

## M02. BASES DE DADES

**UF3. llenguatges SQL: DCL i extensió procedimental**

# GESTIÓ D'USUARIS I PRIVILEGIS

Quins privilegis han de  
tenir?

---

# GESTIÓ D'USUARIS I PRIVILEGIS

En un sistema informàtic les dades constitueixen un recurs valuós que ha d'estar controlat i gestionat estrictament.

Entenem per seguretat d'un sistema el conjunt de mecanismes de protecció enfront d'accessos no autoritzats, ja siguin intencionats o accidentals.

A més, si la informació fa referència a persones i s'emmagatzemen dades de naturalesa sensible ens caldrà saber quines són les obligacions legals que tenim.

# GESTIÓ D'USUARIS I PRIVILEGIS

Quan utilitzem un sistema gestor de bases de dades (SGBD) per accedir a la informació emmagatzemada en una base de dades, primerament cal comprovar quines autoritzacions tenim sobre aquelles dades; d'això s'encarrega el component de seguretat de l'SGBD.

En un sistema d'informació (SI), les diferents aplicacions i usuaris de l'organització fan servir un únic conjunt de dades, anomenat base de dades corporativa, amb l'SGDB. D'una banda, això resol problemes de redundància, inconsistència i independència entre les dades i els programes i, de l'altra, fa que la seguretat esdevingui un dels problemes més importants en aquests entorns.

# GESTIÓ D'USUARIS I PRIVILEGIS. SEGURETAT

- **Confidencialitat:** cal protegir l'ús de la informació per part de persones no autoritzades. Això implica que un usuari només ha de poder accedir a la informació per a la qual té autorització i que a partir d'aquesta informació no podrà inferir altra informació que es consideri secreta.
- **Integritat:** la informació s'ha de protegir de modificacions no autoritzades; això també inclou tant la inserció de dades falses com la destrucció de dades.
- **Disponibilitat:** la informació ha d'estar disponible en el moment que li faci falta a l'usuari.

# GESTIÓ D'USUARIS I PRIVILEGIS. SEGURETAT

Definirem el concepte **amenança** com tot aquell agent hostil que, de manera casual o intencionada i utilitzant una tècnica especialitzada, pot revelar o modificar la informació gestionada pel sistema.

- Amenaces no fraudulentas (accidents casuals)
- Amenaces fraudulentas (violacions intencionades)

# GESTIÓ D'USUARIS I PRIVILEGIS. SEGURETAT

- Amenaces no fraudulent (accidents casuais):
  - desastres naturals o accidentals: normalment són accidents que danyen el maquinari del sistema, com per exemple aquells produïts per terratrèmols, inundacions o foc
  - errors del sistema: corresponen a tots aquells errors accidentals en el maquinari o en el programari que poden conduir a accessos no autoritzats
  - errors humans: corresponen a aquelles errades involuntàries derivades de l'acció dels usuaris en introduir dades o utilitzar aplicacions que treballen sobre aquestes

# GESTIÓ D'USUARIS I PRIVILEGIS. SEGURETAT

- Amenaces fraudulent: generen violacions intencionades i són causades per dos tipus d'usuaris diferents:
  - usuaris autoritzats que abusen dels seus privilegis
  - agents hostils o usuaris impropis que executen accions de vandalisme sobre el programari o el maquinari del sistema o també lectures o escriptures de dades



# GESTIÓ D'USUARIS I PRIVILEGIS. SEGURETAT

Les violacions sobre una base de dades consisteixen en lectures, modificacions o esborraments incorrectes de les dades. Les conseqüències d'aquestes violacions es poden agrupar en tres categories:

1. Lectura inadequada d'informació. Causat per la lectura de dades per part d'usuaris no autoritzats mitjançant un accés intencionat o accidental. S'inclouen les violacions del secret derivades de les deduccions d'informació que es considera secreta.
2. Modificació impròpia de les dades. Correspon a totes les violacions de la integritat de les dades per tractaments o modificacions fraudulentament d'aquestes. Les modificacions impròpies no involucren necessàriament lectures no autoritzades, ja que les dades es poden falsificar sense ser llegides.
3. Denegació de serveis. Correspon a accions que puguin impedir que els usuaris accedeixin a les dades o utilitzin els recursos que tenen assignats.

