

# Constraint

A constraint is a [database](#) examination, which ensures data consistency in the [tables](#) and among each other.

The constraint determines the range of acceptable values for a [column](#) (or columns) or [data set](#) in a database or application. This constraint can be executed automatically and so ensures that [data](#) contents are kept consistent by testing them as they are input.

A constraint can be specified for each [column](#) (or columns) in a table, to guarantee the mechanism described above. Constraints can be [domain-](#) or column-based and the specified conditions must be met when new [data sets](#) are inserted, or existing data sets are modified. They are used to verify data integrity. If a condition is not met, an [exception](#) is raised.

Firebird/InterBase® internally generates a [trigger](#) for each check condition. Constraints can be defined as follows:

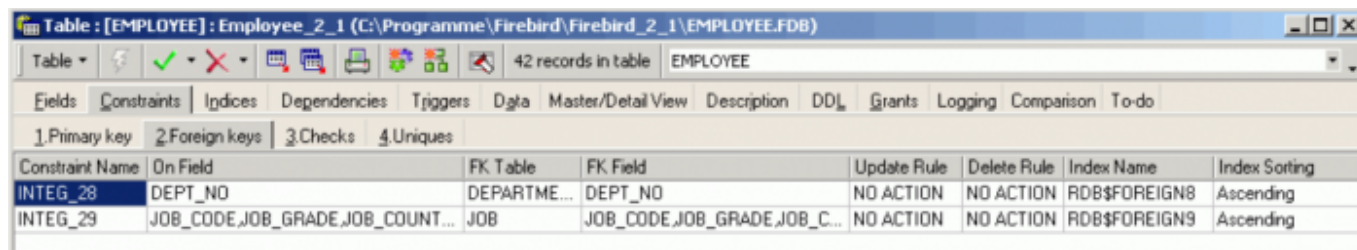
1. **Primary Key/Unique:** Specification of the [unique](#) option forces a unique entry in this [column](#) (these columns) for each data set (i.e. duplicate [field](#) entries are not allowed).



Constraint Name	On Field	Index Name	Index Sorting
INTEG_27	EMP_NO	RDB\$PRIMARY7	Ascending

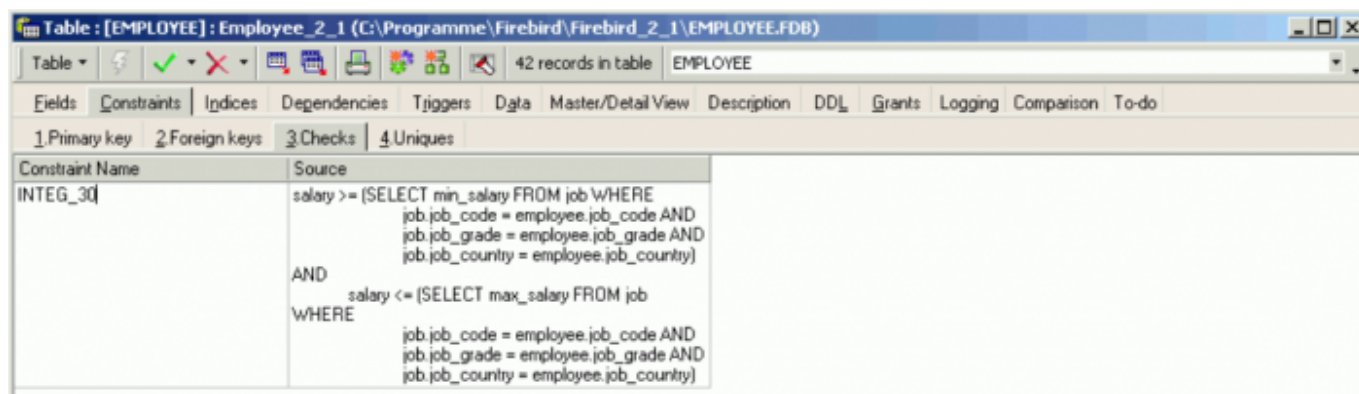
Since Firebird 1.5, in compliance with the SQL-99 standard, [NULLs](#) – even multiple – are now allowed in columns with a [UNIQUE](#) constraint. It is therefore possible to define a [UNIQUE key](#) on a [column](#) that has no [NOT NULL](#) constraint. Please refer to the Firebird 2.0 Language Reference Update chapter, [UNIQUE constraints now allow NULLs](#).

2. **Foreign Key:** The foreign key option determines that the column(s) is/are linked by a [referential integrity](#) relationship to the [primary key](#) of another table (i.e. the input data is only accepted if it already exists in the primary key column(s) in the referenced table).



Constraint Name	On Field	FK Table	FK Field	Update Rule	Delete Rule	Index Name	Index Sorting
INTEG_28	DEPT_NO	DEPARTME...	DEPT_NO	NO ACTION	NO ACTION	RDB\$FOREIGN8	Ascending
INTEG_29	JOB_CODE, JOB_GRADE, JOB_COUNT...	JOB	JOB_CODE, JOB_GRADE, JOB_C...	NO ACTION	NO ACTION	RDB\$FOREIGN9	Ascending

3. **CHECK:** the check option enables each data set to be examined for validation of an [expression](#) specified in brackets. [Check constraints](#) in tables are identical to check constraints in domains.



Constraint Name	Source
INTEG_30	<pre>salary &gt;= (SELECT min_salary FROM job WHERE             job.job_code = employee.job_code AND             job.job_grade = employee.job_grade AND             job.job_country = employee.job_country) AND salary &lt;= (SELECT max_salary FROM job             WHERE             job.job_code = employee.job_code AND             job.job_grade = employee.job_grade AND             job.job_country = employee.job_country)</pre>

Only one constraint is permitted per column. If the column including a constraint is based on a domain also containing a constraint, both constraints are active.

The specification of the keyword `CONSTRAINT` and the name are optional for all constraints. If no name is specified, Firebird/InterBase® generates a name automatically. All constraint names are stored in a system table called `DB$RELATION_CONSTRAINTS`.

Since version 1.5 Firebird allows a `USING INDEX` subclause to be placed at the end of a [primary](#), [unique](#) or [foreign key](#) definition. Please refer to the *Firebird 2.0 Language Reference Update* chapter, [USING INDEX subclause](#) for further information.

It is only necessary to name constraints if they are to be deactivated at a later date using the `ALTER TABLE DROP` statement.

From InterBase® 5 onwards, [cascading referential integrity](#) is also supported.

From:  
<http://ibexpert.com/docu/> - **IBExpert**

Permanent link:  
<http://ibexpert.com/docu/doku.php?id=01-documentation:01-13-miscellaneous:glossary:constraint>

Last update: **2023/08/14 10:45**

