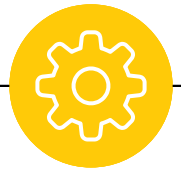
Welcome back



SQL Essentials

Roadmap


RDBMS
ER Model
SQL Server
DDL
DML
Operator
Function
SQL Clause
Join data
Sub Query
Best
Practice

Previous lecture

Data Definition Language

- Fun with database
- SQL data type
- Table In database
- Meaningful data with constraints
- SQL Process

MySQL Workbench

- Structure query language
- Take a look on UI
- First command
- SQL Components

What we will explore today?

DML

- INSERT
- UPDATE
- DELETE

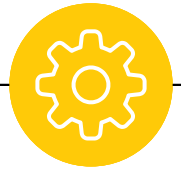
Select

- SELECT Syntax
- TOP & PERCENT
- ORDER
- ALIAS
- DISTINCT
- WHERE
- VIEW
- SELECT INTO

Operator

- Authentic
- Compare
- Logical

Data Manipulation Language (DML)



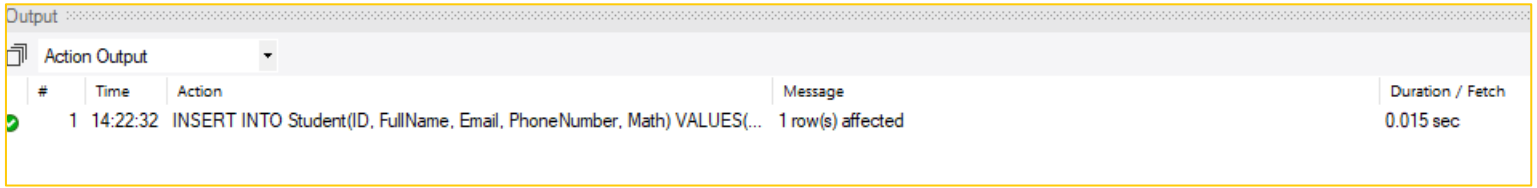
Prepair the table structure

```
DROP DATABASE IF EXISTS LECTURE3_CRUD_DATA;

CREATE DATABASE LECTURE3_CRUD_DATA;

USE LECTURE3_CRUD_DATA;

CREATE TABLE Student(
        ID int PRIMARY KEY,
        FullName varchar(15) NOT NULL, Email char(15) UNIQUE,
        PhoneNumber varchar(12),
        DateOfBirth date default '2000-10-25',
        Math tinyint CHECK (Math >= 0 AND Math <= 10)
);
```

Insert data into table syntax

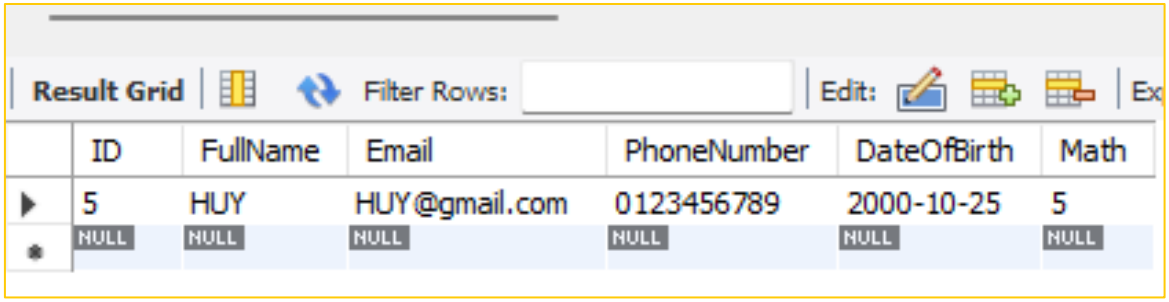
```
INSERT INTO table_name (column1, column2, column3)
VALUES (value1, value2, value3);
```

```
INSERT INTO Student(ID, FullName, Email, PhoneNumber, Math)
VALUES(5, 'HUY', 'HUY@gmail.com', '0123456789', 5);
```



Check it

```
SELECT *
FROM Student
```

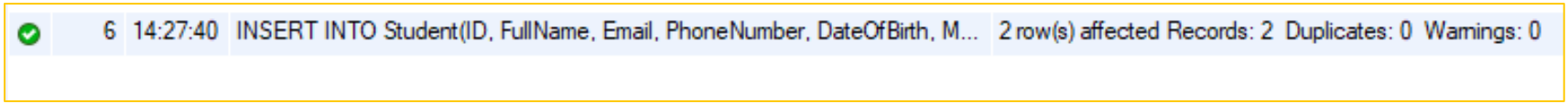


ID	FullName	Email	PhoneNumber	DateOfBirth	Math
5	HUY	HUY@gmail.com	0123456789	2000-10-25	5
NULL	NULL	NULL	NULL	NULL	NULL

