

BASIS DATA



**ALJABAR RELASIONAL
(RELATIONAL ALGEBRA)**

[Aljabar Relasional]

- Yaitu sekumpulan operasi yang digunakan untuk melakukan proses manipulasi data dalam rangka untuk mendapatkan informasi yang diperlukan dari database

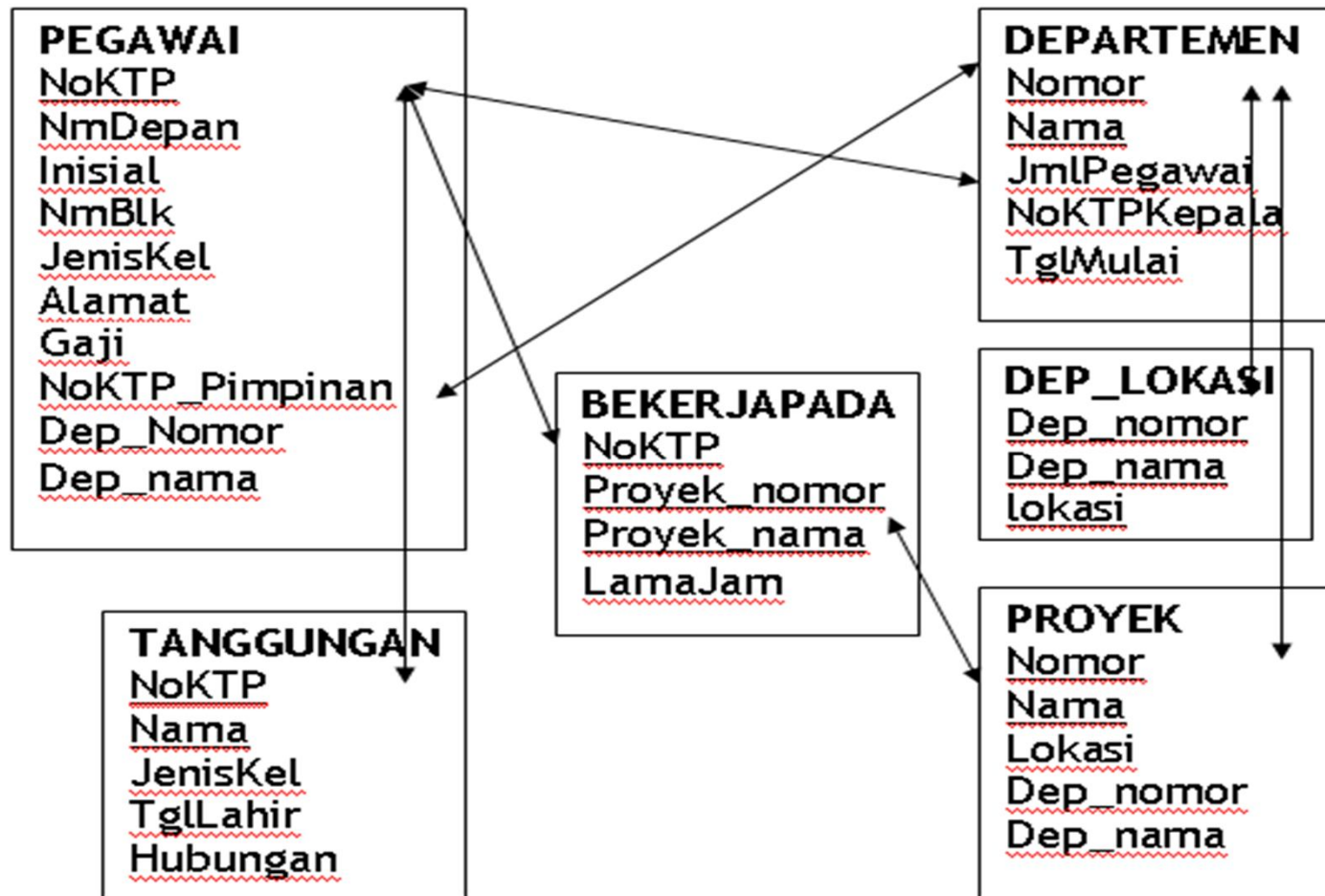
[Aljabar Relasional]

