



# STK371

# Komputasi Statistik 1

*Materi 4. Menyiapkan Data Statistik dalam R*

<https://www.stat.ipb.ac.id/agusms/index.php/stk371/>

- Tipe File:
  - Text File
  - Spreadsheet
  - Database relasional
  - Bigdata
  - dll
- Umumnya data disiapkan dalam bentuk database relasional

# DATABASE RELASIONAL

- Jenis database relasional ini merupakan jenis database yang paling sederhana disbanding database Hierarki (***Hierarchical Database Model***) dan database Jaringan (***Network Database Model***)
- Sebuah database relasional ini dirancang untuk memiliki keterkaitan antar tabelnya
- Ada dua cara menyiapkan database relasional:
  - Melalui Proses Normalisasi
  - Melalui Perancangan ER Diagram



# PROSES NORMALISASI

# APA & KENAPA NORMALISASI DATABASE?

- Sebuah teknik mengelola data dalam database dengan **pendekatan sistematis** untuk **mendekomposisi tabel**
- Dua tujuan utama:
  - Eliminating **redundant** (useless) data.
  - Ensuring **data dependencies** make sense i.e data is logically stored.
- Digunakan untuk memastikan bahwa **database** yang dibuat **berkualitas baik**

# NORMALISASI DATABASE

- Sebuah tabel dikatakan baik (efisien) atau normal jika memenuhi 3 kriteria sbb:
  - Jika ada dekomposisi (penguraian) tabel, maka dekomposisinya harus dijamin aman (*Lossless-Join Decomposition*): setelah tabel tersebut diuraikan/didekomposisi menjadi **tabel-tabel baru**, tabel-tabel baru tersebut bisa **menghasilkan tabel semula** dengan sama persis
  - Terpeliharanya ketergantungan fungsional pada saat perubahan data (*Dependency Preservation*)
  - Tidak memiliki kandidat *key* yang tumpang tindih: Hanya ada satu super key

# NORMALISASI DATABASE

Bentuk-bentuk Normal:

1. Bentuk Normal Tahap Pertama (1st Normal Form / 1NF)
2. Bentuk Normal Tahap Kedua (2nd Normal Form / 2NF)
3. Bentuk Normal Tahap (3rd Normal Form / 3NF)
4. **Boyce-Code Normal Form (BCNF) ← Biasanya cukup sampai bentuk ini**
5. Bentuk Normal Tahap (4th Normal Form / 4NF)
6. Bentuk Normal Tahap (5th Normal Form / 5NF)

# BENTUK NORMAL PERTAMA (1NF)

□ Harus memenuhi 4 aturan:

- It should only have single(atomic) valued attributes/columns.
- Values stored in a column should be of the same domain
- All the columns in a table should have unique names.
- And the order in which data is stored, does not matter.



# BENTUK NORMAL PERTAMA (1NF)

- Ilustrasi:
  - Belum Normal:

Student#	Advisor	Adv-Room	Class1	Class2	Class3
1022	Jones	412	101-07	143-01	159-02
4123	Smith	216	201-01	211-02	214-01

- Normal Pertama:

Student#	Advisor	Adv-Room	Class#
1022	Jones	412	101-07
1022	Jones	412	143-01
1022	Jones	412	159-02
4123	Smith	216	201-01
4123	Smith	216	211-02
4123	Smith	216	214-01

# BENTUK NORMAL PERTAMA (1NF)

- Ilustrasi lain:

BUKAN 1NF

NRP	Nama	Hobi
12020001	Heri Susanto	Sepakbola, Membaca Komik, Berenang
12020013	Siti Zulaiha	Memasak, membuat program komputer
12020018	Heri Susanto	Membuat program Komputer
12020015	Dini Susanti	Menjahit, Membuat Roti

atau:

NRP	Nama	Hobi1	Hobi2	Hobi3
12020001	Heri Susanto	Sepakbola	Membaca Komik	Berenang
12020013	Siti Zulaiha	Memasak	Membuat Program Komputer	
12020018	Heri Susanto	Membuat Program Komputer		
12020015	Dini Susanti	Menjahit	Membuat Roti	

# BENTUK NORMAL PERTAMA (1NF)

- 1NF:

NRP	Nama	Hobi
12020001	Heri Susanto	Sepakbola
12020001	Heri Susanto	Membaca Komik
12020001	Heri Susanto	Berenang
12020013	Siti Zulaiha	Memasak
12020013	Siti Zulaiha	Membuat program Komputer
12020018	Heri Susanto	Membuat Program Komputer
12020015	Dini Susanti	Menjahit
12020015	Dini Susanti	Membuat Roti

## BENTUK NORMAL KEDUA (2NF)

- Bentuk normal 2NF terpenuhi dalam sebuah tabel jika telah memenuhi bentuk 1NF, dan semua atribut selain primary key, secara utuh memiliki Functional Dependency pada primary key
- Sebuah tabel tidak memenuhi 2NF, jika ada atribut yang ketergantungannya (Functional Dependency) hanya bersifat parsial saja (hanya tergantung pada sebagian dari primary key)
- Jika terdapat atribut yang tidak memiliki ketergantungan terhadap primary key, maka atribut tersebut harus dipindah atau dihilangkan

# Functional Dependency

- **Notasi:  $A \rightarrow B$**
- A dan B adalah atribut dari sebuah tabel. Berarti secara fungsional A menentukan B atau B tergantung pada A, jika dan hanya jika ada 2 baris data dengan nilai A yang sama, maka nilai B juga sama
- **Notasi:  $A \nrightarrow B$  atau  $A \not\rightarrow B$**

Adalah kebalikan dari notasi sebelumnya.

