

Entidades débiles:

Son aquellas cuya existencia depende de la existencia de otras instancias de entidad. Por ejemplo, consideremos las entidades EDIFICIO y AULAS supongamos que puede haber aulas identificadas con la misma numeración, pero en edificios diferentes. La numeración de cada aula no identificará completamente cada una de ellas. Para poder identificar completamente un aula es necesario saber también en qué edificio está localizada. Por tanto, la existencia de una instancia de una entidad débil depende de la existencia de una instancia de la entidad fuerte con la que se relaciona.

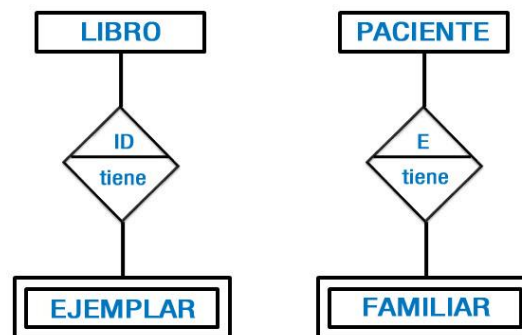
Entidad Débil: es un tipo de entidad cuyas propiedades o atributos no la identifican completamente, sino que sólo la identifican de forma parcial. Esta entidad debe participar en una relación que ayude a identificarla.

En el modelo E/R una entidad débil se representa con el nombre de la entidad encerrado en un rectángulo doble. En el gráfico se muestra la representación de la entidad AULA

Las entidades débiles presentan dos tipos de dependencia

Dependencia en existencia: entre entidades, si desaparece una instancia de entidad fuerte desaparecerán las instancias de entidad débiles que dependan de la primera. La representación de este tipo de dependencia incluirá una E en el interior de la relación débil.

Dependencia en identificación: debe darse una dependencia en existencia y además, una ocurrencia de la entidad débil no puede identificarse por sí misma, debiendo hacerse mediante la clave de la entidad fuerte asociada. La representación de este tipo de dependencia incluirá una ID en el interior de la relación débil.

**Privilegios y Permiso en una DBMS**

A **privilege** in a database management system is the permission to execute certain actions on the database.

Privileges can be permitted to:

1. Access a table
2. Access permission to execute a database command
3. Access another user's object

Privileges make certain actions possible, such as connecting to a database, creating a table, and executing another user's stored procedure.

Privileges are granted to users so they can accomplish a given task. If privileges are not granted with guidance, this can lead to a security bridge on the database.

Categories of privileges

There are two main categories of privileges possible in a database:

1. System privileges
2. Object privileges

1. System privileges

A **system** privilege can include the following:

1. Perform an action on any object of a database.example
2. Create tablespaces in a database
3. Delete the rows of any table in a database

We have close to 60 system privileges in existence in the database management system.

System privileges are mostly granted to administrative personnel and application developers. This privilege is usually not open to end-users of the database.

Below is a table of some popular system privileges.

[CREATE] DATABASE	→	gives user permission to create a new database
[CREATE]EXTERNAL TABLE	→	allows user permission to create external table
[CREATE]GROUP	→	allows user permission to create a group
[CREATE]LIBRARY	→	allows user the permission to create a library
[CREATE]PROCEDURE	→	allows user the permission to create a stored procedure
[CREATE]SCHEMA	→	allows user permission to create a schema
[CREATE]TABLE	→	allows user the permission to create a table
RESTORE	→	allows user the permission to restore system
[CREATE]AGGREGATE	→	allows user to create user-defined aggregate (UDA)

2. Object privileges

Object privilege is the permission to access specific database objects. Object privilege entails performing a specific action on a particular table, function, or package.

The right to delete rows from a table is an object privilege. Object privileges are granted to normal users, unlike system privileges. Some of the popular object privileges are given below.

ABORT	→	allows the user to abort session. applies to groups & users
DELETE	→	allows the user to delete table rows. applies only to tables
DROP	→	allows the user to drop objects. applies to all object type
EXECUTE	→	allows user to run user-defined functions, user-defined aggregate or stored procedure
LIST	→	allows the user to display an object name, either in a list or in another manner. applies to all objects
UPDATE	→	allows users to modify table rows. applies to tables only.
TRUNCATE	→	allows the user to delete all rows from a table .applies to tables only
INSERT	→	allows the user to insert rows into a table. applies only to table
ALTER	→	allows the user to modify object attributes. applies to all objects.

Roles en una DBMS

Hay una serie de usuarios que pueden acceder o recuperar los datos bajo demanda utilizando la aplicación y las interfaces proporcionadas por el Sistema de gestión de base de datos (DBMS). Tipos de usuarios Los usuarios de la base de datos se pueden clasificar en los siguientes grupos:

Usuarios nativos: los usuarios nativos no necesitan ser conscientes de la presencia del sistema de base de datos. Son usuarios finales de la base de datos que trabajan a través de un menú de programas de aplicación, donde siempre se le indica al usuario el tipo y rango de respuesta.

Usuarios en línea: los usuarios en línea pueden comunicarse con las bases de datos directamente a través de un terminal en línea o indirectamente a través de la interfaz de usuario y los programas de aplicación.

Usuarios Sofisticados – Son aquellos usuarios que interactúan con el sistema sin escribir el programa sino que formulan su solicitud en lenguaje de consulta de base de datos. Son los programadores de SQL, que van a tratar directamente con la base de datos. Escriben consultas para eliminar o seleccionar o insertar y actualizar la base de datos.

Usuarios especializados: usuarios especializados que escriben aplicaciones de bases de datos especializadas que no encajan en el marco de procesamiento de bases de datos fraccionarias.

Programador de aplicaciones: los usuarios programadores de aplicaciones que son responsables de desarrollar los programas de aplicación o la interfaz de usuario. Los programas de aplicación podrían estar escritos en lenguaje de alto nivel. Por ejemplo: Java, .net, php, etc.

Administrador de base de datos (DBA) - Es una persona o el grupo a cargo de implementar el sistema de base de datos dentro de la organización. El DBA tiene todos los privilegios permitidos por el DBMS y puede asignar o eliminar los privilegios de los usuarios.

Clasificación de Usuarios DBMS

se clasifica principalmente en tres usuarios:

Los usuarios finales.

Programadores de aplicaciones.

Administrador de base de datos.

La clasificación de los usuarios en DBMS se representa gráficamente a continuación:

