



# LPIC-1. Examen 102. Objectiu 105.3

## LPI 105.3. Gestió de dades SQL Wiki:

[http://acacha.org/mediawiki/index.php/LPI\\_105.3](http://acacha.org/mediawiki/index.php/LPI_105.3)

# Objectius

## 105.3. Gestió de dades SQL



- **Objectiu:** Els candidats han de ser capaços de fer consultes a bases de dades i manipular dades utilitzant consultes SQL bàsiques. Aquest objectiu inclou realitzar consultes que uneixin 2 taules i/o subconsultes.
- **Pes:** 2



### Àrees Clau de Coneixement:

- Utilitzar ordres SQL bàsiques
- Realitzar manipulacions bàsiques de dades



La següent és una llista parcial de fitxers, termes i utilitats utilitzades:

- insert
- update
- select
- delete
- from
- where
- group by
- order by
- join



**Apunts:** LPI 105.3. Gestió de dades SQL

## ♦ SQL (Standard Query Language)

- ♦ Aka Structured Query Language: llenguatge de consultes estructurades
- ♦ Estàndard ANSI/ISO
- ♦ **Llenguatge declaratiu** d'accés a **bases de dades relacionals** que permet realitzar diversos tipus d'operacions en una base de dades.
- ♦ Va ser desenvolupat a IBM per Donald D. Chamberlin i Raymond F. Boyce a principis de la dècada dels 70
- ♦ La versió inicial es va anomenar **SEQUEL**. Es va canviar a SQL, era una marca registrada d'una altra companyia.
- ♦ Vegeu també els **temes d'oposicions d'informàtica**

## ♦ 2 formes d'utilitzar SQL

- ♦ **SQL interactiu**: se'ns proporciona un "interpret d'ordres" SQL i executem consultes SQL.
- ♦ **SQL immers**: aka SQL embedded (encastat o "empotrat"). **No es tracta a LPI**. Similar als guions de shell.

## ♦ Subconjunts de SQL:

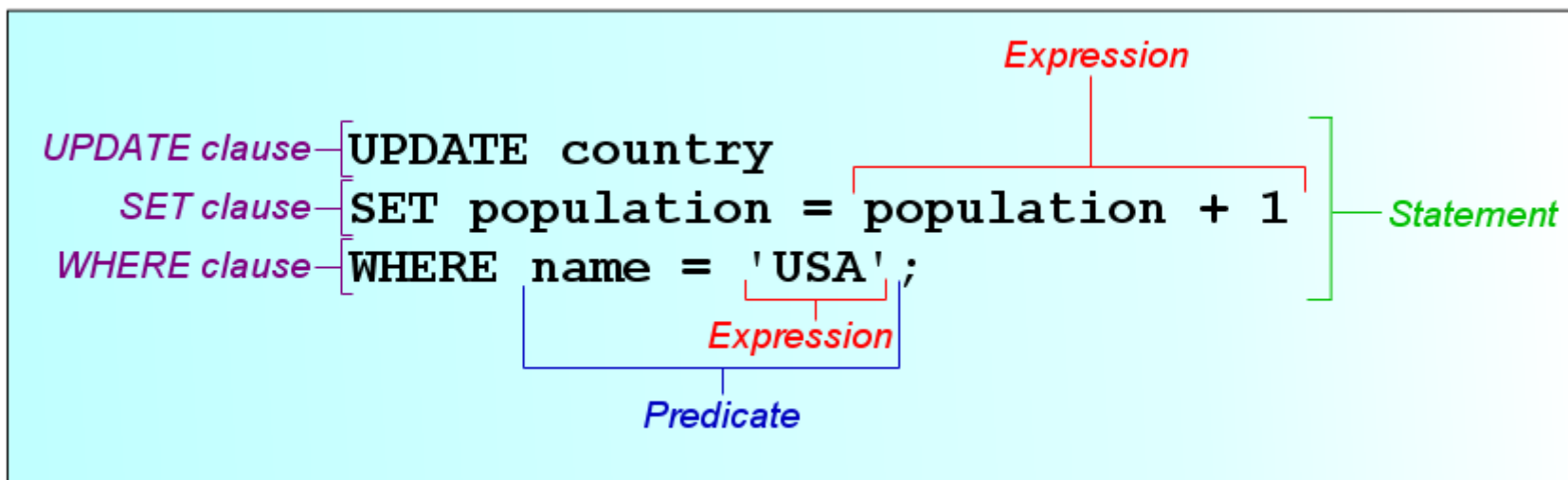
- ♦ **Data Definition Language (DDL)**: Definició de l'estructura de la base de dades. **Clàusules**: CREATE, DROP i ALTER. **NO LPI**
- ♦ **Data Manipulation Language (DML)**: Manipulació de les dades. **Clàusules**: INSERT, UPDATE i DELETE.
- ♦ **Data Control Language (DCL)**: Control d'accés a les dades. **Clàusules**: GRANT i REVOKE. **NO LPI**

