

Algoritma dan Pemrograman

Bab II – Dev Pascal, Variabel, Tipe Data



```
> internet
```

Adam Mukharil Bachtiar, S.Kom.
Universitas Komputer Indonesia

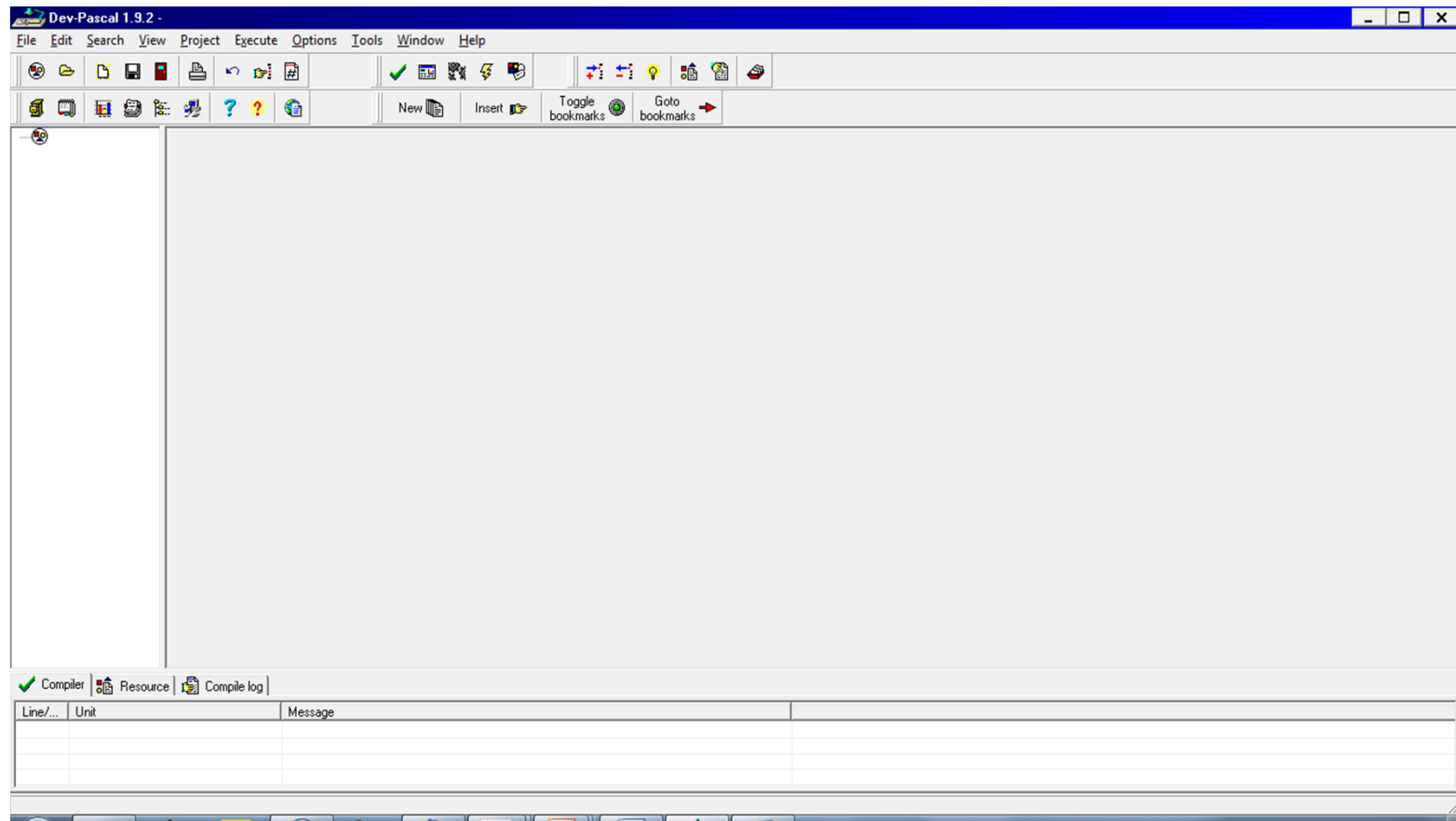
Dev Pascal

Sebuah IDE untuk bahasa PASCAL
keluaran BloodSheed yang sifatnya
freeware.



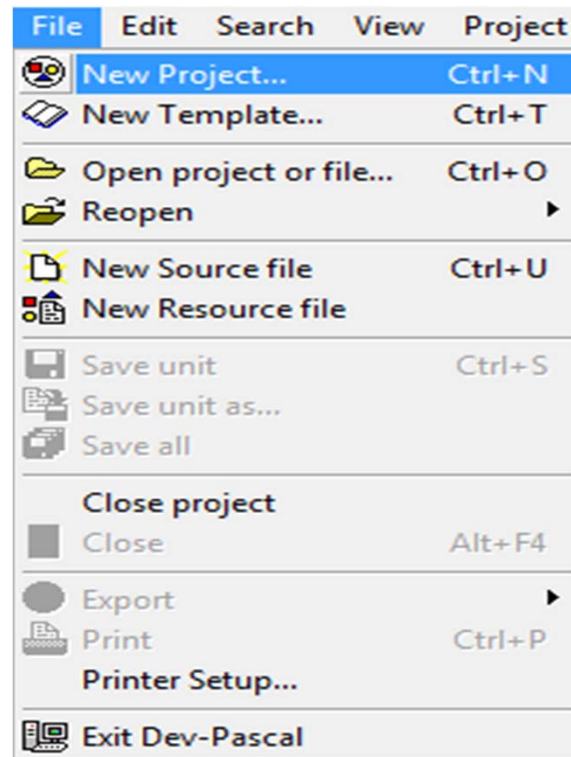
Cara Membuat Program di Dev Pascal

Buka aplikasi Dev Pascal.



