

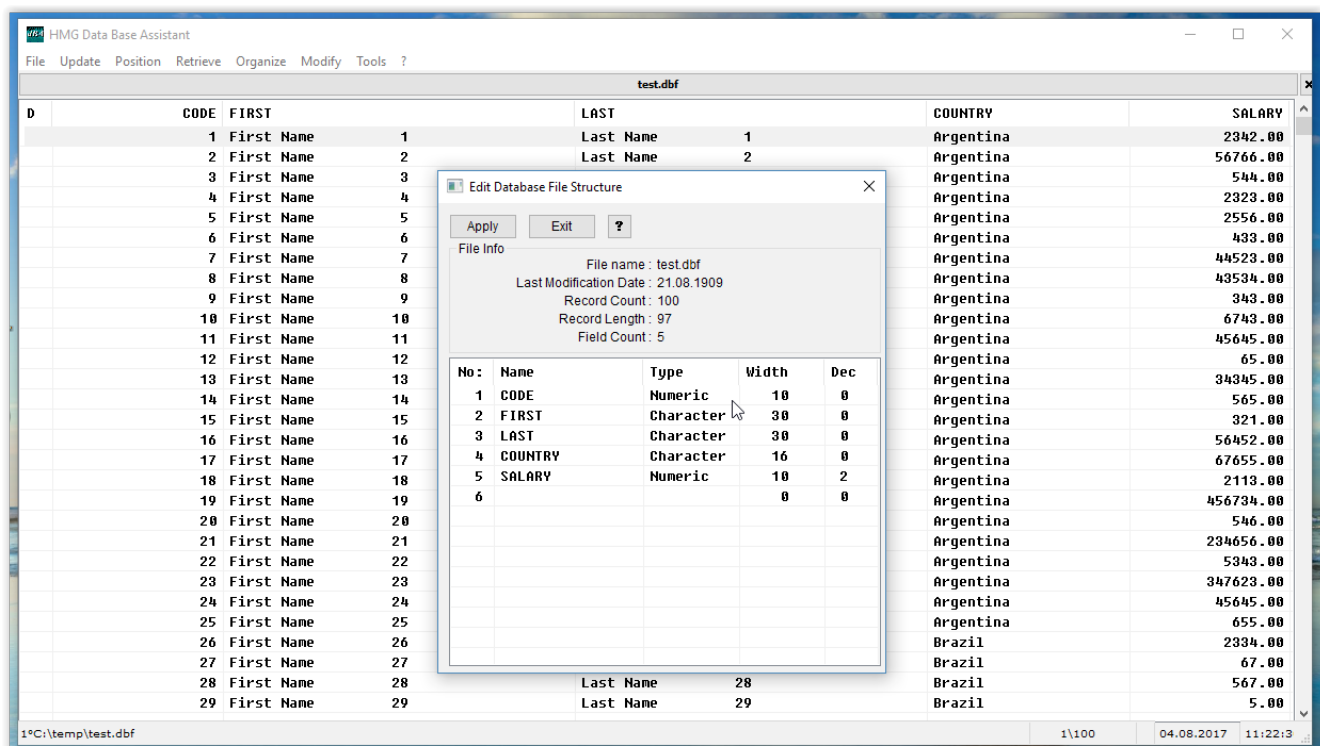
What is the table?

What is the table? Data must be stored somewhere, users of this forum are well aware of the DBF and its capabilities. Ordinary files, which can be easily copied, renamed or deleted, viewing content from your program or a dbf manager. You can create and use indexes, but these are only files with the extension ndx, ntx, cdx and the like. But you do not have to use them, so it's easy to forget to establish a connection and dbf and the indexes become obscure.

MySQL is a relational database, the function is to store data and to make them available on request. The table is one of the database objects where the data is stored. It consists of rows and columns. As a rule, each table should have a primary key. Each column is defined which is the type and length, the type of field can be a number, a string, a date, and a blob (something like a memo). The field value can be something or NULL (no value). Accuracy (the number of digits to be stored) depends on the base. The databases today support UTF8 records so that you are not limited by the character set.

Example NULL values: you have a table designed to record temperature measurements in a place, what will you type if it is not measured? The NULL value remains and you will then recognize it as "not measured".

What does DBF look like? Let's look at the TEST.DBF example from one example



The screenshot shows the HMG Data Base Assistant window. The main window displays the structure of the 'test.dbf' file. A dialog box titled 'Edit Database File Structure' is open, showing the file's metadata and a table of field definitions.

