



Prifysgol Cymru
Y Drindod Dewi Sant
University of Wales
Trinity Saint David

200
1822 - 2022

Introduction to SQL

**SQL Commands (DDL, DML, DCL, TCL, DQL):
Types, Syntax, and Examples**

*Man Geni Addysg Uwch yng Nghymru
Where Higher Education Began in Wales*

www.uwtsd.ac.uk



Prifysgol Cymru
Y Drindod Dewi Sant
University of Wales
Trinity Saint David

200
1822 - 2022

Review of Normalization



<https://forms.office.com/e/AR9Zsh45CV>

Man Geni Addysg Uwch yng Nghymru
Where Higher Education Began in Wales

www.uwtsd.ac.uk

Objectives of SQL

Ideally, a database language should allow a user to:

- create the database and relation structures;
- perform basic data management tasks, such as the insertion, modification, and deletion of data from the relations;
- perform both simple and complex queries.

Our expectations on any database language:

- minimal user effort
- command structure and syntax must be relatively easy to learn.
- Language must be portable; able to use the same command structure and syntax when moved from one DBMS to another.

SQL is intended to satisfy these requirements.

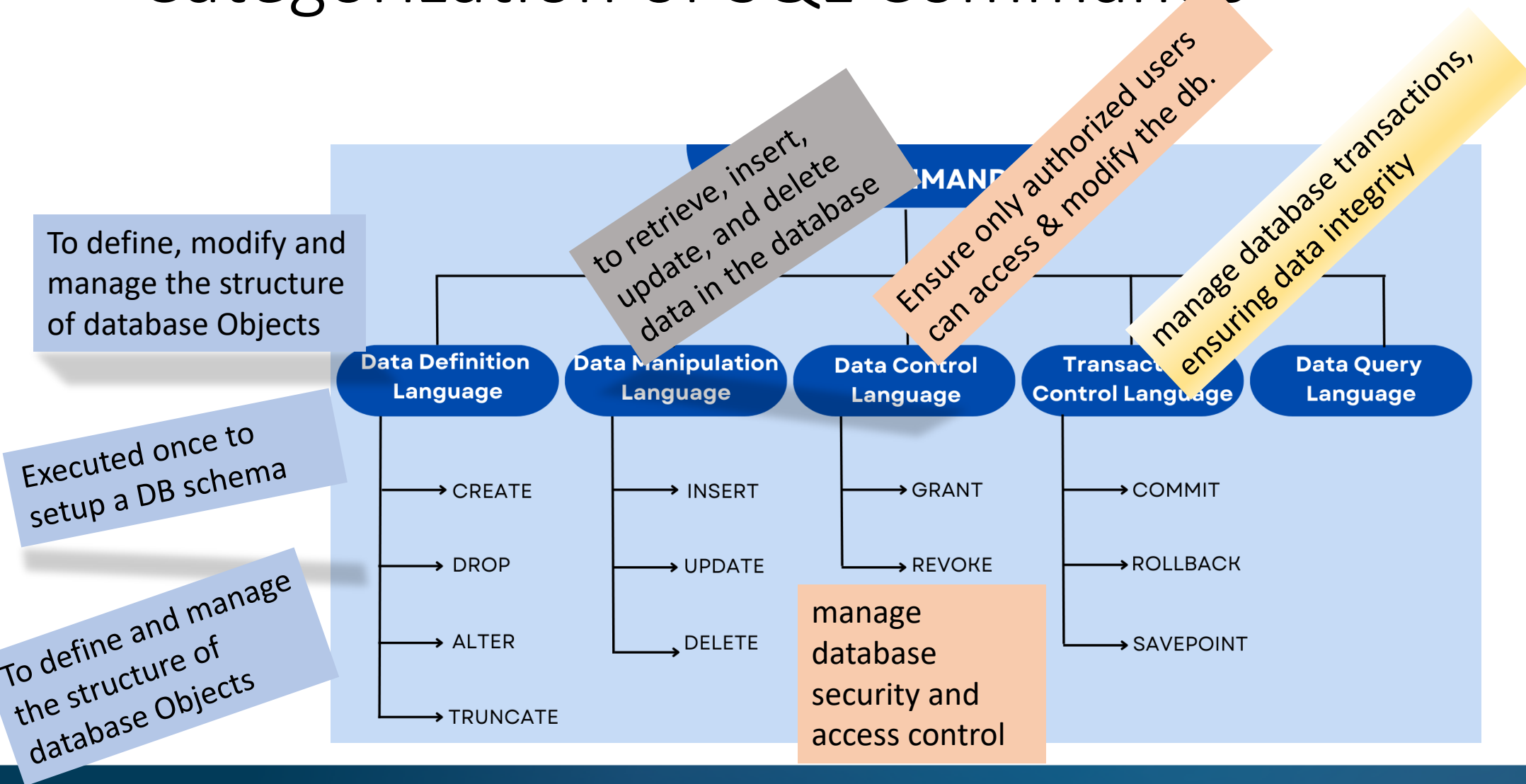
SQL standard has two major components:

- Data Definition Language (DDL) for defining(create) the database structure and controlling access to the data;
- Data Manipulation Language (DML) for retrieving and updating data (populate and query the tables).