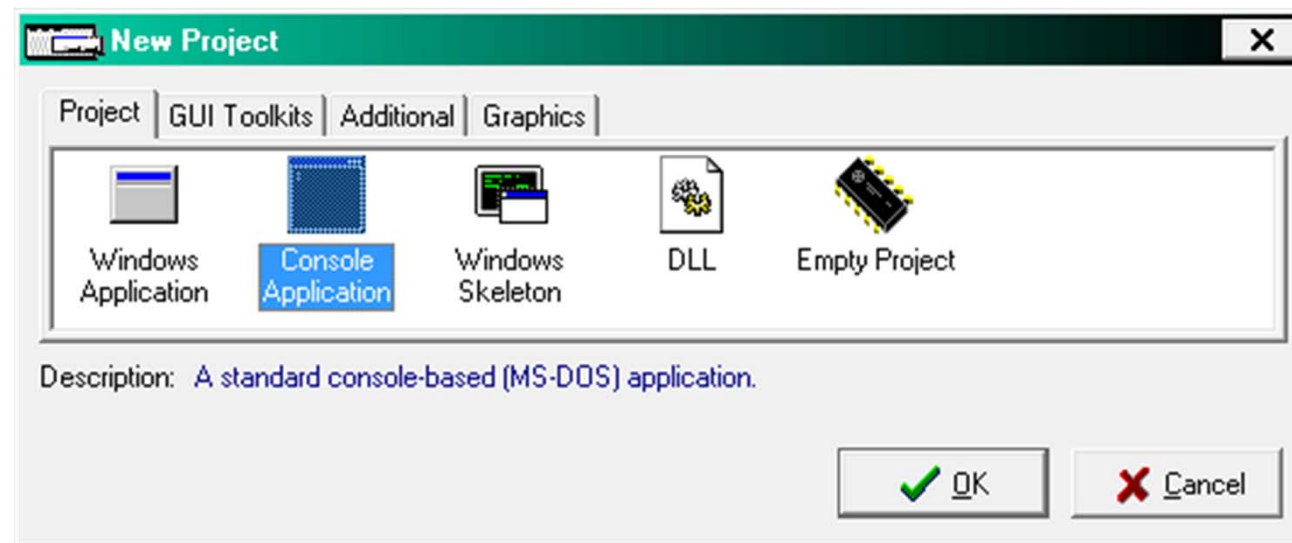
Cara Membuat Program di Dev Pascal

Buat sebuah file baru atau project baru



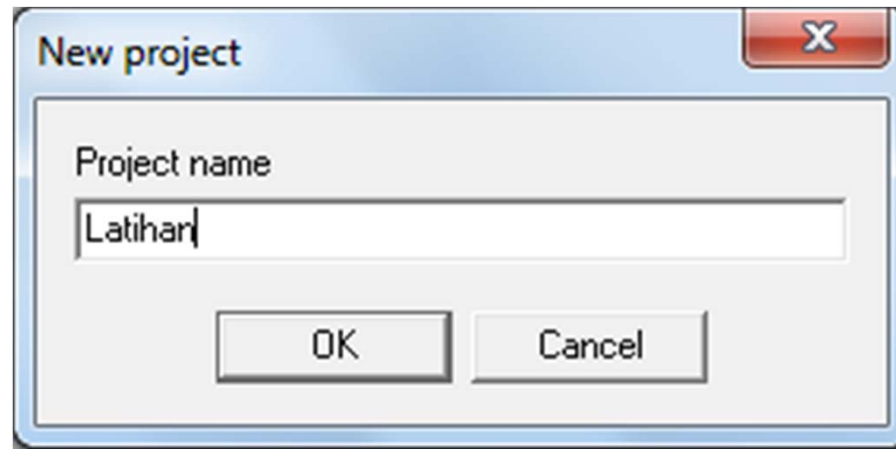
Cara Membuat Program di Dev Pascal

Pilih Console Application lalu klik OK.