File Info:

- File name: test.dbf
- Last Modification Date: 21.08.1909
- Record Count: 100
- Record Length: 97
- Field Count: 5

Field Definitions:

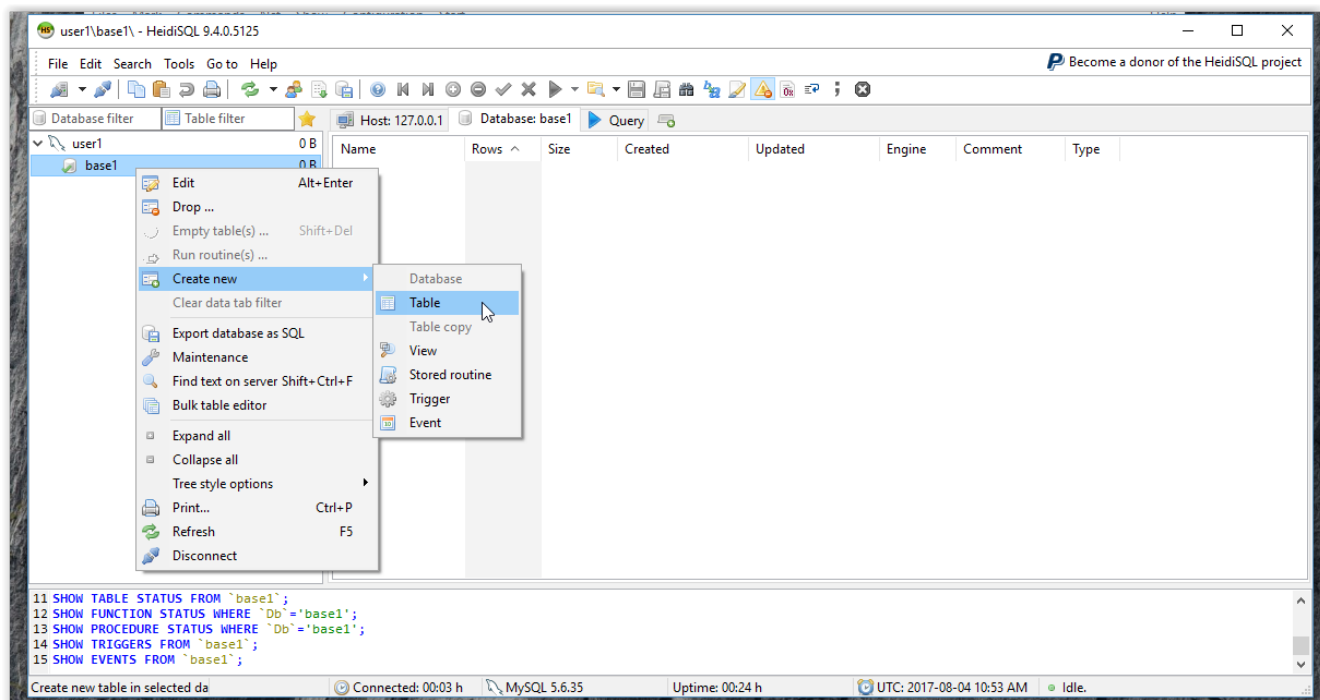
No:	Name	Type	Width	Dec
1	CODE	Numeric	10	0
2	FIRST	Character	30	0
3	LAST	Character	30	0
4	COUNTRY	Character	16	0
5	SALARY	Numeric	10	2
6			0	0

The background window shows the data table with columns: D, CODE, FIRST, LAST, COUNTRY, and SALARY. The data includes records for Argentina and Brazil with various salary values.

It's structured
CODE numerical length 10
FIRST textual length 30
LAST textual length 30
COUNTRY text length 16
SALARY numerical length 10 with 2 decimals

