

Normalisasi





- Normalisasi adalah suatu teknik untuk mengorganisasi data ke dalam tabel-tabel untuk memenuhi kebutuhan pemakai di dalam suatu organisasi.



Pentingnya Normalisasi

- Karena adanya struktur database yang kurang bagus
 - Data yang sama tersimpan di beberapa tempat (file atau record)
 - Ketidakmampuan untuk menghasilkan informasi tertentu
 - Terjadi kehilangan informasi
 - Terjadi adanya **redundansi** (pengulangan) atau **duplikasi** data sehingga memboroskan ruang penyimpanan dan menyulitkan saat proses updating data
 - Adanya **NULL VALUE**



Tujuan Normalisasi

- Tujuan **normalisasi** adalah
 - Untuk menghilangkan kerangkapan data
 - Untuk mengurangi kompleksitas
 - Untuk mempermudah pemodifikasian data
 - menyempurnakan struktur table menjadi lebih baik
- Bentuk normalisasi yang sering digunakan adalah 1st NF, 2nd NF, 3rd NF, dan BCNF



FUNCTIONAL DEPENDENCY (FD)

- Untuk melakukan normalisasi, harus bisa menentukan terlebih dahulu **Functional Dependency (FD)** atau **Ketergantungan Fungsional**, khususnya dalam melakukan dekomposisi rancangan database.
- **Functional Dependency (FD)** dapat disimbolkan dengan:
 $A \rightarrow B$: artinya B memiliki ketergantungan dengan A
- Berarti A secara fungsional menentukan B atau B secara fungsional tergantung pada A.



FUNCTIONAL DEPENDENCY (FD)

- **Contoh:**

	Mata_Kuliah	NRP	Nama	Nilai
row 1	Aplikasi Web	7405040100	Deni Astikapuri	A
row 2	Aplikasi Web	7405040101	Uun Widiatmoko	A
row 3	Basis Data 1	7405040100	Deni Astikapuri	B
row 4	Basis Data 1	7405040102	Wasis Waskito Adi	B
row 5	Basis Data 1	7405040103	Imam Bukhori	A
row 6	Basis Data 2	7405040104	Aswina Rahayu Kurniati	A
row 8	Administrasi Basis Data	7405040101	Uun Widiatmoko	AB

Functional Dependency:

- NRP → Nama
- Mata_Kuliah, NRP → Nilai

Non Functional Dependency:

- Mata_Kuliah → NRP
- NRP → Nilai



Langkah Normalisasi





NORMALISASI 1NF

1st Normal Form (1NF)

- Merubah dari bentuk tabel tidak normal (**unnormalized** table) menjadi bentuk **normal (1NF)**.
- Suatu relasi R disebut **1st NF** jika dan hanya jika kondisi tablenya dari **unnormalized** dirubah ke bentuk **normal** dengan kondisi semua attribute value-nya harus **atomic** (tidak boleh ada attribute yang **composit** / **multivalue**)



Unnormalized table (tabel tidak normal)

Suatu tabel dikatakan **unnormalized** jika :

- a) Mempunyai penggandaan field yang sejenis

Contoh :

Tabel dibawah adalah tabel siswa mengambil mata kuliah

SISWA	NRP	Nama	MK1	MK2	MK3
--------------	------------	-------------	------------	------------	------------

- b) Elemen datanya memungkinkan untuk **null value** (kosong)

Contoh :

Tabel yang mencatat No. SIM yang dimiliki siswa

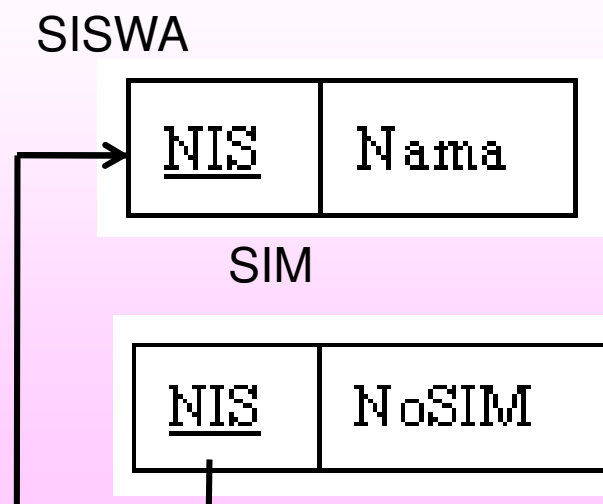
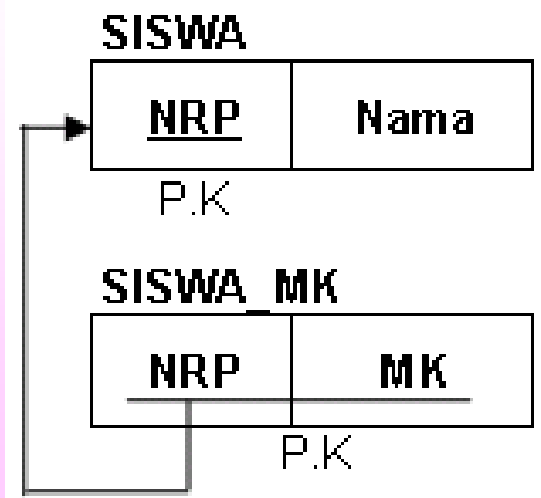
SISWA_SIM

