

SQL

SQL Fundamentals



SQL Fundamentals

- ☐ Introduction to Sample DB
- ☐ SQL Modification: CREATE
- ☐ SQL Basic Queries: SELECT
- ☐ SQL Basic Queries: LIMIT
- ☐ SQL Basic Queries: WHERE
- ☐ SQL Basic Queries: ORDER BY
- ☐ SQL Arithmetic Operation
- ☐ SQL Basic Queries FAQ
- ☐ SQL Modification: INSERT
- ☐ SQL Modification: UPDATE
- ☐ SQL Modification: DELETE

SQL Fundamentals



Introduction to Sample DB



SQL Modification: CREATE



SQL Basic Queries: SELECT



SQL Basic Queries: LIMIT



SQL Basic Queries: WHERE



SQL Basic Queries: ORDER BY



SQL Arithmetic Operation



SQL Basic Queries FAQ



SQL Modification: INSERT



SQL Modification: UPDATE



SQL Modification: DELETE

Hands-On Required :

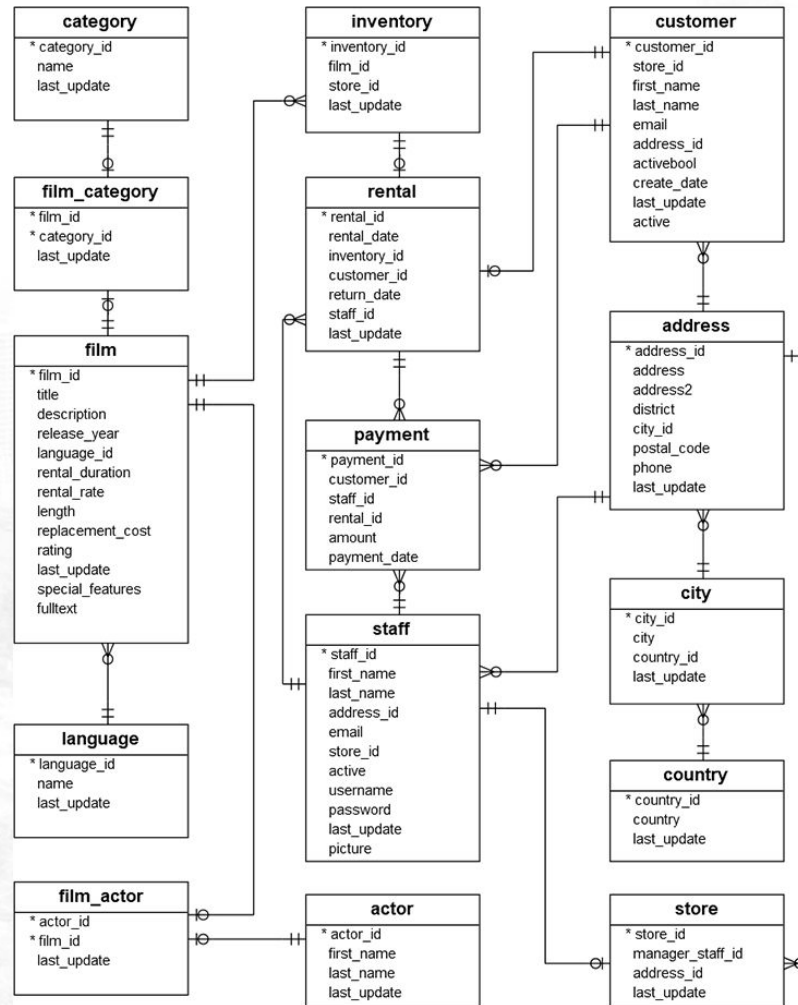
Resources :

- [dvdrental.tar](#)
- [rakafood.zip](#)

Klik disini untuk mengakses
folder Database

Entity Relationship Diagram

DVDRental





RakaFood

service pembelian makanan secara online

- rakamin_customers
- rakamin_orders
- rakamin_merchant
- rakamin_customer_address

SQL Fundamentals



Introduction to Sample DB



SQL Modification: CREATE



SQL Basic Queries: SELECT



SQL Basic Queries: LIMIT



SQL Basic Queries: WHERE



SQL Basic Queries: ORDER BY



SQL Arithmetic Operation



SQL Basic Queries FAQ



SQL Modification: INSERT



SQL Modification: UPDATE



SQL Modification: DELETE

SQL Fundamentals



Introduction to Sample DB



SQL Modification: CREATE



SQL Basic Queries: SELECT



SQL Basic Queries: LIMIT



SQL Basic Queries: WHERE



SQL Basic Queries: ORDER BY



SQL Arithmetic Operation



SQL Basic Queries FAQ



SQL Modification: INSERT



SQL Modification: UPDATE



SQL Modification: DELETE

CREATE - Definisi

CREATE merupakan klausa yang digunakan untuk membuat tabel baru.