Cara Membuat Program di Dev Pascal

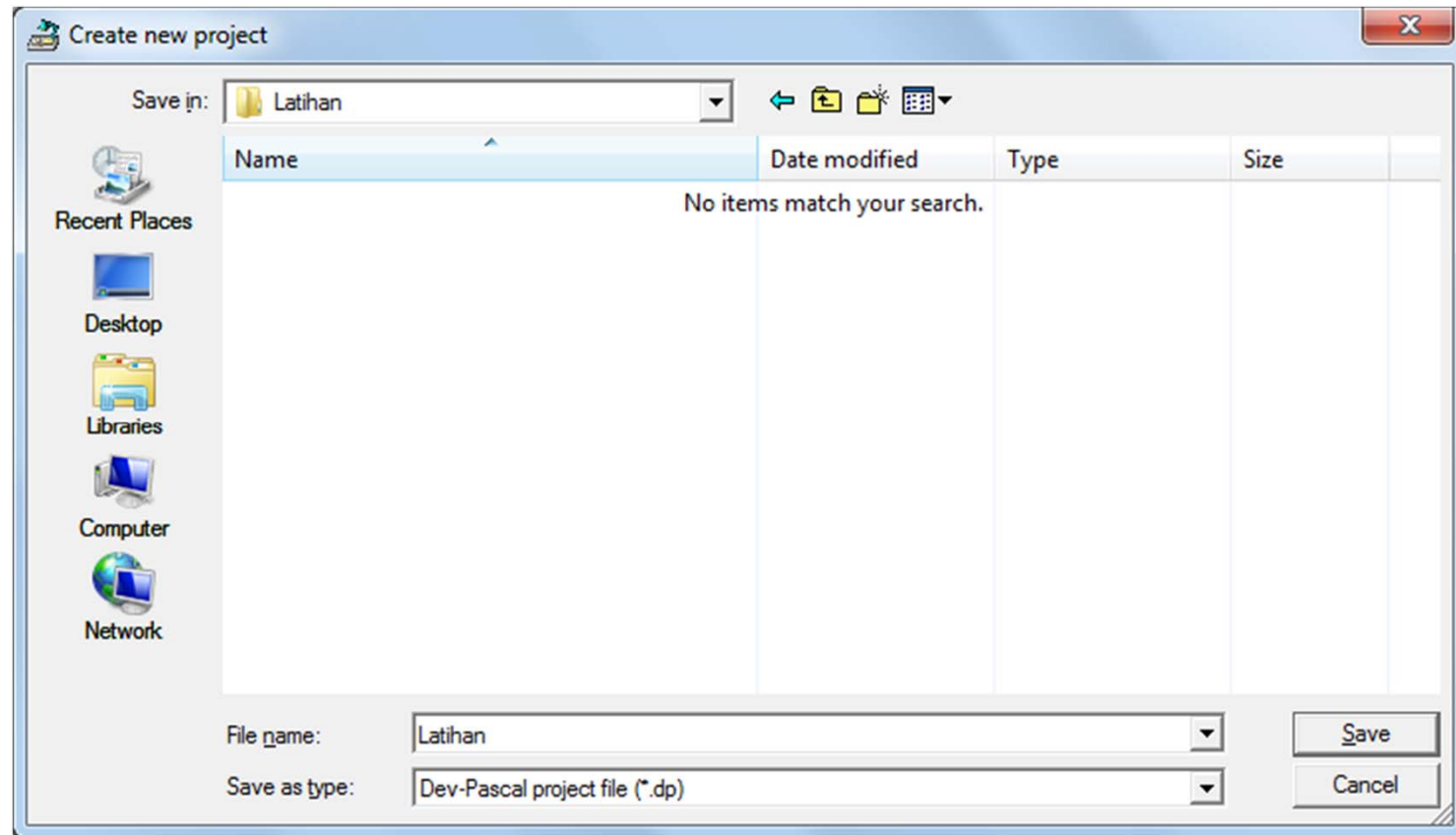
Beri nama project (Boleh mengandung spasi).



WARNING: Nama project samakan dengan nama folder tempat menyimpannya. Satu buah Project disimpan dalam satu buah folder.