Insert multiple data to table

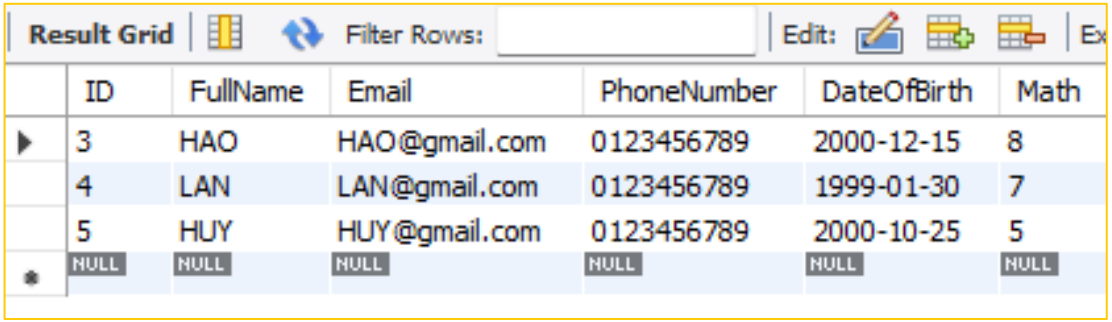
```
INSERT INTO table_name (column1, column2, column3)
VALUES (value1, value2, value3),
       (value4, value5, value6)
```

```
INSERT INTO Student(ID, FullName, Email, PhoneNumber, DateOfBirth, Math)
VALUES(4, 'LAN', 'LAN@gmail.com', '0123456789', '1999-01-30', 7),
      (3, 'HAO', 'HAO@gmail.com', '0123456789', '2000-12-15', 8)
```



Check it

```
SELECT *
FROM Student
```

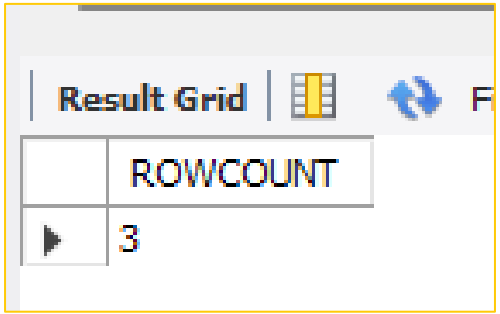


Result Grid | Filter Rows: | Edit:

ID	FullName	Email	PhoneNumber	DateOfBirth	Math
3	HAO	HAO@gmail.com	0123456789	2000-12-15	8
4	LAN	LAN@gmail.com	0123456789	1999-01-30	7
5	HUY	HUY@gmail.com	0123456789	2000-10-25	5
NULL	NULL	NULL	NULL	NULL	NULL

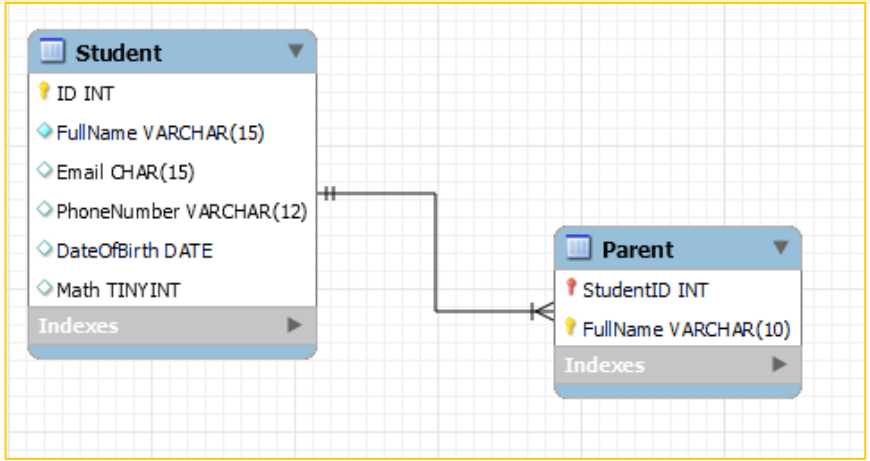
FOUND_ROWS()

```
SELECT * FROM Student
SELECT FOUND_ROWS() AS 'ROWCOUNT'
```



