Font.Size:=18;
        4 : Label1.Font.Size:=22;
        5 : Label1.Font.Size:=24;
    end;
end;


```

D. Tugas Praktikum (diperiksa saat praktikum berikutnya)

1. Buatlah aplikasi pembayaran akses internet pada warnet sederhana (billing Warnet) yang memiliki tampilan sebagai berikut :

a. Form Design

b. Tampilan Aplikasi

2. Gunakan komponen Timer  (pada tab system di component pallette) untuk mengetahui durasi waktu secara real time.
3. Deklarasikan variabel-variabel berikut :

```

public
    { Public declarations }
end;

var
    Form1: TForm1;
    awal : TDateTime;
    akhir: TDateTime;
    SLama: String;
    Lama : Double;
    BiayaPermenit: Double;
    Biaya:Double;
    biayamakan:integer;
    biayaminum:integer;

implementation

{$R *.dfm}

```

4. Untuk menghitung biaya pemakaian akses internet, gunakan kode dibawah ini :

```

procedure TForm1.btnInternetClick(Sender: TObject);
begin
    BiayaPermenit := 50;
    if btnInternet.Caption='START' then
        begin
            Timer1.Enabled:=True;
            awal:=Time;
            edMulai.Text:=TimeToStr (Time);
            edAakhir.Text:='';
            edBiayaInternet.Text:='';
            btnInternet.Caption:='STOP';
        end else
            if btnInternet.Caption='STOP' then
                begin
                    Timer1.Enabled:=False;
                    edAakhir.Text := TimeToStr (Time);
                    //menghitung lama pemakaian sekaligus mendecode-nya menjadi Jam1, Menit1, Detik1, dan MDetik1.
                    DecodeTime(StrToTime(edAakhir.Text)-StrToTime(edMulai.Text), Jam1, Menit1, Detik1, MDetik1);

                    edBiayaInternet.Text := FloatToStr((Jam1 * 3000) + ((Menit1) * 50) + ((Detik1) * 5));
                    edJumInternet.Text:=edBiayaInternet.Text;
                    btnInternet.Caption:='START'
                end;
            end;
        end;

```