MENGAKSES DATA PADA BANYAK TABEL

Join tabel dalam pernyataan SELECT, sintaksnya:

```
<select statement>::=
SELECT
[<select option> [<select option>...]]
{* | <select list>}
[
FROM {<table reference> | <join definition>}
[WHERE <expression> [{<operator> <expression>}...]]
[GROUP BY <group by definition>]
[HAVING <expression> [{<operator> <expression>}...]]
[ORDER BY <order by definition>]
[LIMIT [<offset>,] <row count>]
]
<join definition>::=
{<table reference>, <table reference> [{, <table reference>}...]}
| {<table reference> [INNER | CROSS ] JOIN <table reference> [<join condition>]}
| {<table reference> STRAIGHT_JOIN <table reference>}
| {<table reference> LEFT [OUTER] JOIN <table reference> [<join condition>]}
| {<table reference> RIGHT [OUTER] JOIN <table reference> [<join condition>]}
| {<table reference> NATURAL [{LEFT | RIGHT} [OUTER]] JOIN <table reference>}
<table reference>::=
<table name> [[AS] <alias>]
[{USE | IGNORE | FORCE} INDEX <index name> [{, <index name>}...]]
<join condition>::=
ON <expression> [{<operator> <expression>}...]
| USING (<column> [{, <column>}...])
```

Tabel Buku

KodeBuku	JudulBuku	Copyright
12786	Belajar JavaScript	2004
13331	Algoritma dan Pemrograman	2000
14356	Belajar Pascal	1998
15729	Menghitung Hari	2001
16284	Latihan Java 2 dalam 21 hari	2004
17695	Pemrograman Delphi 6	2005
19264	Kamus Internet	2003
19354	Kamus Inggris-Indonesia	1999

Script:

```
CREATE TABLE Buku
(
KodeBuku SMALLINT NOT NULL PRIMARY KEY,
JudulBuku VARCHAR(60) NOT NULL,
Copyright YEAR NOT NULL
)
ENGINE=INNODB;
```

Tabel Pengarang

KodePengarang	NamaDepan	NamaTengah	NamaBelakang
1006	Budi	Agus	Purwanto
1007	Lidya		Anggraeni
1008	Budi		Mulya
1009	Andi		Purwanto
1010	Seno	Agus	Raharja
1011	Rizki	Indah	Sari
1012	Alan		Kusuma
1013	Mila		Kartika
1014	Andi	Ahmad	Yusuf

Script:

```
CREATE TABLE Pengarang
(
KodePengarang SMALLINT NOT NULL PRIMARY KEY,
NamaDepan VARCHAR(20),
NamaTengah VARCHAR(20),
NamaBelakang VARCHAR(20)
)
ENGINE=INNODB;
```

Tabel PenulisBuku

KodePengarang	KodeBuku
1006	14356
1008	15729
1009	12786
1010	17695
1011	15729
1012	19264
1012	19354
1014	16284

```
CREATE TABLE PenulisBuku
(
KodePengarang SMALLINT NOT NULL,
KodeBuku SMALLINT NOT NULL,
PRIMARY KEY (KodePengarang, KodeBuku),
FOREIGN KEY (KodePengarang) REFERENCES Pengarang (KodePengarang),
FOREIGN KEY (KodeBuku) REFERENCES Buku (KodeBuku)
)
ENGINE=INNODB;
```

FULL JOIN

1. SELECT JudulBuku, Copyright, KodePengarang
   FROM Buku, PenulisBuku
   ORDER BY JudulBuku;

2. SELECT JudulBuku, Copyright, KodePengarang
   FROM Buku AS b, PenulisBuku AS pb
   WHERE b.KodeBuku=pb.KodeBuku
   ORDER BY JudulBuku;

3. SELECT JudulBuku, Copyright, CONCAT_WS(' ', NamaDepan, NamaTengah, NamaBelakang) AS NamaPengarang
   FROM Buku AS b, PenulisBuku AS pb, Pengarang AS p
   WHERE b.KodeBuku=pb.KodeBuku AND pb.KodePengarang=p.KodePengarang AND Copyright<=2000
   ORDER BY JudulBuku;