Create child table

```
CREATE TABLE Parent(
        StudentID int,
        FullName varchar(10) NOT NULL,
        CONSTRAINT FK_Parent_Student FOREIGN KEY (StudentID)
                                        REFERENCES Student (ID),
        CONSTRAINT PK_StudentParent PRIMARY KEY (StudentID, FullName)
);
```



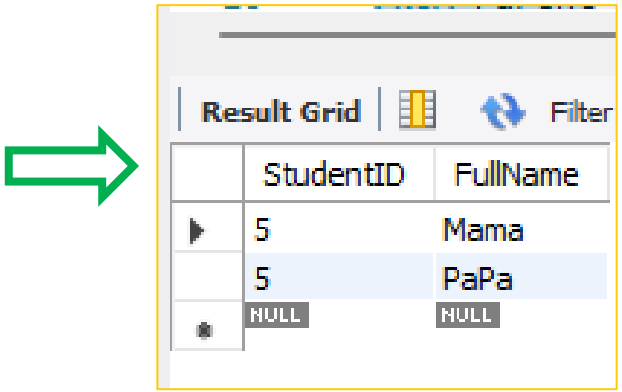
Insert Parent table

```
INSERT INTO Parent(StudentID, FullName)
VALUES(5, 'PaPa'),
      (5, 'Mama')
```

✓	12	14:47:21	INSERT INTO Parent(StudentID, FullName) VALUES(5, 'PaPa'), (5, 'Mama')	2 row(s) affected Records: 2 Duplicates: 0 Warnings: 0

Check data

```
SELECT*
FROM Parent
```



Can we insert into Parent this?

Result Grid | Filter Rows: | Edit:

ID	FullName	Email	PhoneNumber	DateOfBirth	Math
3	HAO	HAO@gmail.com	0123456789	2000-12-15	8
4	LAN	LAN@gmail.com	0123456789	1999-01-30	7
5	HUY	HUY@gmail.com	0123456789	2000-10-25	5
NULL	NULL	NULL	NULL	NULL	NULL

```
INSERT INTO Parent(StudentID, FullName)
VALUES(1, 'Dung')
```

Output

Action Output

#	Time	Action	Message	Duration / Fetch
1	14:51:02	INSERT INTO Parent(StudentID, FullName) VALUES(1, 'Dung')	Error Code: 1452. Cannot add or update a child row: a foreign key constraint ...	0.015 sec

Update syntax

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

```
UPDATE Student
SET FullName = 'HAHA'
```

Writing code with heart

```
UPDATE Student
SET FullName = 'I AM A GOOD BOY'
WHERE ID = 3
```

ID	FullName	Email	PhoneNumber	DateOfBirth	Math
3	I AM A GOOD BOY	HAO@gmail.com	0123456789	2000-12-15	8
4	LAN	LAN@gmail.com	0123456789	1999-01-30	7
5	HUY	HUY@gmail.com	0123456789	2000-10-25	5
NULL	NULL	NULL	NULL	NULL	NULL

