

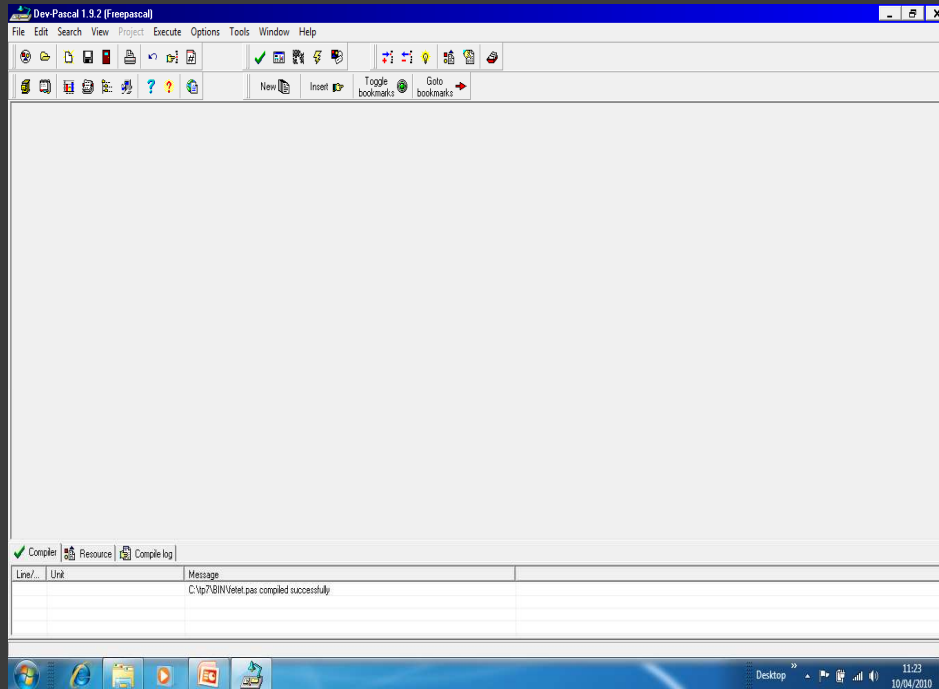
Materi 1

LOGIKA & ALGORITMA I

Pengenalan Pascal/DevPascal

- Turbo Pascal adalah Compiler bahasa pemrograman Pascal.
- Untuk memulai menjalankan Pascal:
 - Cari Folder Pascal ada di C:\TP\BIN\TPX.EXE
 - Jalankan File TPX tersebut

Dev Pascal



Jika menggunakan Dev-pascal, Untuk membuat File baru, Pilih Menu : File->New Source File

Untuk Menyimpan Tekan Keyboard Ctr+S

Untuk Melihat hasil atau menjalankan tekan F9

Struktur program pascal

```
program Project1;  
uses  
    Crt;
```

```
Begin
```

```
{ToDo - .....}
```

```
end.
```

Bagian deklarasi

**Bagian deskripsi
(body/tubuh program)**

Komentar

- Biasakan menuliskan komentar pada program agar memudahkan penelusuran.
- Komentar tidak dijalankan oleh Delphi.
- Ada 3 cara penulisan komentar di Delphi:

- Menggunakan kurung kurawal

Contoh : *{ ini contoh komentar
 boleh lebih dari satu baris }*

- Menggunakan kurung bintang

Contoh : *(* ini contoh komentar
 boleh lebih dari satu baris *)*

- Menggunakan garis miring ganda

Contoh : *// ini contoh komentar
 // kalau 2 baris dituliskan seperti ini*

Separator

- Separator digunakan sebagai pemisah antar dua *token* (statement terkecil yang dieksekusi Delphi yang memiliki arti)
- Pada Delphi, separator menggunakan tanda titik koma (;)
- Contoh:
 - `Size:=20;Price:=10; //atau`
 - `Size:=20;`
`Price:=10;`

Identifier

- Nama untuk konstanta, variabel, tipe, objek, properti, prosedur, fungsi, unit, dll disebut *identifier*
- Aturan penamaan *identifier*:
 1. Maksimum 255 karakter
 2. Karakter pertama **harus** dimulai dengan huruf atau garis bawah
 3. Karakter berikutnya boleh berupa huruf, angka, atau garis bawah
 4. **Non-case sensitif** (huruf besar dan huruf kecil dianggap sama)
 5. Bukan *reserved word* (perintah-perintah standard Delphi)

Catatan:

Pascal/Delphi non-case sensitif, tetapi biasanya menerapkan case sensitif karena bahasa pemrograman lain (mis C,C++) adalah case sensitif.