Perintah dalam SQL

```
CREATE TABLE [IF NOT EXISTS] table_name (
    column1 datatype [constraints],
    column2 datatype [constraints],
    column3 datatype [constraints],
    ....
);
```

```
CREATE TABLE [IF NOT EXISTS]
table_name AS (
    "Tambahkan SELECT query disini"
);
```

Klausa **IF NOT EXISTS** bersifat opsional, jika ditambahkan maka RDBMS tidak akan memberikan sinyal error jika tabel yang kita buat ternyata sudah ada.

Tipe Data

Berikut adalah daftar tipe data yang dapat digunakan untuk persyaratan dalam membuat tabel.

Tipe Data	Deskripsi	Ukuran Penyimpanan
integer	Bilangan bulat	4 bytes
double precision	Variable bilangan dengan presisi.	8 bytes
varchar	Variabel karakter dengan panjang tertentu dengan batasan.	
Timestamp without timezone	Tanggal dan waktu tanpa zonasi waktu.	8 bytes
Boolean	Keadaan benar atau salah	1 byte

Info lebih lanjut:

https://www.tutorialspoint.com/postgresql/postgresql_data_types.htm

Batasan

Constraints atau batasan adalah aturan memaksa yang diterapkan pada kolom-kolom data pada tabel. Berikut adalah jenis-jenis kendala pada PostgreSQL:


Nama Kendala	Deskripsi
NOT NULL	Memastikan bahwa kolom tidak akan memiliki nilai kosong.
UNIQUE	Memastikan bahwa semua nilai pada kolom berbeda satu dengan yang lain.
PRIMARY KEY	Berperan sebagai identifier unik untuk setiap baris data.
FOREIGN KEY	Kendala didefinisikan pada kolom di tabel lain.

Info lebih lanjut:

https://www.tutorialspoint.com/postgresql/postgresql_constraints.htm

Hands-On

Buatlah tabel `rakamin_merchant` yang berisikan id merchant, nama merchant dan status keaktifannya.


tutorial/postgres@local@postgres ▾

Query Editor
Query History

```

1 CREATE TABLE IF NOT EXISTS rakamin_merchant_active (
2     merchant_id INT PRIMARY KEY NOT NULL,
3     merchant_name VARCHAR(50) UNIQUE,
4     active BOOLEAN NOT NULL
5 );

```

Hands-On

Buatlah tabel `rakamin_active_customers` dari tabel `rakamin_customer`.



tutorial/postgres@local@postgres ▾

Query Editor

Query History

```

1 CREATE TABLE IF NOT EXISTS rakamin_active_customers AS(
2     SELECT * FROM rakamin_customer WHERE pengguna_aktif = 1
3 );

```


SQL Fundamentals



Introduction to Sample DB



SQL Modification: CREATE



SQL Basic Queries: SELECT



SQL Basic Queries: LIMIT



SQL Basic Queries: WHERE



SQL Basic Queries: ORDER BY



SQL Arithmetic Operation



SQL Basic Queries FAQ



SQL Modification: INSERT



SQL Modification: UPDATE



SQL Modification: DELETE

SQL Fundamentals



Introduction to Sample DB



SQL Modification: CREATE



SQL Basic Queries: SELECT



SQL Basic Queries: LIMIT



SQL Basic Queries: WHERE



SQL Basic Queries: ORDER BY



SQL Arithmetic Operation



SQL Basic Queries FAQ



SQL Modification: INSERT



SQL Modification: UPDATE



SQL Modification: DELETE

Perintah Dasar SQL

- **Select**, untuk memilih kolom yang ingin diambil
- **Limit**, membatasi jumlah row pada tabel yang diambil
- **Where**, membatasi/filter/mengambil spesifik data yang diambil
- **Order By**, mengurutkan data setelah data diambil

Perintah Dasar 1 - SELECT

Digunakan untuk memilih kolom apa saja yang ingin diambil dari tabel

Deskripsi	Perintah dalam SQL
Ambil semua kolom	<code>SELECT *</code> <code>FROM <nama tabel></code>
Ambil beberapa kolom saja	<code>SELECT kolom_1, kolom_2, ... kolom_10</code> <code>FROM <nama tabel></code>

Perintah Dasar 1 - SELECT

Contoh 1 : Bagaimana isi dari tabel film?