## ♦ LPI només demana conèixer les operacions més habituals dels DML:

- ♦ **Consulta:** Obtenir informació de la base de dades amb la clàusula SELECT. I
  - Inclou les (sub)clàusules opcionals FROM, WHERE, GROUP BY, JOIN o ORDER BY i UNION i subconsultes.
- ♦ **Inserció:** Afegir informació de la base de dades.  
Clàusula INSERT
- ♦ **Modificació:** Modificar informació de la base de dades  
Clàusula UPDATE
- ♦ **Eliminació:** Eliminar informació de la base de dades  
Clàusula DELETE

## ◆ Elements del llenguatge:

- ◆ **Clàusules:** són els components principals de les declaracions o les consultes. Poden ser opcionals
- ◆ **Expressions:** poden produir valors escalars o taules consistents en columnes i files.
- ◆ **Predicats:** especifiquen condicions que permeten limitar els efectes d'una declaració o consulta o permeten controlar el flux del programa



# SQL

- ♦ **Consultes (Querys):** obtenen informació de la base de dades segons uns criteris específics.
- ♦ **Declaracions (Statements):** poden tenir efectes persistents sobre l'estructura o les dades de la base de dades o poden controlar transaccions, el flux del programar, les connexions, diagnòstics, etc.
- ♦ Les declaracions SQL poden acabar en punt i coma (semicolon ; ) que indica el final d'una declaració. No és obligatori a totes les plataformes però està definit a la gramàtica estàndard de SQL.
- ♦ Els espais en blanc extrems són normalment ignorats a SQL.



## ♦ SQL three-valued logic (3VL)

- ♦ Lògica booleana (Boolean truth values) de 3 valors
- ♦ **Nul (Null o unknown)**: indica falta d'informació.
- ♦ Els valors lògics poden ser certs (true), falsos (false) o nuls (null).
- ♦ Cal tenir-ho en compte al fer comparacions i operacions lògiques com ORs o ANDs.
- ♦ Operadors IS NULL i IS NOT NULL
- ♦ Les següents taules lògiques us indiquen quins són els resultats amb lògica de 3 valors:



# SQL

p AND q		p		
		True	False	Unknown
q	True	True	False	Unknown
	False	False	False	False
	Unknown	Unknown	False	Unknown

p OR q		p		
		True	False	Unknown
q	True	True	True	True
	False	True	False	Unknown
	Unknown	True	Unknown	Unknown

p	NOT p
True	False
False	True
Unknown	Unknown

p = q		p		
		True	False	Unknown
q	True	True	False	Unknown
	False	False	True	Unknown
	Unknown	Unknown	Unknown	Unknown

# Gestors de bases de dades

## ♦ SQL és un estàndard de facto

- ♦ La gran majoria de gestors de bases de dades el suporten
- ♦ No tots els gestors de base de dades suporten el mateix SQL. Hi ha diferents estàndards SQL i diferències en les implementacions de diferents fabricants.
- ♦ **Poques diferències!:** tot el que veurem en aquest objectiu és aplicable a qualsevol gestor de base de dades modern

## ♦ Gestors de bases de dades lliures:

- ♦ MySQL
- ♦ PostgreSQL

# MySQL

## ♦ Possiblement el gestor de base de dades lliure més utilitzat

- ♦ Creat per la empresa sueca MySQL AB
- ♦ Comprat a posteriori per Sun Microsystems
- ♦ L'any 2009 Oracle va comprar Sun.
- ♦ La filla del primer desenvolupador es deia My