Contoh identifier

- ⦿ Rata-rata (Tdk boleh, mengandung tanda -)
- ⦿ Nama ku (Tdk boleh, mengandung spasi)
- ⦿ 1warna (Tdk boleh, huruf pertama angka)
- ⦿ Warna1 (boleh)
- ⦿ Rata_rata (boleh)
- ⦿ A*2 (Tdk boleh, mengandung karakter *)
- ⦿ Hasilperkalian (boleh)
- ⦿ Hasil_perkalian (boleh)
- ⦿ Label (Tdk boleh, reserved word)

Menampilkan teks di layar

- Gunakan `write()` dan `writeln()` untuk menampilkan teks di layar console.
- Teks yang akan ditampilkan harus diapit oleh tanda petik tunggal (' ')
- Angka tidak perlu diapit oleh tanda petik tunggal

Contoh:

```
Write('Hello');  
Write('5');  
Write(5);
```

Menampilkan teks di layar

- `write('Hello')` akan menampilkan teks 'Hello' di layar console dan kursor akan berada di belakang huruf terakhir ('o').
- Agar kursor bisa pindah ke baris berikutnya, gunakan perintah `writeln()`
- Coba kedua contoh di bawah ini dan lihat perbedaannya.

Contoh1:

```
Write('Hello');  
Write('apa kabar');
```

Contoh2:

```
Writeln('Hello');  
Writeln('apa kabar');
```

Listing contoh 2 secara lengkap

```
program Project1;
```

```
{ $APPTYPE CONSOLE }
```

```
uses
```

```
    SysUtils;
```

```
begin
```

```
    writeln('Hello');
```

```
    writeln('apa kabar');
```

```
end.
```

Pseudocode vs Pascal

Menampilkan teks di layar

Pseudocode :

- Write('Hello')

Pascal :

- Write('Hello');
- Writeln('Hello');

Catatan :

Perhatikan bahwa di pseudocode kita tidak memperdulikan format tampilan yang akan muncul di layar.

Membaca data dari keyboard

- ⦿ Jika program sebelumnya dijalankan maka hasil *run* langsung hilang, tidak sempat terbaca.
- ⦿ Ini disebabkan karena setelah menampilkan teks, Delphi langsung menjalankan perintah selanjutnya (`end`), sehingga layar console langsung ditutup.
- ⦿ Agar layar console tidak langsung ditutup, trik-nya adalah meminta Delphi menunggu user menekan tombol Enter
- ⦿ Gunakan *readln* untuk menunggu user menekan tombol Enter.

Listing contoh 2 dengan readln

```
program Project1;  
  
uses  
    Crt;  
  
begin  
    writeln('Hello');  
    writeln('apa kabar');  
    readln;  
end.
```

Latihan

Apa yang ditampilkan di layar jika program berikut dijalankan?

```
program Project1;  
uses  
    Crt;  
  
begin  
    write ('Hello ');  
    writeln('apa kabar');  
    writeln; writeln;  
    write ('Kabar baik');  
    readln;  
end.
```

Latihan

Apa yang ditampilkan di layar jika program berikut dijalankan?

```
program Project1;  
uses  
    Crt  
  
begin  
    write ('Hello ', 'apa kabar');  
    writeln;  
    write ('Umur saya ', 25);  
    write (' tahun');  
    readln;  
end.
```


Konstanta

- Konstanta adalah suatu identifier yang nilainya tidak pernah berubah selama program dijalankan
- Harus dituliskan di bagian deklarasi
- Cara penulisan : *const <identifier> = <nilai>*
Contoh : *const pi = 3.14*
- Jika ada lebih dari satu konstanta sebaiknya dideklarasikan sekaligus

