

# IF3111 - Model Relasional

Tricya Widagdo  
Departemen Teknik Informatika  
Institut Teknologi Bandung



# Model Relasional

- E. F. Codd, "A Relational Model of Data for Large Shared Data Banks", 1970.
- Model Relasional terkait dengan tiga aspek: struktur data, integritas data, dan manipulasi data.
- Sistem Manajemen Basis Data Relasional (RDBMS):
  - data dilihat pemakai sebagai sejumlah tabel (*relation*)
  - operator yang dikenakan terhadap data akan membentuk tabel baru, dan paling tidak menangani SELECT, PROJECT, dan JOIN
- Hal yang perlu diperhatikan:
  - tabel merupakan struktur logik atau abstraksi struktur fisik
  - informasi dinyatakan dalam bentuk nilai data
  - semua nilai data bersifat atomik/skalar

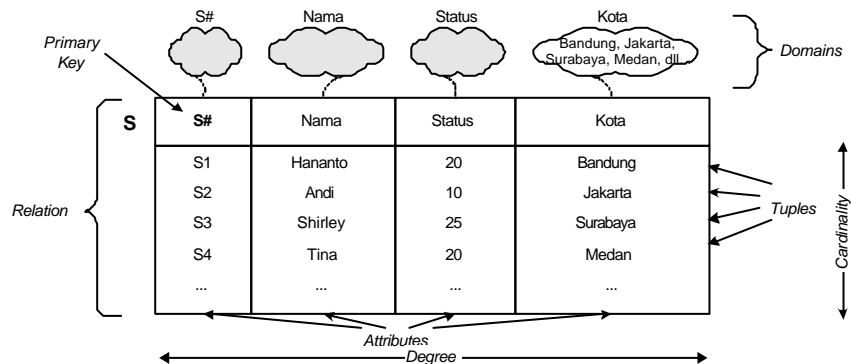


**Model Relasional**, merupakan teori formal yang mendasari sistem relasional, adalah suatu cara untuk melihat data, yaitu cara untuk menampilkan data dan cara untuk memanipulasi data.

**Definisi** RDBMS yang diberikan di atas adalah definisi minimal, yang sudah semakin dikembangkan hingga saat ini.

**Informasi** tidak dinyatakan dalam bentuk pointer. Keterkaitan dinyatakan dalam bentuk kemunculan data yang sama. + contoh dari basis data contoh.

## Struktur Data - Peristilahan



- *Primary key* adalah identifikator unik bagi sebuah tabel
- *Domain* adalah kumpulan nilai yang mungkin bagi sebuah atribut

## Basic Structure

- Formally, given sets  $D_1, D_2, \dots, D_n$  a **relation**  $r$  is a subset of  $D_1 \times D_2 \times \dots \times D_n$   
Thus a relation is a set of n-tuples  $(a_1, a_2, \dots, a_n)$  where each  $a_i \in D_i$
- Example: if  
 $customer-name = \{Jones, Smith, Curry, Lindsay\}$   
 $customer-street = \{Main, North, Park\}$   
 $customer-city = \{Harrison, Rye, Pittsfield\}$   
Then  $r = \{ (Jones, Main, Harrison),$   
 $(Smith, North, Rye),$   
 $(Curry, North, Rye),$   
 $(Lindsay, Park, Pittsfield)\}$   
is a relation over  $customer-name \times customer-street \times customer-city$

## Definisi Relasi

Sebuah relasi,  $R$ , terhadap sekumpulan domain  $D_1, D_2, \dots, D_n$  terdiri dari dua bagian, yaitu kepala (*heading*) dan badan (*body*).

- *Heading* terdiri dari sekumpulan atribut (atau lebih tepatnya pasangan <nama-atribut,nama-domain>) sedemikian sehingga setiap atribut  $A_j$  berkorespondensi dengan sebuah domain  $D_j$ .

Kepala mendefinisikan skema relasi.