## ♦ Instal·lació

- ♦ `$ sudo apt-get install mysql-server`
- ♦ També podeu utilitzar `tasksel` i instal·lar un **LAMP**

## ◆ Clients MySQL

### ◆ Per línia d'ordres:

```
$ sudo mysql -p
[sudo] password for sergi:
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 110
Server version: 5.1.37-1ubuntu5.1 (Ubuntu)
Type 'help;' or '\h' for help. Type '\c' to clear the current
input statement.
mysql>
```

### ◆ Interfícies gràfiques:

- PhpMyAdmin
- Clients de Mysql (paquets mysql-admin, mysql-community, mysql-enterprise, mysql-navigator)

## ◆ Configuració:

```
sudo gedit  
my.cnf
```

## ◆ Accés

- ◆ Usuari per defecte **root sense contrasenya** amb accés només des de **localhost**.

## ◆ Bases de dades

- ◆ Carpeta **/var/lib/mysql**

## ◆ Client

```
sudo apt-get install mysql-client
```

# Sistemes LAMP

- ♦ **Conjunt de programari lliure utilitzat de forma conjunta per crear un servidor web amb suport per a la creació de llocs web dinàmics.**
- ♦ **L'acrònim LAMP:**
  - ♦ **Linux.** Sistema Operatiu.
  - ♦ **Apache.** Servidor web.
  - ♦ **Mysql.** Base de dades.
  - ♦ **Php.** Llenguatge de programació.



## ◆ PhpMyAdmin

- ◆ Aplicació web escrita en PHP per administrar un servidor MySQL a través d'Internet.

```
$ sudo apt-get install phpmyadmin
```

```
http://localhost/phpmyadmin/
```

## ◆ Exercici

- ◆ Accedir a la base de dades Mysql i consultar la base de dades **mysql**. Què conté aquesta base de dades?
- ◆ Què cal fer per canviar les dades d'accés d'un usuari?
- ◆ I si oblidem la contrasenya de root?





## ◆ PostgreSQL

- ◆ Sistema de gestió de base de dades relacionals (ORDBMS).
- ◆ Llicència BSD. No és propietat de cap empresa.

## ◆ Instal·lació

```
$ sudo apt-get install postgresql-8.1
```

- ◆ A l'instal·lar el servidor també s'instal·la el client (**postgresql-client-8.1**)
- ◆ Usuari per defecte postgres i contrasenya psql

```
$ sudo -u postgres psql
```

## ♦ PhpPgAdmin

- ♦ Aplicació web escrita en PHP per administrar un servidor PostgreSQL a través d'Internet.

## ♦ Instal·lació:

```
$ sudo apt-get install phppgadmin
```

## ♦ Exercici

- ♦ Phppgadmin no s'instal·la directament a Apache. Copieu (o enllaceu) el fitxer de configuració **apache.conf** de phppgadmin a la carpeta **sites-available** i afegiu el site amb la comanda **a2ensite**.

```
http://localhost/phppgadmin/
```

# Bases de dades d'exemple

## ♦ Utilitzarem dos:

- ♦ **World:** dades de països
- ♦ **Employees:** Base de dades fictícia empleats d'una empresa
- ♦ <http://dev.mysql.com/doc/index-other.html>

```
$ cd; mkdir world && cd world
$ wget http://downloads.mysql.com/docs/world.sql.gz
$ gunzip world.sql.gz
$ mysql -u root -p
mysql> CREATE DATABASE world;
mysql> USE world;
mysql> SOURCE world.sql;
mysql> SHOW TABLES;
```

**Base de dades world**

# Bases de dades d'exemple

## ♦ Employees

```
$ cd; mkdir employers && cd employers
$ wget http://tinyurl.com/y8cobon
$ tar jxvf employees_db-full-1.0.6.tar.bz2
$ cd employees_db
$ sudo mysql -p < employees.sql
$ sudo mysql -p
mysql> USE employees;
mysql> SHOW TABLES;
```

## ♦ **Querys. SELECT**

- ♦ Les operacions més comunes amb bases de dades són són les consultes (querys) les quals es realitzen amb l'element de llenguatge SELECT. SELECT obté informació d'una o més taules d'una base de dades
- ♦ Les consultes SQL no provoquen canvis persistents a la base de dades. Una excepció és SELECT INTO...
- ♦ Les consultes només indiquen les dades que volem obtenir i no pas com s'han d'obtenir. És el gestor de base de dades (DBMS) el responsable de planificar, optimitzar i realitzar les operacions físiques per tal d'obtenir els resultats demanats.

