

&lt;/

# Operasi Kondisi

/&gt;

} /&gt; [

Andreas M Lbn Gaoḷ, C.S.Kom.

</ Flashback

# Relational Operator Logical Operator



1 0 1 1   0 1 1   0 1   1 0 1 1 0 0 1   1 0   1 1 0 1 1   0 1 1   0 1   1 1 0 1 1 0   1 1 0 1 1 1   1 1 0 1

# </ Relational Operator

=

Equal

<>

Not Equal

>

Greater Than

>=

Greater Than  
or Equal to

<

Less Than

<=

Less Than  
or Equal to

# </ Logical Operator

## NOT

NOT TRUE -> FALSE

NOT FALSE -> TRUE

## AND

TRUE AND TRUE -> TRUE

## OR

FALSE OR FALSE -> FALSE

## XOR

TRUE XOR FALSE -> TRUE

FALSE XOR TRUE -> TRUE



## </ Statement `if` 1 kasus

Pada blok `if`, menjalankan Perintah jika Kondisi Benar

Bentuk umum:

```
if(kondisi) then begin  
    perintah_1;  
    perintah_2;  
    perintah_n;  
end;
```

</ Statement `if` 1 kasus

rupa.pas

&lt;/&gt;

# Statement `If` 2 Kasus

02

} /&gt; [



## </ Statement `if` 2 kasus

Pada blok `else`, menjalankan Perintah jika Kondisi Salah

Bentuk umum:

```
if(kondisi) then begin
    perintah_1;
    perintah_2;
    perintah_n;
end else begin
    perintah_1;
    perintah_2;
    perintah_n;
end.
```

## </ Statement `if` 2 kasus

prodi.pas



# Statement `If` 2 Kasus atau Lebih

03

## </ Statement `if` 2 kasus atau lebih

Jika kondisi pada `if` salah, maka akan mengecek kondisi pada `else if`. Jika benar, perintah akan dijalankan.  
**Prioritas operator kondisi dimulai dari atas ke bawah**

## </ Statement `if` 2 kasus atau lebih

Bentuk umum:

```
if(kondisi_1) then begin
    perintah_1;
    perintah_n;
end else if(kondisi_2) then begin
    perintah_1;
    perintah_n;
end else if(kondisi_3) then begin
    perintah_1;
    perintah_n;
end;
```

# </ Statement `if` 2 kasus atau lebih

Bentuk umum:

```
if(kondisi_1) then begin
    perintah_1;
    perintah_n;
end else if(kondisi_2) then begin
    perintah_1;
    perintah_n;
end else begin
    perintah_1;
    perintah_n;
end;
```

# </ Statement `if` 2 kasus atau lebih

kaya.pas

&lt;/&gt;

Statement `case`

04

} /&gt; [



## </ Statement `case`

case dapat digunakan sebagai alternatif if dan else if jika terlalu banyak yang harus di cek, dan kondisi yang dicek hanya menggunakan 1 variabel

Pada blok `case`, menjalankan perintah jika Kondisi pada benar.

Prioritas kondisi dimulai dari atas ke bawah.

Pada blok `else` akan menjalankan program jika semua kondisi salah

## </ Statement `case`

Bentuk umum:

```
case (variabel) of
    nilai_1 : perintah_1;
    nilai_2 : perintah_2;
    nilai_n : perintah_n;
else begin
    perintah_1;
    perintah_n;
end;
end;
```

</ Statement `case`

hari.pas

# </ Pengayaan

nim2.pas



Soal A.pdf  
Soal B.pdf

1 0 1 1    0 1 1    0 1    1 0 1 1 0 0 1    1 0    1 1 0 1 1    0 1 1    0 1    1 1 0 1 1 0    1 1 0 1 1 1    1 1 0 1

# </ Thanks!

/>

} /> [



**CREDITS:** This presentation template was created by [Slidesgo](#), and includes icons by [Flaticon](#), and infographics & images by [Freepik](#)

Please keep this slide for attribution