



# Algoritma dan Pemrograman

Percabangan



# Opening Keynote



Jika kamu berusaha tidak belajar,  
tidak ada orang yang bisa menolongmu

jika kamu menjadikan diri untuk belajar,  
tidak ada yang bisa menghentikan mu.

# Subtopik



- Pernyataan IF
- Pernyataan IF-ELSE
- Pernyataan ELSIF
- Pernyataan CASE-WHEN
- Pernyataan UNLESS

# Capaian



- Anda memahami pengertian dan fungsi percabangan
- Anda dapat menggunakan Notasi Percabangan dengan Benar
- Anda mampu menggunakan jenis-jenis percabangan sesuai dengan masalah yang ada.

# Syarat Material

Untuk mengikuti rangkaian materi pada slide ini ada prasyarat yang perlu dipenuhi :

- Interpreter Ruby pada media yang akan anda gunakan baik komputer , Handphone atau menggunakan situs daring.
- Memperlajari materi sebelumnya , Struktur Dasar / Materi setaranya .
- Saran, gunakan FOSS ( Free – Open Source Software ) / Perangkat Lunak Bebas Gratis.

# Syarat Mental



- Persiapkan mental anda , jadi pelajar yang proaktif bukan pengemis yang reaktif
- Gunakan Akal dan Daya Kritis anda
- Berasa Ingin tahu dan eksplorasi
- Hadapi masalah , pecahkan serta berani mengotori tangan sendiri
- Jadila penanya yang cerdas , karena belajar dan pahami terlebih dahulu yang anda ingin tanyakan suatu kebermanfaatan.



# Bacaan Lanjutan

- Berfikir Komputasional
- Pseudo-Code dan UML
- Clean Code : DRY , KISS , SOLID , dan lainnya
- Struktur Data
- Pemrograman Berorientasi Objek





**Pemrograman**





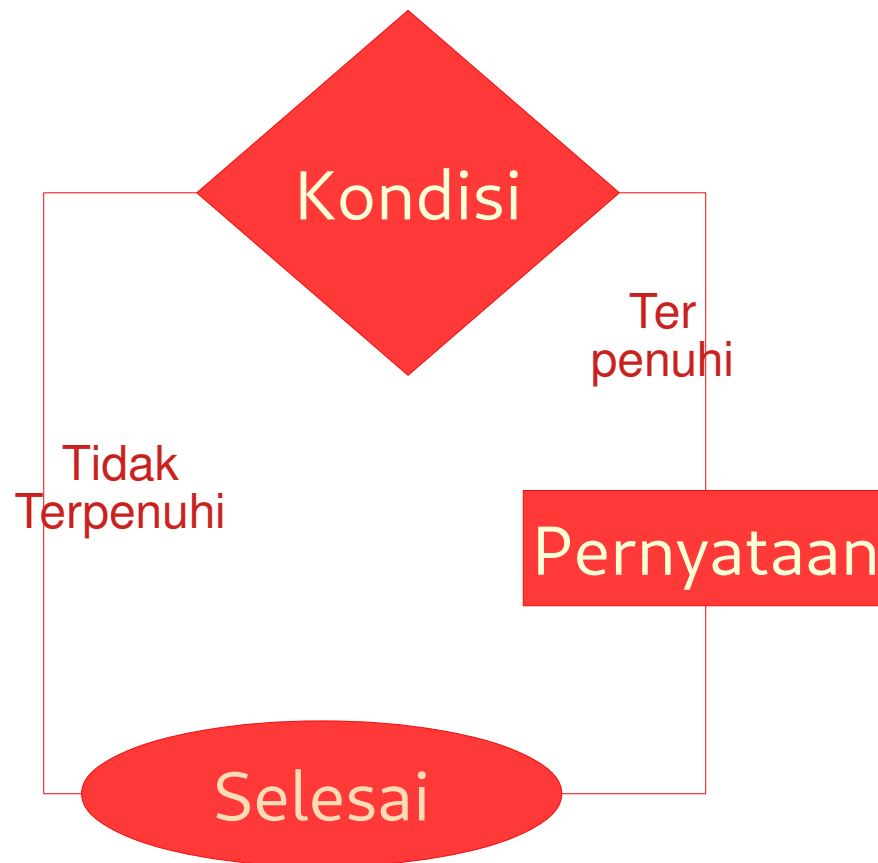
# Percabangan

- Digunakan untuk pengambilan keputusan pada suatu kasus.
- Percabangan melakukan pengujian terhadap suatu kondisi, kemudian menentukan blok atau baris instruksi kode yang akan dieksekusi.
- Pengujian kondisi dilandasi oleh pernyataan boolean , True / False. Dan dihubungkan oleh operator logika.