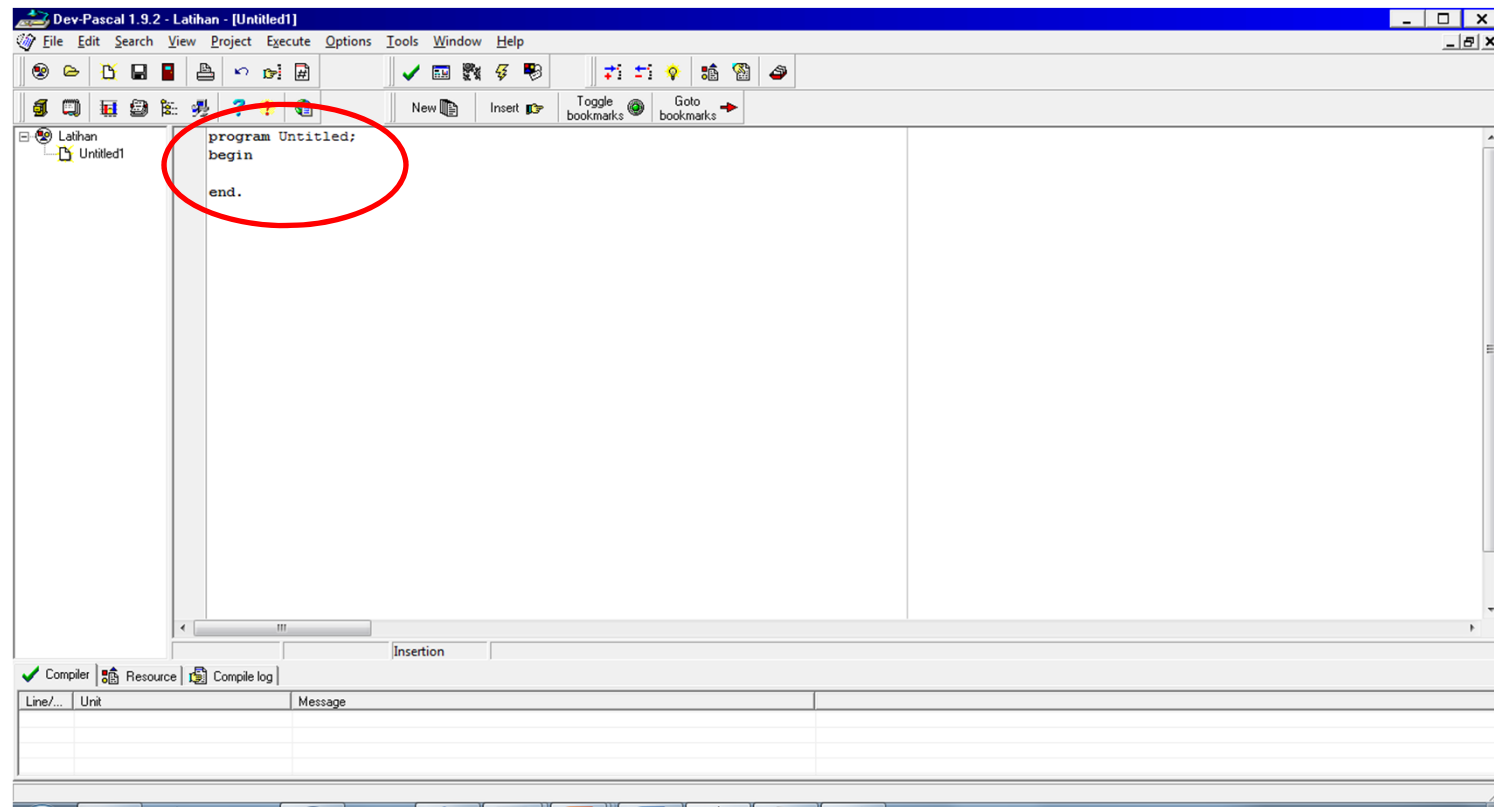
Cara Membuat Program di Dev Pascal

Simpan project pada folder yang sudah disediakan.



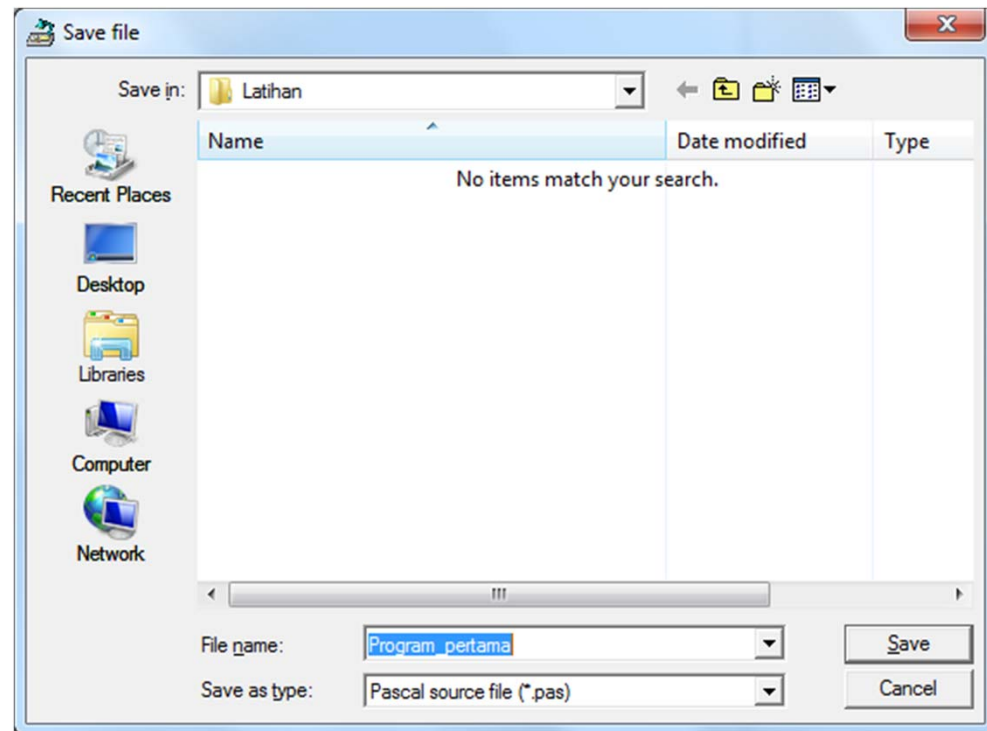
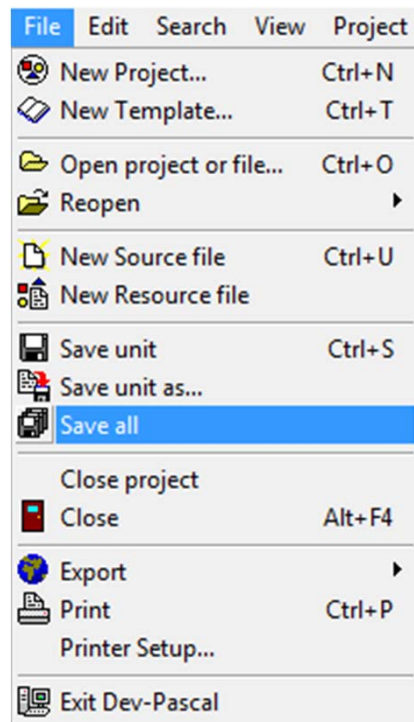
Cara Membuat Program di Dev Pascal

Jika langkahnya benar maka muncul tampilan sebagai berikut.