INNER & CROSS JOIN

Sintaks:

```
<table reference> [INNER | CROSS ] JOIN <table reference> [<join condition>]
<join condition>::=
ON <expression> [{<operator> <expression>}...]
| USING (<column> [{, <column>}...])
```

Empat pernyataan di bawah ini menghasilkan hasil yang sama:

```
SELECT JudulBuku, KodePengarang FROM Buku, PenulisBuku;
SELECT JudulBuku, KodePengarang FROM Buku JOIN PenulisBuku;
SELECT JudulBuku, KodePengarang FROM Buku INNER JOIN PenulisBuku;
SELECT JudulBuku, KodePengarang FROM Buku CROSS JOIN PenulisBuku;
```

4. SELECT JudulBuku, Copyright, pb.KodePengarang
   FROM Buku AS b JOIN PenulisBuku AS pb
   ON b.KodeBuku=pb.KodeBuku
   ORDER BY JudulBuku;

5. SELECT JudulBuku, Copyright, pb.KodePengarang
   FROM Buku JOIN PenulisBuku AS pb
   USING (KodeBuku)
   ORDER BY JudulBuku;

6. SELECT JudulBuku, Copyright, pb.KodePengarang
   FROM Buku as b JOIN PenulisBuku AS pb
   ON b.KodeBuku=pb.KodeBuku
   WHERE Copyright <= 2000
   ORDER BY JudulBuku;

7. SELECT JudulBuku, Copyright, CONCAT_WS(' ', NamaDepan, NamaTengah, NamaBelakang) AS NamaPengarang
   FROM Buku JOIN PenulisBuku USING (KodeBuku)
   JOIN Pengarang USING (KodePengarang)
   WHERE Copyright<=2000
   ORDER BY JudulBuku;

OUTER JOIN

Sintaks Left Join:

```
<table reference> LEFT [OUTER] JOIN <table reference> [<join condition>]
<join condition>::=
ON <expression> [{<operator> <expression>}...]
| USING (<column> [{, <column>}...])
```

Sintaks Right Join:

```
<table reference> RIGHT [OUTER] JOIN <table reference> [<join condition>]
<join condition>::=
ON <expression> [{<operator> <expression>}...]
| USING (<column> [{, <column>}...])
```

8. SELECT JudulBuku, Copyright, KodePengarang
   FROM Buku LEFT JOIN PenulisBuku
   USING (KodeBuku)
   ORDER BY JudulBuku;

9. SELECT JudulBuku, Copyright, CONCAT_WS(' ', NamaDepan, NamaTengah, NamaBelakang) AS NamaPengarang
   FROM Buku AS b LEFT JOIN PenulisBuku AS pb ON b.KodeBuku=pb.KodeBuku
   LEFT JOIN Pengarang AS p ON pb.KodePengarang=p.KodePengarang
   ORDER BY JudulBuku;

Anda juga dapat menerapkannya pada RIGHT JOIN.
Begitu juga untuk NATURAL JOIN, NATURAL LEFT JOIN, dan NATURAL RIGHT JOIN.

JOIN TABEL DALAM PERYATAAN UPDATE

10. UPDATE Pengarang AS p, PenulisBuku AS pb, Buku AS b
    SET NamaBelakang='Purwantoko'
    WHERE p.KodePengarang=pb.KodePengarang AND pb.KodeBuku=b.KodeBuku AND
    JudulBuku='Belajar JavaScript';

JOIN TABEL DALAM PERYATAAN DELETE

11. DELETE pb, b
    FROM Pengarang AS p, PenulisBuku AS pb, Buku AS b
    WHERE p.KodePengarang=pb.KodePengarang AND pb.KodeBuku=b.KodeBuku AND
    NamaBelakang='Yusuf';

MEMBUAT SUBQUERY

12. SELECT CONCAT_WS(' ', NamaDepan, NamaTengah, NamaBelakang) AS NamaPengarang
    FROM Pengarang
    WHERE KodePengarang =
    (
    SELECT pb.KodePengarang
    FROM PenulisBuku AS pb, Buku AS b
    WHERE pb.KodeBuku=b.KodeBuku AND JudulBuku='Belajar JavaScript'
    );