# Pernyataan IF

- Percabangan digunakan untuk menentukan langkah kerja, dalam Bahasa Pemrograman.
- Mengeksekusi suatu blok apabila kondisi terpenuhi.
- Kalau Tidak, maka blok tersebut diabaikan.
- Keluaran keputusan terdiri atas True / False ( Benar / Salah ).

# Penggambaran Diagram Alur Pernyataan IF



# Bentuk Umum

```
1 if kondisi
2   # blok kode
3   #< pernyataan >
4 end
```

# Contoh

```
1 print "Masukan Angka > 4 : "  
2 if gets.to_i > 4  
3   puts "Angka yang dimasukan lebih besar dari 4"  
4 end
```

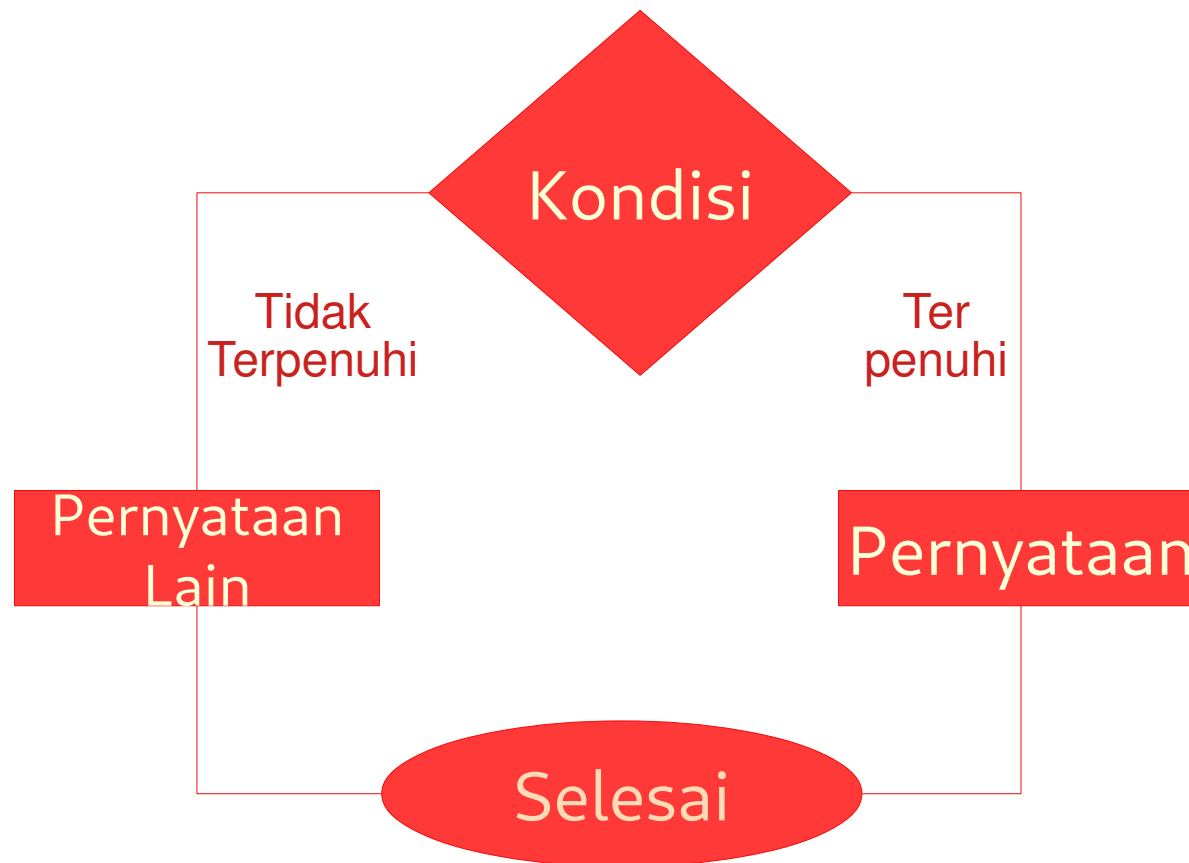
Bila pernyataan memenuhi (benar)  
blok ini yang akan dieksekusi

# Pernyataan IF-ELSE

- Memutuskan mengeksekusi blok kode jika kondisi terpenuhi
- Jika tidak terpenuhi maka akan mengeksekusi kondisi lain. Blok kode pada else.



# Penggambaran Diagram Alur Pernyataan IF-ELSE



# Bentuk Umum

```
1 if kondisi
2     # pernyataan
3 else
4     # pernyataan lain
5 end
```

# Contoh

```
1 print "Persyaratan Naik Rollercoaster (cm): "  
2 tinggi = gets.to_i  
3  
4 #kondisi  
5 if tinggi >= 165  
6   #pernyataan  
7   puts "[*] Anda diperbolehkan menaiki rollercoaster"  
8 else  
9   # pernyataan lain  
10  puts "[!!] Tinggi anda tidak cukup untuk naik rollercoaster"  
11 end
```

Bila pernyataan memenuhi (benar)  
blok ini yang akan dieksekusi

Bila pernyataan tidak memenuhi  
(salah) blok ini yang akan dieksekusi

# Pernyataan IF-ELSIF

- Digunakan untuk memutuskan lebih dari dua kondisi.

