KEY WORDS AND FUNCTION	Examples	Description
ADD	ALTER TABLE customers ADD COLUMN email VACHAR(255)	Adds a column in an existing table
ADD CONSTRAINT	ALTER TABLE persons ADD CONSTRAINT fk_p_d (department_id) REFERENCES departments(id);	Adds a constraint after a table is already created
ALTER TABLE	ALTER TABLE employees ADD COLUMN email VARCHAR(255);	Adds, deletes, or modifies columns in a table
AND	SELECT * FROM users WHERE name = 'Pesho' AND town = 'Sofia';	Only includes rows where both conditions is true
AS	SELECT f_name AS 'First Name' FROM users	Renames a column or table with an alias
BETWEEN	SELECT * FROM products WHERE price BETWEEN 10 AND 20;	Selects values within a given range
CREATE DATABASE	CREATE DATABASE users;	Creates a new database
CREATE TABLE	CREATE TABLE users (id INT PRIMARY KEY, name VARCHAR(45));	Creates a new table in the database
CREATE VIEW	CREATE VIEW view_users_ids AS SELECT id, username FROM users;	Creates a view based on the result set of a SELECT statement
DELETE	DELETE FROM users WHERE salary > 100;	Deletes rows from a table
DESC	SELECT * FROM customers ORDER BY name DESC;	
DISTINCT	SELECT DISTINCT country FROM users;	Selects only distinct (different) values
DROP	ALTER TABLE users DROP COLUMN salary;	Deletes a column, constraint, database, index, table, or view
GROUP BY	SELECT COUNT(*) FROM users GROUP BY department_id;	Groups the result set (used with aggregate functions: COUNT, MAX, MIN, SUM, AVG)
HAVING	SELECT COUNT(*) FROM users GROUP BY department_id HAVING COUNT(*) > 5;	Used instead of WHERE with aggregate functions
IN	SELECT * FROM countries WHERE name IN ('EN', 'BG', 'UK');	Allows you to specify multiple values in a WHERE clause

INNER JOIN	SELECT u.name, d.name FROM users AS u INNER JOIN departments AS d ON u.department_id = d.id;	Returns rows that have matching values in both tables
INSERT INTO	INSERT INTO users(`id`, `name`) VALUES (1, 'Pesh');	Inserts new rows in a table
IS NULL	SELECT name FROM users WHERE address IS NULL;	Tests for empty values
IS NOT NULL	SELECT name FROM users WHERE address IS NOT NULL;	Tests for non-empty values
LEFT JOIN	SELECT u.name, d.name FROM users AS u LEFT JOIN departments AS d ON u.department_id = d.id;	Returns all rows from the left table, and the matching rows from the right table
RIGHT JOIN	SELECT u.name, d.name FROM users AS u RIGHT JOIN departments AS d ON u.department_id = d.id;	Returns all rows from the right table, and the matching rows from the left table
LIKE	SELECT name FROM users WHERE name LIKE 'B%';	Searches for a specified pattern in a column
LIMIT	SELECT name FROM users ORDER BY salary LIMIT 5;	Specifies the number of records to return in the result set
OR	SELECT name FROM users WHERE town = 'Sofia' OR town = 'Plovdiv';	Includes rows where either condition is true
ORDER BY	SELECT * FROM users ORDER BY name;	Sorts the result set in ascending or descending order
TRUNCATE TABLE	TRUNCATE TABLE categories;	Deletes the data inside a table, but not the table itself
UNIQUE		A constraint that ensures that all values in a column are unique
UPDATE	UPDATE users SET salary = salary * 1.2 WHERE department_id = 6;	Updates existing rows in a table
VIEW	CREATE VIEW view_all_emp_names AS SELECT name FROM users;	Creates, updates, or deletes a view
WHERE	SELECT users WHERE name = 'Pesho'	Filters a result set to include only records that fulfill a specified condition