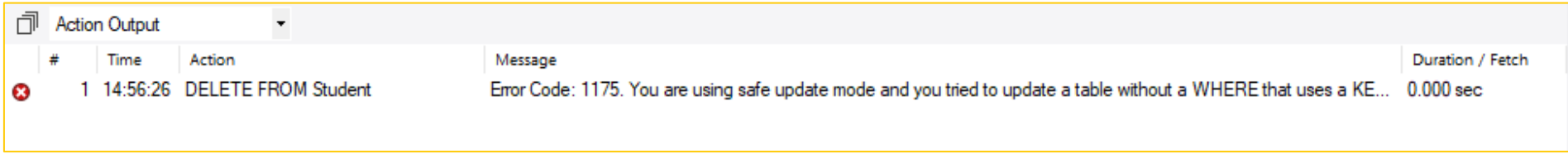
Practice

Modify FullName of Student who has ID = 4

DELETE syntax

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

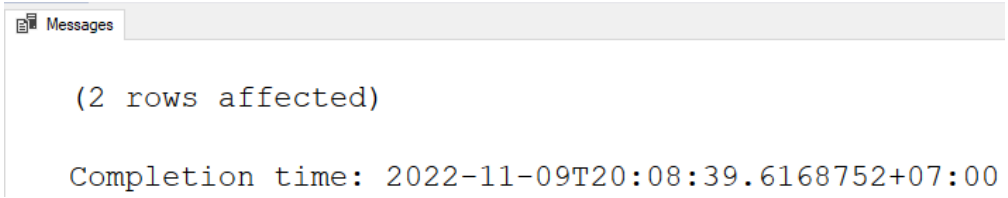
```
DELETE FROM Student
```



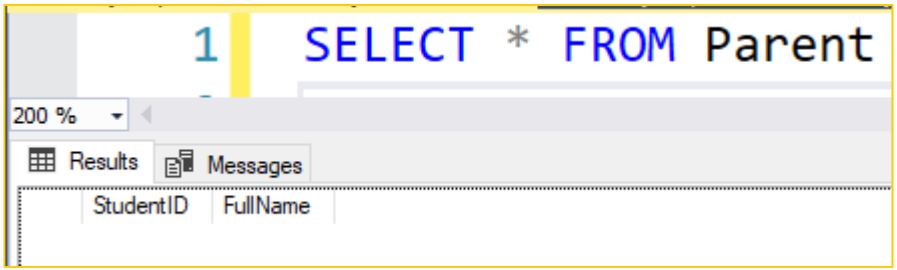
DELETE all references data before delete it self

Delete data & cascade delete

```
DELETE FROM Parent
```



WARNING !



Again... code with heart

```
DELETE FROM Student
WHERE ID = 3
```

ID	FullName	Email	PhoneNumber	DateOfBirth	Math
3	I AM A GOOD BOY	HAO@gmail.com	0123456789	2000-12-15	8
4	LAN	LAN@gmail.com	0123456789	1999-01-30	7
5	HUY	HUY@gmail.com	0123456789	2000-10-25	5
NULL	NULL	NULL	NULL	NULL	NULL

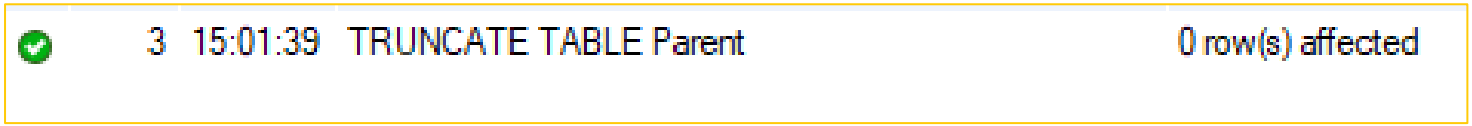
ID	FullName	Email	PhoneNumber	DateOfBirth	Math
4	LAN	LAN@gmail.com	0123456789	1999-01-30	7
5	HUY	HUY@gmail.com	0123456789	2000-10-25	5
NULL	NULL	NULL	NULL	NULL	NULL

Do it by your self

Remove Student who has ID = 4

TRUNCATE

```
TRUNCATE TABLE
Parent
```




```
SELECT *
FROM Parent
```

TRUNCATE vs DELETE

```
TRUNCATE TABLE Parent
```

- DDL
- DROP table & re-create table
- Reset the IDENTITY

```
DELETE FROM Parent
```

- DML
- DELETE all the data in table
- NOT RESET THE IDENTITY

Hi! SELECT

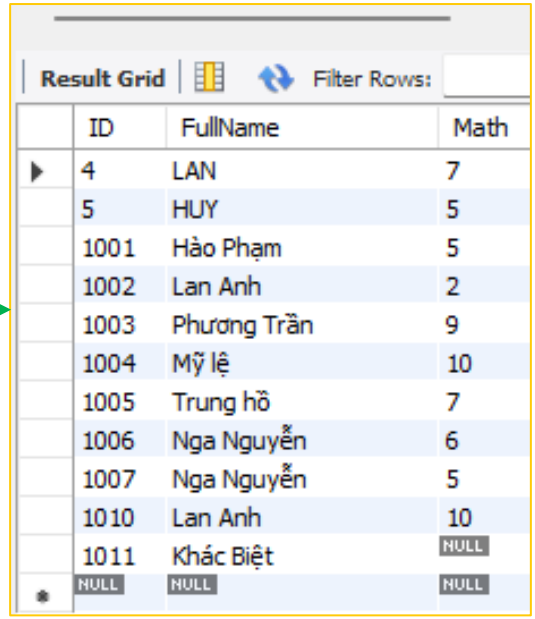
Prepare for practice

```
CREATE TABLE Student(
        ID int PRIMARY KEY,
        FullName varchar(20) NOT NULL,
        Math tinyint CHECK (Math >= 0 AND Math <= 10)
);
```

```
INSERT INTO Student(ID, FullName, Math)
VALUES
        (1007, 'Nga Nguyễn', 5),
        (1006, 'Nga Nguyễn', 6),
        (1005, 'Trung hồ', 7),
        (1001, 'Hào Phạm', 5),
        (1002, 'Lan Anh', 2),
        (1003, 'Phương Trần', 9),
        (1004, 'Mỹ lệ', 10),
        (1010, 'Lan Anh', 10),
        (1011, 'Khác Biệt', NULL);
```

