101 SQL

Apr 2023

Download MySQL

Windows: https://dev.mysql.com/downloads/installer/

MacOS (DMG): https://dev.mysql.com/downloads/mysql/

+ https://dev.mysql.com/downloads/workbench/

macOS說明: https://dev.mysql.com/doc/refman/8.0/en/windows-installation-layout.html

Structured Query Language

- NOT a “programming” language
- Relational database vs non-relational database

mongoDB

Microsoft SQL Server

ORACLE

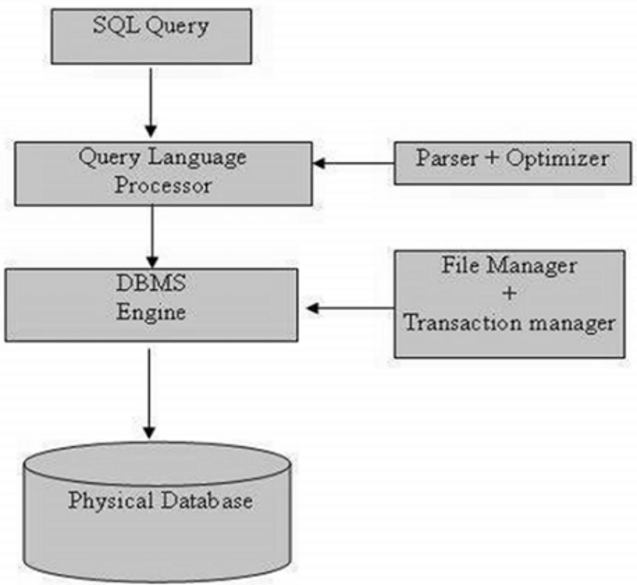
方言: PL/SQL, T-SQL

MariaDB

MySQL

Amazon RDS

PostgreSQL



SQL Query

Query Language Processor

Parser + Optimizer

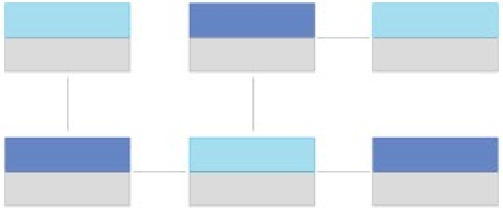
DBMS Engine

File Manager + Transaction manager

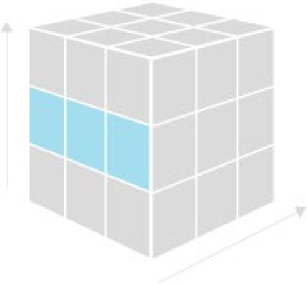
Physical Database

SQL

Relational

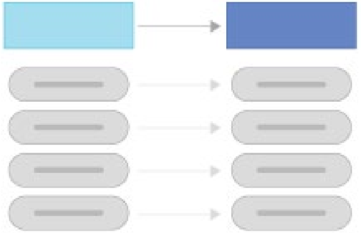


Analytical

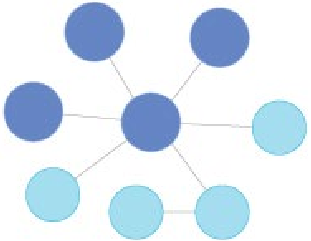


NoSQL

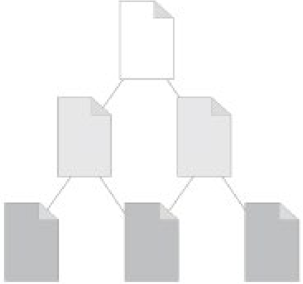
Key - Value



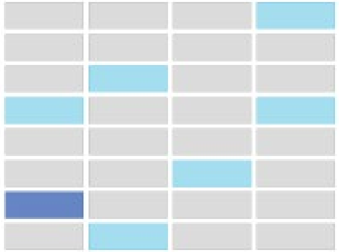
Graph



Document



Wide Column



Create Database

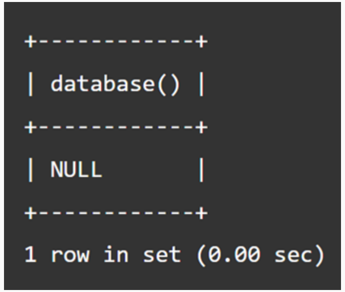
```
CREATE DATABASE testDB;

SHOW DATABASES;
```

Drop database

```
DROP DATABASE testDB;

SHOW DATABASES;
```

Use database

```
/* create another database first */

SELECT database();
```



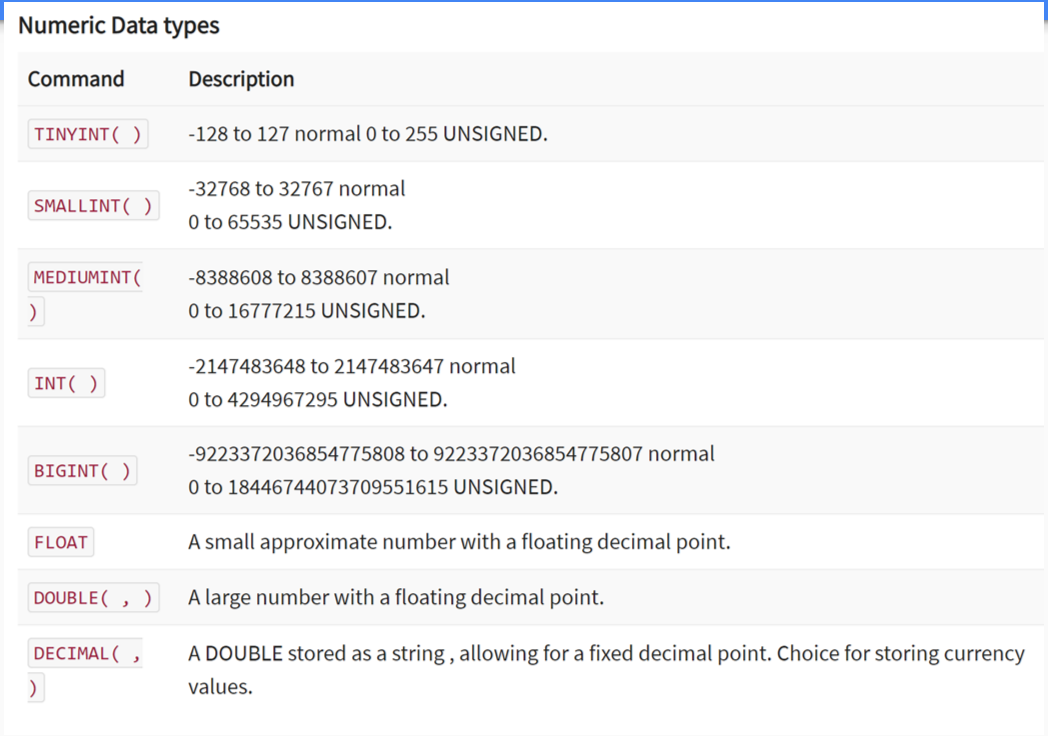
```
+------------+
| database() |
+------------+
| NULL       |
+------------+
1 row in set (0.00 sec)
```

```
USE testDB;

-- select a database

SELECT database();
```

