

## 12 REGLAS DE CODD

El diseño de las bases de datos tradicionalmente se sujeta a una serie de reglas que permiten su diseño y construcción de forma eficiente. Estas reglas son conocidas como “Las 12 Reglas de las Bases de Datos”. Estas reglas fueron propuestas por el mismo creador del lenguaje SQL (Structure Query Language), el Dr. Edgar Frank Codd.

Regla 0: Cualquier sistema que se proclame como relacional, debe ser capaz de gestionar sus bases de datos enteramente mediante sus capacidades relacionales.

Regla 1: La regla de la información, toda la información en la base de datos es representada unidireccionalmente, por valores en posiciones de las columnas dentro de filas de tablas. Toda la información en una base de datos relacional se representa explícitamente en el nivel Lógico exactamente de una manera con valores en tablas.

Regla 2: La regla del acceso garantizado, todos los datos deben ser accesibles sin ambigüedad. Esta regla es esencialmente una nueva exposición del requisito fundamental para las llaves primarias. Dice que cada valor escalar individual en la base de datos debe ser lógicamente direccionable especificando el nombre de la tabla, la columna que lo contiene y la llave primaria.

Regla 3: Tratamiento sistemático de valores nulos, el sistema de gestión de base de datos debe permitir que haya campos nulos. Debe tener una representación de la "información que falta y de la información inaplicable" que es sistemática, distinto de todos los valores regulares.

Regla 4: Catálogo dinámico en línea basado en el modelo relacional, el sistema debe soportar un catálogo en línea, el catálogo relacional debe ser accesible a los usuarios autorizados. Es decir, los usuarios autorizados deben poder tener acceso a la estructura de la base de datos (catálogo).

Regla 5: La regla comprensiva del sublenguaje de los datos, el sistema debe soportar por lo menos un lenguaje relacional que:

Tenga una sintaxis lineal.

Puede ser utilizado de manera interactiva.

Soporte operaciones de definición de datos, operaciones de manipulación de datos (actualización así como la recuperación), seguridad e integridad y operaciones de administración de transacciones.

Regla 6: Regla de actualización, todas las vistas que son teóricamente actualizables deben ser actualizables por el sistema.

Regla 7: Alto nivel de inserción, actualización y borrado, permitiendo el sistema realizar manipulación de datos de alto nivel, es decir, sobre conjuntos de tuplas. Esto significa que los datos no solo se pueden recuperar de una base de datos relacional de filas múltiples y/o de tablas múltiples, sino también pueden realizarse inserciones, actualización y borrados sobre varias tuplas y/o tablas al mismo tiempo (no sólo sobre registros individuales).

Regla 8: Independencia física de los datos, los programas de aplicación y actividades del terminal permanecen inalterados a nivel lógico cuando quiera que se realicen cambios en las representaciones de almacenamiento o métodos de acceso.

Regla 9: Independencia lógica de los datos, los cambios al nivel lógico (tablas, columnas, filas, etc.) no deben requerir un cambio a una solicitud basada en la estructura. La independencia de datos lógica es más difícil de lograr que la independencia física de datos.

Regla 10: Independencia de la integridad, las limitaciones de la integridad se deben especificar por separado de los programas de la aplicación y se almacenan en la base de datos. Debe ser posible cambiar esas limitaciones sin afectar innecesariamente las aplicaciones existentes.

Regla 11: Independencia de la distribución, la distribución de las porciones de la base de datos a las varias localizaciones debe ser invisible a los usuarios de la base de datos. Los usos existentes deben continuar funcionando con éxito:

Cuando una versión distribuida del SGBD se introdujo por primera vez  
Cuando se distribuyen los datos existentes se redistribuyen en todo el sistema.

Regla 12: La regla del orden, si el sistema proporciona una interfaz de bajo nivel de registro, aparte de una interfaz relacional, que esa interfaz de bajo nivel no se pueda utilizar para subvertir el sistema, por ejemplo: sin pasar por seguridad relacional o limitación de integridad. Esto es debido a que existen sistemas anteriormente no relacionales que añadieron una interfaz relacional, pero con la interfaz nativa existe la posibilidad de trabajar no relacionamente.



## REFERENCIA

-Wagner, A. A. (2020, 3 mayo). *Las Doce Reglas de las Bases de Datos*.  
<https://es.linkedin.com/pulse/las-doce-reglas-de-bases-datos-ariel-alejandro-wagner>