Query Editor

Query History

1

2

3

4

5

select *

from film

Data Output

Explain

Messages

Notifications

	<div>film_id</div> <div>[PK] integer</div>	<div>title</div> <div>character varying (255)</div>	<div>description</div> <div>text</div>
1	133	Chamber Italian	A Fateful Reflection of a Moose And a Husband who must Overcome a Monkey in Nigeria
2	384	Grosse Wonderful	A Epic Drama of a Cat And a Explorer who must Redeem a Moose in Australia
3	8	Airport Pollock	A Epic Tale of a Moose And a Girl who must Confront a Monkey in Ancient India
4	98	Bright Encounters	A Fateful Yarn of a Lumberjack And a Feminist who must Conquer a Student in A Jet Boat
5	1	Academy Dinosaur	A Epic Drama of a Feminist And a Mad Scientist who must Battle a Teacher in The Canadian Rockies
6	2	Ace Goldfinger	A Astounding Epistle of a Database Administrator And a Explorer who must Find a Car in Ancient China
7	3	Adaptation Holes	A Astounding Reflection of a Lumberjack And a Car who must Sink a Lumberjack in A Baloon Factory
8	4	Affair Prejudice	A Fanciful Documentary of a Frisbee And a Lumberjack who must Chase a Monkey in A Shark Tank
9	5	African Egg	A Fast-Paced Documentary of a Pastry Chef And a Dentist who must Pursue a Forensic Psychologist in The Gulf of Mexico

Perintah Dasar 1 - SELECT

Contoh 2 :

Apa saja title film yang tersedia untuk disewakan?

Query Editor

Query History

1

select title

2

from film

3

4

5

Data Output

Explain

Messages

Notifications

title

character varying (255)

1

Chamber Italian

2

Grosse Wonderful

3

Airport Pollock

4

Bright Encounters

5

Academy Dinosaur

Perintah Dasar 1 - SELECT

Contoh 3 : Ingin melihat kode film id, judul dan juga tahun rilis dari masing-masing film.

Query Editor

Query History

1

2

3

4

5

```
select film_id, title, release_year
from film
```

Data Output

Explain

Messages

Notifications

	film_id [PK] integer	title character varying (255)	release_year integer
1	133	Chamber Italian	2006
2	384	Grosse Wonderful	2006
3	8	Airport Pollock	2006
4	98	Bright Encounters	2006
5	1	Academy Dinosaur	2006
6	2	Ace Goldfinger	2006
7	3	Adaptation Holes	2006
8	4	Affair Prejudice	2006
9	5	African Egg	2006

Perintah Dasar 1 - SELECT

Memanggil kolom dengan syntax [tabel].[kolom]

Query Editor

Query History

```
1 select
2     film.film_id,
3     film.title,
4     film.release_year
5 from
6     film
```

Data Output

Messages

Explain

Notifications

	<div>film_id</div> <div>[PK] integer</div>	<div>title</div> <div>character varying (255)</div>	<div>release_year</div> <div>integer</div>	
1	133	Chamber Italian	2006	
2	384	Grosse Wonderful	2006	
3	8	Airport Pollock	2006	
4	98	Bright Encounters	2006	
5	1	Academy Dinosaur	2006	
6	2	Ace Goldfinger	2006	

SQL Fundamentals



Introduction to Sample DB



SQL Modification: CREATE



SQL Basic Queries: SELECT



SQL Basic Queries: LIMIT



SQL Basic Queries: WHERE



SQL Basic Queries: ORDER BY



SQL Arithmetic Operation



SQL Basic Queries FAQ



SQL Modification: INSERT



SQL Modification: UPDATE



SQL Modification: DELETE

SQL Fundamentals



Introduction to Sample DB



SQL Modification: CREATE



SQL Basic Queries: SELECT



SQL Basic Queries: LIMIT



SQL Basic Queries: WHERE



SQL Basic Queries: ORDER BY



SQL Arithmetic Operation



SQL Basic Queries FAQ



SQL Modification: INSERT



SQL Modification: UPDATE



SQL Modification: DELETE

Perintah Dasar 2 - LIMIT

Digunakan untuk membatasi jumlah baris/row yang ingin diambil dari tabel (Perintah diletakan di paling bawah)

Deskripsi	Perintah dalam SQL
Ambil n baris teratas	<pre>SELECT * FROM <nama tabel> LIMIT n</pre>
Ambil n baris teratas setelah melangkahi m baris terlebih dahulu	<pre>SELECT kolom_1, kolom_2, ... kolom_10 FROM <nama tabel> LIMIT n OFFSET m</pre>

Perintah Dasar 2 - LIMIT

Contoh 1 :

Apa isi 5 baris teratas dari tabel film?