https://dev.mysql.com/doc/refman/8.0/en/data-types.html
https://www.guru99.com/sql-cheat-sheet.html

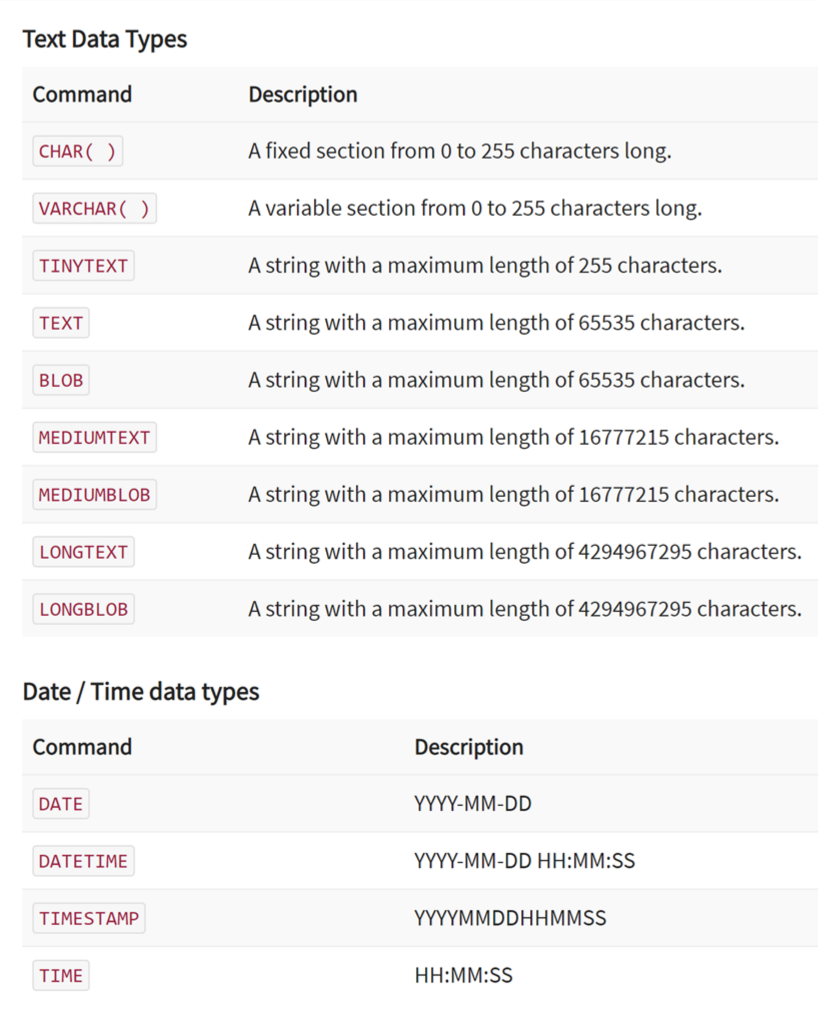
Data types (データタイプ)

numeric:



Numeric Data types

Command	Description
`TINYINT( )`	-128 to 127 normal 0 to 255 UNSIGNED.
`SMALLINT( )`	-32768 to 32767 normal 0 to 65535 UNSIGNED.
`MEDIUMINT( )`	-8388608 to 8388607 normal 0 to 16777215 UNSIGNED.
`INT( )`	-2147483648 to 2147483647 normal 0 to 4294967295 UNSIGNED.
`BIGINT( )`	-9223372036854775808 to 9223372036854775807 normal 0 to 18446744073709551615 UNSIGNED.
`FLOAT`	A small approximate number with a floating decimal point.
`DOUBLE( , )`	A large number with a floating decimal point.
`DECIMAL( , )`	A DOUBLE stored as a string , allowing for a fixed decimal point. Choice for storing currency values.



Text Data Types

Command	Description
`CHAR( )`	A fixed section from 0 to 255 characters long.
`VARCHAR( )`	A variable section from 0 to 255 characters long.
`TINYTEXT`	A string with a maximum length of 255 characters.
`TEXT`	A string with a maximum length of 65535 characters.
`BLOB`	A string with a maximum length of 65535 characters.
`MEDIUMTEXT`	A string with a maximum length of 16777215 characters.
`MEDIUMBLOB`	A string with a maximum length of 16777215 characters.
`LONGTEXT`	A string with a maximum length of 4294967295 characters.
`LONGBLOB`	A string with a maximum length of 4294967295 characters.