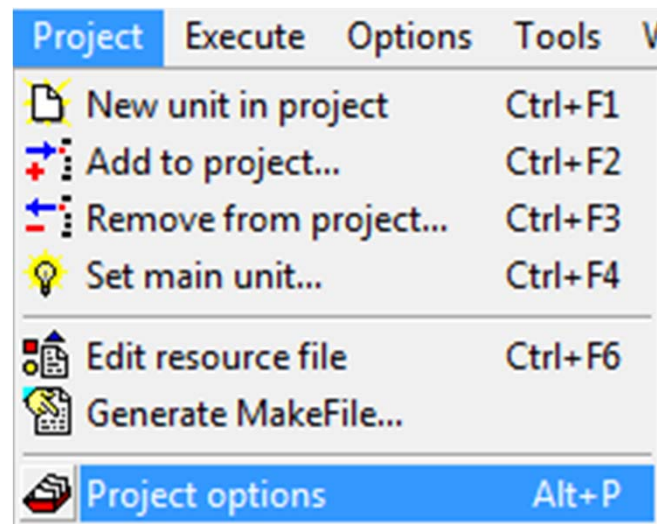
Cara Membuat Program di Dev Pascal

Simpan filenya di folder yang sama dengan project
(Jangan mengandung spasi).



Cara Membuat Program di Dev Pascal

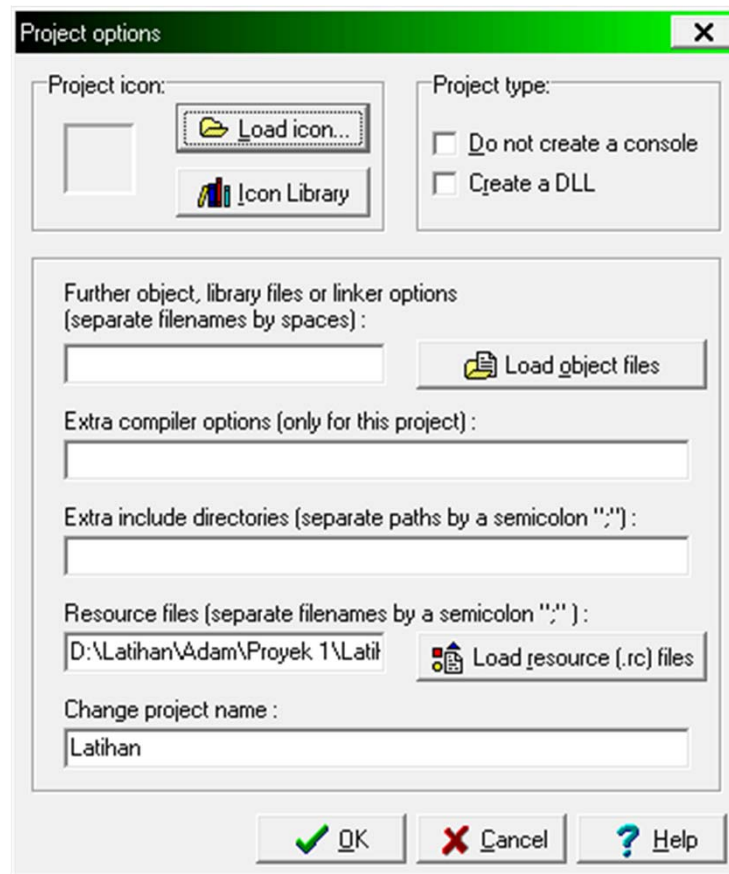
Berikan Icon dengan cara klik menu Project → Project options.



WARNING: Menggunakan icon pada dev pascal menjadi hal yang wajib.

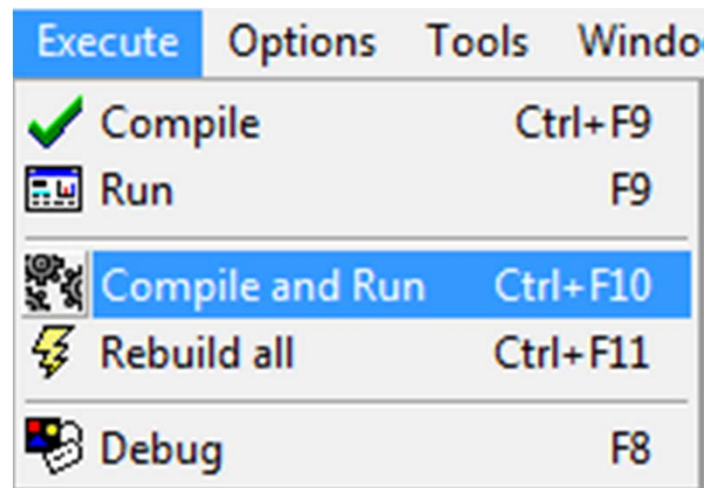
Cara Membuat Program di Dev Pascal

Klik Load icon lalu pilih icon yang diinginkan lalu klik OK.



