

Gramps 4.2 Wiki Manual - Manage Family Trees: CSV Import and Export

 gramps-project.org/wiki/index.php/Gramps_4.2_Wiki_Manual_-_Manage_Family_Trees:_CSV_Import_and_Export

[Previous](#) [Index](#) [Next](#)

 **Languages:** [English](#) • [Deutsch](#) • [français](#) • [Nederlands](#) • [polski](#)

Gramps Spreadsheet Import/Export

This format allows you to import/export a spreadsheet of data all at once. The spreadsheet must be in the Comma Separated Value (CSV) format. Most spreadsheet programs can read and write this format. It is also easy to write by hand. This is the only Gramps import format which allows for merging with existing data.



This spreadsheet format does not allow for 100% export of Gramps data.

It only exports (and imports) a subset of data, namely: people (names, gender, birth, baptism, death, and burial dates/places/sources); marriages (dates/places/sources); relationships (parents and children); and places (title, name, type, latitude, longitude, code, enclosed by, and enclosed date). Notes are not exported, but new notes are appended onto the end of existing notes.



New feature since Gramps 3.3

Previously, sources were not exported, but now they are. Sources are referred to by their title text. You can add further details to a source after importing.



New feature since Gramps 4.2.1

Now you can export and import place information. Also, you can refer to places in marriage and person sections by using a "place id". See below for more details.

There are three main uses for this format:

1. You can export your core Gramps data into a spreadsheet format, edit it with a text or spreadsheet program, and import the changes and additions back into Gramps. This is handy for sending to others to fill in, or for taking on the road when you don't have your full Gramps application.
2. You can import new data into your Gramps database. For example, if you have a set of new people to add to your database, but don't want to hunt and peck your way to finding where they go, you might find it easier to type them into a spreadsheet, and then quickly bring all of them in at once. This is handy if you have a large amount of data that you are cutting and pasting from another application or the web. An example of this is [restoring your Gramps database](#) by loading the Narrative Website into a spreadsheet.
3. You can also import a set of corrections and additions. Say that you have printed out a report, and you are going through it marking corrections. If you make each correction a section of a spreadsheet, you can "script the edits" and then execute them all at once.

Export

To export your database:

1. Start Gramps
2. Select from the menu **Family Trees ->Export...**
3. Select **Next** on the **Saving your data** window.
4. Select **Comma Separated Values Spreadsheet (CSV)** on the **Choose the output format** window
5. On the **Export options** window.
 1. In the top section select which filters to apply to your family tree
 2. From the checkboxes select which items to include in the export (people, marriages, children, places) and whether to Translate headers into the language you are using.

A selected set of fields of your genealogy data will be saved to a .csv file in the format described below. In addition, the people and families are referenced so that the data can be edited and read back in, thereby updating the database.

There are some columns that will be blank, specifically note and source columns. These are listed in the spreadsheet so that you can make notes for the import, but notes are never exported with this tool.



From Gramps 3.3

you can now export source titles; previously no source data was exported.

Your data is broken up into four sections representing places, individuals, marriages, and children. The exported fields and column names are:

Places

Place, Title, Name, Type, Latitude, Longitude, Code, Enclosed_by, Date

Individuals

Person, Lastname, Firstname, Callname, Suffix, Prefix, Title, Gender, Birthdate, Birthplace (or Birthplaceid), Birthsource, Baptismdate, Baptismplace (or Baptismplaceid), Baptismsource, Deathdate, Deathplace (or Deathplaceid), Deathsource, Burialdate, Burialplace (or Burialplaceid), Burialsource, Note

Marriages

Marriage, Husband, Wife, Date, Place (or Placeid), Source, Note

Families

Family, Child

The first column in each area is the Gramps ID. That is what will tie your edits back to the correct data, so don't alter those data. Load this file into your favorite spreadsheet using comma separated, double-quote text delimited, and Text format (any encoding for now). Then you can add or correct data, and save it back out, keeping the same format. You can then import the data back on top of your old data and it will be corrected.

 **LibreOffice allows you to turn off auto-formatting when you open the CSV file.**

If you don't do this, LibreOffice may interpret the dates incorrectly. Change the type of the column to Text rather than Standard. If your spreadsheet program doesn't allow you to format the fields before you get it into columns you need to change the display format of dates in Gramps before you export. You can do this under **Edit -> Preferences -> Display -> Date Format**

Excel allows you to format columns as text when you open the CSV file.