Date / Time data types

Command	Description
`DATE`	YYYY-MM-DD
`DATETIME`	YYYY-MM-DD HH:MM:SS
`TIMESTAMP`	YYYYMMDDHHMMSS
`TIME`	HH:MM:SS

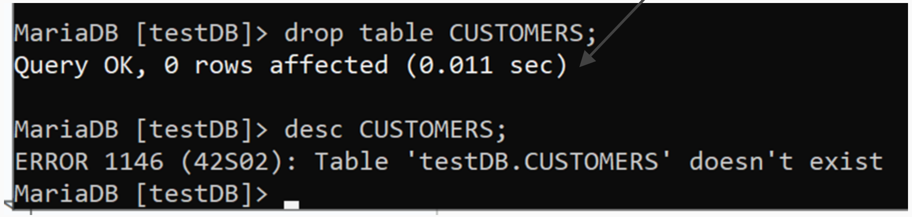
Create table

```
CREATE TABLE CUSTOMERS(
  ID   INT                NOT NULL,
  NAME VARCHAR (20)       NOT NULL,
  AGE  INT                NOT NULL,
  ADDRESS  VARCHAR (75) ,
  SALARY   DECIMAL (18, 2),
  PRIMARY KEY (ID)
);
```

```
show tables;

desc CUSTOMERS;
```

Drop table

DROP TABLE CUSTOMERS;



戻り値: long

```
MariaDB [testDB]> drop table CUSTOMERS;
Query OK, 0 rows affected (0.011 sec)

MariaDB [testDB]> desc CUSTOMERS;
ERROR 1146 (42S02): Table 'testDB.CUSTOMERS' doesn't exist
MariaDB [testDB]> _
```

Insert

```
CREATE TABLE CUSTOMERS(
  ID   INT                  NOT NULL   AUTO_INCREMENT,
  NAME VARCHAR (20)           NOT NULL,
  AGE  INT                  NOT NULL,
  ADDRESS  VARCHAR (75) ,
  SALARY   DECIMAL (18, 2),
  PRIMARY KEY (ID)
);
```

AUTO_INCREMENT=1

```
INSERT INTO CUSTOMERS
(NAME,AGE,ADDRESS,SALARY)
VALUES ('Cody', 25, 'Delhi', 1500.00);

...

INSERT INTO CUSTOMERS
(NAME,AGE,ADDRESS,SALARY)
VALUES (...);
```

Select

```
SELECT * FROM CUSTOMERS;
```

```
SELECT ID, NAME, AGE FROM CUSTOMERS

WHERE SALARY > 1000;
```

```
SELECT * FROM CUSTOMERS

WHERE AGE < 30 AND SALARY = 2000.00;
```

OR

```
SELECT * FROM CUSTOMERS

WHERE NAME = 'Cody';
```

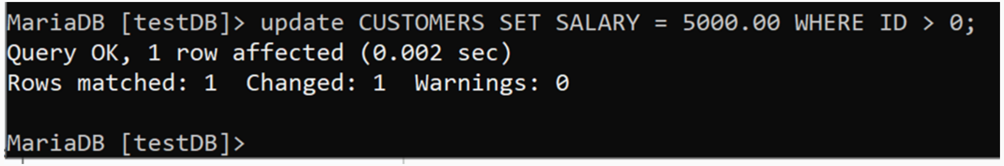
WHERE **NOT** AGE > 30

Update

UPDATE CUSTOMERS SET SALARY = 5000.00
WHERE NAME = 'Cody';

UPDATE CUSTOMERS

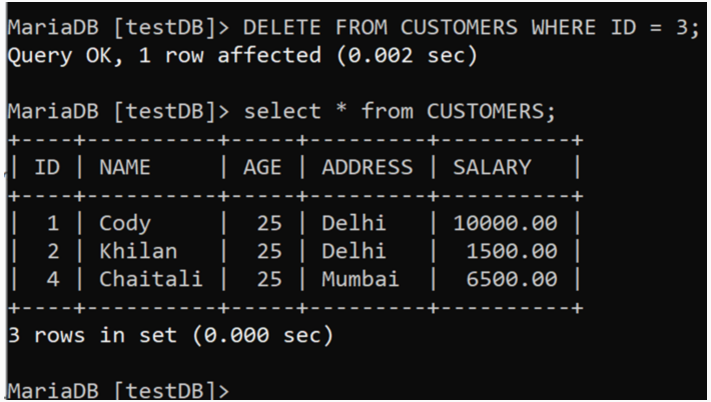
SET SALARY = 10000.00;



```
MariaDB [testDB]> update CUSTOMERS SET SALARY = 5000.00 WHERE ID > 0;
Query OK, 1 row affected (0.002 sec)
Rows matched: 1  Changed: 1  Warnings: 0

MariaDB [testDB]>
```

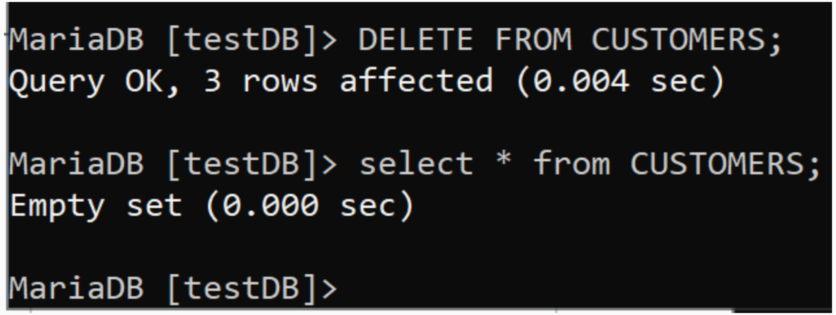
Delete

DELETE FROM CUSTOMERS WHERE ID = 3;