Query Editor

Query History

1

select *

2

from film

3

limit 5

4

5

Data Output

Explain

Messages

Notifications

	<div>film_id</div> <div>[PK] integer</div>	<div>title</div> <div>character varying (255)</div>	<div>description</div> <div>text</div>	<div>release_year</div> <div>integer</div>	<div>language</div> <div>smallint</div>
1	133	Chamber Italian	A Fateful Reflection of a Moose And a Husband who must Overcome a Monkey in Nigeria	2006	
2	384	Grosse Wonderful	A Epic Drama of a Cat And a Explorer who must Redeem a Moose in Australia	2006	
3	8	Airport Pollock	A Epic Tale of a Moose And a Girl who must Confront a Monkey in Ancient India	2006	
4	98	Bright Encounters	A Fateful Yarn of a Lumberjack And a Feminist who must Conquer a Student in A Jet Boat	2006	
5	1	Academy Dinosaur	A Epic Drama of a Feminist And a Mad Scientist who must Battle a Teacher in The Canadian Rockies	2006	

Perintah Dasar 2 - LIMIT

Contoh 2 :

Apa isi baris ke 6-10 dari tabel film?

Query Editor Query History

1 select *

2 from film

3 limit 5 offset 5

Data Output Messages Explain Notifications

	<div>film_id</div> <div>[PK] integer</div>	<div>title</div> <div>character varying (255)</div>	<div>description</div> <div>text</div>	<div>release_date</div> <div>timestamp without time zone</div>
1	2	Ace Goldfinger	A Astounding Epistle of a Database Administrator And a Explorer who must Find a Car in Ancient China	
2	3	Adaptation Holes	A Astounding Reflection of a Lumberjack And a Car who must Sink a Lumberjack in A Baloon Factory	
3	4	Affair Prejudice	A Fanciful Documentary of a Frisbee And a Lumberjack who must Chase a Monkey in A Shark Tank	
4	5	African Egg	A Fast-Paced Documentary of a Pastry Chef And a Dentist who must Pursue a Forensic Psychologist in The Gulf of Mexico	
5	6	Agent Truman	A Intrepid Panorama of a Robot And a Boy who must Escape a Sumo Wrestler in Ancient China	

SQL Fundamentals



Introduction to Sample DB



SQL Modification: CREATE



SQL Basic Queries: SELECT



SQL Basic Queries: LIMIT



SQL Basic Queries: WHERE



SQL Basic Queries: ORDER BY



SQL Arithmetic Operation



SQL Basic Queries FAQ



SQL Modification: INSERT



SQL Modification: UPDATE



SQL Modification: DELETE

SQL Fundamentals



Introduction to Sample DB



SQL Modification: CREATE



SQL Basic Queries: SELECT



SQL Basic Queries: LIMIT



SQL Basic Queries: WHERE



SQL Basic Queries: ORDER BY



SQL Arithmetic Operation



SQL Basic Queries FAQ



SQL Modification: INSERT



SQL Modification: UPDATE



SQL Modification: DELETE

Perintah Dasar 3 - WHERE

Digunakan untuk membatasi / filter pada spesifik kolom tertentu saja (Perintah diletakan tepat setelah *FROM (tabel)*)

Deskripsi	Perintah dalam SQL
Ambil semua kolom dan kolom_1 = xxxx	<pre>SELECT * FROM <nama tabel> WHERE kolom_1 = xxxx</pre>
Ambil beberapa kolom saja dan kolom_2 = yyyy	<pre>SELECT kolom_1, kolom_2, ... kolom_10 FROM <nama tabel> WHERE kolom_2 = yyyy</pre>

Perintah Dasar 3 - WHERE

Contoh 1 :

Film apa saja yang memiliki rental_duration = 3?

