

# E-Commerce

2 SKS | Semester 7 | UNIKOM

Nizar Rabbi Radliya  
nizar@email.unikom.ac.id

## Lesson 6

Database

SQL



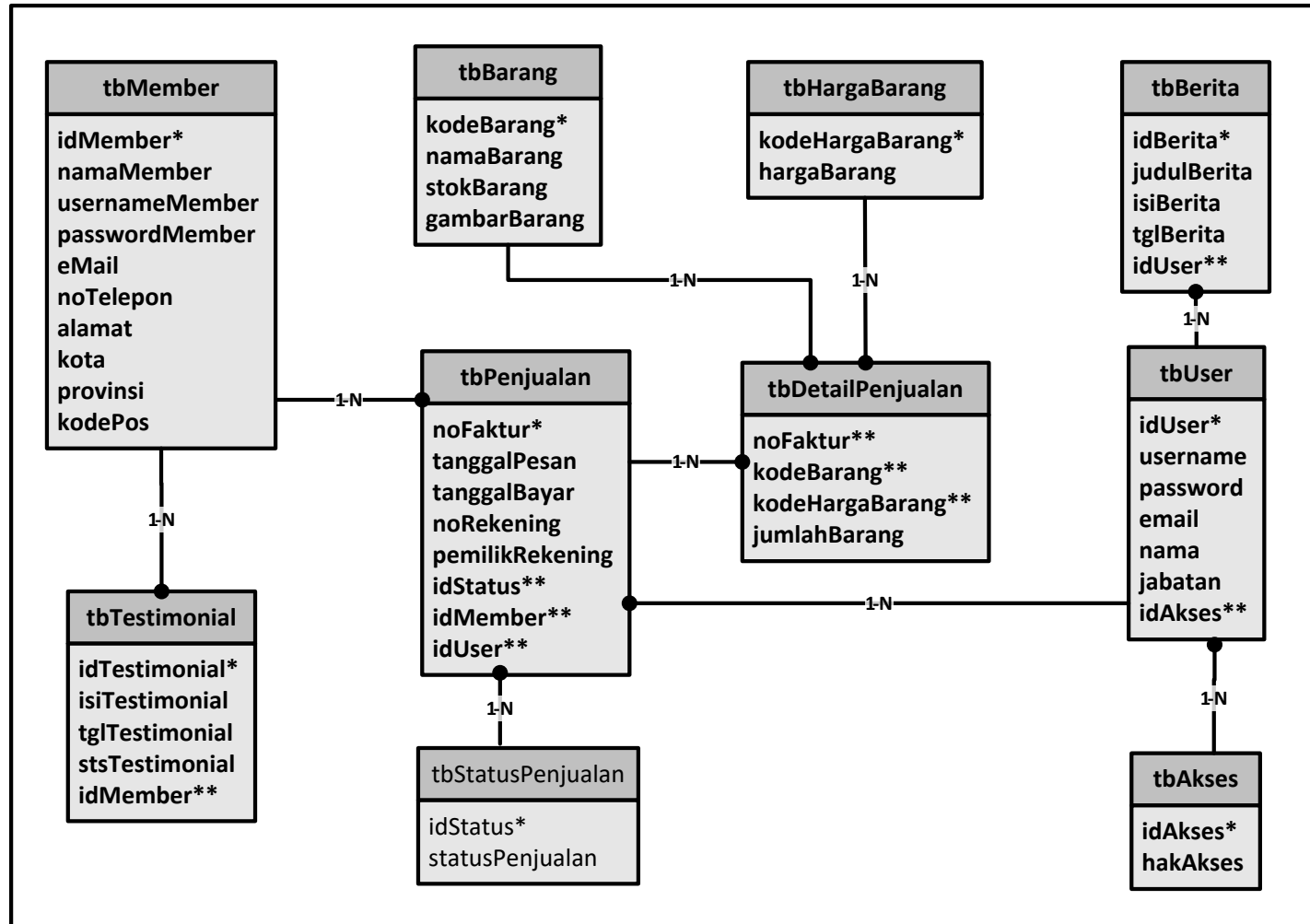
# Database / Basis Data

1. Menghimpun data yang terkait atau saling berhubungan
2. Kumpulan data tersebut terorganisasi dengan aturan tertentu
3. Data tersebut dapat digunakan untuk menghasilkan informasi

*Kumpulan file/tabel yang saling berhubungan dan disimpan pada media elektronik*



# Contoh Relasi Tabel di dalam Database



DATA A	DATA B
12	JK
26	AS
78	BB
45	RP

DATA D	DATA C	DATA A
10	A	78
20	E	45
30	U	12
40	O	26

DATA B	DATA C
JK	?
AS	?
BB	?
RP	?

