SQL

Xbi Juli | 2020

SQL?

- **Structured Query Language**
- SQL is Structured Query Language, which is a computer language for **storing**, **manipulating** and **retrieving** data stored in relational database.
- SQL is the standard language for **Relation** Database System. All relational database management systems like "MySQL, MS Access, Oracle, Sybase, Informix, postgres and SQL Server" use SQL as standard database language.

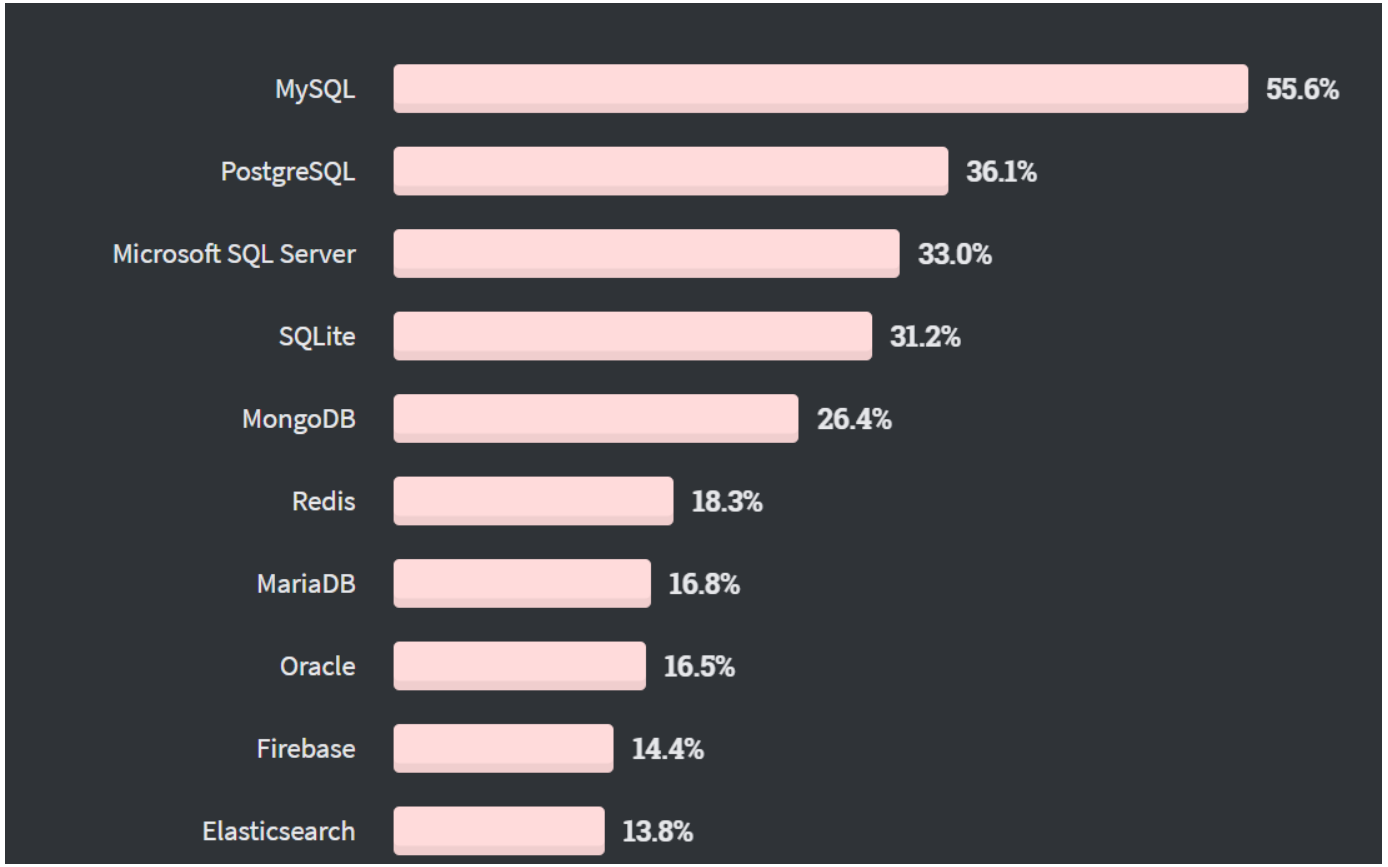
SQL Commands

- **DDL - Data Definition Language**
- **DML - Data Manipulation Language**
- **DCL - Data Control Language**
- **DQL - Data Query Language**

