



MATA KULIAH : Basis Data Lanjut
KODE MATA KULIAH/SKS : SI0054/ 3/1 sks
KURIKULUM : 2017
VERSI : 0.0

DESKRIPSI MATA KULIAH

Mata kuliah ini memberikan pengetahuan kepada mahasiswa mengenai basis data transaksional meliputi model, pengamanan, pengendalian, basis data berorientasi objek, basis data objek relasional, basis data terdistribusi dan aplikasi basis data, serta basis data analytical meliputi Data Warehouse, OLAP, dan Data Mining.



CAPAIAN PEMBELAJARAN

Mahasiswa mampu membangun serta mengelola basis data transaksional dan basisdata analytical untuk menghasilkan informasi melalui sistem ataupun aplikasi yang sesuai dengan permasalahan maupun kebutuhan bisnis.



SUMBER PUSTAKA

1. Connolly, T.M and Carolyn B. 2015. *Database Systems, A Practical Approach to Design, Implementation, and Management, Sixth Edition*. England: Pearson Education Limited. ISBN 10: 1-292-06118-9. ISBN 13:978-1-292-06118-4.
2. Date, CJ. 2000. *An Introduction to Database System Seventh Edition*. New Jersey: Pearson Addison Weesley. ISBN: 979-683-185-6.
3. Fatansyah. 2012. *Basis Data*. Bandung: Informatika. ISBN:978-602-8758-53-6
4. Nugroho, Adi. 2011. *Perancangan dan Implementasi Basis Data*. Yogyakarta: Andi. ISBN 978-979-29-2609-5.
5. Nugroho, Adi. 2014. *Sistem Basis Data ORACLE 10g*. Yogyakarta: Andi. ISBN 978-979-29-4342-9.



Minggu 4

Pertemuan 4



KEMAMPUAN AKHIR YANG DIHARAPKAN

Mahasiswa mampu melakukan pengamanan, pengendalian persaingan, serta back up dan recovery untuk basis data transaksional



POKOK BAHASAN

- Pengamanan Basis Data



MATERI POKOK

- Otoritas basis data
- Jenis-jenis integritas basis data
- Pemaksaan integritas





Jenis integritas basis data

Jenis-jenis integraitas data yang harus dipelihara dalam basis data adalah :

- Integritas entitas (Entity Integrity), tidak ada baris data yang duplikat dalam satu tabel
- Integritas jangkauan (domain integrity), tidak ada item data yang melanggar jangkauan nilai di tiap kolom data
- Integritas Acuan (Referential Integrity), yang menjaga relasi/korespondensi antar tabel
- Integritas Data Antar Tabel (Redundant Data Integrity), data disebuah tabel tidak terulang di tabel yang lain
- Integritas Aturan Nyata (Business Rule Integrity), data di satu tabel harus memiliki nilai yang layak dan dpt diterima di tempat basis data diterapkan.



Integritas Entitas

nim	kode_kul	indeks_nilai
100001	IF-110	D
100002	IF-310	B
100003	KU-234	B
100001	IF-110	D

tidak memenuhi Integritas Entitas, karena nilai semua field dari kedua baris data ini sama (tidak unik)



nim	kode_kul	indeks_nilai	semester
100001	IF-110	D	I/2011
100002	IF-310	B	I/2011
100003	KU-234	B	I/2011
100001	IF-110	D	I/2012

telah memenuhi Integritas Entitas



Integritas Jangkauan (Domain Integrity)

kode_kul	nama_kul	sks	semester
IF-110	Struktur Data	3	1
IF-310	Basis Data	4	12
KU-234	Bahasa Indonesia	2	2
MA-115	Matematika I	A	1

tidak memenuhi
Integritas Jangkauan,
karena nilai semester
seharusnya tidak boleh
lebih besar dari 9

tidak memenuhi Integritas Jangkauan,
karena nilai sks harus numerik



Integritas Acuan (Referential Integrity)

Tabel Kuliah

kode_kul	nama_kul	sks	semester
IF-110	Struktur Data	3	3
IF-310	Basis Data	4	3
KU-234	Bahasa Indonesia	2	2
MA-115	Matematika I	3	1

Tabel Nilai

nim	kode_kul	indeks_nilai
100001	IF-110	D
100002	IF-310	B
100003	KU-234	B
100001	MA-117	A

Mengacu ke tabel Kuliah

tidak memenuhi Integritas Acuan, karena nilai kode_kul di baris ini tidak ada di tabel Kuliah



Integritas Data Antar Tabel (Redundant Data Integrity)

Berikut ini adalah contoh adanya dua buah tabel yang tidak memenuhi Integritas Data Antar Tabel. Katakanlah kita ingin membagi baris-baris data kuliah ke dua buah tabel sesuai dengan program studi yang ada. Tabel pertama kita beri nama Kuliah_D3 yang berisi data kuliah untuk program studi D3, seperti berikut ini:

kode_kul	nama_kul	sks	semester
IF-120	Struktur Data	3	3
IF-320	Basis Data	4	3
KU-234	Bahasa Indonesia	2	2
MA-115	Matematika I	3	1

Sedang tabel kedua kita beri nama Kuliah_S1 yang berisi data kuliah untuk program studi S1, seperti berikut ini:

kode_kul	nama_kul	sks	semester
IF-110	Struktur Data	3	3
IF-310	Basis Data	4	3
KU-234	Bahasa Indonesia	2	2
MA-115	Matematika I	3	1

tidak memenuhi Integritas Data Antar Tabel, karena kedua tabel memiliki baris-baris data yang sama

Integritas Aturan Nyata (Business Rule Integrity)

nim	kode_kul	indeks_nilai
100001	IF-110	E
100002	IF-310	B
100003	KU-234	B
100001	IF-310	A

tidak memenuhi Integritas Aturan Nyata, karena ada aturan : mata kuliah IF-310 hanya boleh diambil seorang mahasiswa jika dia telah lulus untuk mata kuliah IF-110



Pemaksaan Integritas

Langkah melakukan pemaksaan integritas dilakukan melalui :

- Pendefinisian tabel dan struktur penyimpanan
- Pemanfaatan aturan integritas
- Penggunaan mekanisme pembangkitan (trigger) aksi dan prosedur basis data (database procedure)

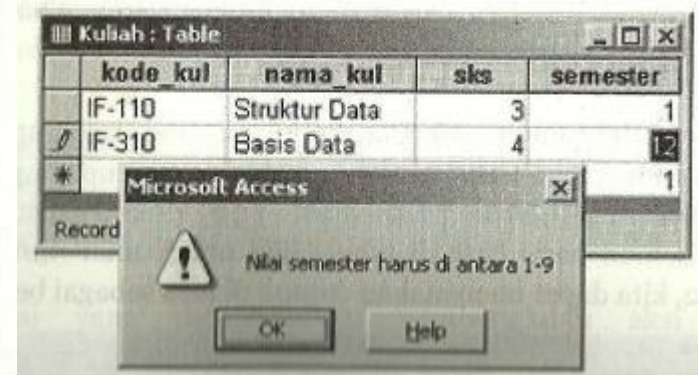
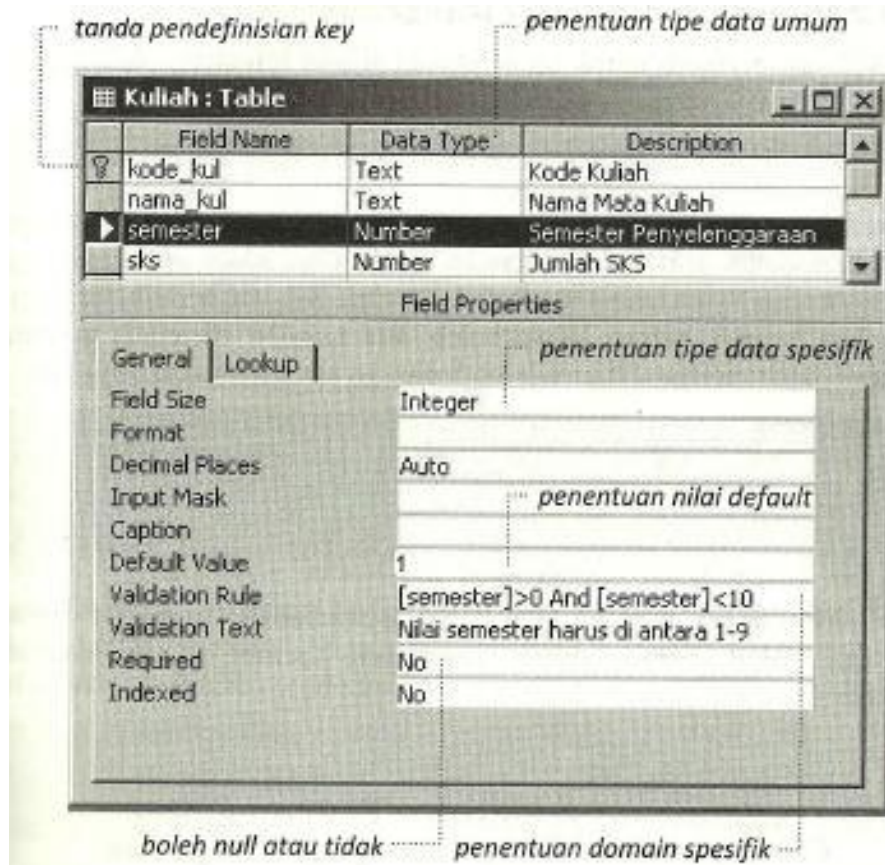


Pendefinisian tabel dan struktur penyimpanan

- a) Pendefinisian type data : create table – date
- b) Pendefinisian domain constraint : constraint – check
- c) Penerapan Nullability : not null
- d) Penggunaan nilai default : default
- e) Penerapan keunikan kolom : unique
- f) Penerapan keunikan baris



Pendefinisian tabel dan struktur penyimpanan



Pemanfaatan aturan integritas

Penerapan aturan integritas dapat digunakan untuk menjaga Integritas Aturan Nyata, sebagai contoh bila ada aturan setiap mhs tidak boleh mengambil mata kuliah sebanyak 20 sks

```
create assertion not_exists
  check (not exist (select sum(sks)
                    from nilai a, kuliah b
                    where a.kode_kul=b.kode_kul
                    group by semester having sum(sks)>20))
```



Penggunaan mekanisme pembangkitan (trigger) aksi dan prosedur basis data (database procedure)

Dalam SQL-92

```
define trigger hapus_mahasiswa on delete of mahasiswa a  
  (delete from nilai where nim=old a.nim))
```

Dalam Oracle

```
create procedure hapus_nilai (old_nim = varchar2)  
as begin  
  delete from nilai where nim=:old_nim;  
end;
```

```
create trigger hapus_mahasiswa after delete on mahasiswa  
for each row hapus_nilai (old.nim);
```

Dalam MySQL

```
create procedure hapus_nilai (old_nim varchar(6))  
begin  
  delete from nilai where nim=old_nim;  
end;
```

```
create trigger hapus_mahasiswa after delete on mahasiswa  
for each row hapus_nilai (old.nim);
```





Managed by BINA NUSANTARA