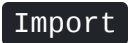
If you don't do this, Excel may interpret the dates incorrectly. Change the type of the column to Text rather than General. One way to do this is to open the CSV from the file menu (select 'Text files' as the type in the file open dialog box). This brings up the 'Text Import Wizard' which allows you to select 'Delimited' by commas, and to select 'Text' for all the columns (select the first, scroll to last and shift-click to select all).

Excel cannot directly save Unicode CSV files.

To save a CSV in Unicode, save to "Unicode Text (*.txt)", then open the file in Notepad++. Using Notepad++ 'Search' and replace menu, change all the tabs ('\t') to commas (','),. Using Notepad++ 'Encoding' menu convert to "UTF-8-BOM" and save the file to CSV (*.csv').

Import

To import your data:

1. Use the file from above, or create a spreadsheet (described below) with genealogical data
2. Start Gramps
3. Create a new Family Tree
4. Select from the menu **Family Trees ->Import...**
5. Select the **Comma Separated Values Spreadsheet (CSV)** file and then select the  button

The merge part of this code will only add or update information to your database, and it always assume that the spreadsheet data is the correct version.

If you load this spreadsheet into LibreOffice, make sure you select each column as type **Text** rather than **Standard**. Standard will reformat your dates and numbers. Also, if you use Excel, you will probably want to select all cells once opened, and change the format of the cells to **Text**.

The spreadsheet is data made up of columns. Each column should have at the top of it the name of what type of data is in the column. You must use special names for the columns. Currently they are:

Place

place - a reference to this place
title - title of place
name - name of place
type - type of place (eg, City, County, State, etc.)
latitude - latitude of place
longitude - longitude of place
code - postal code, etc.
enclosed_by - the reference to another place that encloses this one
date - date that the enclosed_by place was in effect

People

person - a reference to be used for families (marriages, and children)
grampsid - to assign a gramps id to the person
firstname - a person's first name
surname/lastname - a person's last name
callname - a common name (nickname) for the person
prefix - surname prefix (von, de, etc)
suffix - a suffix of a person's name (Jr., Sr.)
title - a person's title (Dr., Mr.)
gender - male or female (you should use the translation for your language)
note - a note for the person's record
birthdate - date of birth
birthplace - place of birth
birthplaceid - place id of birth
birthsource - source title for birth
baptismdate - date of baptism
baptismplace - place of baptism
baptismplaceid - place id of baptism
baptismsource - source title of baptism
deathdate - date of death
deathplace - place of death
deathplaceid - place id of death
deathsource - source title for death
deathcause - cause of death
burialdate - date of burial
burialplace - place of burial
burialplaceid - place id of burial
burialsource - source title of baptism

Marriage

marriage - if you want to reference this from a family, you'll need a matching name here
husband/father/parent1 - the reference of the person above who is the husband
(for female parent1, you'll need to put gender in the person area,
or edit it later in gramps)
wife/mother/parent2 - the reference of the person above who is the wife
(for male parent2, you'll need to put gender in the person area,
or edit it later in gramps)
date - the date of the marriage
place - the place of the marriage
placeid - the place id of the marriage
source - source title of the marriage
note - a note about the marriage/wedding

Family

family - a reference to tie this to a marriage above (required)
child - the reference of the person above who is a child
source - source title of the marriage
note - a note about the family
gender - male or female (you should use the translation for your language)
[You can put gender here, or in person above]

Details

Column names are not case-sensitive. You may use any combination of the columns, in any order. (Actually, you have to at least have a surname and a given name when defining a person, you have to have a marriage and child columns when defining children, and places need a place reference, but that is it.) The column names are the English names given (for now) but the data should be in your language (including the words "male" and "female").

Top-to-bottom order is important in that if you want to reference something in one area to another, the definition **MUST** come first. For example, if you want to define families of people, the individuals must be defined before the families. The same applies to places. So it is usually best to put the Places data first, people next, then marriages and families.

Each of these can go in its own area in a spreadsheet. There is no limit to the number of areas in a sheet, and each area can have any number of rows. Leave a blank row between "areas". Just make sure that areas are not next to each other; they must be above and below one another.

You can have multiple areas of each kind on a spreadsheet. The only limitation is that if you refer to a person, you must do that in a row lower than where that person is described. Likewise, if you refer to a marriage, you must do that in a row lower than where the marriage is described. References to enclosed_by places must already exist in the database, or be defined in rows above in the spreadsheet.

If you use the 'grampsid' as a way to assign specific ids, be *very* careful when importing to a current database. Any data you enter will **overwrite** the data assigned to that grampsid. If you use ids in the place, person, marriage, or family columns that are surrounded by brackets (for example '[I0001]'), the values you use will be interpreted as grampsids as well. If you are adding **new** items, you are encouraged to avoid use of the bracket format or grampsid columns, so as to avoid accidentally overwriting your data. If you are mixing the bracket (or grampsid) methods with plain references (no brackets), put the plain referenced data after the bracket referenced data.