<u>NIS</u>	Nama	No SIM
1	Budi	12345
2	Amin	
3	Irfan	67890
4	Bayu	



Tabel dalam bentuk normal 1NF, jika ...

- Suatu tabel dikatakan berada pada bentuk **normal I** jika ia tidak berada pada bentuk **unnormalized** table, dimana terjadi penggandaan field yang sejenis dan memungkinkan ada field yang null (kosong)





NIS	NAMA
1	BUDI
2	ANIS
3	IRFAN
4	BAYU

NIS	NOSIM
1	12345
3	67890



idSales	namaSales	telepon
ADN006	Oki	312311, 081232456
ADN007	Asfie	768987
ADN008	Tomi	345678, 762009, 0812666777
ADN009	Rizki	711000



idSales	namaSales	telepon
ADN006	Oki	312311
ADN006	Oki	081232456
ADN007	Asfie	768987
ADN008	Tomi	345678
ADN008	Tomi	762009
ADN008	Tomi	0812666777
ADN009	Rizki	711000



NORMALISASI 2NF

2st Normal Form (2NF)

- Normalisasi **2NF**: jika tabel berada dalam bentuk **Normal Pertama (1NF)** dan setiap atribut **bukan kunci bergantung penuh** pada **kunci primer**.
- Sehingga tidak ada atribut bukan kunci yang bergantung pada sebagian (parsial) kunci primer.



Syarat 2st Normal Form (2NF)

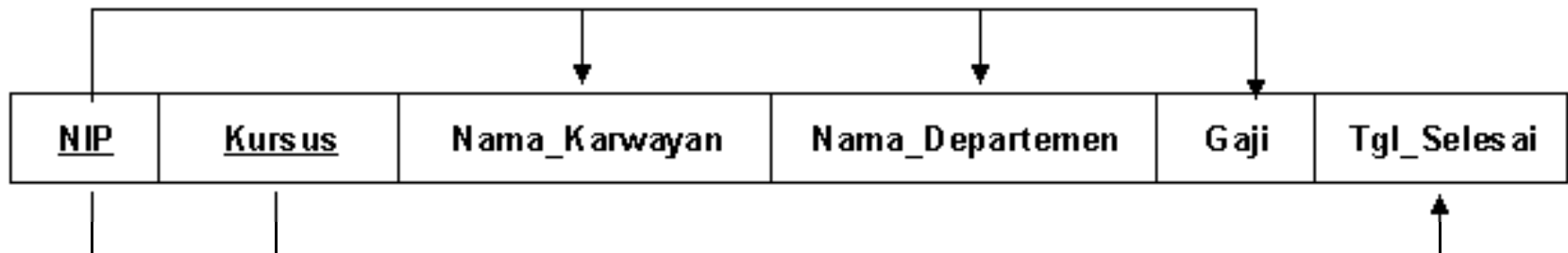
Syarat 2st Normal Form (2NF):

- Memenuhi kriteria tabel **Normal I (1NF)**
- Di dalam tabel tersebut tidak ada **Redundansi / Pengulangan** data dan **Null Value**.
- Field-field yang **bukan PK** adalah **Full Dependent** (bergantung penuh) pada **PK**.