Cara Membuat Program di Dev Pascal

Ketik Sintaks lalu untuk mengcompile dan menjalankan program klik Ctrl + F10 atau klik menu Execute → Compile and Run





NOTASI ALGORITMA

VS

NOTASI PASCAL

Notasi Algoritma

- 1 { ini adalah notasi algoritma } → komentar
- 2 Algoritma judul_algoritma
- 3 {I.S.: → diisi keadaan yang terjadi di awal algoritma}
- 4 {F.S.: → diisi keadaan yang terjadi di akhir algoritma}
- 5
- 6 Kamus/Deklarasi:
- 7 {diisi pendefinisian konstanta}
- 8
- 9 {diisi deklarasi variabel beserta tipe data}
- 10
- 11 Algoritma/Deskripsi:
- 12 {diisi dengan input, proses, dan output}

Notasi Pascal

```
1 { ini adalah notasi pascal} → komentar  
2 program judul_program;  
3  
4 var  
5 {diisi pendefinisian konstanta}  
6  
7 {diisi deklarasi variabel beserta tipe data}  
8  
9 begin  
10 {diisi dengan input, proses, dan output}  
11 end.
```

Notasi Algoritma VS Notasi Pascal

{ ini adalah notasi algoritma } → komentar

Algoritma judul_algoritma

{I.S.: → diisi keadaan yang terjadi di awal algoritma}

{F.S.: → diisi keadaan yang terjadi di akhir algoritma}

Kamus/Deklarasi:

{diisi pendefinisian konstanta}

{diisi deklarasi variabel beserta tipe data}

Algoritma/Deskripsi:

{diisi dengan input, proses, dan output}

{ ini adalah notasi pascal } → komentar

program judul_program;

var

{diisi pendefinisian konstanta}

{diisi deklarasi variabel beserta tipe data}

begin

{diisi dengan input, proses, dan output}

end.

Notasi Algoritma VS Notasi Pascal

NO.	ALGORITMA	PASCAL
1	<u>Kamus</u> :	var
2	<u>Algoritma</u> :	begin end.
3	<u>input</u> (variabel)	readln(variabel); read(variabel);
4	<u>output</u> ('.....')	write('.....'); atau writeln('.....');
5	<u>output</u> ('.....', variabel)	write('.....', variabel); atau writeln('.....', variabel);
6	<u>output</u> (variabel)	write(variabel); atau writeln(variabel);
7	←	:=

Program Pertama Pascal

```
1  program Program_Pertama;
2  uses crt; {pemanggilan unit crt untuk readkey()}
3
4  begin
5      writeln('Selamat Datang');
6      write('Di');
7      writeln(' UNIKOM');
8      writeln('Bandung');
9      writeln();
10     write('Tekan sembarang tombol untuk menutup. ');
11     readkey();
12 end.
```

Kasus Tukar Nilai dengan Variabel Bantu (1)

```
1  Algoritma Tukar_Nilai
2  {I.S.: Nilai variabel a dan b dimasukkan oleh user}
3  {F.S.: Menampilkan hasil penukaran nilai variabel a
4   dan b}
5
6  Kamus:
7   a,b: integer
8   bantu:integer
9
10 Algoritma:
11  output('Masukkan nilai a: ')
12  input(a)
13  output('Masukkan nilai a: ')
14  input(a)
15  bantu←a
16  a←b
17  b←bantu
18  output('Nilai a sekarang : ',a)
19  output('Nilai b sekarang : ',b)
```

Kasus Tukar Nilai dengan Variabel Bantu (2)

```
1  Algoritma Tukar_Nilai
2  {I.S.: Nilai variabel a dan b dimasukkan oleh user}
3  {F.S.: Menampilkan hasil penukaran nilai variabel a
4   dan b}
5
6  Kamus:
7   a,b: integer
8   bantu: integer
9
10 Algoritma:
11  input(a,b)
12  bantu ← a
13  a ← b
14  b ← bantu
15  output('Nilai a sekarang : ',a)
16  output('Nilai b sekarang : ',b)
```

Kasus Tukar Nilai dengan Variabel Bantu

```
1  program Tukar_Nilai;
2  uses crt; {pemanggilan unit crt untuk readkey()}
3
4  var
5      a,b:integer
6      bantu:integer
7
8  begin
9      write('Masukan nilai a: '); readln(a);
10     write('Masukan nilai b: '); readln(b);
11     bantu:=a;
12     a:=b;
13     b:=bantu;
14     write('Nilai a sekarang: ',a);
15     write('Nilai b sekarang: ',b);
16     readkey();
17 end.
```

Kasus Tukar Nilai Tanpa Variabel Bantu

```
1  Algoritma Tukar_Nilai
2  {I.S.: Nilai variabel a dan b dimasukkan oleh user}
3  {F.S.: Menampilkan hasil penukaran nilai variabel a
4   dan b}
5
6  Kamus:
7   a,b: integer
8
9  Algoritma:
10  input(a,b)
11  a←a+b
12  b←a-b
13  a←a-b
14  output('Nilai a sekarang : ',a)
15  output('Nilai b sekarang : ',b)
```

Kasus Tukar Nilai Tanpa Variabel Bantu

```
1  program Tukar_Nilai;
2  uses crt; {pemanggilan unit crt untuk readkey()}
3
4  var
5      a,b:integer
6
7  begin
8      write('Masukan nilai a: '); readln(a);
9      write('Masukan nilai b: '); readln(b);
10     a:=a+b;
11     b:=a-b;
12     a:=a-b;
13     write('Nilai a sekarang: ',a);
14     write('Nilai b sekarang: ',b);
15     readkey();
16 end.
17
```



TIPE DATA ALGORITMA DAN PASCAL

Tipe Data

1. Tipe Data Dasar
2. Tipe Data Bentukan



Tipe Data Dasar

1. Sudah dikenal dalam kehidupan sehari-hari.
2. Disebut sebagai ***predefined data type***.
3. Contoh: bilangan logik, bilangan bulat, bilangan riil, karakter, dan string.