# Functional Dependency

- Ilustrasi: Perhatikan Tabel berikut

<u>Namakul</u>	<u>Nrp</u>	<u>namaMhs</u>	<u>NiHuruf</u>
Struktur Data	980001	Ali Akbar	A
Struktur Data	980004	Indah Susanti	B
Basis Data	980001	Ali Akbar	
Basis Data	980002	Budi Haryanto	
Basis Data	980004	Indah Susanti	
Bahasa Indonesia	980001	Ali Akbar	B
Matematika I	980002	Budi Haryanto	C

# Functional Dependency

## ➤ **Nrp** → **namaMhs**

Karena untuk setiap nilai **nrp** yang sama, maka nilai **namaMhs** juga sama.

## ➤ **{NamaKul, nrp}** → **NiHuruf**

Karena atribut **Nihuruf** tergantung pada **Namakul** dan **nrp** secara bersama-sama. Dalam arti lain untuk **Namakul** dan **nrp** yang sama, maka **NiHuruf** juga sama, karena **Namakul** dan **nrp** merupakan key (bersifat unik).

## ➤ **NRP** ↗ **NamaKul**

## ➤ **Nrp** ↗ **NiHuruf**

# BENTUK NORMAL KEDUA (2NF)

- Ilustrasi (lanjutan dari 1NF)

Students:

<b>Student#</b>	<b>Advisor</b>	<b>Adv-Room</b>
1022	Jones	412
4123	Smith	216

Registration:

<b>Student#</b>	<b>Class#</b>
1022	101-07
1022	143-01
1022	159-02
4123	201-01
4123	211-02
4123	214-01



# BENTUK NORMAL KETIGA (3NF)

- Bentuk normal 3NF terpenuhi
  - jika telah memenuhi bentuk 2NF, dan
  - jika **tidak ada** atribut yang bukan kunci memiliki ketergantungan transitif (tidak secara langsung) pada primary key.

# BENTUK NORMAL KETIGA (3NF)

- Ilustrasi:

2NF

Students:

Student#	Advisor	Adv-Room
1022	Jones	412
4123	Smith	216

3NF

Students:

Student#	Advisor
1022	Jones
4123	Smith

Faculty:

Name	Room	Dept
Jones	412	42
Smith	216	42

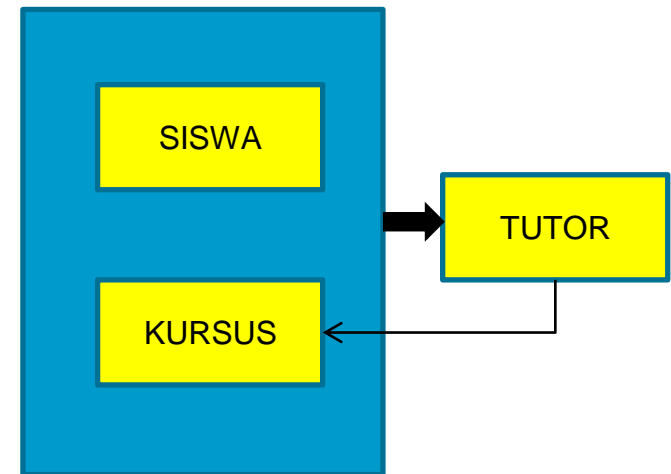
# BOYCE-CODE NORMAL FORM (BCNF)

- Bentuk BCNF terpenuhi dalam sebuah tabel, jika untuk setiap *functional dependency* terhadap setiap atribut atau gabungan atribut dalam bentuk:  $X \rightarrow Y$  maka **X** adalah *super key*
- tabel tersebut harus di-dekomposisi berdasarkan *functional dependency* yang ada, sehingga X menjadi *super key* dari tabel-tabel hasil dekomposisi
- Setiap tabel dalam BCNF merupakan 3NF. Akan tetapi setiap 3NF belum tentu termasuk BCNF . BCNF merupakan perbaikan bagi 3NF yang anomali.