# GESTIÓ D'USUARIS I PRIVILEGIS. NIVELLS DE SEGURETAT

1. Sistema gestor de base de dades: pot ser que alguns usuaris de la base de dades solament tinguin accés a una part limitada de la base de dades. Pot ser que altres usuaris tant sols tinguin autorització per fer consultes però que no puguin modificar les dades. És responsabilitat de l'administrador de l'SGBD que no es violin aquestes restriccions d'autorització.
2. Sistema operatiu: independentment del nivell de seguretat assolit en l'SGBD la debilitat de la seguretat del sistema operatiu pot servir com a mitjà per a accessos no autoritzats a la base de dades.

# GESTIÓ D'USUARIS I PRIVILEGIS. NIVELLS DE SEGURETAT

3. Xarxa: atès que gairebé tots els sistemes de bases de dades permeten l'accés remot mitjançant terminals o xarxes, la seguretat en el nivell de programari de la xarxa és tan important com la seguretat física, tant a Internet com en les xarxes privades de les empreses.
4. Físic: els llocs on estan ubicats els sistemes d'informació cal que estiguin adequadament protegits contra l'entrada d'intrusos.
5. Humà: els usuaris han d'estar degudament autoritzats per reduir la possibilitat que algun doni accés a intrusos a canvi de suborns o d'altres favors.

# GESTIÓ D'USUARIS I PRIVILEGIS. IDENTIFICACIÓ I AUTENTICACIÓ

La primera acció que cal fer per assolir la seguretat d'un sistema d'informació és la capacitat de verificar la identitat dels usuaris. Aquest procés està format per dues parts:

- Identificació: implica la manera en què l'usuari proporciona la seva identitat al sistema (veure qui és). Segons els requisits operacionals, una identitat pot descriure un individu, més d'un individu, o un o més individus només durant un període de temps.
- Autenticació: és la manera en què un individu estableix la validesa de la seva identitat (verificar que l'usuari és qui diu que és).

# GESTIÓ D'USUARIS I PRIVILEGIS. CONTROL D'ACCÉS

Definim **control d'accés** com el conjunt de funcions de l'SGBD per assegurar que els accessos al sistema estan d'acord amb les regles establertes per la política de protecció fixada pel model de negoci.

El control d'accés es pot considerar format per dos components:

- Polítiques d'accés: defineixen els principis pels quals s'autoritza un usuari o es denega l'accés específic a un objecte de la base de dades.
- Mecanismes de seguretat: formats per tots aquells procediments que s'aplicaran a les consultes amb l'objectiu que els usuaris compleixin els principis anteriors.

# GESTIÓ D'USUARIS I PRIVILEGIS. AUDITORIA

L'auditoria correspon a un conjunt de mecanismes per saber qui ha fet què, és a dir, portar un registre de qui fa tots els canvis i consultes a la base de dades. Més que un mecanisme de seguretat és un mecanisme per detectar el culpable.

S'utilitza per als casos següents:

- La investigació d'una activitat sospitosa.
- El monitoratge d'activitats específiques de la base de dades.

# GESTIÓ D'USUARIS I PRIVILEGIS. AUDITORIA

El sistema d'auditoria ha de permetre diferents formes d'utilització:

- Auditar sentències. L'auditoria indicarà quan i qui ha utilitzat un tipus de sentència correcta. Per exemple, auditar totes les insercions o esborraments.
- Auditar objectes. El sistema auditarà cada vegada que es faci una operació sobre un objecte determinat.
- Auditar sentències sobre objectes, una versió combinada de les dues anteriors.
- Auditar usuaris o grups.