Bilangan Logik

1. Nama Tipe : **boolean**.
2. Nilai : benar (*true*) atau salah (*false*).
3. Bisa dinyatakan dengan 0 dan 1 (**pendefinisian benar dan salah tergantung pembuat algoritma**).



Operasi Bilangan Logik

x	<u>not</u> x
true	false
false	true

x	y	x <u>and</u> y	x <u>or</u> y	x <u>xor</u> y
true	true	true	true	false
true	false	false	true	true
false	true	false	true	true
false	false	false	false	false

Bilangan Bulat

1. Nama Tipe : **integer**.
2. Nilai : - (~) sampai dengan + ~ (tanpa koma).
3. Operasi aritmatika : + , - , * , / , **div (hasil bagi bulat)**, **mod (sisanya)**.
4. Operasi Perbandingan : < , ≤ , > , ≥ , = , ≠.



Bilangan Riil

1. Nama Tipe : real.
2. Nilai : - (\sim) sampai dengan + \sim .
3. Operasi aritmatika : + , - , * , /.
4. Operasi Perbandingan : < , \leq , > , \geq , = , \neq .



Karakter

1. Nama Tipe : char.
2. Nilai : semua huruf ('a'..'z', 'A'..'Z'), semua angka desimal (0..9), tanda baca, operator aritmatika dan karakter khusus (simbol **ASCII**).
3. Operasi Perbandingan : < , ≤ , > , ≥ , = , ≠.



String

1. Nama Tipe : string.
2. Nilai : deretan karakter diapit ' '.
3. Operasi Penyambungan : +.
4. Operasi Perbandingan : < , ≤ , > , ≥ , = , ≠.



Tipe Bentukan

1. Tipe dasar yang diberi nama tipe baru.
2. Tipe struktur.



Tipe Dasar yang Diberi Nama Tipe Baru

1. Alasan: agar lebih mudah diingat dan ***High Readability***.
2. Kata kunci : **type**.
3. Contoh: **type**

pecahan : **real** {tanda : bisa diganti =}



Tipe Terstruktur (Record)

1. Kumpulan data (field) yang memiliki tipe data berbeda.
2. Contoh Pendeklarasian:

type

Mahasiswa : record

< NIM : integer, {0..9}

Nama : string, {'A'..'Z', 'a'..'z'}

Nilai : real {0..100} >



Tipe Terstruktur (Record)

3. Jika mhs1 bertipe Mahasiswa maka cara mengakses tiap field adalah sebagai berikut:
 - a. mhs1.NIM
 - b. mhs1>Nama
 - c. mhs1.Nilai



Perbandingan Tipe Data

Algoritma	Pascal	Rentang Pascal
<u>boolean</u>	boolean	true dan false
<u>integer</u>	byte	0..255
	shortint	-128..127
	word	0..65535
	integer	-32768..32767
	longint	-2147483648..2147483647
<u>real</u>	real	$2.9 \times 10^{-39} \dots 1.7 \times 10^{38}$
	single	$1.5 \times 10^{-45} \dots 3.4 \times 10^{38}$
	double	$5.0 \times 10^{-324} \dots 1.7 \times 10^{308}$
	extended	$3.4 \times 10^{-4932} \dots 1.1 \times 10^{4932}$
<u>char</u>	char	
<u>string</u>	string	
	string[n]	
<u>type</u>	type	
varrecord: <u>record</u>	varrecord=record	
< field1:type1,	field1:type1;	
field2:type2,	field2:type2;	
...	...	
fieldn:typen>	fieldn:typen;	
	end;	

Perbandingan Operator

Algoritma	Pascal
+	+
-	-
*	*
/	/
<u>div</u>	div
<u>mod</u>	mod

Algoritma	Pascal
<u>not</u>	not
<u>and</u>	and
<u>or</u>	or
<u>xor</u>	xor

Algoritma	Pascal
<	<
≤	<=
>	>
≥	>=
=	=
≠	<>

Algoritma	Pascal
<u>type</u>	type
<u>const</u>	const
<u>true</u>	true
<u>false</u>	false
{ komentar }	{ komentar }
	(* komentar *)