Introduction to MySQL

Databases: Explosion

Stack Overflow Developer Survey 2020 Most Popular Databases



MySQL Powers eCommerce

MySQL Architecture

```
Create table tutorials_tbl(
tutorial_id INT NOT NULL
AUTO_INCREMENT,  tutorial_title
VARCHAR(100) NOT NULL,
tutorial_author VARCHAR(40) NOT
NULL,  submission_date DATE,
PRIMARY KEY ( tutorial_id ) );
```

INT, VARCHAR, DATE are data types

DDL-CREATE

DDL - ALTER

Syntax:

```
ALTER TABLE table_name action1[,action2,…]
```

Using MySQL ALTER TABLE statement to add a new column into a table

```
ALTER TABLE table_name
ADD COLUMN column_name DECIMAL(2,1) NULL
AFTER  column_name;
```

DDL - DROP

Syntax:

Here is generic SQL syntax to drop a MySQL table:

```
DROP TABLE table_name;
```

DML - INSERT

Syntax:

Here is generic SQL syntax of INSERT INTO command to insert data into MySQL table:

```
INSERT INTO table_name ( field1, field2,...fieldN )
VALUES ( value1, value2,...valueN );
```

To insert string data types, it is required to keep all the values in double or single quote, for example:- **"value"**.

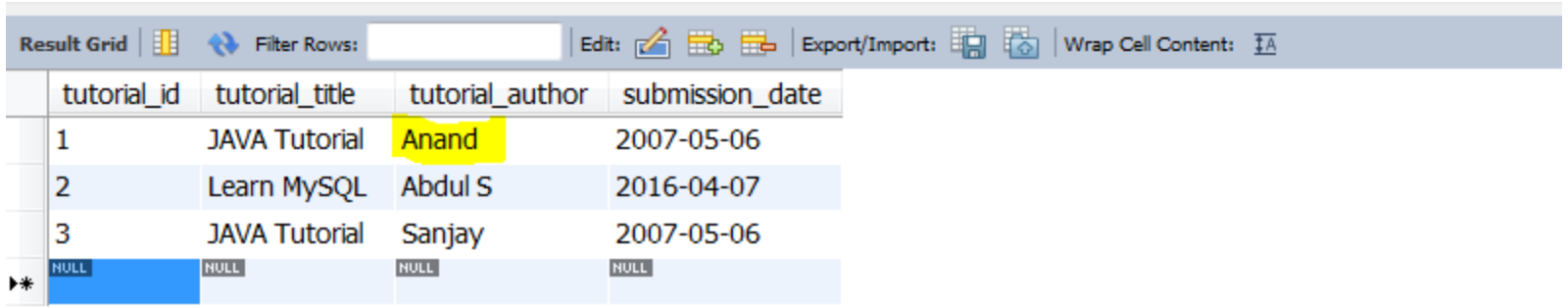
DML– UPDATE

Syntax:

```
UPDATE table_name
SET
    column_name1 = expr1,
    column_name2 = expr2,
    ...
WHERE
    condition;
```

DML – UPDATE

```
update tutorials_tbl
set tutorial_author = 'Anand'
where tutorial_id = 1;
```



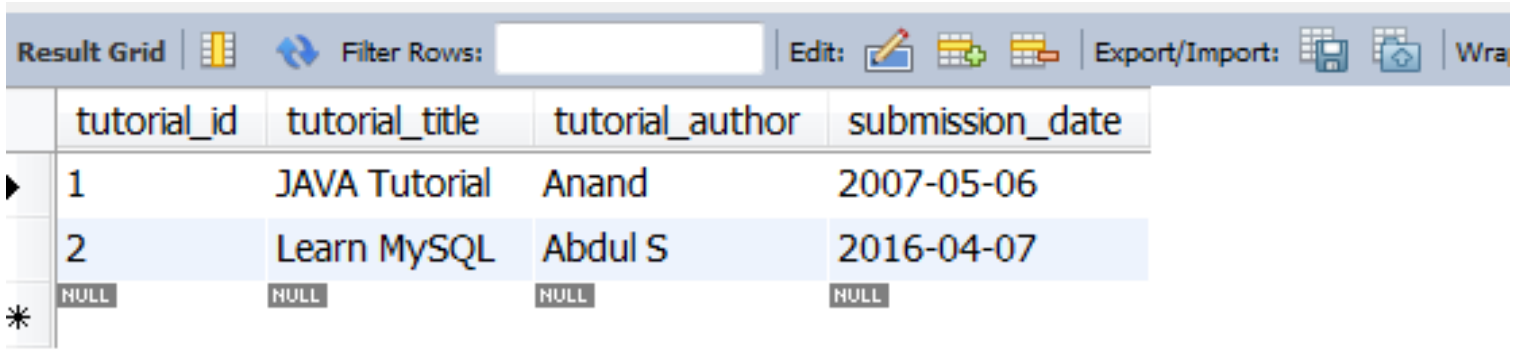
tutorial_id	tutorial_title	tutorial_author	submission_date
1	JAVA Tutorial	Anand	2007-05-06
2	Learn MySQL	Abdul S	2016-04-07
3	JAVA Tutorial	Sanjay	2007-05-06
NULL	NULL	NULL	NULL