Query Editor

Query History

1

2

3

4

5

```
select film_id, title, rental_duration
from film
where rental_duration = 3
```

Data Output

Explain

Messages

Notifications

	<div>film_id</div> <div>[PK] integer</div>	<div>title</div> <div>character varying (255)</div>	<div>rental_duration</div> <div>smallint</div>	
1		2 Ace Goldfinger	3	
2		6 Agent Truman	3	
3		9 Alabama Devil	3	
4		17 Alone Trip	3	
5		21 American Circus	3	
6		23 Anaconda Confessions	3	
7		25 Angels Life	3	
8		26 Annie Identity	3	
9		37 Arizona Bang	3	

Perintah Dasar 3 - WHERE

Kita juga bisa menggunakan tipe data lain selain integer

Tipe data	Perintah dalam SQL
string	<pre>SELECT * FROM <nama tabel> WHERE kolom_1 = 'xxxx'</pre>
date	<pre>SELECT kolom_1, kolom_2, ... kolom_10 FROM <nama tabel> WHERE kolom_2 = 'yyyy-mm-dd'</pre>
null	<pre>SELECT kolom_1, kolom_2, ... kolom_10 FROM <nama tabel> WHERE kolom_3 IS NULL</pre>

Perintah Dasar 3 - WHERE

Contoh 2 : Bagaimana deskripsi dari film Frost Head?

Query Editor

Query History

1

select *

2

from film

3

where title = 'Frost Head'

Data Output

Messages

Explain

Notifications

	<div>film_id</div> <div>[PK] integer</div>	<div>title</div> <div>character varying (255)</div>	<div>description</div> <div>text</div>
1	341	Frost Head	A Amazing Reflection of a Lumberjack And a Cat who must Discover a Husband in A MySQL Convention

Perintah Dasar 3 - WHERE

Contoh 3 : Adakah order rental yang terjadi pada rental_date = 2005-07-29 08:40:36 ?

Query Editor

Query History

1

select *

2

from rental

3

where rental_date = '2005-07-29 08:40:36'

4

5

Data Output

Explain

Messages

Notifications

	rental_id [PK] integer	rental_date timestamp without time zone	inventory_id integer	customer_id smallint	return_date timestamp without time zone	staff_id smallint	last_update timestamp without time zone
1	8477	2005-07-29 08:40:36	3571	354	2005-08-06 08:28:36	2	2006-02-16 02:30:53
2	8478	2005-07-29 08:40:36	3742	162	2005-08-01 10:23:36	1	2006-02-16 02:30:53

Perintah Dasar 3 - WHERE

Contoh 4 : Adakah alamat dengan address2 yang kosong?

Query Editor Query History

```
1 select *
2 from address
3 where address2 is null
```

Data Output Explain Messages Notifications

	address_id [PK] integer	address character varying (50)	address2 character varying (50)	district character varying (20)	city_id smallint	postal_code character varying (10)	phone character varying (20)	last_update timestamp
1	1	47 MySakila Drive	[null]	Alberta	300			2006-10-04 15:20:29
2	2	28 MySQL Boulevard	[null]	QLD	576			2006-10-04 15:20:29
3	3	23 Workhaven Lane	[null]	Alberta	300		14033335568	2006-10-04 15:20:29
4	4	1411 Lillydale Drive	[null]	QLD	576		6172235589	2006-10-04 15:20:29

SQL Fundamentals



Introduction to Sample DB



SQL Modification: CREATE



SQL Basic Queries: SELECT



SQL Basic Queries: LIMIT



SQL Basic Queries: WHERE



SQL Basic Queries: ORDER BY



SQL Arithmetic Operation



SQL Basic Queries FAQ



SQL Modification: INSERT



SQL Modification: UPDATE



SQL Modification: DELETE

SQL Fundamentals



Introduction to Sample DB



SQL Modification: CREATE



SQL Basic Queries: SELECT



SQL Basic Queries: LIMIT



SQL Basic Queries: WHERE



SQL Basic Queries: ORDER BY



SQL Arithmetic Operation



SQL Basic Queries FAQ



SQL Modification: INSERT



SQL Modification: UPDATE



SQL Modification: DELETE

Perintah Dasar 4 - ORDER BY

Digunakan untuk mengurutkan data dari besar ke kecil (Descending) atau kecil ke besar (Ascending) pada kolom tertentu setelah data diambil.

Deskripsi	Perintah dalam SQL
Ambil semua kolom dan diurutkan berdasarkan kolom_1 dari kecil ke besar (ascending).	<pre>SELECT * FROM <nama tabel> ORDER BY kolom_1 ASC</pre>
Ambil beberapa kolom saja dan diurutkan berdasarkan kolom_1 dari besar ke kecil (descending)	<pre>SELECT kolom_1, kolom_2, ... kolom_10 FROM <nama tabel> ORDER BY kolom_1 DESC</pre>