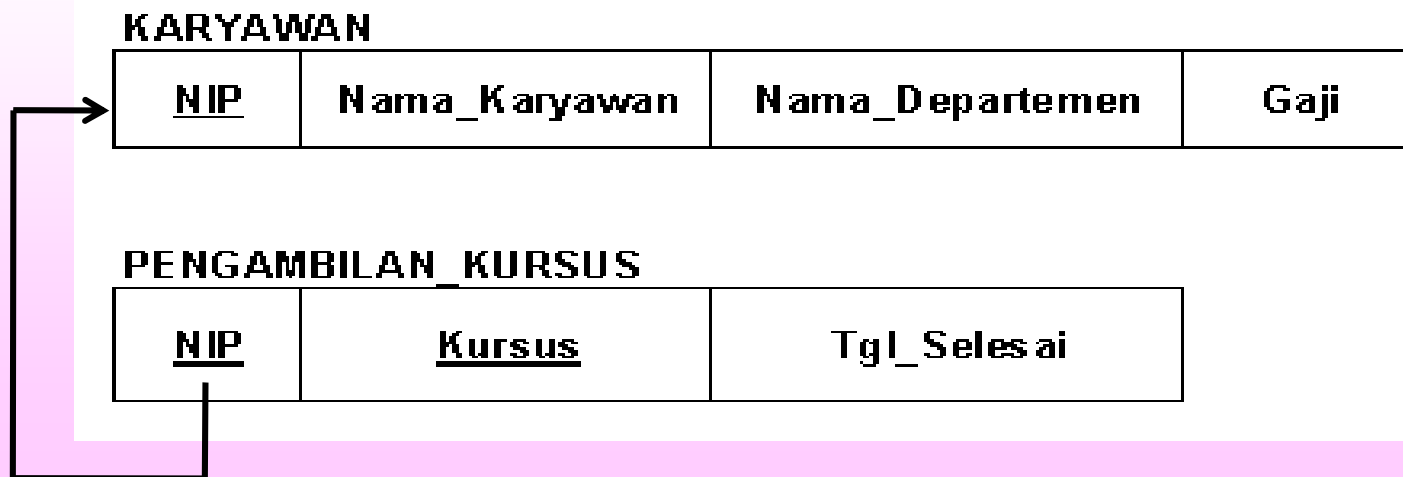


Contoh 2NF:

- Suatu format tabel Normal I (**1NF**) :



- Bentuk Normal II (**2NF**) : (Decompose)





Ilustrasi 2NF

- Suatu format tabel Normal I (**1NF**) :
(menghilangkan Redundansi)

<u>NIP</u>	Nama_Karyawan	Nama_Departemen	Gaji	<u>Kursus</u>	Tgl_Selesai
25210021	Ali Topan	Geologi Komputasi	2.000.000	AutoCAD Map	8-Oct-2002
25210021	Ali Topan	Geologi Komputasi	2.000.001	Potoshop	9-Oct-2002
25210022	James Bond	Pengeboran	1.250.000	3D MAX	9-Oct-2002
25210023	Cici Faramida	Geofisika Eksplorasi	1.500.000	3D MAX	9-Oct-2002
25210023	Cici Faramida	Geofisika Eksplorasi	1.500.001	ArcView	10-Dec-2002
25210024	Siti Nurhaliza	Sistem Informasi	2.500.000	Oracle	21-Sep-2002
25210024	Siti Nurhaliza	Sistem Informasi	2.500.001	SQL Server	21-Sep-2003

- Bentuk Normal II (**2NF**) : (Decompose)

KARYAWAN

<u>NIP</u>	Nama_Karyawan	Nama_Departemen	Gaji
25210021	Ali Topan	Geologi Komputasi	2.000.000
25210022	James Bond	Pengeboran	1.250.000
25210023	Cici Faramida	Geofisika Eksplorasi	1.500.000
25210024	Siti Nurhaliza	Sistem Informasi	2.500.000

PENGAMBILAN KURSUS

<u>NIP</u>	<u>Kursus</u>	Tgl_Selesai
25210021	AutoCAD Map	8-Oct-2002
25210021	Potoshop	9-Oct-2002
25210022	3D MAX	9-Oct-2002
25210023	3D MAX	9-Oct-2002
25210023	ArcView	10-Dec-2002
25210024	Oracle	21-Sep-2002
25210024	SQL Server	21-Sep-2003



NORMALISASI 3NF

3rd Normal Form (3NF)

- Suatu relasi R disebut **normal III (3rd NF)** jika berada dalam bentuk **normal II (2nd NF)** dan **tidak** dijumpai adanya **ketergantungan transitif (Transitive Dependency)**.
- **Kebergantungan Transitif (Transitive Dependency)** adalah ketergantungan fungsional antara 2 (atau lebih) atribut **bukan key** (kunci).



Syarat 3NF

Syarat 3NF:

- Harus berada dalam bentuk **normal II (2NF)**.
- Ketergantungan field-field yang bukan PK adalah harus secara **mutlak (full-dependent)**. Artinya **harus tidak ada transitive dependency** (ketergantungan secara transitif).