$\{ \langle A_1:D_1 \rangle, \langle A_2:D_2 \rangle, \dots, \langle A_n:D_n \rangle \}$

- *Body* terdiri dari sekumpulan *tuples*. Setiap *tuple* terdiri dari sekumpulan pasangan <nama-atribut:nilai-atribut> yang mewakili setiap nama-atribut yang muncul pada kepala.

$\{ \langle A_1:v_{i1} \rangle, \langle A_2:v_{i2} \rangle, \dots, \langle A_n:v_{in} \rangle \},$

$i = 1, 2, \dots, m; m = \text{jumlah tuple}$



**Kepala relasi** ekuivalen dengan baris judul kolom pada tabel, sedangkan badan relasi ekuivalen dengan baris-baris data yang terkandung di dalam tabel.

Pada bagian kepala biasanya hanya dituliskan nama atributnya, sedangkan pada bagian badan biasanya hanya dituliskan nilai atributnya yang diapit tanda kurung () untuk setiap tuple (dengan asumsi urutan penulisan nilai atribut sesuai dengan urutan penulisan nama atribut di bagian kepala).

## Attribute Types

- Each attribute of a relation has a name
- The set of allowed values for each attribute is called the **domain** of the attribute
- Attribute values are (normally) required to be **atomic**, that is, indivisible
  - E.g. multivalued attribute values are not atomic
  - E.g. composite attribute values are not atomic
- The special value *null* is a member of every domain
- The null value causes complications in the definition of many operations
  - we shall ignore the effect of null values in our main presentation and consider their effect later

## Relation

### Variable vs Value

- *Relation Variable* adalah sebuah relasi yang skemanya terdefinisi  
 $R = (A_1, A_2, \dots, A_n)$  adalah sebuah skema relasi (*variable*)
- *Relation Value* adalah nilai yang dikandung suatu relasi pada suatu waktu; disebut juga instans  
 $r(R)$  adalah sebuah instans (*value*) dari skema  $R$

Berdasarkan keberadaannya, relasi terbagi 3:

- *Base Relation* adalah relasi yang skemanya terdefinisi dan benar-benar ada pada basis data
- *Derived Relation* adalah relasi yang diturunkan dari relasi-relasi lainnya dengan menggunakan ekspresi relasional
- *View* adalah *derived relation* yang memiliki nama



Untuk selanjutnya, istilah relasi yang berdiri sendiri akan ekuivalen dengan istilah *relation value*.

**Base Relation** berkorespondensi dengan level konseptual pada level abstraksi data (arsitektur ANSI/SPARC), sedangkan **view** berkorespondensi dengan level eksternal.

## Properti Relasi

- Tidak terdapat *tuple* yang terduplikasi
- *Tuple* tidak terurut dari atas ke bawah
  - » akibatnya, tidak dikenal pengalamatan posisi (seperti *tuple* pertama, *tuple* berikutnya, dll.)
- Atribut tidak terurut dari kiri ke kanan
  - » akibatnya, juga tidak dikenal pengalamatan posisi atribut (seperti atribut pertama, atribut berikutnya, dll.)
- Semua nilai atribut adalah atomik
  - » menghasilkan relasi yang ternormalisasi (*normalized*)

Setiap relasi memiliki *predicate* yang merupakan kriteria kebenaran *update* terhadap relasi.

**Predikat** menjamin bahwa setiap saat sebuah relasi hanya mengandung *tuple* yang mengakibatkan evaluasi predikat bernilai benar.



## Jenis Relasi

- Relasi bernama (*named relation*): variabel relasi yang terdefinisi di basis data.
- Relasi dasar (*based relation*)
- Relasi turunan (*derived relation*)
- *Expressible relation*: yang diperoleh dari sekumpulan relasi bernama dengan menggunakan ekspresi relasional.
- *View*
- *Snapshot*: relasi turunan yang memuat definisi + data yang di *update* secara periodik.
- Hasil *query*: relasi turunan akhir.
- Hasil antara (*intermediate*): relasi turunan antara.
- Relasi tersimpan (*stored*): penyimpanan fisik



Jenis yang diberikan disini adalah jenis yang dapat muncul pada sistem relasional. Tetapi, tidak semua sistem mendukung semua jenis yang diberikan.