Prepare for practice

```sql
CREATE TABLE Student(
    ID int PRIMARY KEY,
    FullName varchar(20) NOT NULL,
    Math tinyint CHECK (Math >= 0 AND Math <= 10)
);

INSERT INTO Student(ID, FullName, Math)
VALUES(1007, 'Nga Nguyễn', 5),
      (1006, 'Nga Nguyễn', 6),
      (1005, 'Trung hồ', 7),
      (1001, 'Hào Phạm', 5),
      (1002, 'Lan Anh', 2),
      (1003, 'Phương Trần', 9),
      (1004, 'Mỹ lệ', 10),
      (1010, 'Lan Anh', 10),
      (1011, 'Khác Biệt', NULL);

SELECT * FROM Student
```

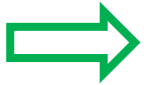


Result Grid | Filter Rows:

ID	FullName	Math
4	LAN	7
5	HUY	5
1001	Hào Phạm	5
1002	Lan Anh	2
1003	Phương Trần	9
1004	Mỹ lệ	10
1005	Trung hồ	7
1006	Nga Nguyễn	6
1007	Nga Nguyễn	5
1010	Lan Anh	10
1011	Khác Biệt	NULL
NULL	NULL	NULL

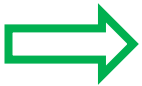
SELECT syntax

```
SELECT ID, FullName
FROM Student
```



	ID	FullName
▶	4	LAN
	5	HUY
	1001	Hào Phạm
	1002	Lan Anh
	1003	Phương Trần
	1004	Mỹ lệ
	1005	Trung hồ
	1006	Nga Nguyễn
	1007	Nga Nguyễn
	1010	Lan Anh
	1011	Khác Biệt
*	NULL	NULL

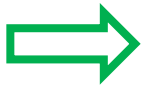
```
SELECT *
FROM Student
```



	ID	FullName	Math
▶	4	LAN	7
	5	HUY	5
	1001	Hào Phạm	5
	1002	Lan Anh	2
	1003	Phương Trần	9
	1004	Mỹ lệ	10
	1005	Trung hồ	7
	1006	Nga Nguyễn	6
	1007	Nga Nguyễn	5
	1010	Lan Anh	10
	1011	Khác Biệt	NULL
*	NULL	NULL	NULL

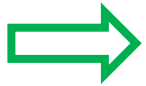
Alias

```
SELECT ID, FullName AS 'CỘT NÈ'
FROM Student
```



	ID	CỘT NÈ
▶	4	LAN
	5	HUY
	1001	Hào Phạm
	1002	Lan Anh
	1003	Phương Trần
	1004	Mỹ lệ
	1005	Trung hồ
	1006	Nga Nguyễn
	1007	Nga Nguyễn
	1010	Lan Anh
	1011	Khác Biệt
*	NULL	NULL

```
SELECT T.FullName FullNameAlias
FROM Student T
```



	FullNameAlias
▶	LAN
	HUY
	Hào Phạm
	Lan Anh
	Phương Trần
	Mỹ lệ
	Trung hồ
	Nga Nguyễn
	Nga Nguyễn
	Lan Anh
	Khác Biệt

ORDER BY

```
SELECT *
FROM Student
ORDER BY FullName
```

```
SELECT *
FROM Student
ORDER BY FullName ASC
```

ID	FullName	Math
1001	Hào Phạm	5
5	HUY	5
1011	Khác Biệt	NULL
4	LAN	7
1002	Lan Anh	2
1010	Lan Anh	10
1004	Mỹ lệ	10
1006	Nga Nguyễn	6
1007	Nga Nguyễn	5
1003	Phương Trần	9
1005	Trung hồ	7
NULL	NULL	NULL

```
SELECT *
FROM Student
ORDER BY FullName DESC
```

ID	FullName	Math
1005	Trung hồ	7
1003	Phương Trần	9
1006	Nga Nguyễn	6
1007	Nga Nguyễn	5
1004	Mỹ lệ	10
1002	Lan Anh	2
1010	Lan Anh	10
4	LAN	7
1011	Khác Biệt	NULL
5	HUY	5
1001	Hào Phạm	5
NULL	NULL	NULL