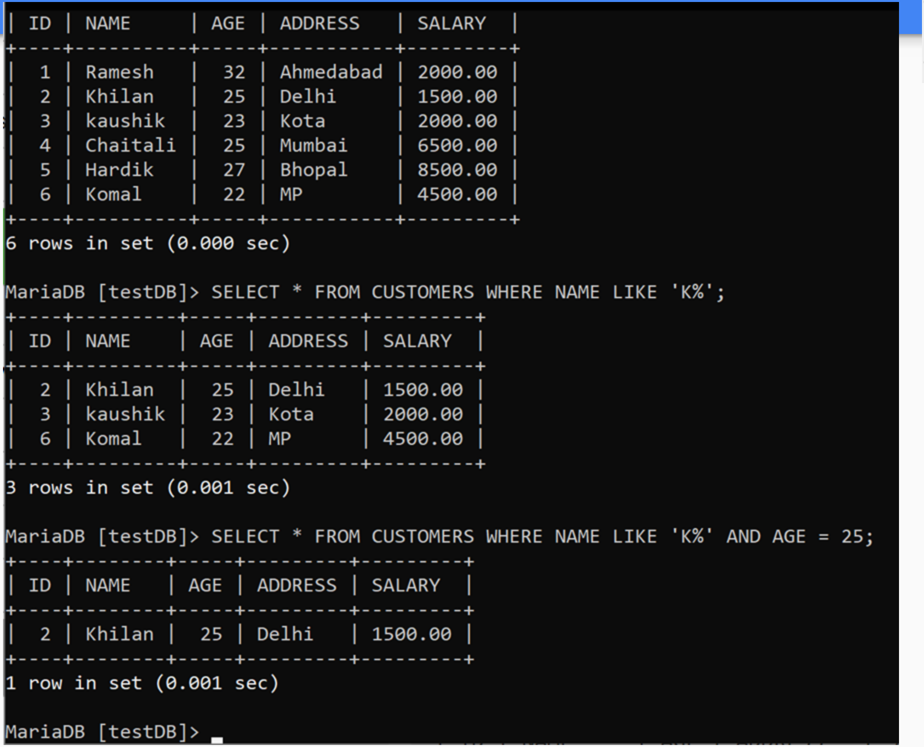
```
MariaDB [testDB]> DELETE FROM CUSTOMERS WHERE ID = 3;
Query OK, 1 row affected (0.002 sec)

MariaDB [testDB]> select * from CUSTOMERS;
+----+----------+-----+---------+----------+
| ID | NAME     | AGE | ADDRESS | SALARY   |
+----+----------+-----+---------+----------+
|  1 | Cody     |  25 | Delhi   | 10000.00 |
|  2 | Khilan   |  25 | Delhi   |  1500.00 |
|  4 | Chaitali |  25 | Mumbai  |  6500.00 |
+----+----------+-----+---------+----------+
3 rows in set (0.000 sec)

MariaDB [testDB]>
```

DELETE FROM CUSTOMERS;

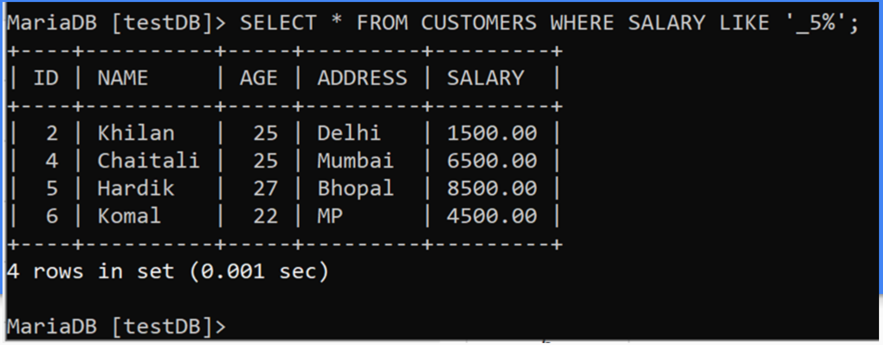


```
MariaDB [testDB]> DELETE FROM CUSTOMERS;
Query OK, 3 rows affected (0.004 sec)

MariaDB [testDB]> select * from CUSTOMERS;
Empty set (0.000 sec)

MariaDB [testDB]>
```

“LIKE”



```
| ID | NAME     | AGE | ADDRESS   | SALARY  |
+----+----------+-----+-----------+---------+
|  1 | Ramesh   |  32 | Ahmedabad | 2000.00 |
|  2 | Khilan   |  25 | Delhi     | 1500.00 |
|  3 | kaushik  |  23 | Kota      | 2000.00 |
|  4 | Chaitali |  25 | Mumbai    | 6500.00 |
|  5 | Hardik   |  27 | Bhopal    | 8500.00 |
|  6 | Komal    |  22 | MP        | 4500.00 |
+----+----------+-----+-----------+---------+
6 rows in set (0.000 sec)

MariaDB [testDB]> SELECT * FROM CUSTOMERS WHERE NAME LIKE 'K%';
+----+---------+-----+---------+---------+
| ID | NAME    | AGE | ADDRESS | SALARY  |
+----+---------+-----+---------+---------+
|  2 | Khilan  |  25 | Delhi   | 1500.00 |
|  3 | kaushik |  23 | Kota    | 2000.00 |
|  6 | Komal   |  22 | MP      | 4500.00 |
+----+---------+-----+---------+---------+
3 rows in set (0.001 sec)

MariaDB [testDB]> SELECT * FROM CUSTOMERS WHERE NAME LIKE 'K%' AND AGE = 25;
+----+--------+-----+---------+---------+
| ID | NAME   | AGE | ADDRESS | SALARY  |
+----+--------+-----+---------+---------+
|  2 | Khilan |  25 | Delhi   | 1500.00 |
+----+--------+-----+---------+---------+
1 row in set (0.001 sec)

MariaDB [testDB]> _
```



```
MariaDB [testDB]> SELECT * FROM CUSTOMERS WHERE SALARY LIKE '_5%';
+----+----------+-----+---------+---------+
| ID | NAME     | AGE | ADDRESS | SALARY  |
+----+----------+-----+---------+---------+
|  2 | Khilan   |  25 | Delhi   | 1500.00 |
|  4 | Chaitali |  25 | Mumbai  | 6500.00 |
|  5 | Hardik   |  27 | Bhopal  | 8500.00 |
|  6 | Komal    |  22 | MP      | 4500.00 |
+----+----------+-----+---------+---------+
4 rows in set (0.001 sec)

MariaDB [testDB]>
```