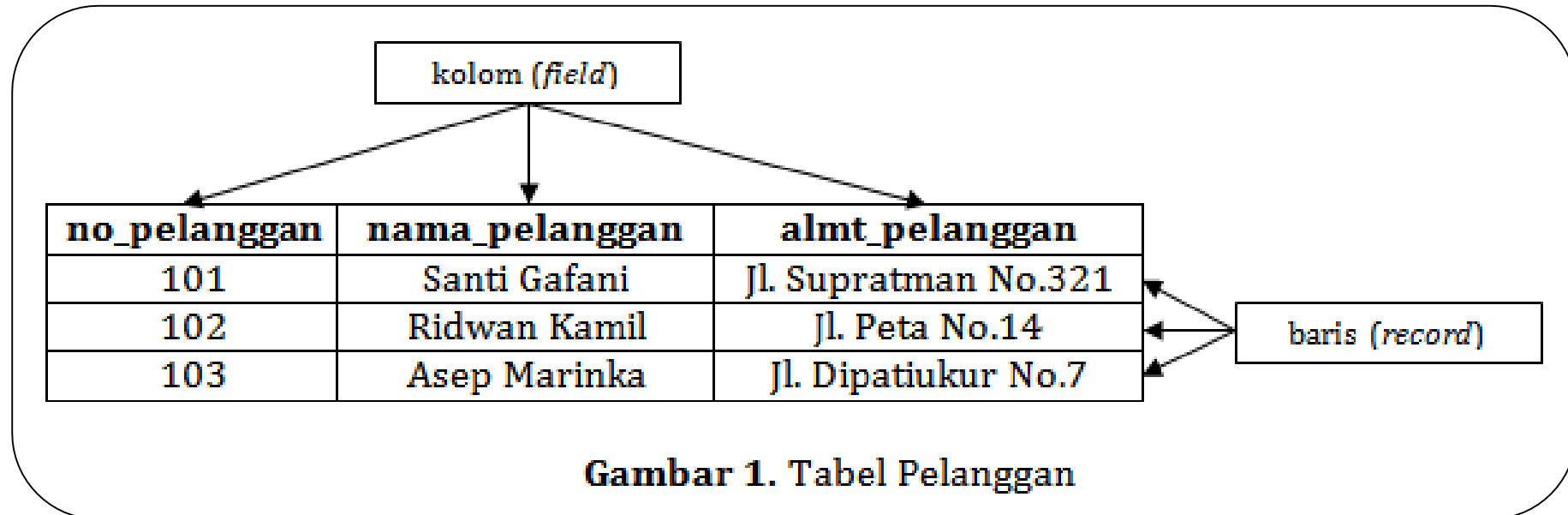
NIP	NAMA DOSEN
107	BAMBANG
115	HERWAN
116	DODI
120	NANI

