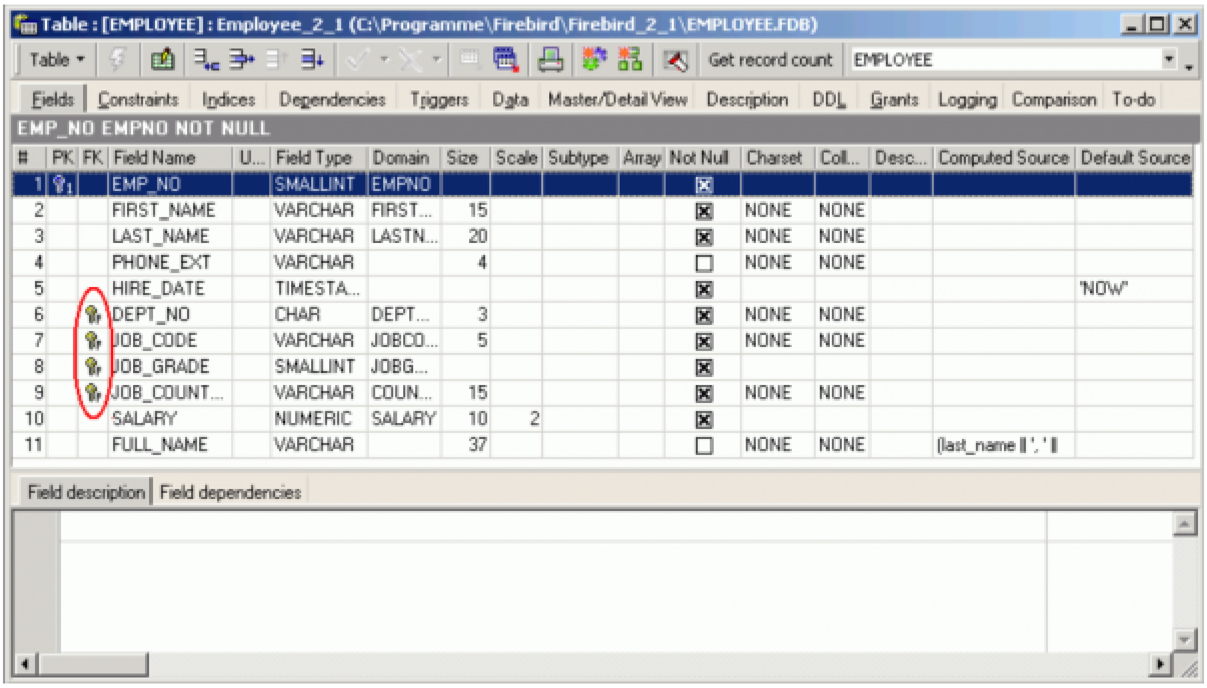
Foreign key

A foreign key is composed of one or more columns that reference a primary key. *Reference* means here that when a value is entered in a foreign key, Firebird/InterBase® checks that the value also exists in the referenced primary key. This is used to maintain domain integrity.

A foreign key is vital for defining relationships in the database. It can be specified in the IBExpert Table Editor (started from the DB Explorer) on the Constraints page.



EMP_NO EMPNO NOT NULL

#	PK	FK	Field Name	U...	Field Type	Domain	Size	Scale	Subtype	Array	Not Null	Charset	Coll...	Desc...	Computed Source	Default Source
1	🔑1		EMP_NO		SMALLINT	EMPNO					☒					
2			FIRST_NAME		VARCHAR	FIRST...	15				☒	NONE	NONE			
3			LAST_NAME		VARCHAR	LASTN...	20				☒	NONE	NONE			
4			PHONE_EXT		VARCHAR		4				☐	NONE	NONE			
5			HIRE_DATE		TIMESTA...						☒					'NOW'
6		🔑	DEPT_NO		CHAR	DEPT...	3				☒	NONE	NONE			
7		🔑	JOB_CODE		VARCHAR	JOBCO...	5				☒	NONE	NONE			
8		🔑	JOB_GRADE		SMALLINT	JOBG...					☒					
9		🔑	JOB_COUNT...		VARCHAR	COUN...	15				☒	NONE	NONE			
10			SALARY		NUMERIC	SALARY	10	2			☒					
11			FULL_NAME		VARCHAR		37				☐	NONE	NONE		(last_name || ', ' ||	