IDENTIFIER

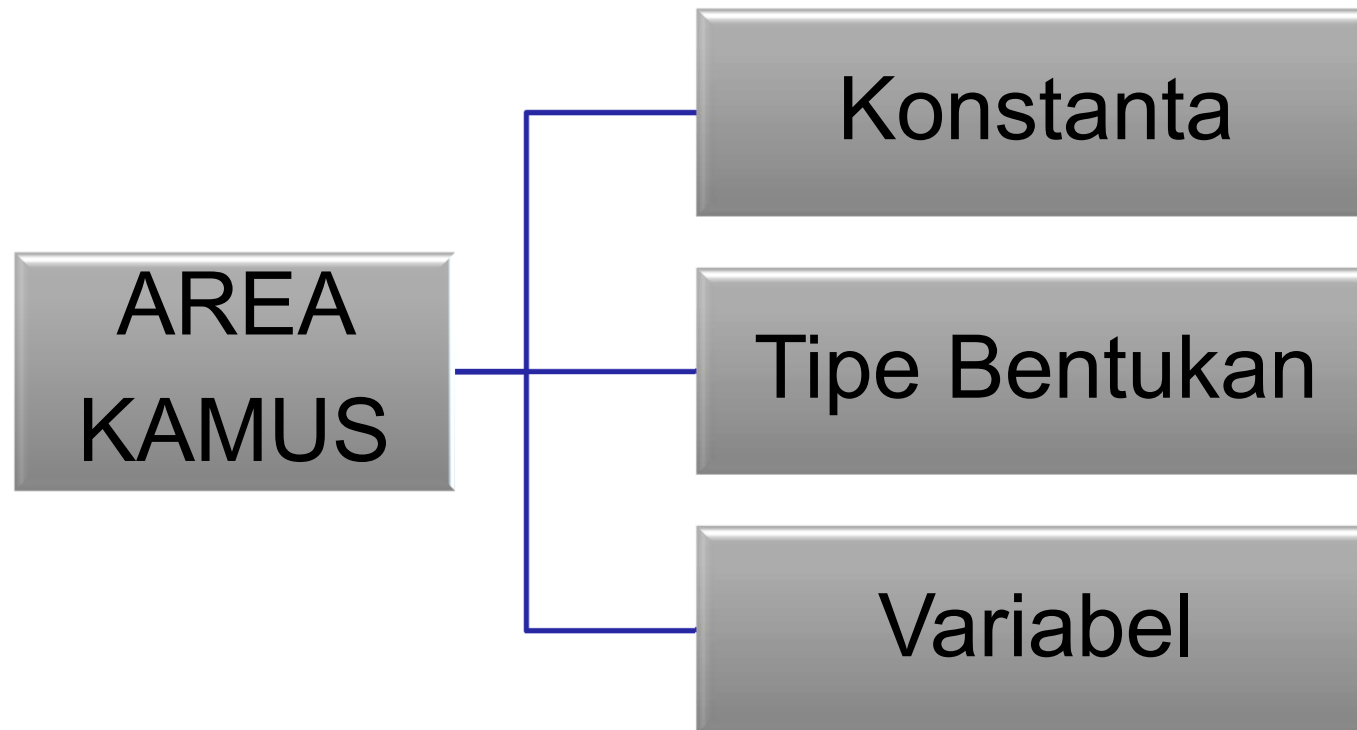
(Pengenal / Nama)

Identifier (Pengenalan>Nama)

Digunakan untuk mengacu sesuatu di dalam algoritma atau program



Identifier (Pengenalan>Nama)



Aturan Penamaan

1. Nama harus dimulai dengan alfabet dan **tidak boleh** diawali dengan **angka**.
2. Huruf besar dan kecil tidak dibedakan (PASCAL) → saran: **konsisten**.
3. Karakter penyusun identifikasi **hanya boleh** alfabet, angka, dan '_'.

```
> int
```
4. Identifikasi **tidak boleh** mengandung operator aritmatika, relasional, tanda baca, dan karakter khusus lainnya.

Aturan Penamaan

5. Tidak boleh mengandung spasi
6. Panjang identifier tidak dibatasi akan tetapi pilihlah nama yang **sederhana dan mudah diingat**.

Variabel VS Konstanta

Variabel dan konstanta sama sama digunakan untuk menampung sebuah nilai. Perbedaannya terletak dari sifat penyimpanannya. Variabel memungkinkan perubahan nilai di dalamnya selama program dieksekusi sedangkan konstanta akan tetap menyimpan nilainya secara permanen.

Variabel VS Konstanta

Deklarasi variabel

Nama_variabel:tipe_data

Contoh: x,y:integer

Deklarasi Konstanta

type

const nama_konstanta = nilai_konstanta

Contoh:

type

const phi =3.14

Notasi Matematis → Notasi Algoritma

1. Notasi prefix → $*79$, $*+a/bc-d*ef$
- 2. Notasi infix → $7*9$, $a+b/c*d-e*f$**
3. Notasi postfix → $68*$, $abc/+def*-*$



Notasi Matematis → Notasi Algoritma

1. $luas = \frac{1}{2}(alas \cdot tinggi) \rightarrow luas \leftarrow 1/2 * (alas * tinggi)$

2. $a = \frac{10b + 3c}{5d} \rightarrow a \leftarrow (10 * b + 3 * c) / (5 * d)$



GAMES

1. Deklarasikan sebuah tipe bentukan untuk menampung data KTP, data SIM, dan data dosen!
2. Tuliskan notasi matematis ini menjadi notasi algoritma:

a.
$$m = \frac{a-b}{3ac} \left(1 - \frac{bcd}{fgh} \right)$$

b.
$$x = \frac{-b + 2c^2 + 4abc}{2c(3a + 4c)}$$