SELECT * FROM CUSTOMERS WHERE NAME LIKE 'K%';

% _ ~~* ?~~

WHERE AGE LIKE '%2'

WHERE NAME LIKE '%k%'

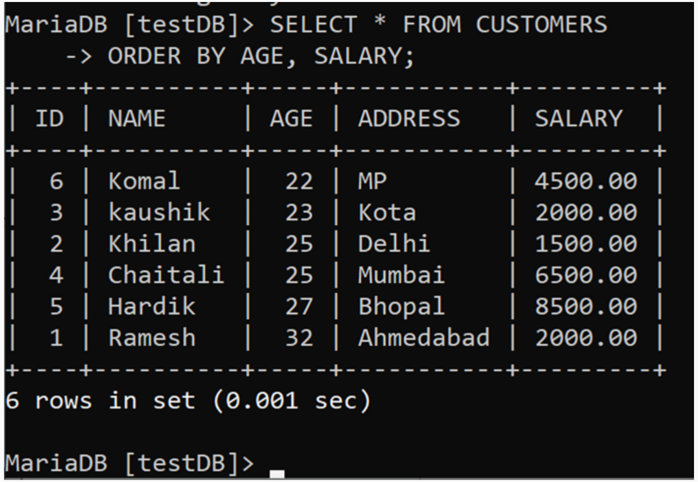
WHERE SALARY LIKE '_5%0'

LIMIT

```
/* MySQL: */

SELECT * FROM CUSTOMERS LIMIT 3;
```

Order by

SELECT * FROM CUSTOMERS
ORDER BY AGE, SALARY;



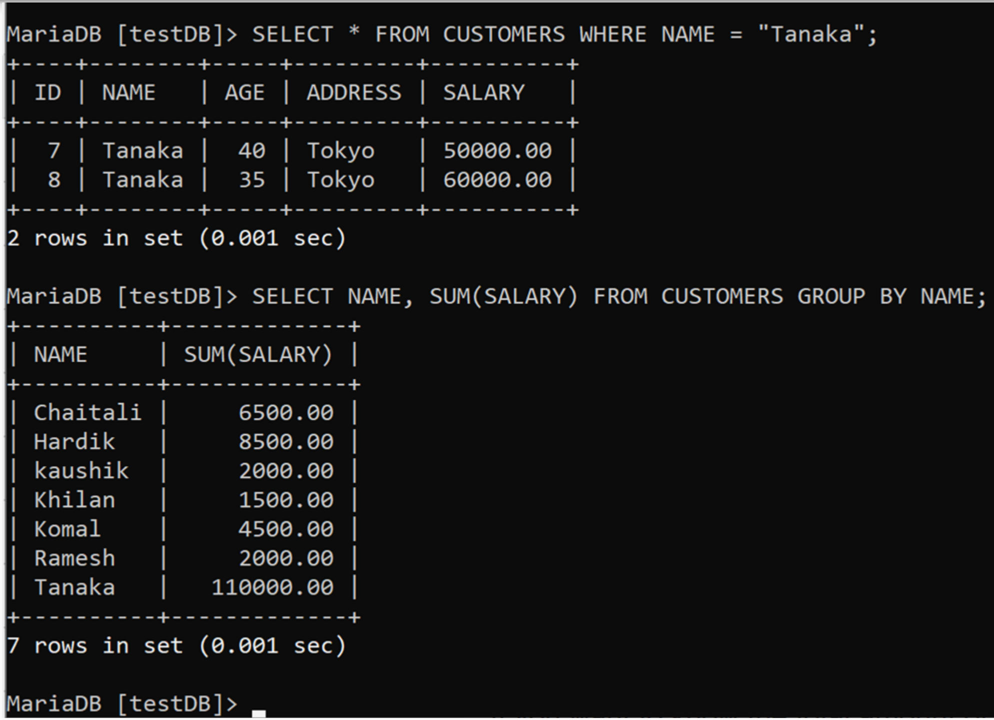
```
MariaDB [testDB]> SELECT * FROM CUSTOMERS
    -> ORDER BY AGE, SALARY;
+----+----------+-----+-----------+---------+
| ID | NAME     | AGE | ADDRESS   | SALARY  |
+----+----------+-----+-----------+---------+
|  6 | Komal    |  22 | MP        | 4500.00 |
|  3 | kaushik  |  23 | Kota      | 2000.00 |
|  2 | Khilan   |  25 | Delhi     | 1500.00 |
|  4 | Chaitali |  25 | Mumbai    | 6500.00 |
|  5 | Hardik   |  27 | Bhopal    | 8500.00 |
|  1 | Ramesh   |  32 | Ahmedabad | 2000.00 |
+----+----------+-----+-----------+---------+
6 rows in set (0.001 sec)

MariaDB [testDB]> _
```

SELECT * FROM CUSTOMERS
ORDER BY NAME **DESC**; ↓

ASC ↑

Group by

```
SELECT NAME, COUNT(NAME),
SUM(SALARY)
FROM CUSTOMERS
WHERE AGE > 18
GROUP BY NAME
ORDER BY SUM(SALARY) DESC;
```