Contoh 3NF:

- Bentuk Normal ke Dua (2NF) :**

Tabel di samping sudah masuk dalam bentuk Normal 2. Akan tetapi kita lihat bahwa field **Nama** dan **Nilai** adalah **Full-Dependent** terhadap **NRP** yang bertindak sebagai **PK**. Berbeda dengan field **Keterangan** di atas yang **Dependent** kepada **NRP** akan tetapi **Tidak Mutlak**. Ia lebih dekat ketergantungannya dengan field **Nilai**. Karena field **Nilai** **Dependent** kepada **NRP** dan field **Keterangan** **Dependent** kepada **Nilai**, maka field **Keterangan** juga dependent kepada **NRP**. Ketergantungan yang demikian ini yang dinamakan **Transitive-Dependent** (dependent secara transitif atau **samar/tidak langsung**). Untuk itu dilakukan **Normalisasi III (3NF)**.

<u>NRP</u>	Nama	Nilai	Ke terangan
1	Budi	75	Baik
2	Amin	95	Istimewa
2	Irfan	85	Cukup baik
3	Bayu	40	Kurang

- Bentuk Normal ke Tiga (3NF) :**

<u>NRP</u>	Nama	Nilai
1	Budi	75
2	Amin	95
2	Irfan	85
3	Bayu	40

<u>Nilai</u>	Keterangan
90	Istimewa
80	Baik
70	Cukup baik
60	Lumayan
0	Kurang



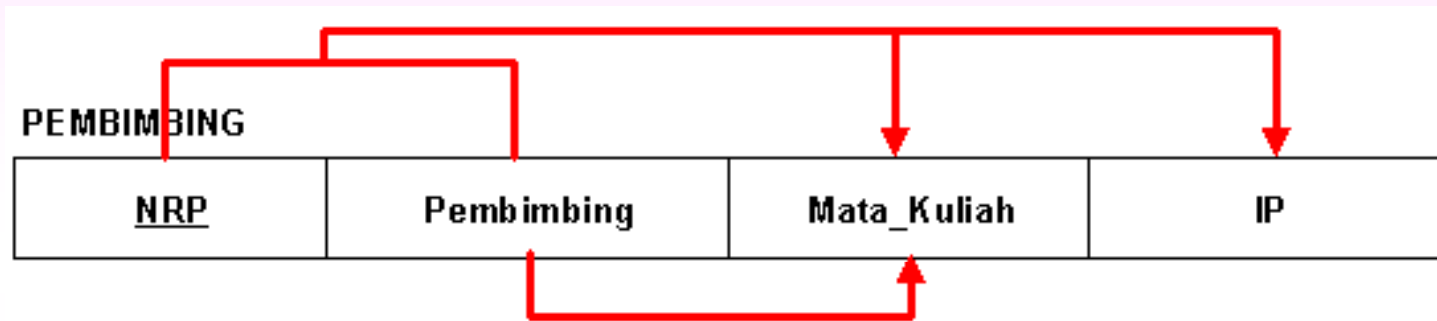
Boyce-Codd Normal Form (BCNF)

- Secara praktis, tujuan rancangan database adalah cukup sampai pada 3NF. Akan tetapi untuk kasus-kasus tertentu kita bisa mendapatkan rancangan yang lebih baik lagi apabila bisa mencapai ke BCNF.
- BCNF ditemukan oleh: **R.F. Boyce** dan **E.F. Codd**
- Suatu relasi R dikatakan dalam bentuk **BCNF**: jika dan hanya jika setiap **Atribut Kunci (Key)** pada suatu relasi adalah **Kunci Kandidat (Candidate Key)**.
- **Kunci Kandidat (Candidate Key)** adalah atribut-atribut dari entitas yang mungkin dapat digunakan sebagai kunci (key) atribut.
- **BCNF** hampir sama dengan **3NF**, dengan kata lain setiap **BCNF** adalah **3NF**.