La informació que s'acostuma a emmagatzemar quan es fa una tasca d'auditoria és el nom de l'usuari, l'identificador de la sessió, l'identificador del terminal, el nom de l'objecte al qual s'ha accedit, l'operació executada o intentada, el codi complet de l'operació, la data i l'hora.

# GESTIÓ D'USUARIS I PRIVILEGIS. FUNCIONS DEL DBMS ADMIN

Les funcions de l'administrador de la base de dades inclouen:

1. Definició de l'esquema. L'administrador crea l'esquema original de la base de dades escrivint un conjunt d'instruccions de definició de dades.
2. Definició de l'estructura i del mètode d'accés. Referent al programari client emprat i les diferents activitats relacionades amb l'emmagatzematge i recuperació utilitzant diferents estàndards.
3. Modificació de l'esquema i l'organització física. Els administradors de la base de dades fan canvis en l'esquema i l'organització física per reflectir les necessitats canviants dins de l'organització, o per fer alteracions en l'organització física per millorar-ne el rendiment.



# GESTIÓ D'USUARIS I PRIVILEGIS. FUNCIONS DEL DBMS ADMIN

4. Concessió d'autorització per a l'accés a les dades. La concessió de diferents tipus d'autorització permet a l'administrador de la base de dades determinar a quines parts de la base de dades pot accedir cada usuari: la informació d'autorització es manté en una estructura de l'esquema especial que el sistema de base de dades consulta quan s'intenta fer l'accés a les dades.

# GESTIÓ D'USUARIS I PRIVILEGIS. FUNCIONS DEL DBMS ADMIN

5. Manteniment rutinari. Alguns exemples d'activitats rutinàries de manteniment de l'administrador són:
  - a. Còpia de seguretat periòdica de la base de dades (cinta o servidors remots) per prevenir la pèrdua de dades a causa de desastres naturals.
  - b. Verificació de l'espai lliure necessari al disc per a les operacions habituals i incrementar-lo en cas que sigui necessari.
  - c. Supervisió de les tasques que s'executen a la base de dades i verificar que el rendiment no es degrada per tasques molt costoses iniciades per alguns usuaris.

# GESTIÓ D'USUARIS I PRIVILEGIS

Instal·lació de MySQL Server

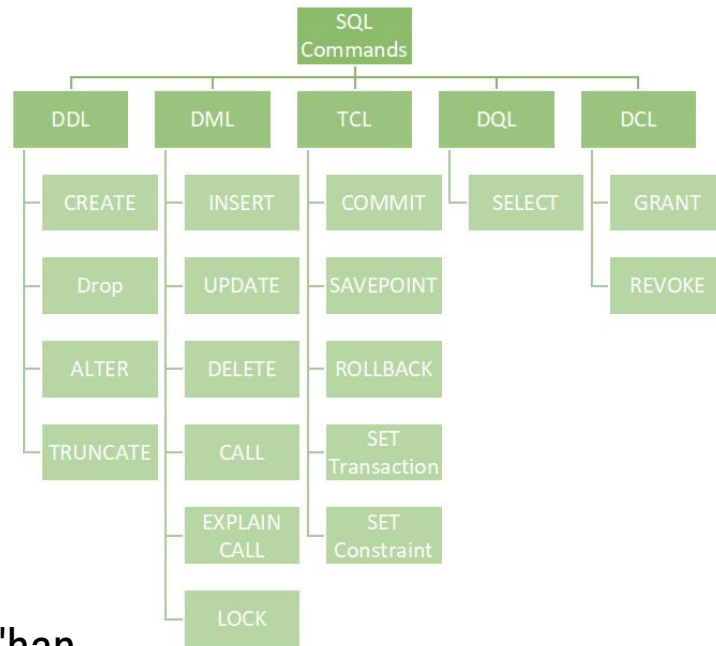
---

# MYSQL SERVER. SQL STATEMENTS

1. DDL (Data Definition Language)
2. DQL (Data Query Language)
3. DML (Data Manipulation Language)
4. DCL (Data Control Language)
5. TCL (Transaction Control Language)

## DCL (Data Control Language)

- GRANT, permet atorgar permisos.
- REVOKE, elimina els permisos que prèviament s'han concedit.