DELETE

Syntax:

```
DELETE FROM table
[WHERE conditions]
```

```
Delete from tutorials_tbl
where tutorial_id = 3;
```



DML – SELECT

Syntax:

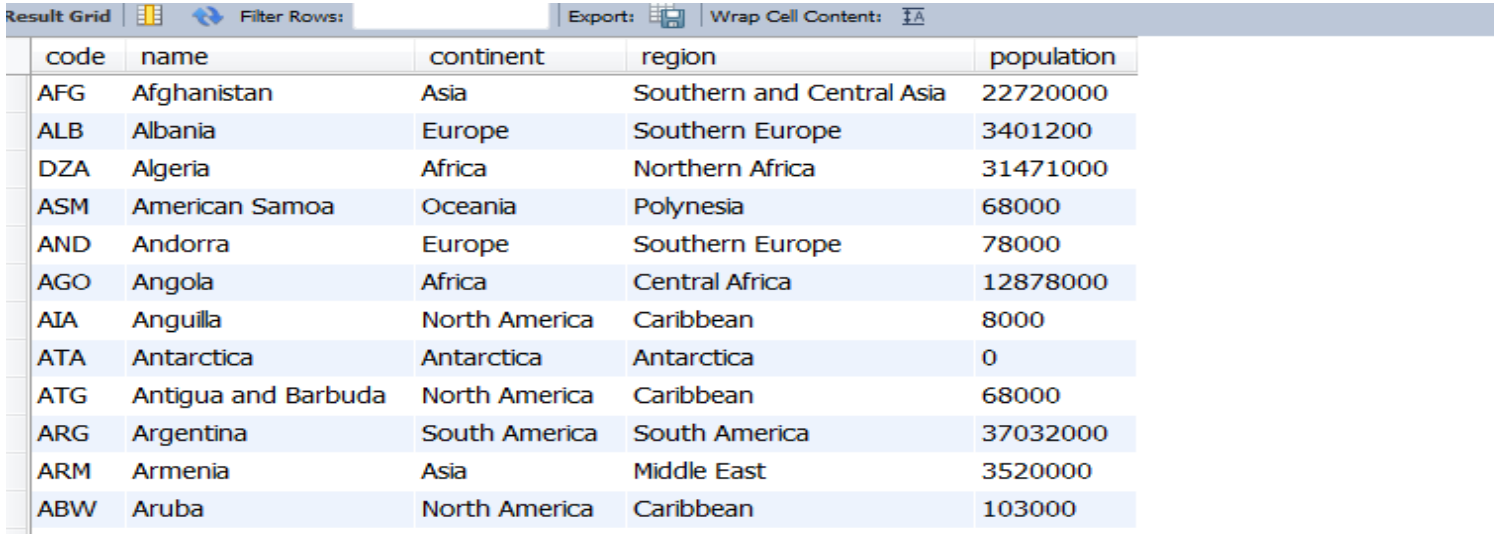
Here is generic SQL syntax of SELECT command to fetch data from MySQL

table:

```
SELECT field1, field2,...fieldN
FROM table_name1, table_name2...
[WHERE Clause] [OFFSET M ][LIMIT N]
```

DML Statements – SELECT with ORDER BY

```
select code, name, continent, region, population
from country
order by name asc;
```