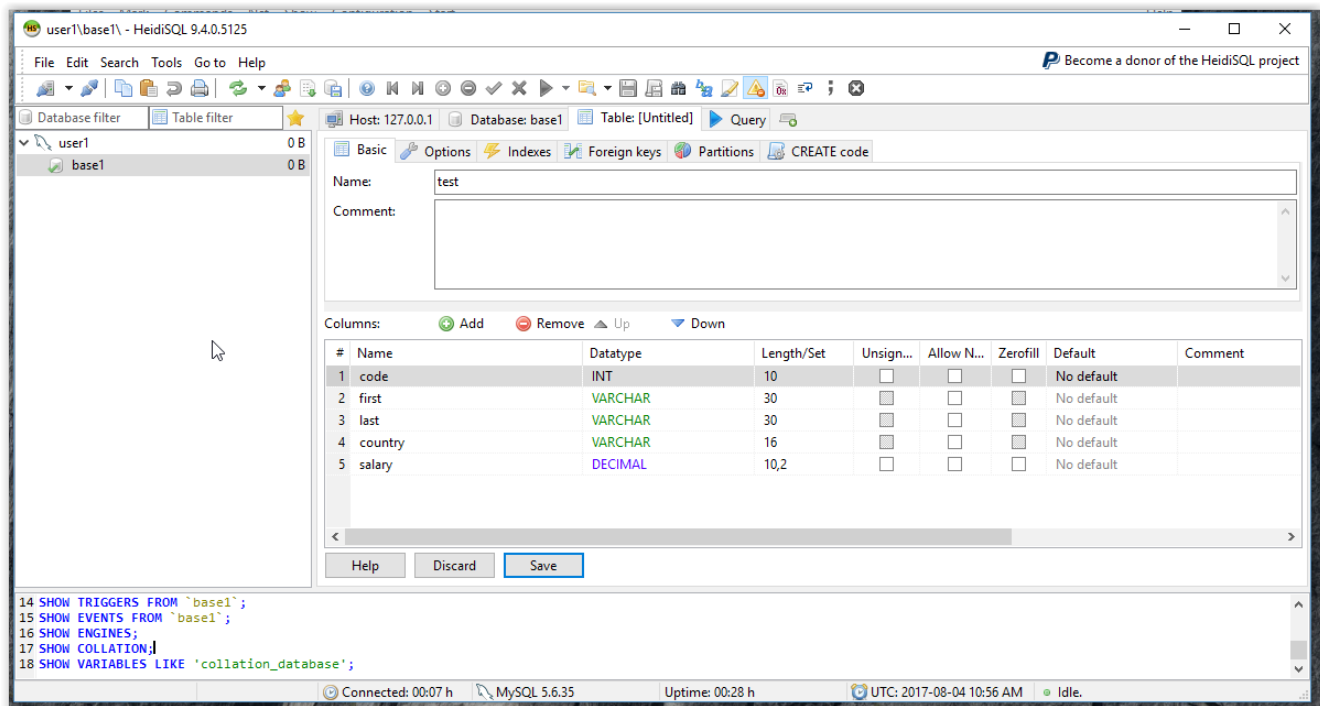
Create a table

Sign up via HeidiSQL



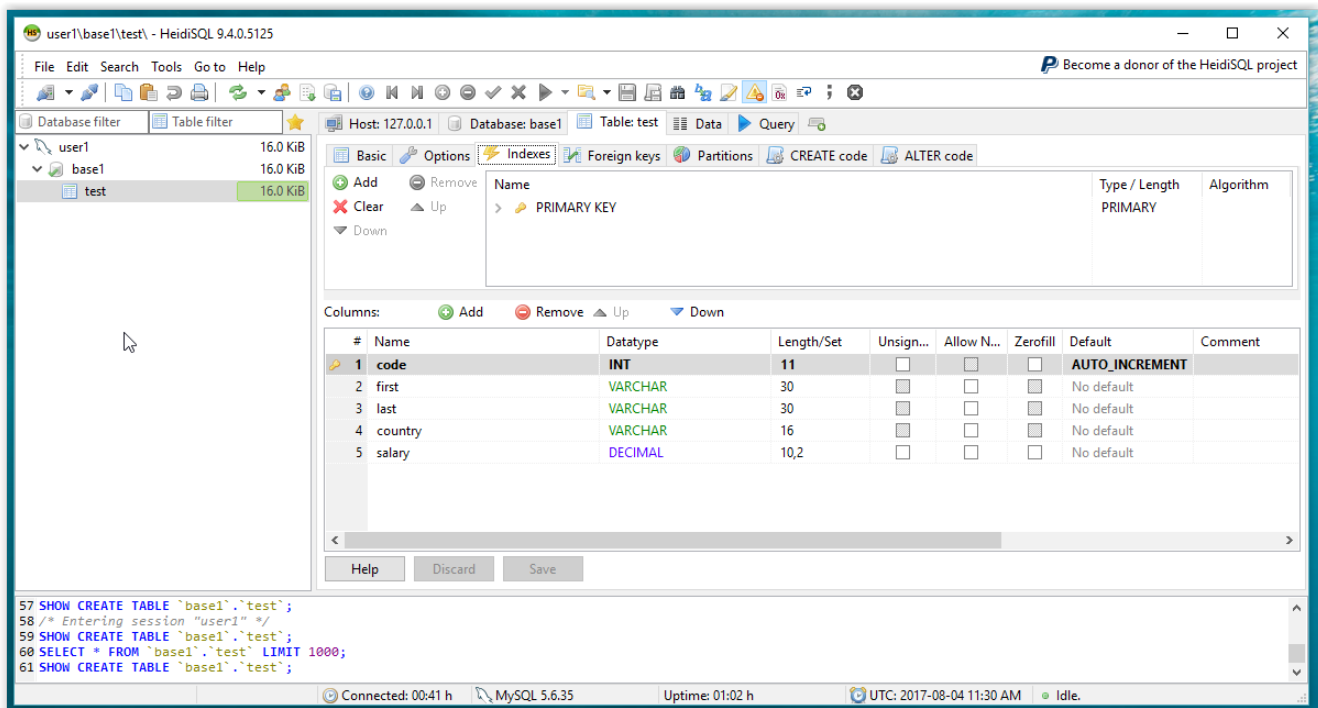
Right-click the database, you will get a menu and click Create new → Table. Enter the name of the table and then you can define the field. The Add button is for creating a new field

You need to enter the field name and select Datatype, then Length and can do something else. Clicking on the Default field opens the option to define a value, you can specify that the value of the field AUTO_INCREMENT, but you define that column in the Index tab as PRIMARY KEY.