NIM	NAMA MHS	NIP
234	ANDIN	120
235	ERLAN	115
236	ULIL	107
237	OKAN	116

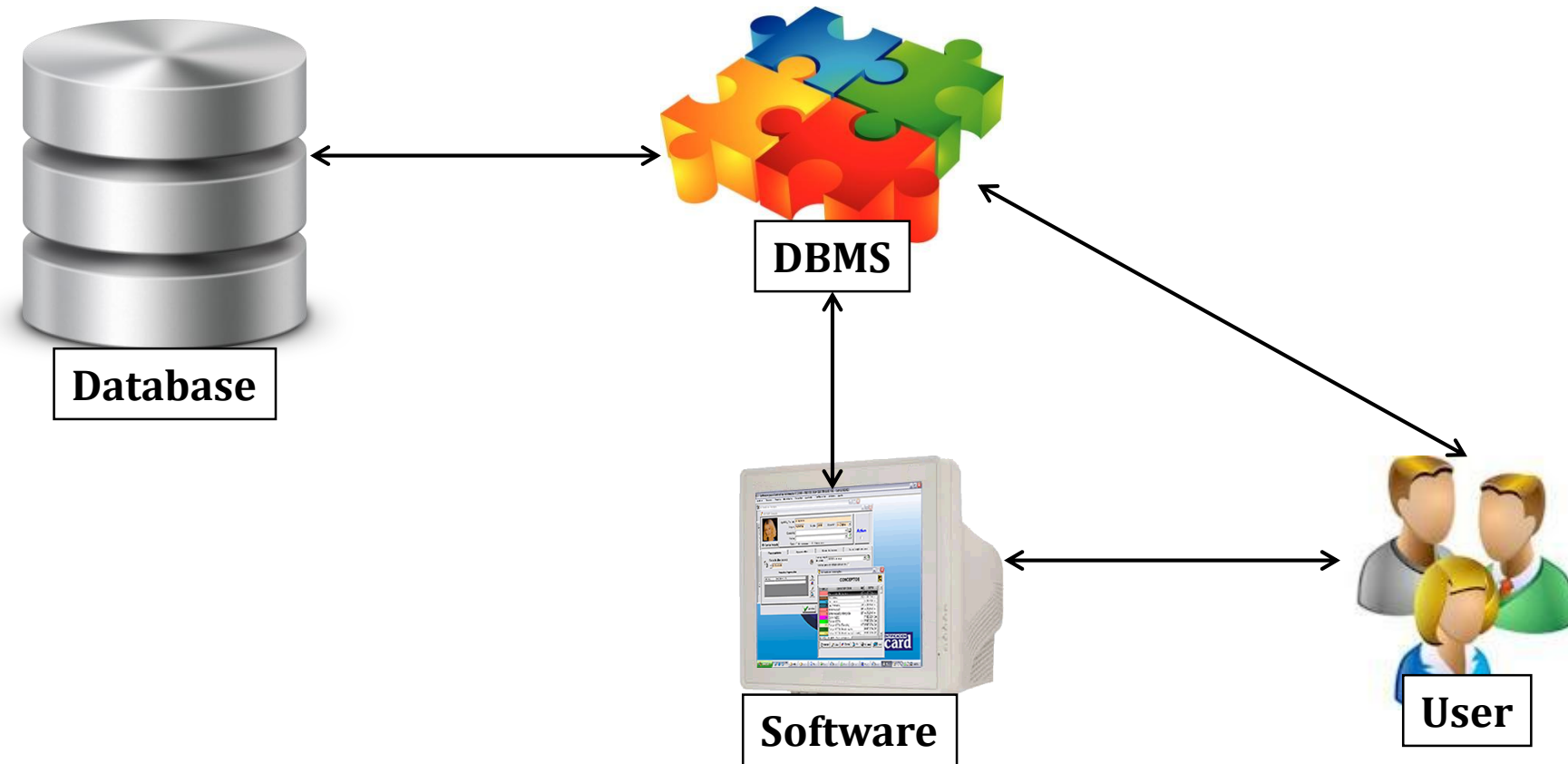
NAMA MHS	NAMA DOSEN
ANDIN	?
ERLAN	?
ULIL	?
OKAN	?

# Elemen Database

1. Tabel
2. Kolom
3. Baris
4. Kunci



# Sistem Basis Data



# Introduction to SQL

SQL is a standard language for accessing and manipulating databases.

What is SQL?

1. SQL stands for Structured Query Language
2. SQL lets you access and manipulate databases
3. SQL is an ANSI (American National Standards Institute) standard





# Introduction to SQL

## What Can SQL do?

1. SQL can execute queries against a database
2. SQL can retrieve data from a database
3. SQL can insert records in a database
4. SQL can update records in a database
5. SQL can delete records from a database
6. SQL can create new databases
7. SQL can create new tables in a database
8. SQL can create stored procedures in a database
9. SQL can create views in a database
10. SQL can set permissions on tables, procedures, and views



