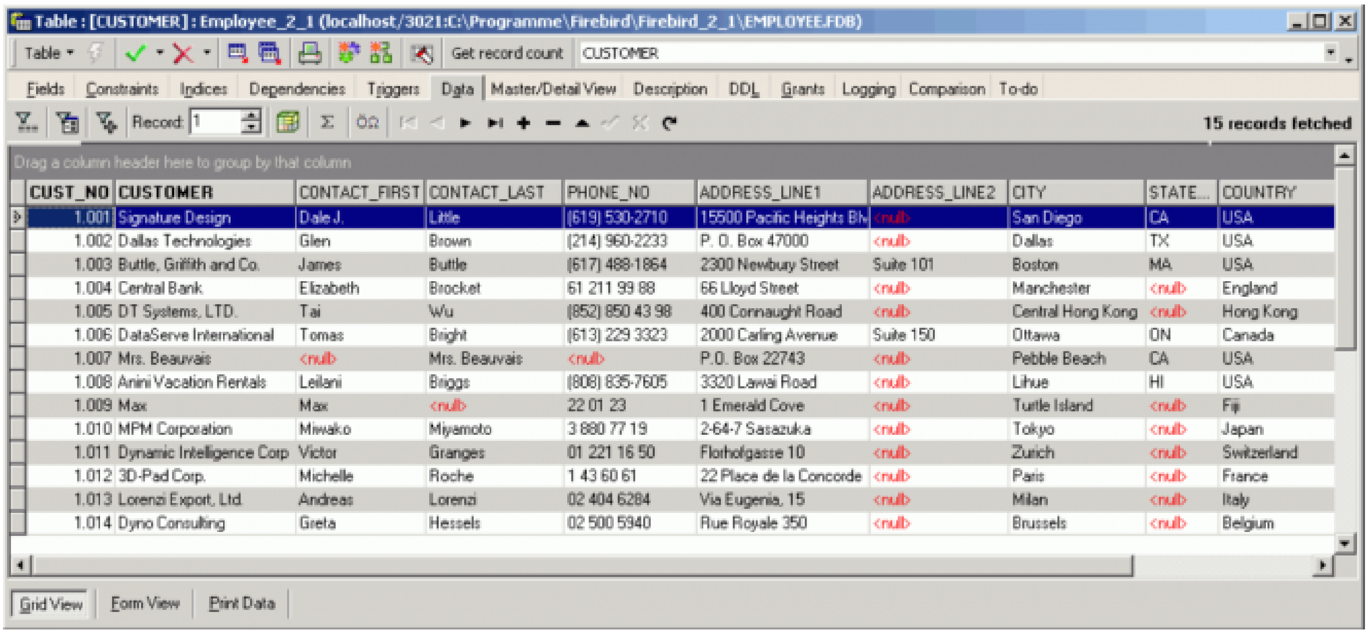
NULL

NULL is the term used to describe a data field without a value, i.e. the field has been left blank because the information is either not known or not relevant for this record/data set. The NULL value can be stored in text, numeric and date data types.

A relational database is able to store NULL values as data content. A NULL value does not mean numerical zero. For example, a product can have zero sales (0) or unknown sales (<null>), and just because a customer's telephone number is not known (<null>), this does not mean that the customer has no telephone, and he most certainly will not have the telephone number “0”!