## Basis Data Relasional

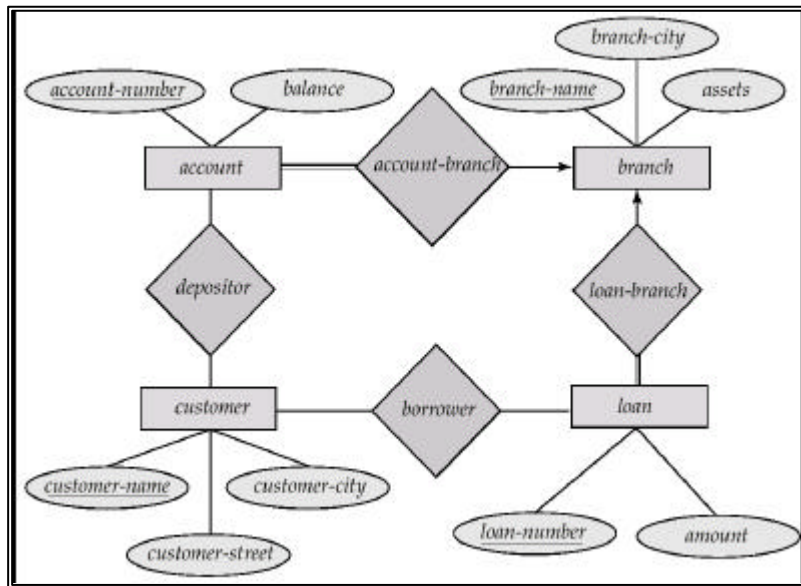
Definisi Formal:

adalah sebuah basis data yang dilihat oleh pemakainya sebagai sekumpulan relasi, yaitu variabel relasi, yang ternormalisasi dengan derajat yang beragam.

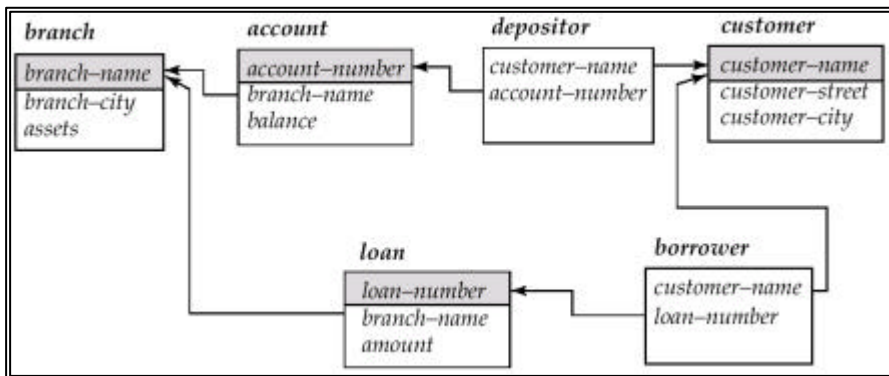
Relasi merupakan subset dari hasil perkalian kartesian antara domain-domain yang terdefinisi bagi relasi tersebut.



## E-R Diagram for the Banking Enterprise



## Schema Diagram for the Banking Enterprise



## Bahasa Query

- Bahasa yang digunakan untuk mengambil informasi dari basis data
- Kategori bahasa:
  - procedural
  - non-procedural
- Bahasa relasional murni:
  - Relational Algebra
  - Tuple Relational Calculus
  - Domain Relational Calculus
- Bahasa relasional murni merupakan dasar pembangunan bahasa query yang banyak digunakan