## ♦ SELECT. Sintaxi

```
SELECT
  [ALL | DISTINCT | DISTINCTROW ]
  select_expr [, select_expr ...]
  [FROM table_references
  [WHERE where_condition]
  [GROUP BY {col_name | expr | position}
   [ASC | DESC], ...]
  [HAVING where_condition]
  [ORDER BY {col_name | expr | position}
   [ASC | DESC], ...]
  [LIMIT {[offset,] row_count | row_count OFFSET offset}]
```

- ♦ El resultat de la consulta és guarda en una **taula de resultat** o **conjunt resultat (result table o result-set)**
- ♦ Després de la paraula clau SELECT, sempre s'indica la llista de camps (**select\_expr**) que volem mostrar.
- ♦ Es pot utilitzar l'asterisc (\*) per tal d'indicar que volem mostrar totes les columnes implicades a les taules de la consulta.

## ♦ Les clàusules opcionals són:

- ♦ **FROM:** Indica de quina taula o taules seran obtingudes les dades. La clàusula FROM pot incloure subclàusules opcionals JOIN per especificar les normes per a unir taules entre si.
- ♦ **WHERE:** aquesta clàusula inclou un predicat de comparació, que restringeix el conjunt de files retornades per la consulta. La clàusula WHERE elimina totes les files del conjunt resultat per les quals les condicions del predicat no són avaluades com a certes (true).
- ♦ **GROUP BY:** s'utilitza per agrupar les files del conjunt resultat amb valors comuns en conjunts de files més petites. GROUP BY s'utilitza sovint amb funcions d'agregació o s'utilitza per eliminar files duplicades. La clàusula WHERE sempre s'aplica abans que la GROUP BY.



# Consultes

- ♦ **HAVING**: aquesta clàusula inclou un predicat que s'utilitza per filtrar files resultat d'una agrupació amb GROUP BY. També s'utilitzen sovint funcions d'agregació.
- ♦ **ORDER BY**: ordena les files del conjunt resultat i també permet especificar en quina direcció han de ser ordenades (ASC, ordre ascendent o DESC, ordre descendent). Sinó s'utilitza ORDER BY, aleshores l'ordre del conjunt resultat no està definit a priori.
- ♦ **Consulteu la wiki del curs per a trobar més exemples**

## ♦ S'utilitzen en expressions

- ♦ Sobretot a les clàusules WHERE

Operador	Descripció
=	Igual
<> o !=	No igual o diferent
>	Major
>=	Major o igual
<	Menor
<=	Menor que
BETWEEN	Indica un rang que inclou els extrems
LIKE	Busca un patró
IN	Indica una llista de valors possibles

## ♦ Funcions d'agregació

- ♦ Utilitzades amb les subclàusules GROUP BY i HAVING

Funció	Descripció
AVG()	Calcula la mitja.
COUNT()	Conta el número de files
FIRST()	Mostra el primer valor
LAST()	Mostra l'últim valor
MAX()	Retorna el valor més gran
MIN()	Retorna el valor més petit
SUM()	Retorna la suma dels camps

- ♦ Existeixes **més funcions** però no entren als objectius de LPI

# JOIN

- ♦ S'utilitza per tal d'obtenir dades de dos o més taules segons les relacions entre certes columnes d'aquestes taules.
- ♦ Les taules sovint estan relacionades entre si mitjançant claus:
  - **Clau primària:** és una columna o combinació de columnes amb un únic valor per cada fila. Les claus primàries identifiquen de forma unívoca un registre o fila d'una taula.
  - **Claus foranies:** són camps d'una taula que contenen una clau primària d'un altra taula.
- ♦ Alguns gestors de bases de dades permeten especificar restriccions segons les claus.