CUST_NO	CUSTOMER	CONTACT_FIRST	CONTACT_LAST	PHONE_NO	ADDRESS_LINE1	ADDRESS_LINE2	CITY	STATE...	COUNTRY
1.001	Signature Design	Dale J.	Little	(619) 530-2710	15500 Pacific Heights Blv	<null>	San Diego	CA	USA
1.002	Dallas Technologies	Glen	Brown	(214) 960-2233	P. O. Box 47000	<null>	Dallas	TX	USA
1.003	Buttle, Griffith and Co.	James	Buttle	(617) 488-1864	2300 Newbury Street	Suite 101	Boston	MA	USA
1.004	Central Bank	Elizabeth	Brocket	61 211 99 88	66 Lloyd Street	<null>	Manchester	<null>	England
1.005	DT Systems, LTD.	Tai	Wu	(852) 850 43 98	400 Connaught Road	<null>	Central Hong Kong	<null>	Hong Kong
1.006	DataServe International	Tomas	Bright	(613) 229 3323	2000 Carling Avenue	Suite 150	Ottawa	ON	Canada
1.007	Mrs. Beauvais	<null>	Mrs. Beauvais	<null>	P.O. Box 22743	<null>	Pebble Beach	CA	USA
1.008	Anini Vacation Rentals	Leilani	Briggs	(808) 835-7605	3320 Lawai Road	<null>	Lihue	HI	USA
1.009	Max	Max	<null>	22 01 23	1 Emerald Cove	<null>	Turtle Island	<null>	Fiji
1.010	MPM Corporation	Miwako	Miyamoto	3 880 77 19	2-64-7 Sasazuka	<null>	Tokyo	<null>	Japan
1.011	Dynamic Intelligence Corp	Victor	Granges	01 221 16 50	Florhofgasse 10	<null>	Zurich	<null>	Switzerland
1.012	3D-Pad Corp.	Michelle	Roche	1 43 60 61	22 Place de la Concorde	<null>	Paris	<null>	France
1.013	Lorenzi Export, Ltd.	Andreas	Lorenzi	02 404 6284	Via Eugenia, 15	<null>	Milan	<null>	Italy
1.014	Dyno Consulting	Greta	Hessels	02 500 5940	Rue Royale 350	<null>	Brussels	<null>	Belgium

A NULL value can occur for the following reasons:

- The value is not yet known, but will be added at a future date.
- The value is not yet available for some reason, e.g. the date of receipt of payment.
- The value is not important, e.g. the credit card expiry date of someone who has paid cash.

Firebird/InterBase® does not use a special byte sequence to indicate a NULL, but administrates this information internally. NULL values can influence query contents considerably, for example, when a column average is calculated. The values filled by the NULL value, i.e. empty fields, are not taken into consideration. A field containing the value 0 is included in the calculation of the average.

Examples from the Firebird 1.5 Quick Start Guide:

- 1 + 2 + 3 + NULL = NULL
- not (NULL) = NULL
- 'Home ' || 'sweet ' || NULL = NULL

```
if (a = b) then
   MyVariable = 'Equal';
else
   MyVariable = 'Not equal';
```

After executing this code, MyVariable will be Not equal if both a and b are NULL. The reason is that the expression a = b yields NULL if at least one of them is NULL. In an if…then context, NULL behaves like FALSE. So the then block is skipped, and the else block executed.

```
if (a <> b) then
   MyVariable = 'Not equal';
else
   MyVariable = 'Equal';
```

Here, MyVariable will be Equal if a is NULL and b isn't, or vice versa. The explanation is analogous to that of the previous example.

```
FirstName || ' ' || LastName
```

will return NULL if either FirstName or LastName is NULL.

Think of NULL as UNKNOWN and all these strange results suddenly start to make sense! If the value of Number is unknown, the outcome of 1 + 2 + 3 + Number is also unknown (and therefore NULL). If the content of MyString is unknown, then so is MyString || YourString (even if YourString is non-NULL). Etcetera.

New to Firebird 2.0: NULLs are now "lowest" for SORTS

NULL is now treated as the lowest possible value for ordering purposes and sets ordered on nullable criteria are sorted accordingly. Thus: .

- for ascending sorts NULLs are placed at the beginning of the result set,
- for descending sorts NULLs are placed at the end of the result set.

Important: In former versions, NULLs were always at the end. If you have client code or PSQL definitions that rely on the legacy NULLs placement, it will be necessary to use the NULLS LAST option in your ORDER BY clauses for ascending sorts.

Please also refer to the *Firebird 2.0.4*. Release Notes for further information regarding Enhancements to NULL logic in Firebird 2.