# Penggambaran Diagram Alur Pernyataan IF-ELSIF



# Bentuk Umum

```
1 if kondisi_pertama
2     # pernyataan
3 elsif kondisi_kedua
4     # pernyataan
5 else
6     # pernyataan
7 end
```



# Contoh

```
1 print " Suhu saat senja (celsius): "  
2 temperatur_senja = gets.to_f  
3  
4 if durasi_hujan => 30.2  
5     puts " Kehangatan senja menyelimuti naunganya"  
6 elsif durasi_hujan < 25.2  
7     puts " Senja dengan aura sejuk yang mengundang beku"  
8 else  
9     puts " Senja yang adem "  
10 end
```

Bila pernyataan memenuhi (benar) blok ini yang akan dieksekusi

Bila pernyataan memenuhi (benar) blok ini yang akan dieksekusi  
Dan tidak memenuhi Kondisi sebelumnya

Ini dieksekusi jika semua sebelumnya Kondisi tidak terpenuhi

# Pernyataan CASE-WHEN

- Pernyataan ini mengeksekusi bila nilai cocok dengan kondisi yang diberikan.
- Percabangan ini , tidak berlandasan dengan true/false , melainkan kecocokan nilai.
- Digunakan untuk kondisi banyak pilihan.
- Bentuk alternatif dari IF-ELSIF.

# Penggambaran Diagram alur Penyataan CASE-WHEN



# Bentuk Umum

```
1 case < nilai >
2     when < pilihan >
3         # blok eksekusi
4     when < pilihan >
5         # blok eksekusi
6     else
7         # blok eksekusi
8 end
```

# Contoh

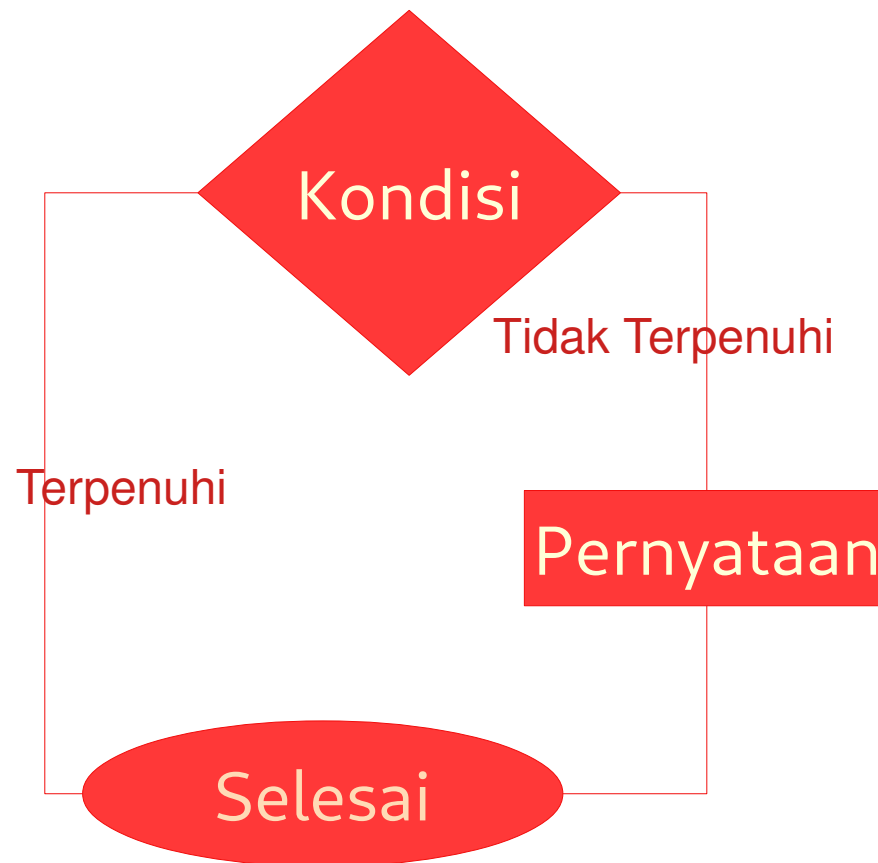
```
1 print "Kondisi lingkungan saat ini :"  
2 kondisi_lingkungan = gets.to_s.downcase  
3 case kondisi_tubuh  
4     when "panas"  
5         puts "Cari tempat sejuk "  
6     when "dingin"  
7         puts "Cari tempat hangat"  
8     else  
9         puts "Mungkin kamu demam"  
10 end
```