Contoh :

```
const pi = 3.14;  
pajak = 10;
```

Latihan

Apa yang ditampilkan di layar jika program berikut dijalankan?

```
program Project1;  
uses  
    Crt;  
  
Const a=20;  
  
begin  
    write ('Hello ', 'apa kabar');  
    writeln;  
    write ('Umur saya ', a);  
    write (' tahun');  
    readln;  
end.
```

Variabel

- Variabel adalah suatu *identifier* yang nilainya boleh berubah selama program dijalankan
- Harus dituliskan di bagian deklarasi
- Tipe data harus ditentukan saat deklarasi
- Cara penulisan : *var <identifier> : <tipe data>*
- Jika ada lebih dari satu variabel sebaiknya dideklarasikan sekaligus

Contoh :

```
var  pi : real;  
    pajak, a, b : integer;
```

Tipe Data di pascal

- ◉ *simple*
 - *ordinal*
 - ◉ *integer*
 - ◉ *character*
 - ◉ *Boolean*
 - *real*
- ◉ *string*
- ◉ *structured*
 - *set*
 - *array*
 - *record*
 - *file*
 - *...dll*

**Baca help di pascal untuk
keterangan lebih lanjut**

Tipe Data Integer

- Tipe data **Integer** digunakan untuk menampung data bilangan bulat.
- Varian dari tipe data integer yg dikenal Pascal a.l:

Tipe	Rentang Nilai
Integer / Longint	-2147483648...2147483647
Shortint	-128...127
Smallint	-32768...32767
Int64	$-2^{63} \dots 2^{63}-1$
Word	0...65535
Byte	0...255

Tipe Data Real

- Tipe data **Real** digunakan untuk menampung data bilangan real / floating-point.
- Varian dari tipe data real yg dikenal Delphi a.l:

Tipe	Rentang Nilai
Single	$1.5 \times 10^{-45} \dots 3.4 \times 10^{38}$
Real / Double	$5.0 \times 10^{-324} \dots 1.7 \times 10^{308}$
Extended	$3.6 \times 10^{-4951} \dots 1.1 \times 10^{4932}$
Comp	$-2^{63}+1 \dots 2^{63}-1$
Currency	$-922337203685477.5808 \dots 922337203685477.5807$

Tipe Data Character

- Tipe data character digunakan untuk menampung data sebanyak 0 buah atau 1 buah karakter
- Varian dari tipe data character yg dikenal a.l:

Tipe	Rentang Nilai
Char / AnsiChar	8-bit ASCII = 256 karakter
WideChar	16-bit Unicode = 65536 karakter

Catatan :

Daftar karakter ASCII dan Unicode dapat dilihat di berbagai aplikasi berbasis Windows, misalnya Microsoft Word (pilih Insert > Symbol...)

Tipe Data String

- Tipe data string digunakan untuk menampung data sebanyak 0 buah, 1 buah karakter atau lebih
- Varian dari tipe data string yg dikenal Delphi a.l:

Tipe	Panjang maksimum
ShortString	255 karakter
AnsiString	$\sim 2^{31}$ karakter
WideString	$\sim 2^{30}$ karakter

Catatan :

Fungsi **string** secara default memiliki sifat seperti tipe data AnsiString

Operator Assignment

- **Operator assignment**, digunakan untuk memberikan nilai kepada suatu variabel. Simbol yang digunakan **`:=`**

- Contoh :

Pajak := 10;

Nama := 'Jamilah';

- Untuk tipe karakter (*char*) dan tipe *string*, nilai yang akan dimasukkan ke variabel harus dibatasi dengan tanda petik tunggal (' ')

Latihan

Apa yang ditampilkan di layar jika program berikut dijalankan?

```
program Project1;  
uses  
    Crt;  
  
Var umur : integer;  
    nama : string;  
  
Begin  
    umur := 20;  
    nama := 'Iwan';  
    write ('Hello apa kabar');  
    writeln;  
    write ('Umur saya ', umur, ' tahun');  
    write ('Nama saya ', nama);  
    readln;  
end.
```

Pseudocode vs Pascal

Deklarasi & Assignment

Pseudocode :

- Deklarasi:

- const pi = 3.14

- Deklarasi:

- A : real

- $A \leftarrow B+C$

Pascal :

- `const pi = 3.14;`

- `var A:real;`

- `A := B+C;`