Perintah Dasar 4 - ORDER BY

Query Editor Query History

```
1 select *
2 from customer
```

Data Output Explain Messages Notifications

	customer_id [PK] integer	store_id smallint	first_name character varying (45)	last_name character varying (45)
1	524	1	Jared	Ely
2	1	1	Mary	Smith
3	2	1	Patricia	Johnson
4	3	1	Linda	Williams
5	4	2	Barbara	Jones
6	5	1	Elizabeth	Brown
7	6	2	Jennifer	Davis
8	7	1	Maria	Miller
9	8	2	Susan	Wilson
10	9	2	Margaret	Mass



Query Editor Query History

```
1 select *
2 from customer
3 order by first_name
```

Data Output Explain Messages Notifications

	customer_id [PK] integer	store_id smallint	first_name character varying (45)	last_name character varying (45)
1	375	2	Aaron	Selby
2	367	1	Adam	Gooch
3	525	2	Adrian	Clary
4	217	2	Agnes	Bishop
5	389	1	Alan	Kahn
6	352	1	Albert	Crouse
7	568	2	Alberto	Henning
8	454	2	Alex	Gresham
9	439	2	Alexander	Fennell
10	422	2	Alfred	Coelho

Perintah Dasar 4 - ORDER BY

Query Editor Query History

```
1 select *
2 from customer
```

Data Output Explain Messages Notifications

	customer_id [PK] integer	store_id smallint	first_name character varying (45)	last_name character varying (45)
1	524	1	Jared	Ely
2	1	1	Mary	Smith
3	2	1	Patricia	Johnson
4	3	1	Linda	Williams
5	4	2	Barbara	Jones
6	5	1	Elizabeth	Brown
7	6	2	Jennifer	Davis
8	7	1	Maria	Miller
9	8	2	Susan	Wilson
10	9	2	Margaret	Mass



Query Editor Query History

```
1 select *
2 from customer
3 order by first_name desc
```

Data Output Explain Messages Notifications

	customer_id [PK] integer	store_id smallint	first_name character varying (45)	last_name character varying (45)
1	479	1	Zachary	Hite
2	174	2	Yvonne	Watkins
3	190	2	Yolanda	Weaver
4	212	2	Wilma	Richards
5	359	2	Willie	Markham
6	219	2	Willie	Howell
7	303	2	William	Satterfield
8	578	2	Willard	Lumpkin
9	469	2	Wesley	Bull
10	115	1	Wendy	Barber

SQL Fundamentals



Introduction to Sample DB



SQL Modification: CREATE



SQL Basic Queries: SELECT



SQL Basic Queries: LIMIT



SQL Basic Queries: WHERE



SQL Basic Queries: ORDER BY



SQL Arithmetic Operation



SQL Basic Queries FAQ



SQL Modification: INSERT



SQL Modification: UPDATE



SQL Modification: DELETE

SQL Fundamentals



Introduction to Sample DB



SQL Modification: CREATE



SQL Basic Queries: SELECT



SQL Basic Queries: LIMIT



SQL Basic Queries: WHERE



SQL Basic Queries: ORDER BY



SQL Arithmetic Operation



SQL Basic Queries FAQ



SQL Modification: INSERT



SQL Modification: UPDATE



SQL Modification: DELETE

Penggunaan Operator Aritmatika

Tidak hanya menggunakan angka saja, namun kita juga bisa menggunakan operator aritmatika di dalam query.

Contoh 5 : Ambil kolom **customer_id**, **rental_id**, **amount** pada tabel **payment**, kali kolom **amount** dengan 14000

Query Editor Query History

```
1 select customer_id, rental_id, amount, amount * 14000
2 from payment
```

Data Output Explain Messages Notifications

	customer_id smallint	rental_id integer	amount numeric (5,2)	?column? numeric
1	341	1520	7.99	111860.00
2	341	1778	1.99	27860.00
3	341	1849	7.99	111860.00
4	341	2829	2.99	41860.00
5	341	3130	7.99	111860.00
6	341	3382	5.99	83860.00
7	342	2190	5.99	83860.00
8	342	2914	5.99	83860.00
9	342	3081	2.99	41860.00
10	342	1547	4.00	60060.00

Penggunaan Operator Aritmatika

Contoh 5 : Hasil pembagian staff_id dengan rental_id dari tabel payment.

Query Editor		Query History	
1	select	staff_id, rental_id, staff_id / rental_id	
2	from	payment	

Data Output		Messages	Explain	Notifications
	staff_id smallint	rental_id integer	?column? integer	
1	2	1520	0	
2	1	1778	0	
3	1	1849	0	
4	2	2829	0	
5	2	3130	0	
6	1	3382	0	
7	2	2190	0	

?