# JOIN

## ♦ Hi ha diferents tipus de JOINS:

- ♦ **JOIN o INNER JOIN:** Mostra les files que tenen al menys una coincidència a les 2 taules. Si no s'indica JOIN i es separen les taules per coma (","), aleshores s'està fent un INNER JOIN.
- ♦ **LEFT JOIN:** Mostra totes les files de la taula de l'esquerre inclòs si no hi ha cap coincidència a la taula de la dreta.
- ♦ **RIGHT JOIN:** Mostra totes les files de la taula de la dreta inclòs si no hi ha cap coincidència a la taula de l'esquerre.
- ♦ **FULL JOIN:** Mostra les files que tenen una coincidència al menys a una de les taules.
- ♦ Alguns gestors de bases de dades com SQL proporcionen més tipus de JOINS.

# JOIN

## ♦ Sintaxi

```
SELECT column_name(s)
FROM table_name1
[INNER|LEFT|RIGHT|FULL] JOIN table_name2
ON table_name1.column_name=table_name2.column_name
```

- ♦ MySQL no admet FULL JOINS, s'han de fer amb un UNION de les consultes LEFT JOIN i RIGHT JOIN.
- ♦ Proveu els **exemples de la wiki del curs**

# Subconsultes

- ♦ Les subconsultes són consultes que estan incloses en altres consultes. Normalment es troben a les expressions de les clàusules WHERE.
- ♦ En els cas de que el resultat de la subconsulta sigui un escalar es pot utilitzar directament la consulta (entre “()”) en expressions.
- ♦ S'utilitza la paraula reservada IN si la subconsulta retorna un conjunt de valors (no és un escalar)
- ♦ NOTA: MySQL fins a la versió 4.1 no suportava subconsultes
- ♦ Vegeu [exemples a la wiki del curs](#)



## ♦ Ordres de manipulació de dades

- ♦ Les clàusules que es permeten manipular dades són ( Data Manipulation Language o DML):
  - **INSERT**: Permet afegir dades a les taules de la base de dades.
  - **UPDATE**: Permet modificar dades ja existents a les taules de la base de dades.
  - **DELETE**: Permet eliminar dades de les taules de la base de dades.
  - L'últim estandard SQL (y MySQL) també incorpora **TRUNCATE** (buida una taula)

# Ordres de manipulació de dades

## ➤ INSERT

➤ Sintaxi:

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

## ➤ UPDATE

➤ Sintaxi:

```
UPDATE table_name  
SET column1=value, column2=value2,...  
WHERE some_column=some_value
```

## ➤ DELETE

➤ Sintaxi

```
DELETE FROM table_name  
WHERE some_column=some_value
```

- Sense la clàusula WHERE els canvis s'apliquen a tots els registres!
- Consulteu els [exemples de la wiki del curs](#)

## ♦ Us heu fixat en els apunts de la wiki?

- ♦ Mediawiki us proporciona dos eines que poden ser molt útils per a treballar amb SQL:
- ♦ **Interwiki links:** podeu utilitzar l'espai de noms `[[SQL:]]` per a fer enllaços a la web:
  - <http://www.w3schools.com/sql>
  - `[[SQL:SELECT]]`--> [http://www.w3schools.com/sql/sql\\_SELECT.asp](http://www.w3schools.com/sql/sql_SELECT.asp)
- ♦ **Extensió Syntax highlight**
  - Permet colorejar codi SQL



## Reconeixement 3.0 Unported

### Sou lliure de:



copiar, distribuir i comunicar públicament l'obra



fer-ne obres derivades

### Amb les condicions següents:



**Reconeixement.** Heu de reconèixer els crèdits de l'obra de la manera especificada per l'autor o el llicenciador (però no d'una manera que suggereixi que us donen suport o rebeu suport per l'ús que feu l'obra).

- Quan reutilitzeu o distribuïu l'obra, heu de deixar ben clar els termes de la llicència de l'obra.
- Alguna d'aquestes condicions pot no aplicar-se si obteniu el permís del titular dels drets d'autor.
- No hi ha res en aquesta llicència que menyscabi o restringeixi els drets morals de l'autor.

Advertiment

Els drets derivats d'usos legítims o altres limitacions reconegudes per llei no queden afectats per l'anterior  
Això és un resum fàcilment llegible del text legal (la llicència completa).

<http://creativecommons.org/licenses/by/3.0/deed.ca>