Secara umum dibagi menjadi 2:

- operasi himpunan (UNION, INTERSECTION, DIFFERENCE, dan CARTESIAN PRODUCT)
- operasi yang dikembangkan secara khusus untuk database relational. (SELECT, PROJECT dan JOIN)

Istilah Relasi, dalam bahasan ini dipergunakan untuk penamaan tabel beserta datanya baik yang murni maupun yang sudah dilakukan modifikasi dengan operasi-operasi aljabar relasional.

[Contoh Database]



[Operasi SELECT (Selection)-1]

- Untuk memilih baris tertentu dari sebuah himpunan baris data (record) yang memenuhi kondisi dan membuang baris yang lain.
- Notasi

$\sigma_{\text{<kondisi pilihan>}} (\text{<nama relasi>})$

[Operasi SELECT (Selection)-2]

Contoh

- $\sigma_{\text{dep_nomor}=4}(\text{PEGAWAI})$
untuk memilih sub himpunan pegawai yang bekerja departemen nomor 4
- $\sigma_{\text{gaji}>30000 \text{ AND dep_nomor}=3}(\text{PEGAWAI})$
untuk memilih sub himpunan pegawai yang memiliki gaji lebih dari 30000 yang bekerja di departemen 3

[Operasi Project (Projection) -1]

- Untuk memilih attribut (kolom) tertentu dari himpunan / subhimpunan dan membuang yang lain
- Notasi

$\pi_{\langle \text{daftar attribut} \rangle} (\langle \text{nama relasi} \rangle)$

[Operasi Project (Projection) -2]

- Jika <daftar atribut> tidak menyertakan primary key, maka dimungkinkan akan terjadi duplikasi. Duplikasi ini akan dihilangkan sehingga hanya sekali yang muncul.

Contoh

- Untuk memilih atribut JenisKel dan Gaji dari tabel Pegawai:

$\pi_{\text{JenisKel, Gaji}} (\text{Pegawai})$

[Urutan Operasi - 1]

- Sebuah operasi bisa dituliskan dalam bentuk beberapa ekspresi aljabar relasional dengan mengelompokkan untuk tiap-tiap operasi dan memberi nama

Contoh

$\pi_{nmDepan, nmBlk, gaji} (\sigma_{dep_nomor=5}(PEGAWAI))$

Mengambil informasi nama depan, nama belakang dan gaji dari pegawai-pegawai yang bekerja di departemen nomor 5

[Urutan Operasi - 2]

bisa ditulis dalam bentuk:

$PEG_DEP5 \leftarrow \sigma_{dep_nomor=5} (PEGAWAI)$

(tabel pegawai dipilah dulu hanya dengan mengambil yg memenuhi dep_nomor=5 dan disimpan dalam relasi dengan nama PEG_DEP5)

$HASIL \leftarrow \pi_{nmDepan, nmBlk, gaji} (PEG_DEP5)$

(kemudian setelah itu, relasi PEG_DEP5 dipilah hanya diambil kolom-kolom nmDepan, nmBlk dan gaji)

[Perubahan Nama Atribut]

- Untuk mengubah nama atribut dari sebuah relasi yang merupakan hasil dari operasi aljabar relasional, ditulis nama atribut baru dalam huruf besar

Contoh

$TEMP \leftarrow \sigma_{\text{dep_nomor}=5} (PEGAWAI)$

$RBARU (NAMADEPAN, NAMABELAKANG, GAJI) \leftarrow \pi_{nmdepan, nmblk, gaji} (TEMP)$

[Operasi Himpunan - 1]