If you are entering the data in a text file, and if you wish to have a comma inside one of the values, like "Clinton, Co., MO" then you need place the entire value in double-quotes and put the first double-quote right after the preceding comma. For example:

```
marriage, parent1, parent2, place
m1, p1, p2, "Clinton, Co., MO"
m2, p3, p4, "Havertown, PA"
```

A spreadsheet program will do this automatically for you.

Here is an example spreadsheet in LibreOffice, but any spreadsheet program should work.

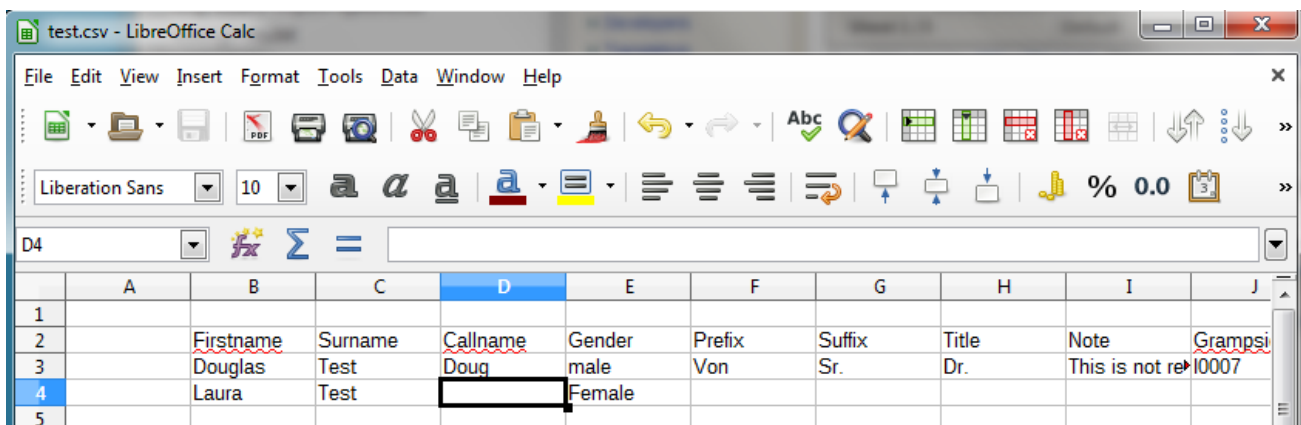


Fig. 5.1.1

Notice that the data need not begin in the first column, nor in the first row.

And here is the resulting data in Gramps:

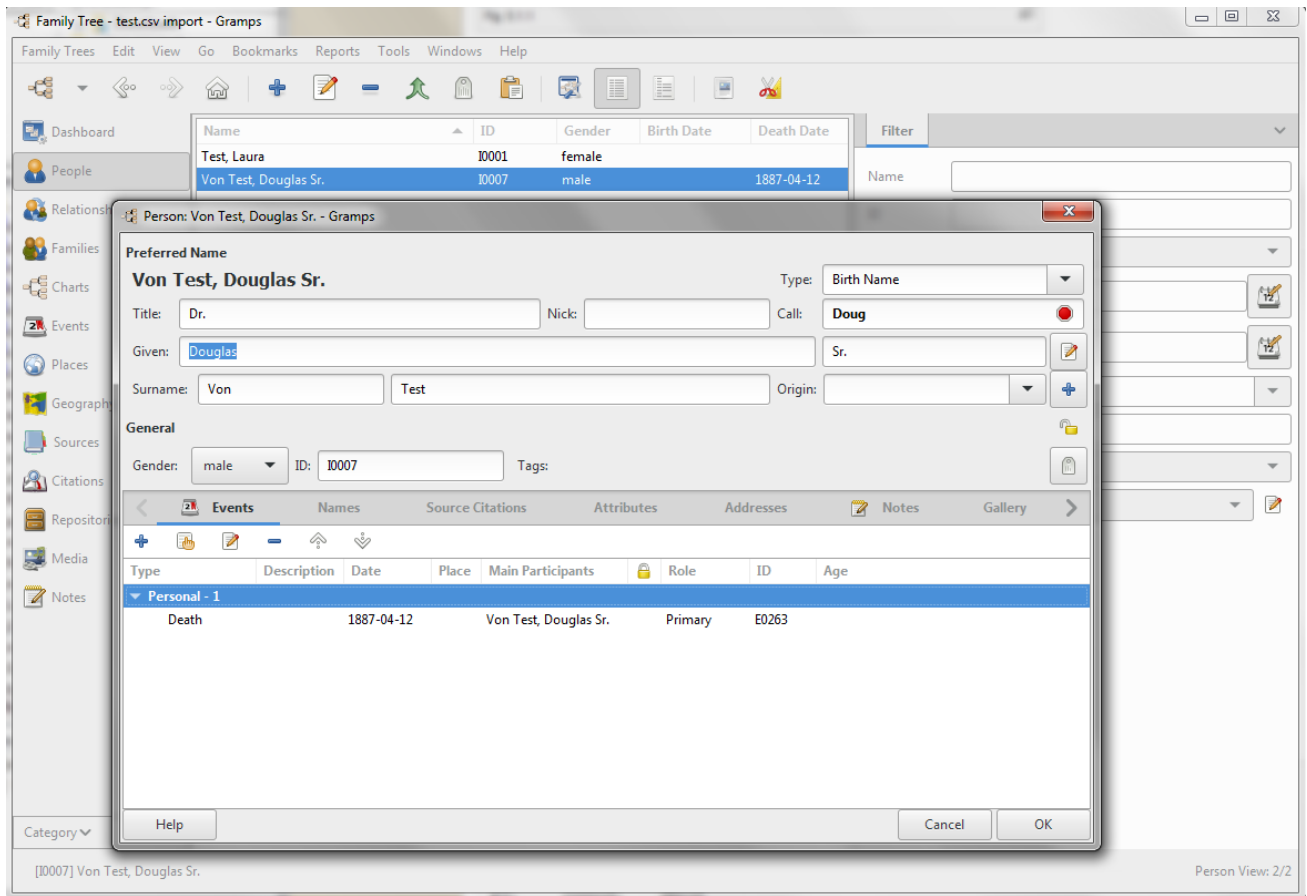


Fig. 5.1.2

Example CSV with multiple areas

Here is an example of a CSV text spreadsheet with multiple areas:


```
Place, Title, Name, Type
[P0001], Michigan, Michigan, State
L1, Canada, Canada, Country
L2, USA, USA, Country
```

```
Firstname, Surname, Birthdate, Birth place id
John,      Tester,  11/11/1965, L1
Sally,     Tester,  01/26/1973, L1
```

```
Person, Firstname, Surname, Birthdate, Birth place id
p1,      Tom,      Smith,  22 Jan, 1970, [P0001]
p2,      Mary,     Jones
p3,      Jonnie,   Smith
p5,      James,    Loucher
p6,      Penny,    Armbruster
[P0002], Tim,     Sparklet
```

```
Marriage, Husband, Wife
m1,      p1,      p2
m2,      p5,      p6
```

```
Family, Child
m1,      p3
m1,      p6
m2,      [P0002]
```

If you cut and paste that into a file (or use the [Import Gramplet](#)), you can import it directly.

A date can be any valid Gramps date, including dates formats like "26 JAN 1973" or "26.1.1973".

If you make your references be Gramps IDs inside square brackets, then you can refer to people already in the database, like this:

```
Person, Firstname, Lastname
joe's boy, Harry, Smith
```

```
Family, Child
[F1524], joe's boy
```

```
Husband, Wife
[I0123], [I0562]
```

```
firstname, surname
Timothy, Jones
```

```
place, enclosed_by
[P0001], [P0002]
```

This example would create and add Harry Smith to the previously existing family in Gramps, family F1524.

Also, this example would marry two previously existing people, I0123, and I0562.

This also creates a person named Timothy Jones who is not related to anyone.

Finally, this also make place P0001 be enclosed by place P0002.

Real world example

The screenshot shows a spreadsheet titled 'gen5-test.csv' in OpenOffice Calc. The spreadsheet is organized into several sections. The first section, starting at row 3, lists 16 individuals with their first and last names. The second section, starting at row 23, lists 8 marriages with the names of the husband and wife and the date of marriage. The third section, starting at row 33, lists 8 children with their family ID and their own ID.