2 / 1520 =

0.00131578947

SQL Fundamentals



Introduction to Sample DB



SQL Modification: CREATE



SQL Basic Queries: SELECT



SQL Basic Queries: LIMIT



SQL Basic Queries: WHERE



SQL Basic Queries: ORDER BY



SQL Arithmetic Operation



SQL Basic Queries FAQ



SQL Modification: INSERT




SQL Modification: UPDATE



SQL Modification: DELETE

SQL Fundamentals

- ☒ Introduction to Sample DB
- ☒ SQL Modification: CREATE
- ☒ SQL Basic Queries: SELECT
- ☒ SQL Basic Queries: LIMIT
- ☒ SQL Basic Queries: WHERE
- ☒ SQL Basic Queries: ORDER BY

- ☒ SQL Arithmetic Operation
- ☒  SQL Basic Queries FAQ
- ☐ SQL Modification: INSERT
- ☐ SQL Modification: UPDATE
- ☐ SQL Modification: DELETE

F.A.Q

- Apakah SQL case sensitive? Huruf kecil / besar sama tidak?**
 Keyword di SQL tidak case sensitive. Namun untuk nama tabel / kolom mengikuti aturan RDBMS yang digunakan, ada yang case sensitive dan ada juga yang tidak.
- Apakah harus menggunakan semicolon (;) di akhir query?**
 Semicolon menandakan akhir dari satu query. Penggunaan semicolon ini wajib jika ingin menjalankan lebih dari satu query sekaligus. Jika hanya ingin menjalankan satu query, tergantung dari aturan RDBMS yang digunakan, biasanya tidak perlu.
- Apakah ada ketentuan tertentu untuk indentasi?**
 Tidak ada. Indentasi biasanya hanya untuk keperluan kosmetik/estetika agar query lebih mudah dipahami.

SQL Fundamentals



Introduction to Sample DB



SQL Modification: CREATE



SQL Basic Queries: SELECT



SQL Basic Queries: LIMIT



SQL Basic Queries: WHERE



SQL Basic Queries: ORDER BY



SQL Arithmetic Operation



SQL Basic Queries FAQ



SQL Modification: INSERT




SQL Modification: UPDATE



SQL Modification: DELETE

SQL Fundamentals

- ☒ Introduction to Sample DB
- ☒ SQL Modification: CREATE
- ☒ SQL Basic Queries: SELECT
- ☒ SQL Basic Queries: LIMIT
- ☒ SQL Basic Queries: WHERE
- ☒ SQL Basic Queries: ORDER BY

- ☒ SQL Arithmetic Operation
- ☒ SQL Basic Queries FAQ
-  ☐ SQL Modification: INSERT
- ☐ SQL Modification: UPDATE
- ☐ SQL Modification: DELETE

INSERT - Definisi

INSERT merupakan klausa untuk melakukan penambahan data baru kedalam sebuah tabel.

Perintah Dalam SQL

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

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

Hands-On

Inputkan data kedalam tabel `rakamin_merchant_active`, yang valuenya berupa:

- `merchant_id = 1`
- `merchant_name = 'merchant_1'`
- `active = true`



tutorial/postgres@local@postgres ▾

Query Editor

Query History

```
1  INSERT INTO rakamin_merchant_active
2  VALUES (1, 'merchant_1', TRUE);
```

Hands-On

Inputkan data kedalam tabel rakamin_merchant_active, yang valuenya berupa:

- merchant_id -> 2, 3, dan 4
- merchant_name -> 'merchant_2', 'merchant_3' dan 'merchant_4'
- active -> TRUE, FALSE, TRUE



tutorial/postgres@local@postgres ▾

Query Editor

Query History

```













1  INSERT INTO rakamin_merchant_active
2  VALUES
3      (2, 'merchant_2', TRUE),
4      (3, 'merchant_3', FALSE),
5      (4, 'merchant_4', TRUE);

```

SQL Fundamentals

- | | | | |
|-------------------------------------|-----------------------------|-------------------------------------|--------------------------|
| <input checked="" type="checkbox"/> | Introduction to Sample DB | <input checked="" type="checkbox"/> | SQL Arithmetic Operation |
| <input checked="" type="checkbox"/> | SQL Modification: CREATE | <input checked="" type="checkbox"/> | SQL Basic Queries FAQ |
| <input checked="" type="checkbox"/> | SQL Basic Queries: SELECT | <input checked="" type="checkbox"/> | SQL Modification: INSERT |
| <input checked="" type="checkbox"/> | SQL Basic Queries: LIMIT | <input type="checkbox"/> | SQL Modification: UPDATE |
| <input checked="" type="checkbox"/> | SQL Basic Queries: WHERE | <input type="checkbox"/> | SQL Modification: DELETE |
| <input checked="" type="checkbox"/> | SQL Basic Queries: ORDER BY | | |