ORDER BY multiple columns

```
SELECT *
FROM Student
ORDER BY FullName ASC
```

ID	FullName	Math
1001	Hào Phạm	5
5	HUY	5
1011	Khác Biệt	NULL
4	LAN	7
1002	Lan Anh	2
1010	Lan Anh	10
1004	Mỹ lệ	10
1006	Nga Nguyễn	6
1007	Nga Nguyễn	5
1003	Phương Trần	9
1005	Trung hồ	7
NULL	NULL	NULL

```
SELECT *
FROM Student
ORDER BY FullName ASC,
    Math DESC
```

ID	FullName	Math
1001	Hào Phạm	5
5	HUY	5
1011	Khác Biệt	NULL
4	LAN	7
1010	Lan Anh	10
1002	Lan Anh	2
1004	Mỹ lệ	10
1006	Nga Nguyễn	6
1007	Nga Nguyễn	5
1003	Phương Trần	9
1005	Trung hồ	7
NULL	NULL	NULL

SELECT TOP (LIMIT)

```
SELECT *
FROM Student
LIMIT 3;
```

	ID	FullName	Math
▶	4	LAN	7
	5	HUY	5
	1001	Hào Phạm	5
*	NULL	NULL	NULL

```
SELECT *
FROM Student
ORDER BY FullName
LIMIT 3;
```

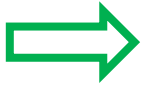
	ID	FullName	Math
▶	1001	Hào Phạm	5
	5	HUY	5
	1011	Khác Biệt	NULL
*	NULL	NULL	NULL

```
SELECT *
FROM Student
ORDER BY FullName DESC
LIMIT 3;
```

	ID	FullName	Math
▶	1005	Trung hồ	7
	1003	Phương Trần	9
	1006	Nga Nguyễn	6
*	NULL	NULL	NULL

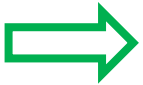
DISTINCT

```
SELECT FullName
FROM Student
```



FullName
LAN
HUY
Hào Phạm
Lan Anh
Phương Trần
Mỹ lệ
Trung hồ
Nga Nguyễn
Nga Nguyễn
Lan Anh
Khác Biệt

```
SELECT DISTINCT FullName
FROM Student
```



FullName
LAN
HUY
Hào Phạm
Lan Anh
Phương Trần
Mỹ lệ
Trung hồ
Nga Nguyễn
Khác Biệt

DISTINCT multiple column

```
SELECT DISTINCT FullName
FROM Student
```

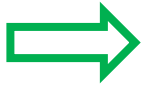
FullName
LAN
HUY
Hào Phạm
Lan Anh
Phương Trần
Mỹ lệ
Trung hồ
Nga Nguyễn
Khác Biệt

```
SELECT DISTINCT ID, FullName
FROM Student
```

ID	FullName
4	LAN
5	HUY
1001	Hào Phạm
1002	Lan Anh
1003	Phương Trần
1004	Mỹ lệ
1005	Trung hồ
1006	Nga Nguyễn
1007	Nga Nguyễn
1010	Lan Anh
1011	Khác Biệt
NULL	NULL

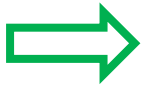
WHERE

```
SELECT FullName, Math
FROM Student
```



FullName	Math
LAN	7
HUY	5
Hào Phạm	5
Lan Anh	2
Phương Trần	9
Mỹ lệ	10
Trung hồ	7
Nga Nguyễn	6
Nga Nguyễn	5
Lan Anh	10
Khác Biệt	NULL

```
SELECT FullName, Math
FROM Student
WHERE Math > 5
```