Writing SQL Commands

- An SQL statement consists of **reserved words** and **user-defined words**.
- SQL require the use of a statement terminator to mark the end of each SQL statement (usually the semicolon “;” is used)
- Most components of an SQL statement are case-insensitive.
- important exception: literal character data must be typed exactly as it appears in the database.
- For example, if we store a person’s surname as “SMITH” and then search for it using the string “Smith,” the row will not be found.

Categorization of SQL Commands



Common DDL Commands

SQL CREATE DATABASE

CREATE DATABASE testDB;

DROP DATABASE

- **DROP DATABASE** testDB;

DDL Commands in SQL with Examples

SQL Command	Code Snippet	Output
CREATE TABLE	CREATE TABLE Employees (EmployeeID INT PRIMARY KEY, FirstName VARCHAR(50), LastName VARCHAR(50), Department VARCHAR(50));	New “Employees” table created with specified columns.
ALTER TABLE	ALTER TABLE Employees ADD Email VARCHAR(100);	“Email” column added to the “Employees” table.
DROP TABLE	DROP TABLE Employees;	“Employees” table and its data deleted.

```
DROP TABLE IF EXISTS branch;
```

```
CREATE TABLE branch  
(branchNo char(5) PRIMARY KEY,  
street varchar(35),  
city varchar(10),  
postcode varchar(10)  
);
```

DML Commands in SQL with Examples

SQL Command	Code Snippet	Output
SELECT	SELECT FirstName, LastName FROM Employees WHERE Department = 'Sales';	Retrieves the first and last names of employees in the “Sales” department.
INSERT	INSERT INTO Employees (FirstName, LastName, Department) VALUES ('John', 'Doe', 'HR');	New employee record added to the “Employees” table.
UPDATE	UPDATE Employees SET Salary = Salary * 1.1 WHERE Department = 'Engineering';	Salary of employees in the “Engineering” department increased by 10%.
DELETE	DELETE FROM Employees WHERE Department = 'Finance';	Employees in the “Finance” department deleted.

1. Without specifying column name

If you want to specify all column values, you can specify or ignore the column values.

```
INSERT INTO EMPLOYEE VALUES (6, 'Marry', 'Canada', 600000, 48);
```

2. By specifying column name

```
INSERT INTO EMPLOYEE (EMP_ID, EMP_NAME, AGE) VALUES (7, 'Jack', 40);
```

Updating single record

Update the column EMP_NAME and set the value to 'Emma' in the row where SALARY is 500000.

```
UPDATE EMPLOYEE SET EMP_NAME = 'Emma' WHERE SALARY = 500000;
```

Updating multiple records

```
UPDATE EMPLOYEE
```

```
SET EMP_NAME = 'Kevin', City = 'Boston'
```

```
WHERE EMP_ID = 5;
```

Without use of WHERE clause: If you want to update all row from a table, then you don't need to use the WHERE clause. In the EMPLOYEE table, update the column EMP_NAME as 'Harry'.

```
UPDATE EMPLOYEE
```

```
SET EMP_NAME = 'Harry';
```

Activity

- CREATE DATABASE called dream home
- Create the following tables :
 - Branch (branchNo, street, city, postcode)
 - Staff (staffNo, fName, IName, position, sex, DOB, salary, branchNo)
 - PropertyForRent (propertyNo, street, city, postcode, type, rooms, rent, ownerNo, staffNo,branchNo)
 - Client (clientNo, fName, IName, telNo, prefType, maxRent, eMail)
 - PrivateOwner (ownerNo, fName, IName, address, telNo, eMail, password)
 - Viewing (clientNo, propertyNo, viewDate, comment)

Literals

- Literals are constants that are used in SQL statements.
- All nonnumeric data values must be enclosed in single quotes;
- all numeric data values must not be enclosed in single quotes.