SQL Fundamentals

- | | | | |
|---|-----------------------------|--|--------------------------|
|  | Introduction to Sample DB |  | SQL Arithmetic Operation |
|  | SQL Modification: CREATE |  | SQL Basic Queries FAQ |
|  | SQL Basic Queries: SELECT |  | SQL Modification: INSERT |
|  | SQL Basic Queries: LIMIT |   | SQL Modification: UPDATE |
|  | SQL Basic Queries: WHERE |  | SQL Modification: DELETE |
|  | SQL Basic Queries: ORDER BY | | |

UPDATE

Nilai-nilai dalam tabel di setiap kolom dapat diubah sesuai kebutuhan. Dengan klausa UPDATE, kita dapat mengubah nilai pada kolom-kolom tertentu.

Perintah dalam SQL

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


Hands-On

Ubah status aktif dari merchant_3 menjadi aktif pada tabel rakamin_merchant_active

- merchant_id = 1
- merchant_name = 'merchant_1'
- active = true

tutorial/postgres@local@postgres ▾

Query Editor Query History

```
139
140 SELECT * FROM rakamin_merchant_active;
141
142
```

Data Output Explain Messages Notifications

	merchant_id [PK] integer	merchant_name character varying (50)	active boolean
1	1	merchant_1	true
2	2	merchant_2	true
3	3	merchant_3	false
4	4	merchant_4	true

tutorial/postgres@local@postgres ▾

Query Editor Query History

```
1 UPDATE rakamin_merchant_active
2 SET active = TRUE
3 WHERE merchant_name = 'merchant_3';
```

Data Output Explain Messages Notifications

UPDATE 1

Query returned successfully in 41 msec.

SQL Fundamentals



Introduction to Sample DB



SQL Modification: CREATE



SQL Basic Queries: SELECT



SQL Basic Queries: LIMIT



SQL Basic Queries: WHERE



SQL Basic Queries: ORDER BY



SQL Arithmetic Operation



SQL Basic Queries FAQ



SQL Modification: INSERT









SQL Modification: UPDATE



SQL Modification: DELETE

SQL Fundamentals

-  Introduction to Sample DB
-  SQL Modification: CREATE
-  SQL Basic Queries: SELECT
-  SQL Basic Queries: LIMIT
-  SQL Basic Queries: WHERE
-  SQL Basic Queries: ORDER BY

-  SQL Arithmetic Operation
-  SQL Basic Queries FAQ
-  SQL Modification: INSERT
-  SQL Modification: UPDATE
-   SQL Modification: DELETE

DELETE


Untuk menghapus data pada sebuah tabel, dapat menggunakan klausa DELETE

Perintah dalam SQL

```
DELETE FROM table_name  
WHERE condition
```

HANDS-ON

Hapus baris dari tabel `rakamin_merchant_active` dengan `merchant_id = 2`

 tutorial/postgres@local@postgres ▾


[Query Editor](#)
[Query History](#)

```

139
140 SELECT * FROM rakamin_merchant_active;
141
142
        
```

[Data Output](#)
[Explain](#)
[Messages](#)
[Notifications](#)

	merchant_id [PK] integer	merchant_name character varying (50)	active boolean
1	1	merchant_1	true
2	2	merchant_2	true
3	4	merchant_4	true
4	3	merchant_3	true

 tutorial/postgres@local@postgres ▾

[Query Editor](#)
[Query History](#)

```

1 DELETE FROM rakamin_merchant_active
2 WHERE merchant_id = 2;
        
```

[Data Output](#)
[Explain](#)
[Messages](#)
[Notifications](#)

DELETE 1

Query returned successfully in 41 msec.

NOTES

Jika ingin mengubah atau menghapus data tertentu menggunakan **UPDATE** atau **DELETE**, jangan lupa untuk menambahkan klausa **WHERE** sebab akan berakibat **FATAL**, karena dapat memanipulasi keseluruhan data.

SQL Fundamentals



Introduction to Sample DB



SQL Modification: CREATE



SQL Basic Queries: SELECT



SQL Basic Queries: LIMIT



SQL Basic Queries: WHERE



SQL Basic Queries: ORDER BY



SQL Arithmetic Operation



SQL Basic Queries FAQ



SQL Modification: INSERT



SQL Modification: UPDATE



SQL Modification: DELETE