- **UNION: notasi $\rightarrow R \cup S$**

Relasi yang menggabungkan semua baris di R atau S dengan meniadakan duplikasi

- **INTERSECTION: notasi $\rightarrow R \cap S$**

Relasi yang terdiri dari baris yang ada di R dan juga ada di S

- **DIFFERENCE: notasi $\rightarrow R - S$**

Relasi yang terdiri dari semua baris di R, tetapi tidak ada di S

Relasi yang dioperasikan dengan operasi-operasi di atas harus memiliki jumlah atribut dan jenis yang sama (union compatible)

[Operasi Himpunan - 2]

- CARTESIAN PRODUCT $\rightarrow R \times S$

Relasi yang terdiri dari kombinasi baris yang terdapat di R dan S. Yang mana setiap baris di R digabungkan dengan setiap baris di S

Contoh

Mengambil nama-nama pegawai wanita beserta nama-nama tanggungan masing-masing:

[Operasi Himpunan - 3]

Jawab

```
PEG_WAN  $\leftarrow \sigma_{\text{JenisKel}='p'} (\text{PEGAWAI})$   
NAM_PEG_WAN  $\leftarrow \pi_{\text{nmDep}, \text{nmBlk}, \text{NoKTP}} (\text{PEG_WAN})$   
TG_PEG  $\leftarrow \text{NAM_PEG_WAN} \times \text{TANGGUNGAN}$   
AKTUAL_TG  $\leftarrow \sigma_{\text{NoKTP}=\text{Tg\_NoKTP}} (\text{TG_PEG})$   
HASIL  $\leftarrow \pi_{\text{nmDepan}, \text{nmBlk}, \text{Tg\_Nama}} (\text{AKTUAL_TG})$ 
```

[Operasi Join - 1]

- Digunakan untuk mengkombinasikan baris-baris yang berhubungan dari dua relasi menjadi baris-baris tunggal.
- Notasi:
$$R \mid x \mid \langle \text{kondisi join} \rangle S$$
- Kondisi join dalam bentuk:
$$\langle \text{kondisi} \rangle \text{ AND } \langle \text{kondisi} \rangle \text{ AND } \dots \text{ AND } \langle \text{kondisi} \rangle$$
- Operator perbandingan: $\{=, <, \leq, >, \geq, \neq\}$

[Operasi Join - 2]

Contoh

Mendapatkan nama departemen beserta nama pimpinan masing-masing

```
DEPT (dep_nomor, dep_nama, dep_nama,  
dep_noKTP, dep_TglMulai, dep_JmlPegawai) ←  
DEPARTEMEN
```

```
PIM_DEP ← DEPT |x|dep_noKTP=noKTP PEGAWAI
```

```
HASIL ←  $\pi_{\text{dep\_nama, nmDepan, nmBlk}}$  (PIM_DEP)
```


[Operasi Join - 3]

- Operasi join dengan kondisi join secara umum disebut dengan *Theta Join*.
- Kondisi join yang operator pembandingannya berupa =, disebut dengan *Equijoin*

[Natural Join - 1]

- Yakni operasi equijoin yang mana yang mana pasangan atribut-attribut yang di'join'kan **memiliki nama yang sama**. Jika tidak sama, perlu dilakukan pengubahan nama atribut (rename) terlebih dahulu
- Notasi:

$$R \underset{(<\text{list 1}>),(<\text{list 2}>)}{*} S$$

- List 1 adalah atribut-attribut R dan list 2 adalah atribut-attribut S. List 1 dan list 2 membentuk perbandingan equality dari pasangan atribut yang memiliki nama yang sama, yang mana semua perbandingan tersebut di 'AND' kan

[Natural Join - 2]

Contoh

DEPT (dep_nomor, dep_nama, dep_jmlPegawai,
dep_noKTP) \leftarrow DEPARTEMEN

PROY_DEPT \leftarrow PROYEK * DEPT