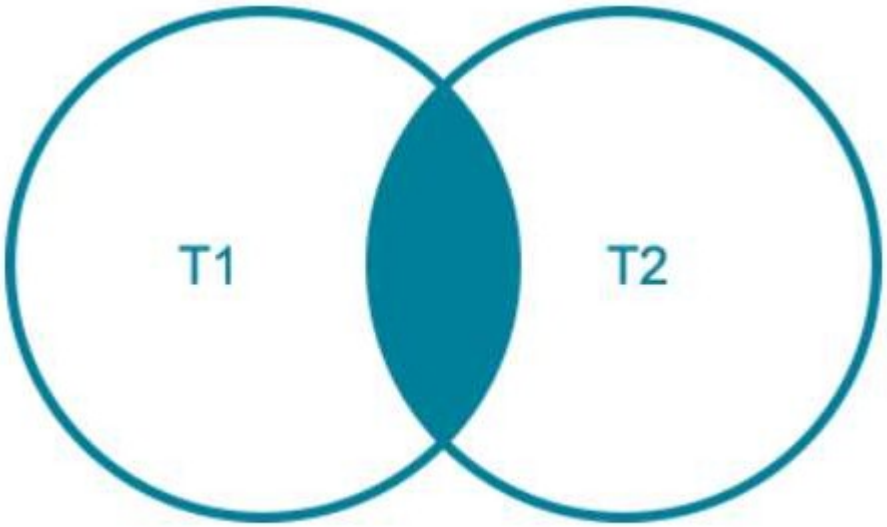


Result Grid | Filter Rows: | Export: | Wrap Cell Content:

code	name	continent	region	population
AFG	Afghanistan	Asia	Southern and Central Asia	22720000
ALB	Albania	Europe	Southern Europe	3401200
DZA	Algeria	Africa	Northern Africa	31471000
ASM	American Samoa	Oceania	Polynesia	68000
AND	Andorra	Europe	Southern Europe	78000
AGO	Angola	Africa	Central Africa	12878000
AIA	Anguilla	North America	Caribbean	8000
ATA	Antarctica	Antarctica	Antarctica	0
ATG	Antigua and Barbuda	North America	Caribbean	68000
ARG	Argentina	South America	South America	37032000
ARM	Armenia	Asia	Middle East	3520000
ABW	Aruba	North America	Caribbean	103000

INNER JOIN

The following Venn diagram illustrates how the MySQL INNER JOIN clause works. The rows in the result set must appear in both tables: T1 and T2.



MySQL INNER JOIN Venn Diagram

```
SELECT column_name(s)
FROM table1
INNER JOIN table2
ON table1.column_name = table2.column_name;
```

DML Statements – SELECT with GROUP BY

```
use world;
```

Execute the highlighted **Select** command to see the impact of aggregate function

```
120
121   -- group by
122 •  select continent, region, sum(population) from country
123   group by continent, region;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

continent	region	sum(population)
Asia	Eastern Asia	1507328000
Asia	Middle East	188380700
Asia	Southeast Asia	518541000
Asia	Southern and Central Asia	1490776000
Europe	Baltic Countries	7561900
Europe	British Islands	63398500
Europe	Eastern Europe	307026000
Europe	Nordic Countries	24166400

DML Statements – SELECT with GROUP BY and HAVING

```
SELECT
    ordernumber,
    SUM(quantityOrdered) AS itemsCount,
    SUM(priceeach) AS total
FROM
    orderdetails
GROUP BY ordernumber
HAVING total > 1000;
```

DML Statements – SELECT with GROUP BY and HAVING clause

Execute the below **SELECT** command in world database.

```
127 • select continent, region, sum(population) as population from country where continent = 'Asia'
128   group by continent, region having sum(population) = 1507328000;
```

esult Grid | Filter Rows: | Export: | Wrap Cell Content:

continent	region	population
Asia	Eastern Asia	1507328000

DML Statements – SELECT with GROUP BY and HAVING clause

Execute the below **SELECT** command in world database.

```
130 • select continent, region, sum(population) as population from country
131   group by continent, region having sum(population) > 1000000;
132
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

continent	region	population
Asia	Eastern Asia	1507328000
Asia	Middle East	188380700
Asia	Southeast Asia	518541000
Asia	Southern and Central Asia	1490776000
Europe	Baltic Countries	7561900
Europe	British Islands	63398500
Europe	Eastern Europe	307026000
Europe	Nordic Countries	24166400


..SQL opgave

Mysql opret skoledatabase

PHP PDO

Øvelse grundstof

AF.opgave

Skoledatabase