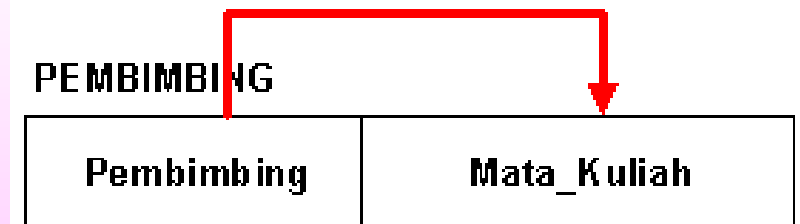
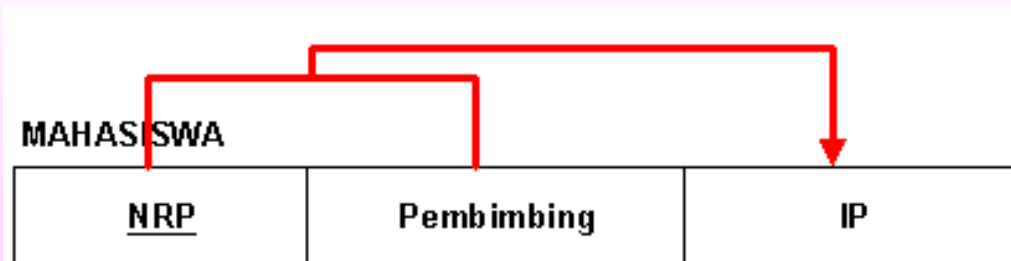


Contoh BCNF

- Normal II (**2NF**) :



- Normal III (**3NF**) atau BCNF





SOAL

- Normalisasikan Tabel Tidak Normal berikut sampai bentuk 3NF

KD_PELANGGAN	NM_PELANGGAN	KD_PROPERTI	ALM_PROPERTI	TG_PINJAM	TG_SELESAI	BIAYA	KD_PEMILIK	NM_PEMILIK
C1	JONI	K2	JL.SIDAKARYA 30	1 AGUSTUS 20013	30 AGUSTUS 2013	2500000	A1	RIAN
		K3	JL.BUAYA 22	1 NOPEMBER 2013	30 NOPEMBER 2013	3000000	A2	ROBI
C23	LEO	K2	JL.SIDAKARYA 30	1 JANURAI 2013	30 JANUARI 2013	2500000	A1	RIAN
		K9	JL. DRUPADI 21	1 JULI 2013	30 JULI 2013	3750000	A2	ROBI
		K3	JL.BUAYA 22	1 FEBRUARI 2013	30 FEBRUARI 2013	3000000	A2	ROBI



BENTUK NORMAL 1NF

KD_PELANGGAN	NM_PELANGGAN	KD_PROPRTI	ALM_PROPRTI	TG_PINJAM	TG_SELESAI	BIAYA	KD_PEMILIK	NM_PEMILIK
C1	JONI	K2	JL.SIDAKARYA 30	1 AGUSTUS 20013	30 AGUSTUS 2013	2500000	A1	RIAN
C1	JONI	K3	JL.BUAYA 22	1 NOPEMBER 2013	30 NOPEMBER 2013	3000000	A2	ROBI
C23	LEO	K2	JL.SIDAKARYA 30	1 JANURAI 2013	30 JANUARI 2013	2500000	A1	RIAN
C23	LEO	K9	JL. DRUPADI 21	1 JULI 2013	30 JULI 2013	3750000	A2	ROBI
C23	LEO	K3	JL.BUAYA 22	1 FEBRUARI 2013	30 FEBRUARI 2013	3000000	A2	ROBI



2NF

kd_pelanggan->nm_pelanggan

Kd_property-

>alm_property,kd_pemilik,nm_pemilik,biaya

a

Kd_pelanggan,kd_property-

>tgl_pinjam,tgl_selesai



3NF

Kd_property->alm_properti,kd_pemilik,biaya

Kd_pemilik->nm_pemilik



Normalisasikan Sampai 3NF

NoProyek	NamaProyek	NoPegawai	NamaPegawai	Golongan	BesarGaji
NP001	BRR	Peg01	Anton	A	1.000.000
		Peg02	Paula	B	900.000
		Peg06	Koko	C	750.000
NP002	PEMDA	Peg01	Anton	A	1.000.000
		Peg12	Sita	B	900.000
		Peg14	Yusni	B	900.000



<u>No-Mhs</u>	<u>Nm-Mhs</u>	<u>Jurusan</u>	<u>Kd-MK</u>	<u>Nama-MK</u>	<u>Kd-Dosen</u>	<u>Nm Dosen</u>	<u>Nilai</u>
2683	<u>Welli</u>	MI	MI350	<u>Manajemen Basis Data</u>	B104	<u>Ati</u>	A
2683	<u>Welli</u>	MI	MI465	<u>Analisis Pro. Sistem</u>	B317	<u>Dita</u>	B
5432	<u>Bakri</u>	AK	MI350	<u>Manajemen Basis Data</u>	B104	<u>Ati</u>	C
5432	<u>Bakri</u>	AK	AKN201	<u>Akuntansi Keuangan</u>	D310	<u>Lia</u>	B
5432	<u>Bakri</u>	AK	MKT300	<u>Dasar Pemasaran</u>	B212	Lola	A