# BOYCE-CODE NORMAL FORM (BCNF)

Siswa	Kursus	Tutor
Anwar	Bahasa Perancis	Pierre
Anwar	Bahasa Inggris	Richard
Budi	Bahasa Perancis	Pierre
Cecep	Bahasa Inggris	Suzanne

- ▶ Tabel tersebut tidak melanggar 3NF
- ▶ Seorang siswa dapat mengambil sejumlah kursus
- ▶ Tutor hanya mengajar pada sebuah kursus, satu kursus bisa diajar oleh banyak tutor
- ▶ Kunci kandidat:
  - ▶ Siswa + Kursus
  - ▶ Kursus + Tutor



Hubungan ketergantungan  
Jika baris Cecep dihapus,  
Informasi Suzanne sebagai  
tutor bahasa Inggris akan hilang

# BOYCE-CODE NORMAL FORM (BCNF)

- Dekomposisi:

Siswa	Tutor
Anwar	Pierre
Anwar	Richard
Budi	Pierre
Cecep	Suzanne

Tutor	Kursus
Pierre	Bahasa Perancis
Richard	Bahasa Inggris
Suzanne	Bahasa Inggris



# PERANCANGAN ER DIAGRAM

# ENTITY RELATIONSHIP (ER) DIAGRAM

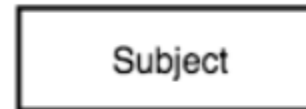
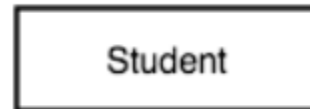
- ER Diagram merupakan representasi data secara visual yang menggambarkan bagaimana data berhubungan dengan yang lain
- Dalam ER Model, data dipisahkan ke dalam entiti, atribut dan hubungan antar entiti, dan semuanya direpresentasikan secara visual menggunakan ER diagram



- Data dijamin memenuhi bentuk 3NF

# KOMPONEN ER DIAGRAM

- **Entity**

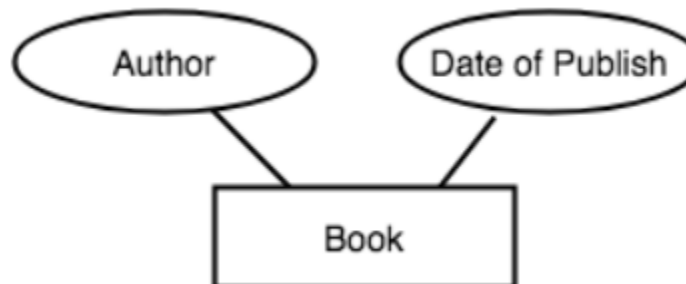


- **Relationships between Entities**



Relationship

- **Attributes**





# ER Diagram: Relationship

- There are three types of relationship that exist between Entities.

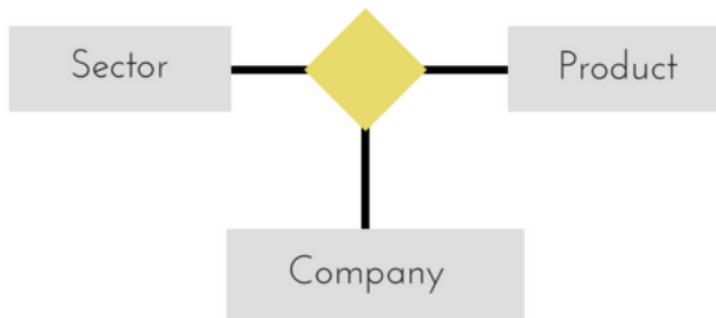
- Binary Relationship



- Recursive Relationship

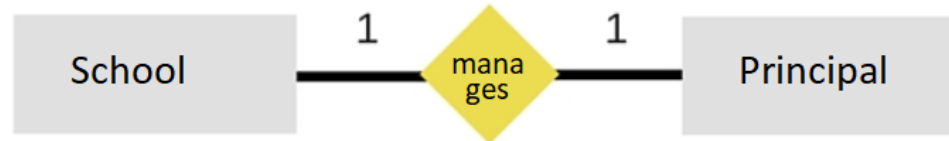


- Ternary Relationship

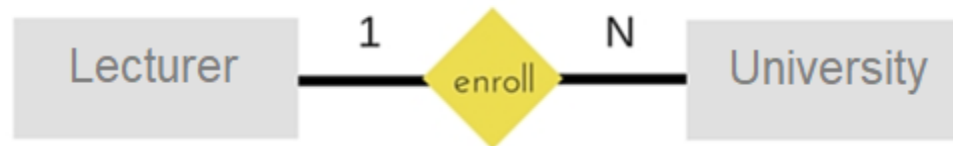


# ER Diagram: Binary Relationship

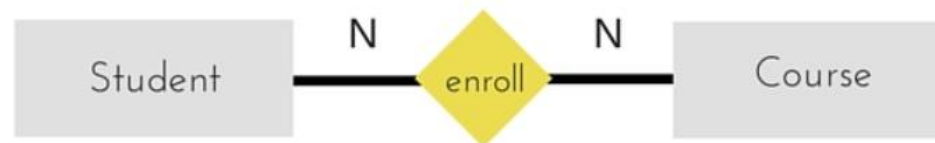
- **One to One Relationship**



- **One to Many Relationship / Many to One Relationship**



- **Many to Many Relationship**



# ER Diagram: Binary Relationship

- Setiap entiti akan menjadi tabel
- Relasi m-n akan menjadi tabel
  - **One to One Relationship: 2 tabel**
  - **One to Many Relationship / Many to One Relationship: 2 tabel**
  - **Many to Many Relationship: 3 tabel**



***Selesai...***