Person	Firstname	Lastname	Marriage	Husband	Wife	Date	Family	Child
1	Peter	Blank	A	1	2	'28 jan 1725	A	[10084]
2	Anna Maria	Kiefer	B	3	4	4 may 1732	B	[10085]
3	Georg	Schmidt	C	5	6	07/02/30	C	[10086]
4	Barbara	Goering	D	7	8	08/17/27	D	[10087]
5	Johann	Kiefer	E	9	10		E	[10088]
6	Fides	Federer	F	11	12	05/06/24	F	[10089]
7	Sebastian	Schelli	G	13	14	09/24/36	G	[10090]
8	Magdalena	Schlegel	H	15	16	11/10/38	H	[10091]
9	Joseph	Oser						
10	Anna Maria	Adam						
11	Johann Marti	Mueller						
12	Anna	Knosp						
13	Phillip	Joggesst						
14	Maria Eva	Kiefer						
15	Bernard	Koenig						
16	Anna Maria	Kutz						

Fig. 5.1.3

In this example, I had an entire generation to enter, 16 names with marriage dates. The children I already had in the database. I entered them into LibreOffice:

Notice that you can use numbers or strings as the reference names between areas. In the person area, I used the numbers 1 through 16. That made it easy to refer to them in the second area of marriages. The marriages are labeled with the letters A through H.

Also note that the children in the third area are existing people as indicated by the brackets around the Gramps IDs.

Saving as CSV and importing into Gramps produces the far right-hand column in the tree:

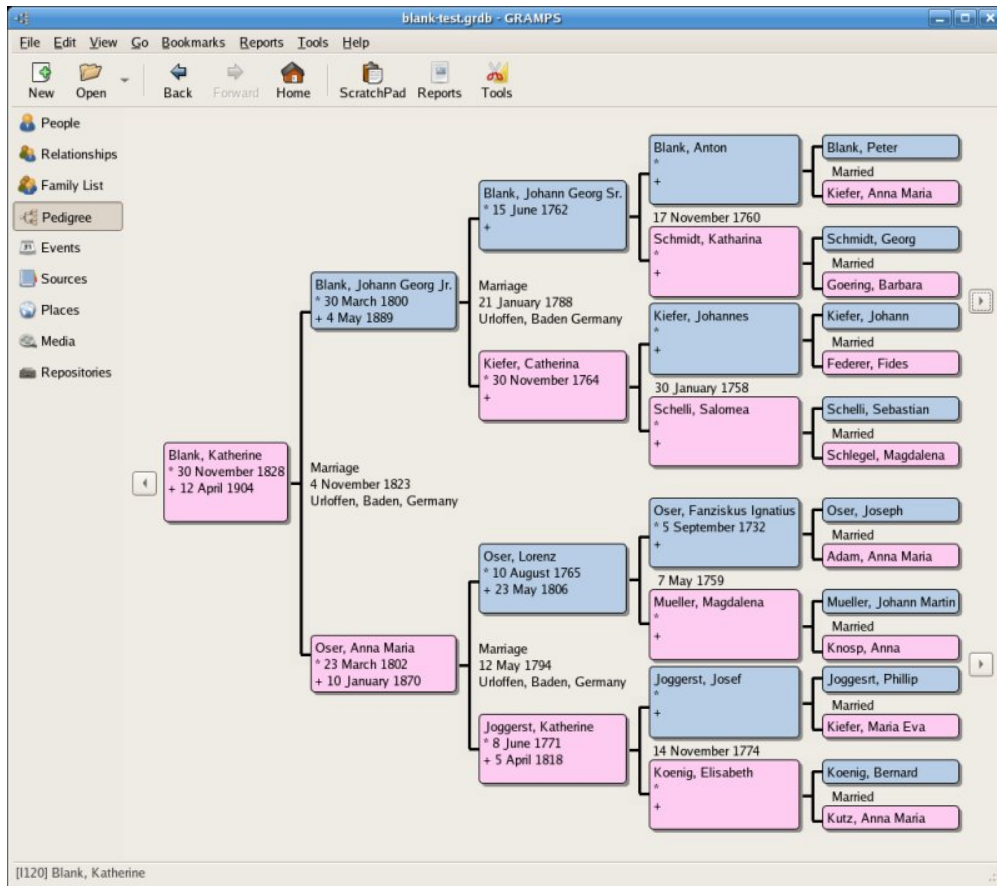


Fig. 5.1.4 Saving as CSV and importing into Gramps produces the far right-hand column in the tree.

[Previous](#) [Index](#) [Next](#)



Special copyright notice: All edits to this page need to be under two different copyright licenses:

- GNU Free Documentation License 1.2 - see [wiki copyright](#)
- GPL - see [manual copyright](#)

These licenses allow the Gramps project to maximally use this wiki manual as free content in future Gramps versions. If you do not agree with this dual license, then do not edit this page. You may only link to other pages within the wiki which fall only under the GFDL license via external links (using the syntax: [https://www.gramps-project.org/...]), not via internal links.

Also, only use the known [Typographical conventions](#)