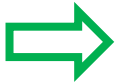


FullName	Math
LAN	7
Phương Trần	9
Mỹ lệ	10
Trung hồ	7
Nga Nguyễn	6
Lan Anh	10

VIEW

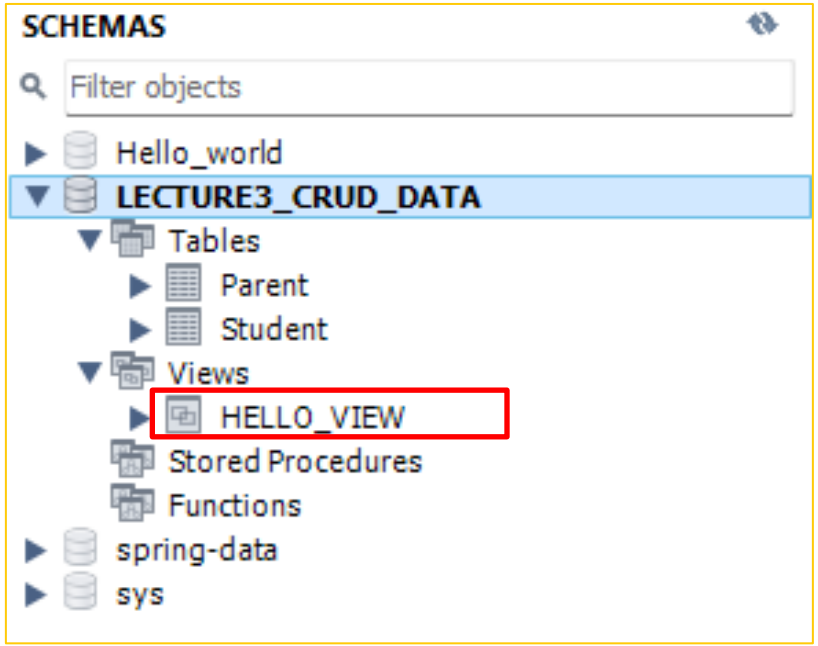
```
CREATE VIEW
HELLO_VIEW AS
SELECT FullName, Math
FROM Student
WHERE Math > 5
```

```
SELECT *
FROM HELLO_VIEW
```



	FullName	Math
▶	LAN	7
	Phương Trần	9
	Mỹ lệ	10
	Trung hồ	7
	Nga Nguyễn	6
	Lan Anh	10

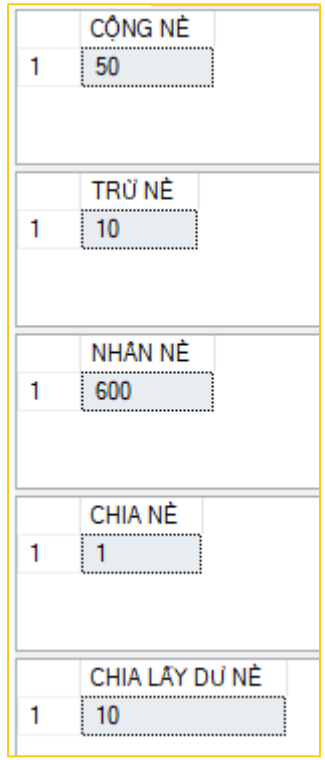
View location



SQL Operators

Arithmetic operators

```
SELECT 30 + 20 AS 'CỘNG NÈ';
SELECT 30 - 20 AS 'TRỪ NÈ';
SELECT 30 * 20 AS 'NHÂN NÈ';
SELECT 30 / 20 AS 'CHIA NÈ';
SELECT 30 % 20 AS 'CHIA LẤY DƯ NÈ';
```



	CỘNG NÈ
1	50

	TRỪ NÈ
1	10

	NHÂN NÈ
1	600

	CHIA NÈ
1	1

	CHIA LẤY DƯ NÈ
1	10

Comparison operators

Operator	Description
=	Equal to
>	Greater than
<	Less than
>=	Greater than or equal to
<=	Less than or equal to
<> or !=	Not equal to

Try it

```
SELECT FullName, Math
FROM Student
WHERE Math > 5
```

FullName	Math
LAN	7
Phương Trần	9
Mỹ lệ	10
Trung hồ	7
Nga Nguyễn	6
Lan Anh	10

ID	FullName	Math
4	LAN	7
5	HUY	5
1001	Hào Phạm	5
1002	Lan Anh	2
1003	Phương Trần	9
1004	Mỹ lệ	10
1005	Trung hồ	7
1006	Nga Nguyễn	6
1007	Nga Nguyễn	5
1010	Lan Anh	10
1011	Khác Biệt	NULL
NULL	NULL	NULL

```
SELECT ID, FullName, Math
FROM Student
WHERE FullName = 'Lan Anh'
```

ID	FullName	Math
1002	Lan Anh	2
1010	Lan Anh	10
NULL	NULL	NULL

Logical operators

AND

```
SELECT ID, FullName
FROM Student
WHERE FullName = 'Lan Anh' AND ID = 1010
```

	ID	FullName
▶	1010	Lan Anh
*	NULL	NULL

	ID	FullName	Math
▶	4	LAN	7
	5	HUY	5
	1001	Hào Phạm	5
	1002	Lan Anh	2
	1003	Phương Trần	9
	1004	Mỹ lệ	10
	1005	Trung hồ	7
	1006	Nga Nguyễn	6
	1007	Nga Nguyễn	5
	1010	Lan Anh	10
	1011	Khác Biệt	NULL
*	NULL	NULL	NULL

OR

```
SELECT ID, FullName
FROM Student
WHERE FullName = 'Lan Anh' OR ID = 1005
```