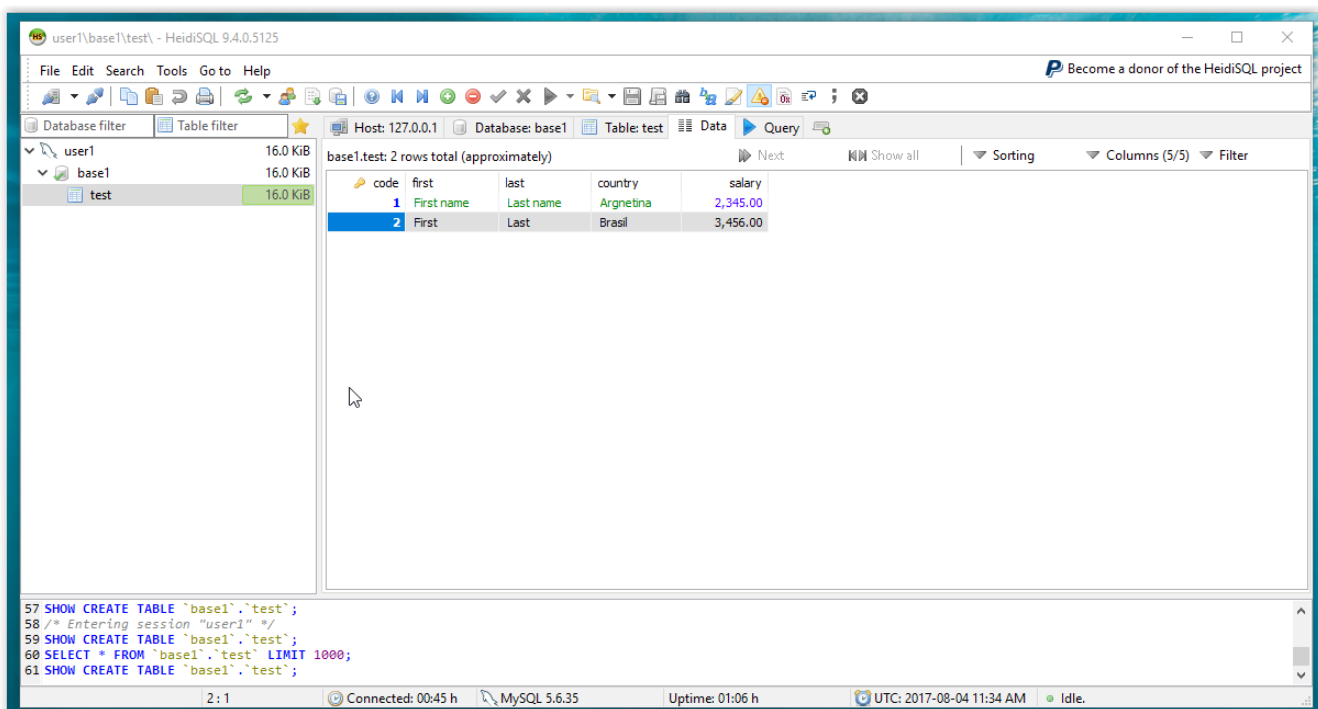


If everything is as a rule by clicking the Save button, the table will be created. That's tab Basic. You have Indexes tab to define Indexes and Foreign keys to define External Keys. See the CREATE code tab and you will see the SQL command for what you did with a few clicks. From now on, everything will work through SQL queries.

Tabs above are for server, database, data, and query creation. Select the Data tab and you will see the data (now there are no because the table is just created). By clicking the (+) button, one line will become colored and you will be able to enter the data.



Once you enter the data, they will be in the database and will be shown as in the picture



Now that database data can be tried by SQL, the Query tab serves for this. In this interface you need to enter the SQL command "select * from test" and click on the button ► (Execute SQL) or F9 and we will get

As you see you have multiple windows, the command above is in the top, in the upper right you have help (you need to click on the content and then open it), and the bottom is the query result

Command **SELECT * FROM test;** Is the simplest form of the SELECT command and means "show all fields from the test table"