# Pernyataan **UNLESS**

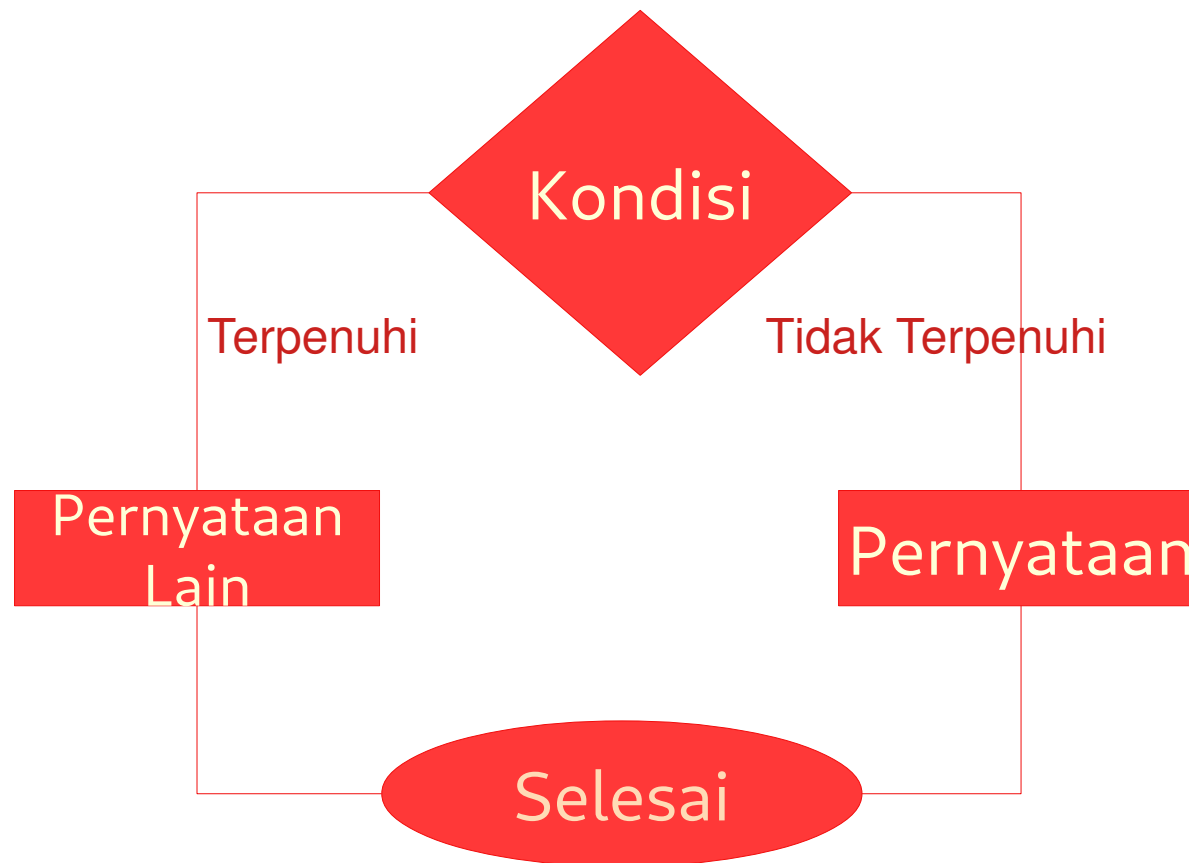
- Negasi dari pernyataan IF dan IF-ELSE, dieksekusi bila salah
- Tidak memiliki banyak situasi seperti elsif.
- Hanya bisa tidak memiliki / satu else sebagai pernyataan alternatif.



# Penggambaran Diagram Alur pernyataan UNLESS



# Penggambaran diagram alur Pernyataan UNLESS-ELSE



# Bentukan Umum

```
1 unless <kondisi>  
2   <pernyataan>  
3 end
```

# Bentukan Umum

```
1 unless <kondisi>  
2   <pernyataan>  
3 else  
4   <pernyataan>  
5 end
```

# Contoh

```
1 sakit = false
2 unless sakit
3   puts " Sehat "
4 else
5   puts " Sakit "
6 end
```



**Terima Kasih**



# Dalam nama perjuangan

- Materi ini dibuat sebagai bentuk perlawanan terhadap komersialisasi pendidikan dan pengendalian informasi
- Menjunjung kebebasan informasi dan pencerdasan umum
- Hak cipta bebas merdeka , setiap orang dianjurkan dan dinasehatkan untuk mengopi ,mencetak , mengganda, menyebarkan isi serta materi – materi didalamnya.