ID	FullName
1002	Lan Anh
1005	Trung hồ
1010	Lan Anh
NULL	NULL

ID	FullName	Math
4	LAN	7
5	HUY	5
1001	Hào Phạm	5
1002	Lan Anh	2
1003	Phương Trần	9
1004	Mỹ lệ	10
1005	Trung hồ	7
1006	Nga Nguyễn	6
1007	Nga Nguyễn	5
1010	Lan Anh	10
1011	Khác Biệt	NULL
NULL	NULL	NULL

BETWEEN - AND

```
SELECT ID, FullName
FROM Student
WHERE ID BETWEEN 1003 AND 1005
```

ID	FullName
1003	Phương Trần
1004	Mỹ lệ
1005	Trung hỗ
NULL	NULL

ID	FullName	Math
4	LAN	7
5	HUY	5
1001	Hào Phạm	5
1002	Lan Anh	2
1003	Phương Trần	9
1004	Mỹ lệ	10
1005	Trung hỗ	7
1006	Nga Nguyễn	6
1007	Nga Nguyễn	5
1010	Lan Anh	10
1011	Khác Biệt	NULL
NULL	NULL	NULL

LIKE

```
SELECT ID, FullName
FROM Student
WHERE FullName LIKE '%U%';
```

ID	FullName
5	HUY
1003	Phương Trần
1005	Trung hồ
1006	Nga Nguyễn
1007	Nga Nguyễn
NULL	NULL

ID	FullName	Math
4	LAN	7
5	HUY	5
1001	Hào Phạm	5
1002	Lan Anh	2
1003	Phương Trần	9
1004	Mỹ lệ	10
1005	Trung hồ	7
1006	Nga Nguyễn	6
1007	Nga Nguyễn	5
1010	Lan Anh	10
1011	Khác Biệt	NULL
NULL	NULL	NULL

SQL Wildcards

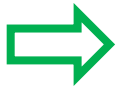
LIKE Operator	Description
WHERE Name LIKE 'a%'	Finds any values that start with "a"
WHERE Name LIKE '%a'	Finds any values that end with "a"
WHERE Name LIKE '%or%'	Finds any values that have "or" in any position
WHERE Name LIKE '_r%'	Finds any values that have "r" in the second position
WHERE Name LIKE 'a_%'	Finds any values that start with "a" and are at least 2 characters in length
WHERE Name LIKE 'a__%'	Finds any values that start with "a" and are at least 3 characters in length
WHERE Name LIKE 'a%o'	Finds any values that start with "a" and ends with "o"

NOT

```
SELECT ID, FullName
FROM Student
WHERE ID NOT BETWEEN 1003 AND 1005
```

ID	FullName
4	LAN
5	HUY
1001	Hào Phạm
1002	Lan Anh
1006	Nga Nguyễn
1007	Nga Nguyễn
1010	Lan Anh
1011	Khác Biệt
NULL	NULL

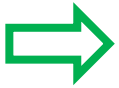
```
SELECT ID, FullName, Math
FROM Student
WHERE NOT Math > 5
```



ID	FullName	Math
5	HUY	5
1001	Hào Phạm	5
1002	Lan Anh	2
1007	Nga Nguyễn	5
NULL	NULL	NULL

IS NULL

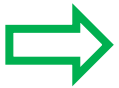
```
SELECT ID, FullName, Math
FROM Student
WHERE Math IS NULL
```



Results | Messages

	ID	FullName	Math
1	1011	Khác Biệt	NULL

```
SELECT ID, FullName, Math
FROM Student
WHERE Math IS NOT NULL
```



Results | Messages

	ID	FullName	Math
1	1001	Hào Phạm	5
2	1002	Lan Anh	2
3	1003	Phương Trần	9
4	1004	Mỹ lệ	10
5	1005	Trung hồ	7
6	1006	Nga Nguyễn	6
7	1007	Nga Nguy?n	5
8	1010	Lan Anh	10

Results | Messages

	ID	FullName	Math
1	1001	Hào Phạm	5
2	1002	Lan Anh	2
3	1003	Phương Trần	9
4	1004	Mỹ lệ	10
5	1005	Trung hồ	7
6	1006	Nga Nguyễn	6
7	1007	Nga Nguy?n	5
8	1010	Lan Anh	10
9	1011	Khác Biệt	NULL

Big picture

```
SELECT column_data
FROM source
    JOIN source2
WHERE condition
GROUP BY
HAVING condition
ORDER BY sort [ASC|DESC]
```

Thank you!

*Any **questions** ?*

Extra Resources

Name	Link
became SQL god?	https://www.w3schools.com/sql/default.asp