String Functions		
CHAR_LENGTH()	SELECT CHAR_LENGTH('String')	Returns the length of a string (in characters)
CONCAT()	SELECT CONCAT(f_name, ' ', l_name) AS 'Full name'	Adds two or more expressions together
CONCAT_WS()	ECT CONCAT_WS("-", name, department) AS 'Name and departme	Adds two or more expressions together with a separator
FORMAT()	SELECT FORMAT(6000.5634, 2);	Formats a number to a format like "#,###,###.##", rounded to a specified number of decimal places
INSERT()	SELECT INSERT("MySTRING", 1, 3, "test");	Inserts a string within a string at the specified position and for a certain number of characters
LEFT()/RIGHT()	SELECT LEFT('Pesho', 1)	Extracts a number of characters from a string (starting from left)
LENGTH()	SELECT LENGTH('PESHO')	Returns the length of a string (in bytes)
LOCATE()	SELECT LOCATE("o", "Pesho");	Returns the position of the first occurrence of a substring in a string
LOWER()/UPPER()	LOWER('MyNAME')	Converts a string to lower-case
LTRIM()/RTRIM()	LTRIM(' some text')	Removes leading spaces from a string
POSITION()	POSITION('b', 'abcd')	Returns the position of the first occurrence of a substring in a string
REPEAT()	REPEAT('some str', 3)	Repeats a string as many times as specified
REVERSE()	REVERSE('some str')	Reverses a string and returns the result
SUBSTR()	SUBSTR('Some str', 2, 4)	Extracts a substring from a string (starting at any position)
TRIM()	TRIM('Some str')	Removes leading and trailing spaces from a string
Math Functions		
ABS()	SELECT ABS(-243.5);	Returns the absolute value of a number
AVG()	SELECT AVG(salary) FROM users	Returns the average value of an expression
CEIL()/CEILING()	SELECT CEIL(1.2)	Returns the smallest integer value that is >= to a number
COUNT()	SELECT COUNT(id) FROM products	Returns the number of records returned by a select query
FLOOR()	SELECT FLOOR(1.2)	Returns the largest integer value that is <= to a number
MIN()/MAX()	SELECT MAX(salary) FROM users;	Returns the maximum value in a set of values
PI()	SELECT PI()	Returns the value of PI
POW()/POWER()	SELECT POW(4,2)	Returns the value of a number raised to the power of another number
ROUND()	SELECT ROUND(1.3)	Rounds a number to a specified number of decimal places
SUM()	SELECT SUM(salary, bonus)	Calculates the sum of a set of values

Date Functions	Date Functions	Date Functions
ADDDATE()	SELECT ADDDATE("2017-06-15", INTERVAL 10 DAY);	Adds a time/date interval to a date and then returns the date
ADDTIME()	SELECT ADDTIME("2017-06-15 09:34:21", "2");	Adds a time interval to a time/datetime and then returns the time/datetime
DATE()	SELECT DATE("2017-06-15");	Extracts the date part from a datetime expression
DATEDIFF()	SELECT DATEDIFF("2017-06-25", "2017-06-15");	Returns the number of days between two date values
DATE_FORMAT()	DATE_FORMAT(date, format)	Formats a date
DAY()	SELECT DAY("2017-06-15");	Returns the day of the month for a given dates
DAYOFWEEK()	SELECT DAYOFWEEK("2017-06-15");	Returns the weekday index for a given date
EXTRACT()	SELECT EXTRACT(MONTH FROM "2017-06-15");	Extracts a part from a given date
MINUTE()	SELECT MINUTE("2017-06-20 09:34:00");	Returns the minute part of a time/datetime
MONTH()	SELECT MONTH("2017-06-15");	Returns the month part for a given date
NOW()	SELECT NOW()	
TIMESTAMP()	TIMESTAMPDIFF(unit,datetime_expr1,datetime_expr2);	
WEEK()	SELECT WEEK("2017-06-15");	Returns the week number for a given date
YEAR()	SELECT YEAR("2017-06-15");	Returns the year part for a given dates
STR_TO_DATE()	("August 10 2017", "%M %d %Y");SELECT STR_TO_DATE("24 08 2017", "%d %m %Y");	
IF()	IF(salary > 5000, 'OK' , 'NOT OK')	
IFNULL()	IFNULL(`name`, '')	