- **SELECT** [DISTINCT | ALL] { * | [columnExpression [AS newName]] [, . . .] }
- **FROM** TableName [alias] [, . . .]
- [WHERE condition]
- [GROUP BY columnList]
- [HAVING condition]
- [ORDER BY columnList]

Mandatory

Rest all are optional

Branch

branchNo	street	city	postcode
B005	22 Deer Rd	London	SW1 4EH
B007	16 Argyll St	Aberdeen	AB2 3SU
B003	163 Main St	Glasgow	G11 9QX
B004	32 Manse Rd	Bristol	BS99 1NZ
B002	56 Clover Dr	London	NW10 6EU

Staff

staffNo	fName	lName	position	sex	DOB	salary	branchNo
SL21	John	White	Manager	M	1-Oct-45	30000	B005
SG37	Ann	Beech	Assistant	F	10-Nov-60	12000	B003
SG14	David	Ford	Supervisor	M	24-Mar-58	18000	B003
SA9	Mary	Howe	Assistant	F	19-Feb-70	9000	B007
SG5	Susan	Brand	Manager	F	3-Jun-40	24000	B003
SL41	Julie	Lee	Assistant	F	13-Jun-65	9000	B005

PropertyForRent

propertyNo	street	city	postcode	type	rooms	rent	ownerNo	staffNo	branchNo
PA14	16 Holhead	Aberdeen	AB7 5SU	House	6	650	CO46	SA9	B007
PL94	6 Argyll St	London	NW2	Flat	4	400	CO87	SL41	B005
PG4	6 Lawrence St	Glasgow	G11 9QX	Flat	3	350	CO40		B003
PG36	2 Manor Rd	Glasgow	G32 4QX	Flat	3	375	CO93	SG37	B003
PG21	18 Dale Rd	Glasgow	G12	House	5	600	CO87	SG37	B003
PG16	5 Novar Dr	Glasgow	G12 9AX	Flat	4	450	CO93	SG14	B003

Client

clientNo	fName	lName	telNo	prefType	maxRent	eMail
CR76	John	Kay	0207-774-5632	Flat	425	john.kay@gmail.com
CR56	Aline	Stewart	0141-848-1825	Flat	350	astewart@hotmail.com
CR74	Mike	Ritchie	01475-392178	House	750	mritchie01@yahoo.co.uk
CR62	Mary	Tregear	01224-196720	Flat	600	maryt@hotmail.co.uk

PrivateOwner

ownerNo	fName	lName	address	telNo	eMail	password
CO46	Joe	Keogh	2 Fergus Dr, Aberdeen AB2 7SX	01224-861212	jkeogh@lhh.com	*****
CO87	Carol	Farrel	6 Achray St, Glasgow G32 9DX	0141-357-7419	cfarrel@gmail.com	*****
CO40	Tina	Murphy	63 Well St, Glasgow G42	0141-943-1728	tinam@hotmail.com	*****
CO93	Tony	Shaw	12 Park Pl, Glasgow G4 0QR	0141-225-7025	tony.shaw@ark.com	*****

Viewing

clientNo	propertyNo	viewDate	comment
CR56	PA14	24-May-13	too small
CR76	PG4	20-Apr-13	too remote
CR56	PG4	26-May-13	
CR62	PA14	14-May-13	no dining room
CR56	PG36	28-Apr-13	

Registration

clientNo	branchNo	staffNo	dateJoined
CR76	B005	SL41	2-Jan-13
CR56	B003	SG37	11-Apr-12
CR74	B003	SG37	16-Nov-11
CR62	B007	SA9	7-Mar-12

<pre>exercise 1 SELECT propertyNO, type, rent FROM propertyForRent; SELECT * FROM branch; SELECT fname, lname, telNo from client;</pre>	<pre>-- exercise 2 SELECT propertyNo, street, city, type, rent AS monthlyRent, rent *12 AS annualRent FROM propertyForRent;</pre> Calculated fields
<pre>-- exercise 3 SELECT CONCAT(lName, ", ", fName) AS fullName FROM staff;</pre>	<pre>-- exercise 4 SELECT * FROM propertyForRent WHERE rent <= 400; SELECT clientNo, fName,lName FROM client WHERE preftype = "house" ; SELECT * FROM viewing WHERE viewDate > "2015-05-15" ;</pre>
<pre>-- exercise 5 SELECT fName, lname, branchNo FROM staff WHERE position = "assistant" AND salary = 9000; SELECT * FROM propertyForRent WHERE RENT < 400 AND CITY ="Glasgow" ; --SELECT * FROM propertyForRent WHERE RENT < 400 AND CITY = "Glasgow" ; SELECT * FROM viewing WHERE viewDate >= '2015-04-01' AND viewDate <= '2015-04-31';</pre>	<pre>---exercise 6 SELECT * FROM viewing ORDER BY ViewDate DESC; SELECT * FROM viewing ORDER BY clientNo, ViewDate;</pre>

Further reading

- Chapter 6 from Connolly,T.,&begg,C.(2005). Database systems: a practical approach to design, Implementation and management, Boston, MA: pearson Addison-Wesley