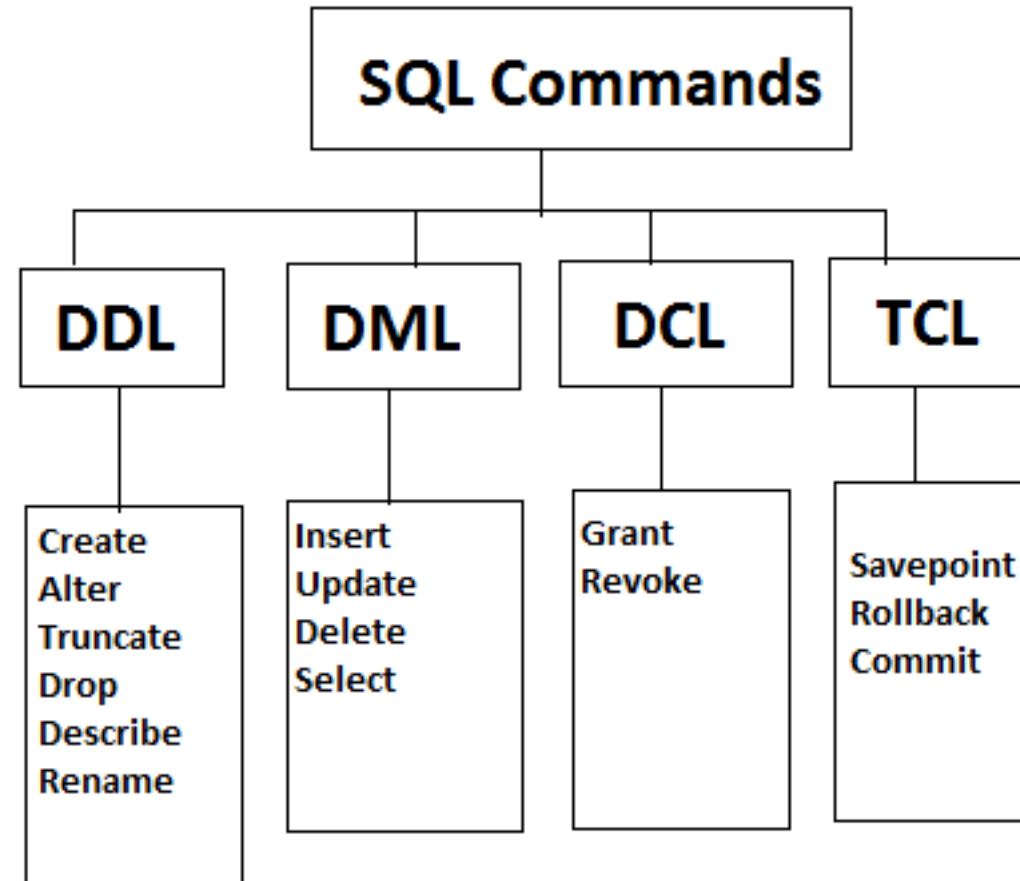
# Introduction to SQL

## Using SQL in Your Web Site

To build a web site that shows data from a database, you will need:

1. An RDBMS database program (i.e. MS Access, SQL Server, MySQL)
2. To use a server-side scripting language, like PHP or ASP
3. To use SQL to get the data you want
4. To use HTML / CSS to style the page





.....

# SQL SELECT Statement

The SELECT statement is used to select data from a database.

The data returned is stored in a result table, called the result-set.

SELECT Syntax:

```
SELECT column1, column2, ...  
FROM table_name;
```

The SELECT statement is used to select data from a database.

The data returned is stored in a result table, called the result-set.

SELECT Syntax:

```
SELECT * FROM table_name;
```



# SQL WHERE Clause

The WHERE clause is used to filter records.

The WHERE clause is used to extract only those records that fulfill a specified condition.

```
SELECT column1, column2, ...  
FROM table_name  
WHERE condition;
```

Operator	Description
=	Equal
<>	Not equal. <b>Note:</b> In some versions of SQL this operator may be written as !=
>	Greater than
<	Less than
>=	Greater than or equal
<=	Less than or equal
BETWEEN	Between an inclusive range
LIKE	Search for a pattern
IN	To specify multiple possible values for a column



# SQL AND, OR and NOT Operators

The WHERE clause can be combined with AND, OR, and NOT operators.

The AND and OR operators are used to filter records based on more than one condition:

1. The AND operator displays a record if all the conditions separated by AND is TRUE.
2. The OR operator displays a record if any of the conditions separated by OR is TRUE.