# MYSQL SERVER. SETUP

## 1. Instal·lació:

```
$sudo apt update
```

```
$sudo apt install mysql-server
```

//opcional [securitzar](#) el server

## 2. Accés al MySQL server:

```
$mysql -u user -p
```

(ens demanarà introduir password)

## 3. Mostrar les bases de dades actuals

```
$show databases;
```

```
mysql> show databases;
+-----+
| Database |
+-----+
| information_schema |
| mysql |
| performance_schema |
| sys |
+-----+
4 rows in set (0,00 sec)
```

# MYSQL SERVER. CREATING AND SELECTING A DATABASE

## 1. Creació de la database (base de dades)

```
CREATE DATABASE [IF NOT EXISTS] database_name  
[CHARACTER SET charset_name]  
[COLLATE collation_name];
```

Server Character Set  
and Collation

```
CREATE DATABASE db_name CHARACTER SET latin1 COLLATE latin1_swedish_ci;
```

## 2. Accedir a la database (base de dades)

```
$use database_name;
```

## 3. Mostrar les taules de la database en ús

```
$show tables;
```

# MYSQL SERVER. CREATING A TABLE

## 4. Creació d'una taula

```
CREATE TABLE [IF NOT EXISTS] table_name(  
    column1 datatype,  
    column2 datatype,  
    column3 datatype  
);
```

```
CREATE TABLE pet (  
    name VARCHAR(20),  
    owner VARCHAR(20),  
    species VARCHAR(20),  
    sex CHAR(1),  
    birth DATE,  
    death DATE  
);
```

# MYSQL SERVER. CREATING A TABLE

5. Visualització de l'estructura de la taula

```
$describe table_name;
```

```
mysql> describe pet;
```

Field	Type	Null	Key	Default	Extra
name	varchar(20)	YES		NULL	
owner	varchar(20)	YES		NULL	
species	varchar(20)	YES		NULL	
sex	char(1)	YES		NULL	
birth	date	YES		NULL	
death	date	YES		NULL	

6 rows in set (0,01 sec)



# MYSQL SERVER. CREATING A TABLE

## 6. Modificació d'una taula (afegir columnes)

```
ALTER TABLE table_name ADD column_name columntype;
```

```
ALTER TABLE pet ADD id int;  
ALTER TABLE pet ADD PRIMARY KEY (id);  
ALTER TABLE pet MODIFY COLUMN sex char NOT NULL;  
ALTER TABLE pet MODIFY COLUMN id INT auto_increment;  
ALTER TABLE pet DROP COLUMN sex;
```

# MYSQL SERVER. CREATING A TABLE

## 7. Creació d'una taula (estructura general)

```
CREATE TABLE [IF NOT EXISTS] table_name(  
    column1 datatype NOT NULL AUTO_INCREMENT,  
    column2 datatype NOT NULL,  
    column3 datatype DEFAULT NULL,  
    PRIMARY KEY (column1),  
    FOREIGN KEY (column2) REFERENCES tbl_name (col3)  
);
```

[Definicions de la taula](#)

# MYSQL SERVER. CREATING A TABLE

## 8. Inserció de valors en una taula

```
INSERT INTO table_name [(col_name [, col_name] ...)]  
    { {VALUES | VALUE} (value_list) [, (value_list)] ... }
```

```
INSERT INTO pet VALUES  
('Puffball','Diane','hamster','f','1999-03-30',NULL);
```

[Tipus d'insercions](#)