

Codd's 12 Rules

by Sudheer Sharma - Thursday, January 15, 2009

<http://dwhnotes.com/data-base/codds-12-rules>

What do you think of this post?

[Awesome \(3\)](#) [Interesting \(2\)](#) [Useful \(1\)](#)

Codd's 12 rules are a set of thirteen rules (numbered zero to twelve) proposed by Edgar F. Codd, a pioneer of the relational model for databases, designed to define what is required from a database management system in order for it to be considered *relational*, i.e., an RDBMS.

Rule 1: The information Rule.

“All information in a relational data base is represented explicitly at the logical level and in exactly one way – by values in tables.”

Everything within the database exists in tables and is accessed via table access routines.

Rule 2: Guaranteed access Rule.

“Each and every datum (atomic value) in a relational data base is guaranteed to be logically accessible by resorting to a combination of table name, primary key value and column name.”

To access any data-item you specify which column within which table it exists, there is no reading of characters 10 to 20 of a 255 byte string.

Rule 3: Systematic treatment of null values.

“Null values (distinct from the empty character string or a string of blank characters and distinct from zero or any other number) are supported in fully relational DBMS for representing missing information and inapplicable information in a systematic way, independent of data type.”

If data does not exist or does not apply then a value of NULL is applied, this is understood by the RDBMS as meaning non-applicable data.

Rule 4: Dynamic on-line catalog based on the relational model.

“The data base description is represented at the logical level in the same way as-ordinary data, so that authorized users can apply the same relational language to its interrogation as they apply to the regular data.”

The Data Dictionary is held within the RDBMS, thus there is no-need for off-line volumes to tell you the structure of the database.

Rule 5: Comprehensive data sub-language Rule.

“A relational system may support several languages and various modes of terminal use (for example, the fill-in-the-blanks mode). However, there must be at least one language whose statements are expressible, per some well-defined syntax, as character strings and that is comprehensive in supporting all the following items

- Data Definition
- View Definition
- Data Manipulation (Interactive and by program).
- Integrity Constraints
- Authorization.

Every RDBMS should provide a language to allow the user to query the contents of the RDBMS and also manipulate the contents of the RDBMS.

Rule 6: View updating Rule

“All views that are theoretically updatable are also updatable by the system.”

Not only can the user modify data, but so can the RDBMS when the user is not logged-in.

Rule 7: High-level insert, update and delete.

“The capability of handling a base relation or a derived relation as a single operand applies not only to the retrieval of data but also to the insertion, update and deletion of data.”

The user should be able to modify several tables by modifying the view to which they act as base tables.

Rule 8: Physical data independence.

“Application programs and terminal activities remain logically unimpaired whenever any changes are made in either storage representations or access methods.”

The user should not be aware of where or upon which media data-files are stored

Rule 9: Logical data independence.

“Application programs and terminal activities remain logically unimpaired when information-preserving

changes of any kind that theoretically permit un-impairment are made to the base tables.”

User programs and the user should not be aware of any changes to the structure of the tables (such as the addition of extra columns).

Rule 10: Integrity independence.

“Integrity constraints specific to a particular relational data base must be definable in the relational data sub-language and storable in the catalog, not in the application programs.”

If a column only accepts certain values, then it is the RDBMS which enforces these constraints and not the user program, this means that an invalid value can never be entered into this column, whilst if the constraints were enforced via programs there is always a chance that a buggy program might allow incorrect values into the system.

Rule 11: Distribution independence.

“A relational DBMS has distribution independence.”

The RDBMS may spread across more than one system and across several networks; however to the end-user the tables should appear no different to those that are local.

Rule 12: Non-subversion Rule.

“If a relational system has a low-level (single-record-at-a-time) language, that low level cannot be used to subvert or bypass the integrity Rules and constraints expressed in the higher level relational language (multiple-records-at-a-time).”

The RDBMS should prevent users from accessing the data without going through the Oracle data-read functions.

What do you think of this post?

[Awesome \(3\)](#) [Interesting \(2\)](#) [Useful \(1\)](#)