The NOT operator displays a record if the condition(s) is NOT TRUE.

```
SELECT column1, column2, ...  
FROM table_name  
WHERE condition1 AND condition2 OR NOT condition3 ...;
```



## SQL ORDER BY Keyword

The ORDER BY keyword is used to sort the result-set in ascending or descending order.

The ORDER BY keyword sorts the records in ascending order by default. To sort the records in descending order, use the DESC keyword.

```
SELECT column1, column2, ...  
FROM table_name  
ORDER BY column1, column2, ... ASC|DESC;
```



# SQL INSERT INTO Statement

The INSERT INTO statement is used to insert new records in a table.

It is possible to write the INSERT INTO statement in two ways.

The first way specifies both the column names and the values to be inserted:

```
INSERT INTO table_name (column1, column2, column3, ...)  
VALUES (value1, value2, value3, ...);
```

If you are adding values for all the columns of the table, you do not need to specify the column names in the SQL query. However, make sure the order of the values is in the same order as the columns in the table. The INSERT INTO syntax would be as follows:

```
INSERT INTO table_name  
VALUES (value1, value2, value3, ...);
```





# SQL UPDATE Statement

The UPDATE statement is used to modify the existing records in a table.

```
UPDATE table_name  
SET column1 = value1, column2 = value2, ...  
WHERE condition;
```

**Note:** Be careful when updating records in a table! Notice the WHERE clause in the UPDATE statement. The WHERE clause specifies which record(s) that should be updated. If you omit the WHERE clause, all records in the table will be updated!



# SQL DELETE Statement

The DELETE statement is used to delete existing records in a table.

```
DELETE FROM table_name  
WHERE condition;
```

**Note:** *Be careful when deleting records in a table! Notice the WHERE clause in the DELETE statement. The WHERE clause specifies which record(s) that should be deleted. If you omit the WHERE clause, all records in the table will be deleted!*



# SQL Joins

A JOIN clause is used to combine rows from two or more tables, based on a related column between them.

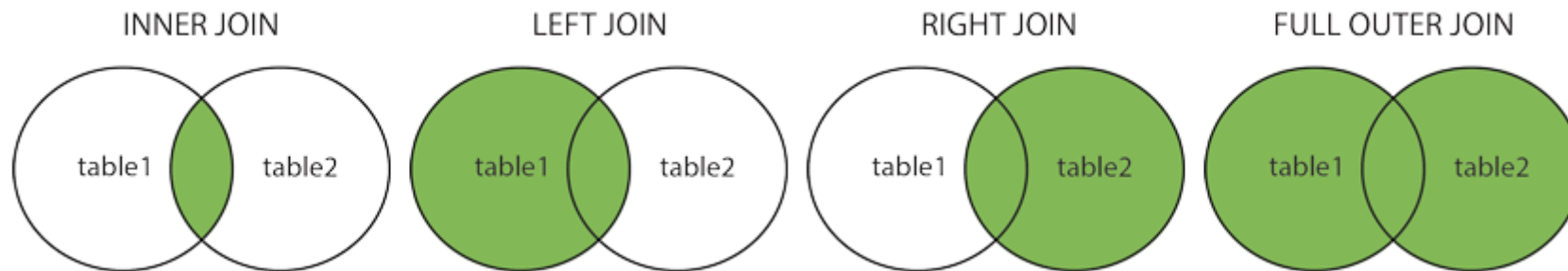
Here are the different types of the JOINS in SQL:

(INNER) JOIN: Returns records that have matching values in both tables

LEFT (OUTER) JOIN: Return all records from the left table, and the matched records from the right table

RIGHT (OUTER) JOIN: Return all records from the right table, and the matched records from the left table

FULL (OUTER) JOIN: Return all records when there is a match in either left or right table



# SQL Joins

A JOIN clause is used to combine rows from two or more tables, based on a related column between them.

Here are the different types of the JOINS in SQL:

(INNER) JOIN: Returns records that have matching values in both tables

LEFT (OUTER) JOIN: Return all records from the left table, and the matched records from the right table

RIGHT (OUTER) JOIN: Return all records from the right table, and the matched records from the left table

FULL (OUTER) JOIN: Return all records when there is a match in either left or right table



NEXT:

PHP & MySQL