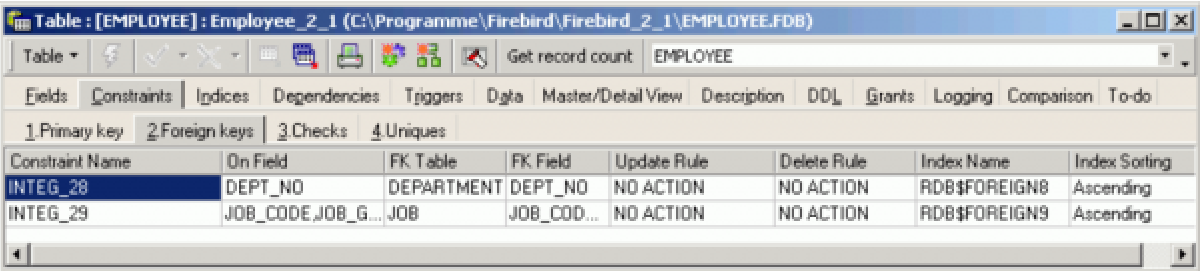
Foreign keys are used mainly for so-called reference tables. In a table storing, for example, employees, it needs to be determined which department each employee belongs to. Possible entries for the department number of each EMPLOYEE data set are contained in the DEPARTMENT table. As the EMPLOYEE table refers to the DEPT_NO as the primary key for the DEPARTMENT table, there is a foreign key relationship between the EMPLOYEE table and the DEPARTMENT table. Foreign key relationships are automatically checked in Firebird/InterBase®, and data sets with a non-existent department number cannot be saved.

When a primary key:foreign key relationship links to a single row in another table, what is known as a virtual row is created. The columns in that second table provide additional description about the primary key of the first table. This is also known as a 1:1 relationship.

A foreign key can also point to itself. Firebird enables you to reference recursive data and even represent tree structures in this way.

Foreign keys and their system names can be defined and viewed on the IBExpert Table Editor / Constraints page.

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.



Constraint Name	On Field	FK Table	FK Field	Update Rule	Delete Rule	Index Name	Index Sorting
INTEG_28	DEPT_NO	DEPARTMENT	DEPT_NO	NO ACTION	NO ACTION	RDB$FOREIGN8	Ascending
INTEG_29	JOB_CODE,JOB_G...	JOB	JOB_COD...	NO ACTION	NO ACTION	RDB$FOREIGN9	Ascending

A primary key does not have to reference a foreign key. However a unique index is insufficient; a unique constraint needs to be defined (this definition also causes a unique index to be automatically generated).

When defining a foreign key, it is necessary to specify update and delete rules. Please refer to Referential integrity and Cascading referential integrity for further information.

SQL syntax:

```
ALTER TABLE MASTER
ADD CONSTRAINT UNQ_MASTER UNIQUE (FIELD_FOR_FK);
```

Foreign key names are limited to 32 characters up until InterBase® 6 and Firebird 1.5; InterBase® 7 allows 64 characters. IBExpert therefore recommends limiting table names to 14 characters, so that the foreign key name can include both related table names: prefix FK plus two separators plus both table names, e.g. FK_Table1_Table2.

Please note however that this is not an Firebird/InterBase® restriction, but purely an IBExpert recommendation to enable a clear and logical naming convention for foreign keys.

Note: if data has already been input in a table which is to subsequently be assigned a foreign key, this will not be allowed by Firebird/InterBase®, as it violates the principle of referential integrity. It is however possible to filter and delete the old data (where no reference to a primary key has been made) using a SELECT statement and committing. It is important to then disconnect and reconnect the database in IBExpert, for this to work.

New to Firebird 2.0: Creating foreign key constraints no longer requires exclusive access - Now it is possible to create foreign key constraints without needing to get an exclusive lock on the whole database.

Should you wish to delete constraints defined for a unique, foreign or secondary key, use the IBExpert Table Editor. Alternatively you can find a list of all constraints specified in a database in the system table, RDB$RELATION_NAME.

back to top of page