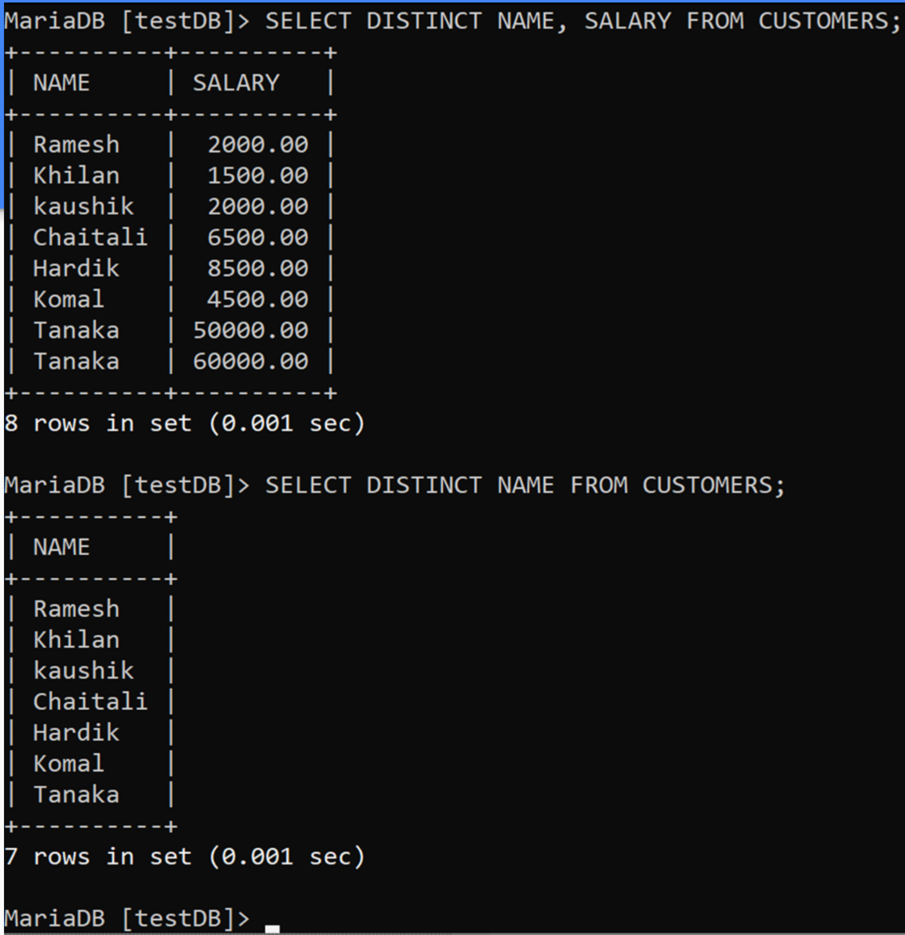
```
MariaDB [testDB]> SELECT * FROM CUSTOMERS WHERE NAME = "Tanaka";
+----+--------+-----+---------+----------+
| ID | NAME   | AGE | ADDRESS | SALARY   |
+----+--------+-----+---------+----------+
|  7 | Tanaka |  40 | Tokyo   | 50000.00 |
|  8 | Tanaka |  35 | Tokyo   | 60000.00 |
+----+--------+-----+---------+----------+
2 rows in set (0.001 sec)

MariaDB [testDB]> SELECT NAME, SUM(SALARY) FROM CUSTOMERS GROUP BY NAME;
+----------+-------------+
| NAME     | SUM(SALARY) |
+----------+-------------+
| Chaitali |     6500.00 |
| Hardik   |     8500.00 |
| kaushik  |     2000.00 |
| Khilan   |     1500.00 |
| Komal    |     4500.00 |
| Ramesh   |     2000.00 |
| Tanaka   |   110000.00 |
+----------+-------------+
7 rows in set (0.001 sec)

MariaDB [testDB]> _
```

Select distinct

SELECT **DISTINCT** NAME FROM CUSTOMERS;

MAX()

SELECT MIN(AGE) AS youngest
FROM CUSTOMERS;



```
MariaDB [testDB]> SELECT DISTINCT NAME, SALARY FROM CUSTOMERS;
+----------+----------+
| NAME     | SALARY   |
+----------+----------+
| Ramesh   |  2000.00 |
| Khilan   |  1500.00 |
| kaushik  |  2000.00 |
| Chaitali |  6500.00 |
| Hardik   |  8500.00 |
| Komal    |  4500.00 |
| Tanaka   | 50000.00 |
| Tanaka   | 60000.00 |
+----------+----------+
8 rows in set (0.001 sec)

MariaDB [testDB]> SELECT DISTINCT NAME FROM CUSTOMERS;
+----------+
| NAME     |
+----------+
| Ramesh   |
| Khilan   |
| kaushik  |
| Chaitali |
| Hardik   |
| Komal    |
| Tanaka   |
+----------+
7 rows in set (0.001 sec)

MariaDB [testDB]> _
```

Constraints (制約)

- NOT NULL
- DEFAULT
- UNIQUE
- PRIMARY
- FOREIGN
- CHECK
- INDEX

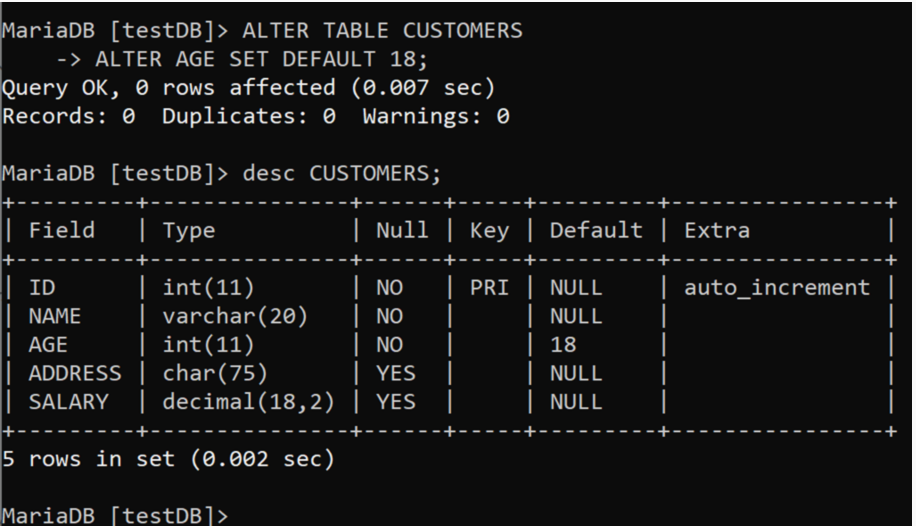
default

```
CREATE TABLE CUSTOMERS_2(
  ID   INT              NOT NULL    AUTO_INCREMENT,
  NAME VARCHAR (20)          NOT NULL,
  AGE  INT                NOT NULL CHECK (AGE >= 18),
 ADDRESS VARCHAR (75)  DEFAULT 'Tokyo',
  SALARY   DECIMAL (18, 2) UNIQUE,
  PRIMARY KEY (ID)
);
```

```
CREATE TABLE ORDERS (
            CUSTOMERS_NAME  VARCHAR(20),
            OrderNumber int NOT NULL
AUTO_INCREMENT  PRIMARY  KEY,
            OrderDate DATE DEFAULT CURDATE()
);

ALTER TABLE ORDERS RENAME COLUMN
CUSTOMERS_NAME  TO NAME;
create index idx_name on CUSTOMERS(NAME);
ALTER TABLE ORDERS ADD FOREIGN KEY (NAME)
REFERENCES CUSTOMERS (NAME);
```

