



UNIVERSIDAD NACIONAL AUTÓNOMA DE  
MÉXICO

---

FACULTAD DE INGENIERÍA

BASES DE DATOS

GRUPO:1

PROFESOR: ING. FERNANDO ARREOLA  
FRANCO

ALUMNO: CALVILLO MARTÍNEZ ERIK  
JONATHAN

TAREA 10

"REGLAS DE CODD"



# Reglas de Codd

En 1984 Edgar F. Codd, creador de del Modelo Relacional, publicó las 12 Reglas que un verdadero Sistema Relacional de Bases de Datos debería cumplir, en la práctica algunas de estas reglas son difíciles de implementar. Codd se percató de que existían bases de datos en el mercado las cuales decían ser relacionales, pero lo único que hacían era guardar la información en las tablas, sin estar estas tablas literalmente normalizadas; gracias a las 12 reglas un sistema podrá considerarse “más relacional” cuanto más siga estas reglas.

## Regla 0. Regla de fundación

Para que este sistema califique como un sistema de gestión de base de datos relacional RDBMS ese sistema debe de usar su estructura relacional exclusivamente para gestionar la base de datos.

## Regla 1. Regla de la información

Toda la información es representada unidireccionalmente por los valores de las posiciones de las columnas dentro de las filas de las tablas, exactamente de una manera; con valores en tablas.

## Regla 2. Regla de garantía de acceso o regla del acceso garantizado

Todos los datos deben ser accesibles sin ambigüedad. Esta regla es un replanteamiento del requerimiento fundamental para las llaves primarias. Cada ítem de datos debe ser lógicamente accesible al ejecutar una búsqueda que combine el nombre de la tabla, su clave primaria, y el nombre de la columna. Esto significa que dado un nombre de tabla, dado el valor de la clave primaria, y dado el nombre de la columna requerida, deberá encontrarse uno y solamente un valor. Por esta razón la definición de claves primarias para todas las tablas es prácticamente obligatoria.

- Cada tabla debe tener su nombre único. No puede haber dos filas iguales.
- No se permiten los duplicados.
- Todos los datos en una columna deben ser del mismo tipo.

## Regla 3. Tratamiento sistemático de los valores nulos

“La información inaplicable o faltante puede ser representada a través de valores nulos”. Un RDBMS (Sistema Gestor de Bases de Datos Relacionales) debe ser capaz de soportar el uso de valores nulos en el lugar de columnas cuyos valores sean desconocidos o inaplicables.

## Regla 4. Catálogo dinámico en línea basado en el modelo relacional

El sistema debe soportar un catálogo relacional en línea y por línea el cuál es accesible por los usuarios autorizados por medio del lenguaje de consulta regular. Eso es, el usuario debe se capaz de acceder a la estructura de la base de datos (catálogo) usando el mismo lenguaje de consulta que usa para acceder a sus datos.

## **Regla 5. Regla comprensiva del sublenguaje de los datos**

Significa que debe haber por lo menos un lenguaje relacional con una sintaxis bien definida que pueda ser usado para administrar completamente la base de datos.

Este lenguaje debe contar con las siguientes características:

- Soportar una sintaxis lineal
- Puede ser usado para interactividad y dentro de programas de aplicación.
- Soporte operaciones de definición de datos, de manipulación de datos, seguridad e integridad y operaciones de administración de transacciones.

## **Regla 6. La regla de la actualización de vistas**

Todas las vistas que son teóricamente actualizables, deben ser actualizables por el sistema mismo. La mayoría de las RDBMS permiten actualizar vistas simples, pero deshabilitan los intentos de actualizar vistas complejas. La actualización debe de ser automática, sin necesidad de que el usuario tenga que estar actualizando manualmente.

## **Regla 7. Inserción, actualización y borrado de alto nivel**

Esto significa que los datos pueden ser restaurados de una base de datos relacional en sets de datos estructurados desde múltiples filas y/o múltiples tablas. Esta regla defina que las operaciones de inserción, actualización y eliminación deben de ser soportadas por algún set restaurable preferiblemente de sólo una simple columna en una simple tabla.

## **Regla 8. Independencia física de los datos**

El comportamiento de los programas de aplicación y de la actividad de usuarios vía terminales debería ser predecible basados en la definición lógica de la base de datos, y éste comportamiento debería permanecer inalterado, independientemente de los cambios en la definición física de ésta. El modelo relacional es un modelo lógico de datos, y oculta las características de su representación física.

## **Regla 9. Independencia física de los datos**

La independencia lógica de los datos especifica que los programas de aplicación y las actividades de terminal deben ser independientes de la estructura lógica, por lo tanto los cambios en la estructura lógica no deben alterar o modificar estos programas de aplicación. Cuando se modifica el esquema lógico preservando información no es necesario modificar nada en niveles superiores. La independencia de datos lógica es más difícil de lograr que la independencia física de datos.

## **Regla 10. Independencia de integridad**

Todas las restricciones de integridad deben ser definibles en los datos, y almacenables en el catálogo, no en el programa de aplicación.

Las reglas de integridad son:

1. Ningún componente de una clave primaria puede tener valores en blanco o nulos.
2. Para cada valor de clave foránea deberá existir un valor de clave primaria concordante. La combinación de estas reglas asegura que haya Integridad referencial.

El objetivo de las bases de datos no es sólo almacenar los datos, sino también sus relaciones y evitar que estas (limitantes) se codifiquen en los programas. Por tanto en una BDR se deben poder definir limitantes de integridad. Una BDR tiene integridad de entidad. Es decir, toda tabla debe tener una clave primaria. Una BDR tiene integridad referencial. Es decir, toda clave externa no nula debe existir en la relación donde es primaria.

## Regla 11. independencia de distribución

El soporte para bases de datos distribuidas significa que una colección arbitraria de relaciones, bases de datos corriendo en una mezcla de distintas máquinas y distintos sistemas operativos y que esté conectada por una variedad de redes, pueda funcionar como si estuviera disponible como en una única base de datos en una sola máquina.

Esta regla es responsable de tres tipos de transparencia de distribución:

1. Transparencia de localización. El usuario tiene la impresión de que trabaja con una BD local. (aspecto de la regla de independencia física)
2. Transparencia de fragmentación. El usuario no se da cuenta de que la relación con que trabaja está fragmentada. (aspecto de la regla de independencia lógica de datos).
3. Transparencia de replicación. El usuario no se da cuenta de que pueden existir copias (réplicas) de una misma relación en diferentes lugares.

## Regla 12. Regla de la no subversión

Si un sistema relacional tiene un lenguaje de bajo nivel (un registro de cada vez), ese bajo nivel no puede ser usado para saltarse (subvertir) las reglas de integridad y los limitantes expresados en los lenguajes relacionales de más alto nivel (una relación (conjunto de registros) de cada vez).

Es evidente que para la creación de una base de datos lo debemos hacer de una manera lógica y con una secuencia sistemática. El objetivo de las bases de datos no es sólo almacenar los datos, sino también sus relaciones y evitar que estas (limitantes) se codifiquen en los programas. Así mismo evitar que los datos sean repetitivos. A su vez las bases de datos cuentan con niveles, soporte, operaciones de definición de datos, de manipulación de datos, seguridad e integridad y operaciones de administración.

## Referencias

- [1] "Las 12(+1) Leyes de Codd", 2022. [Online]. Disponible en:  
<https://platzi.com/tutoriales/1566-bd/4120-las-12-1-leyes-de-codd/>
- [2] Reglas de codd(modelo relacional). [Online]. Disponible en:  
<https://usacdatospb.files.wordpress.com/2015/09/grupo-4.pdf>.