ALTER, MODIFY

ALTER TABLE CUSTOMERS
ALTER AGE SET DEFAULT 18;

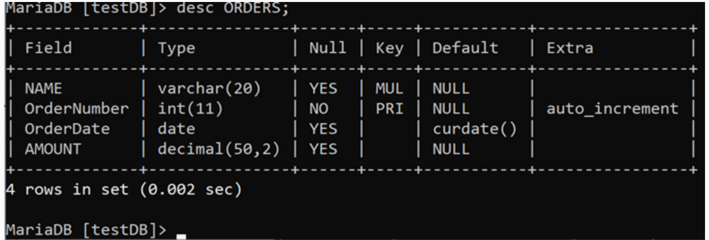


```
MariaDB [testDB]> ALTER TABLE CUSTOMERS
    -> ALTER AGE SET DEFAULT 18;
Query OK, 0 rows affected (0.007 sec)
Records: 0  Duplicates: 0  Warnings: 0

MariaDB [testDB]> desc CUSTOMERS;
+---------+---------------+------+-----+---------+----------------+
| Field   | Type          | Null | Key | Default | Extra          |
+---------+---------------+------+-----+---------+----------------+
| ID      | int(11)       | NO   | PRI | NULL    | auto_increment |
| NAME    | varchar(20)   | NO   |     | NULL    |                |
| AGE     | int(11)       | NO   |     | 18      |                |
| ADDRESS | char(75)      | YES  |     | NULL    |                |
| SALARY  | decimal(18,2) | YES  |     | NULL    |                |
+---------+---------------+------+-----+---------+----------------+
5 rows in set (0.002 sec)

MariaDB [testDB]>
```

ALTER TABLE CUSTOMERS
ALTER ADDRESS DROP DEFAULT;

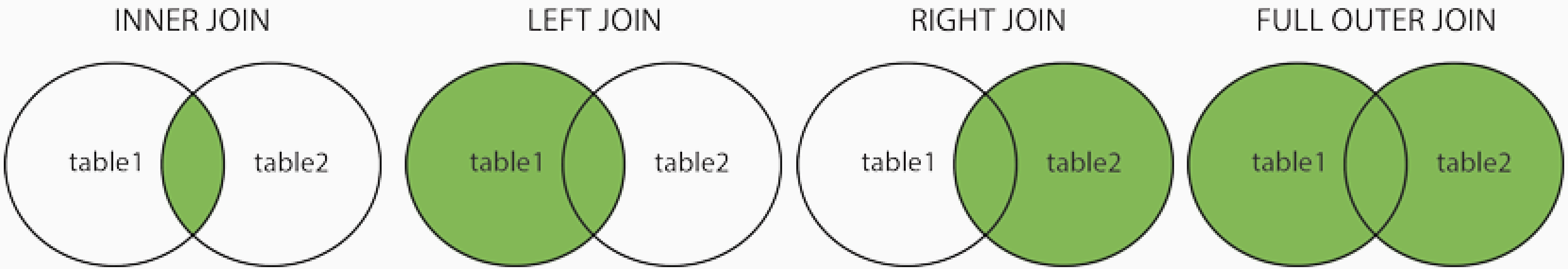
ALTER TABLE CUSTOMERS
MODIFY SALARY DECIMAL (30, 2);

ALTER TABLE ORDERS
ADD AMOUNT DECIMAL (50, 2);

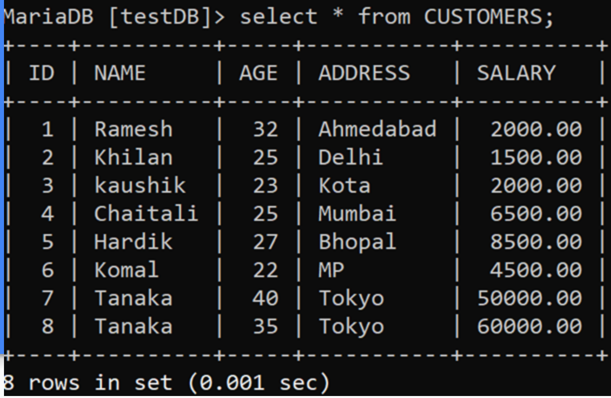


```
MariaDB [testDB]> desc ORDERS;
+-------------+---------------+------+-----+-----------+----------------+
| Field       | Type          | Null | Key | Default   | Extra          |
+-------------+---------------+------+-----+-----------+----------------+
| NAME        | varchar(20)   | YES  | MUL | NULL      |                |
| OrderNumber | int(11)       | NO   | PRI | NULL      | auto_increment |
| OrderDate   | date          | YES  |     | curdate() |                |
| AMOUNT      | decimal(50,2) | YES  |     | NULL      |                |
+-------------+---------------+------+-----+-----------+----------------+
4 rows in set (0.002 sec)

MariaDB [testDB]>
```

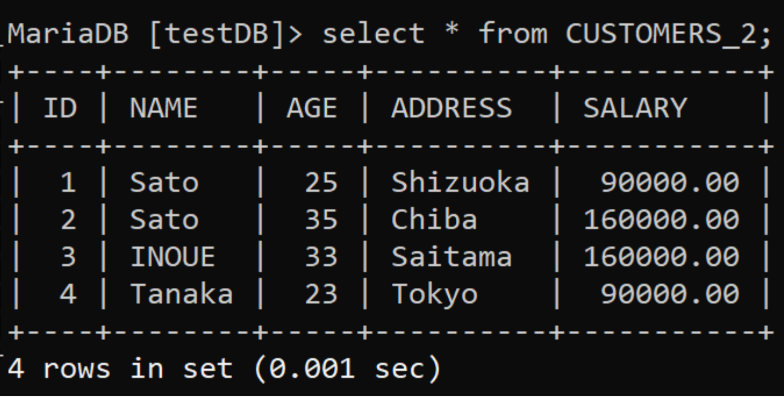
Join


INNER JOIN
table1
table2
LEFT JOIN
table1
table2
RIGHT JOIN
table1
table2
FULL OUTER JOIN
table1
table2

Inner Join



```
MariaDB [testDB]> select * from CUSTOMERS;
+----+----------+-----+-----------+----------+
| ID | NAME     | AGE | ADDRESS   | SALARY   |
+----+----------+-----+-----------+----------+
|  1 | Ramesh   |  32 | Ahmedabad |  2000.00 |
|  2 | Khilan   |  25 | Delhi     |  1500.00 |
|  3 | kaushik  |  23 | Kota      |  2000.00 |
|  4 | Chaitali |  25 | Mumbai    |  6500.00 |
|  5 | Hardik   |  27 | Bhopal    |  8500.00 |
|  6 | Komal    |  22 | MP        |  4500.00 |
|  7 | Tanaka   |  40 | Tokyo     | 50000.00 |
|  8 | Tanaka   |  35 | Tokyo     | 60000.00 |
+----+----------+-----+-----------+----------+
8 rows in set (0.001 sec)
```

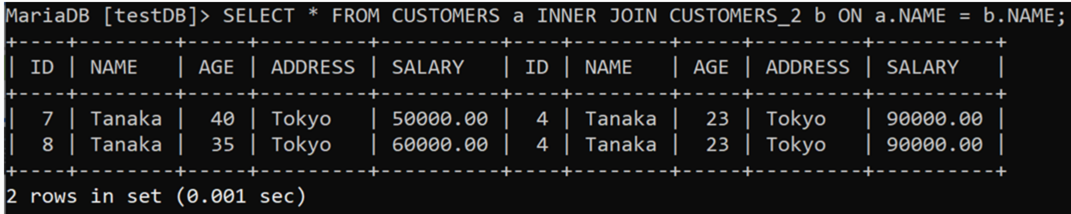


```
MariaDB [testDB]> select * from CUSTOMERS_2;
+----+--------+-----+----------+-----------+
| ID | NAME   | AGE | ADDRESS  | SALARY    |
+----+--------+-----+----------+-----------+
|  1 | Sato   |  25 | Shizuoka |  90000.00 |
|  2 | Sato   |  35 | Chiba    | 160000.00 |
|  3 | INOUE  |  33 | Saitama  | 160000.00 |
|  4 | Tanaka |  23 | Tokyo    |  90000.00 |
+----+--------+-----+----------+-----------+
4 rows in set (0.001 sec)
```

```
CREATE TABLE CUSTOMERS_2(
  ID   INT     NOT NULL    AUTO_INCREMENT,
  NAME     VARCHAR (20)       NOT NULL,
  AGE  INT   NOT NULL  CHECK (AGE >= 18),
  ADDRESS VARCHAR (75)  DEFAULT 'Tokyo',
  SALARY   DECIMAL (18, 2) UNIQUE,
  PRIMARY KEY (ID)
);
```

```
SELECT * FROM CUSTOMERS a
INNER JOIN CUSTOMERS_2 b
ON a.NAME = b.NAME;
```

```
INSERT INTO CUSTOMERS_2 (NAME, AGE,
ADDRESS, SALARY) VALUES (...);
```



```
MariaDB [testDB]> SELECT * FROM CUSTOMERS a INNER JOIN CUSTOMERS_2 b ON a.NAME = b.NAME;
+----+--------+-----+---------+----------+----+--------+-----+---------+----------+
| ID | NAME   | AGE | ADDRESS | SALARY   | ID | NAME   | AGE | ADDRESS | SALARY   |
+----+--------+-----+---------+----------+----+--------+-----+---------+----------+
|  7 | Tanaka |  40 | Tokyo   | 50000.00 |  4 | Tanaka |  23 | Tokyo   | 90000.00 |
|  8 | Tanaka |  35 | Tokyo   | 60000.00 |  4 | Tanaka |  23 | Tokyo   | 90000.00 |
+----+--------+-----+---------+----------+----+--------+-----+---------+----------+
2 rows in set (0.001 sec)
```

Left join, right join

```
SELECT * FROM CUSTOMERS a
LEFT JOIN CUSTOMERS_2 b
ON a.NAME = b.NAME;
```

```
SELECT * FROM CUSTOMERS a
RIGHT JOIN CUSTOMERS_2 b
ON a.NAME = b.NAME;
```

Full join

```
SELECT * FROM CUSTOMERS
FULL JOIN CUSTOMERS_2;
```

Union

```
SELECT ID, NAME, AGE FROM
CUSTOMERS a
UNION
SELECT ID, NAME, AGE FROM
CUSTOMERS_2 b;
```

union all: allow duplicate values

SQL injection

```
$name = "hacker'; DELETE FROM CUSTOMERS;";

mysql_query("SELECT * FROM CUSTOMERS WHERE name='{$name}'");
```

PHP: mysql_query() does not allow multiple queries in a single function call

Homework

有感情地朗读并背诵全文