



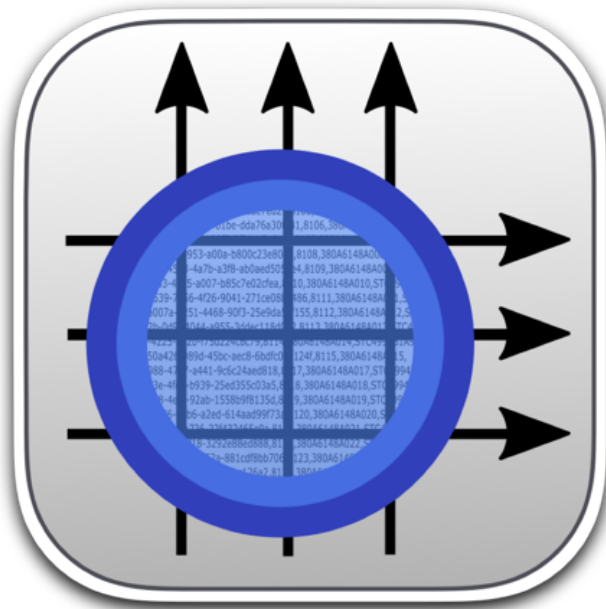
Roberto Panetta



Tension Software

Data Merge 1.1

User Guide



We Make Software - Ecleti.com

Data merge © 2019-2025 Ecleti - Roberto Panetta all rights reserved

Every effort has been made to ensure that the information in this manual is accurate.

Ecleti is not responsible for printing or clerical errors.

Other company and product names mentioned herein are trademarks of their respective companies.

Welcome to Data Merge

Create complex merge operations

Data Merge can execute complex merge operations between data and forms.

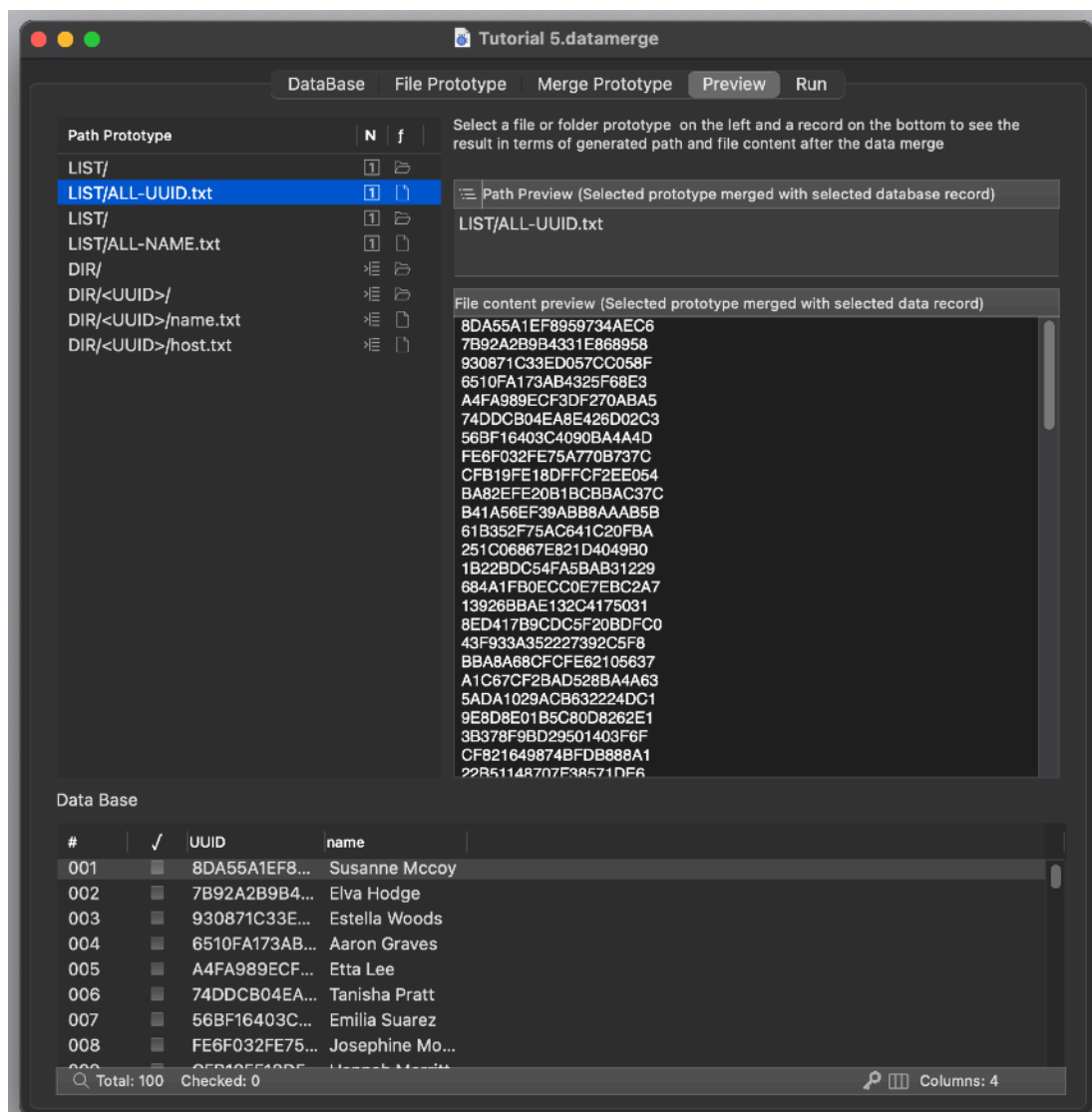
Data Merge can generate the standard usual merge:

You have one form letter and generate a different letter for each record substituting in the form you made each <filed> tag with the content of the <field> value for any record.

Data merge can also generate more complex merge operations:

From a single Data merge document it can generate multiple files and folders, nested at any level, each with file name and content polled from your records, or also generations single files with all the records content merged sequential inside single file for all the records.

All this mixed and aggregated as you like through simple rules editable via point and click with a minimum of text input required



Data merge can be used also as code generator, creating long repetitive code where you need to transform a long sequence of records in structured code.

Think as example to a long 'switch case' sequence or to complex web pages merging all in one page more records data plus linked multiple other pages

Data merge can be used also to generate configuration files merging data from records inside a configuration form (for Apache as example if you need to generate many different config files and you need to change them often) or for example also BIND9 to maintain a name server or POSTFIX to maintain a mail server long alias email list.

You can mix 'one file per record' with 'all records in one file' inside the same document allowing the same document to generate multiple different merging all in one click

The list of examples can be infinite.

Data merge can save you hours of work.

You update your database and then press 'Generate'

Data Merge can generate hundreds or thousands of file in few seconds mixing different form and an unlimited size of data.

One click and the job is done, fast and error free.

If you have often to create code with a standard structure with different values inside Data Merge is a dream coming true.

Getting Started with Data Merge

Using the 'Data Merge Tutorial' documents

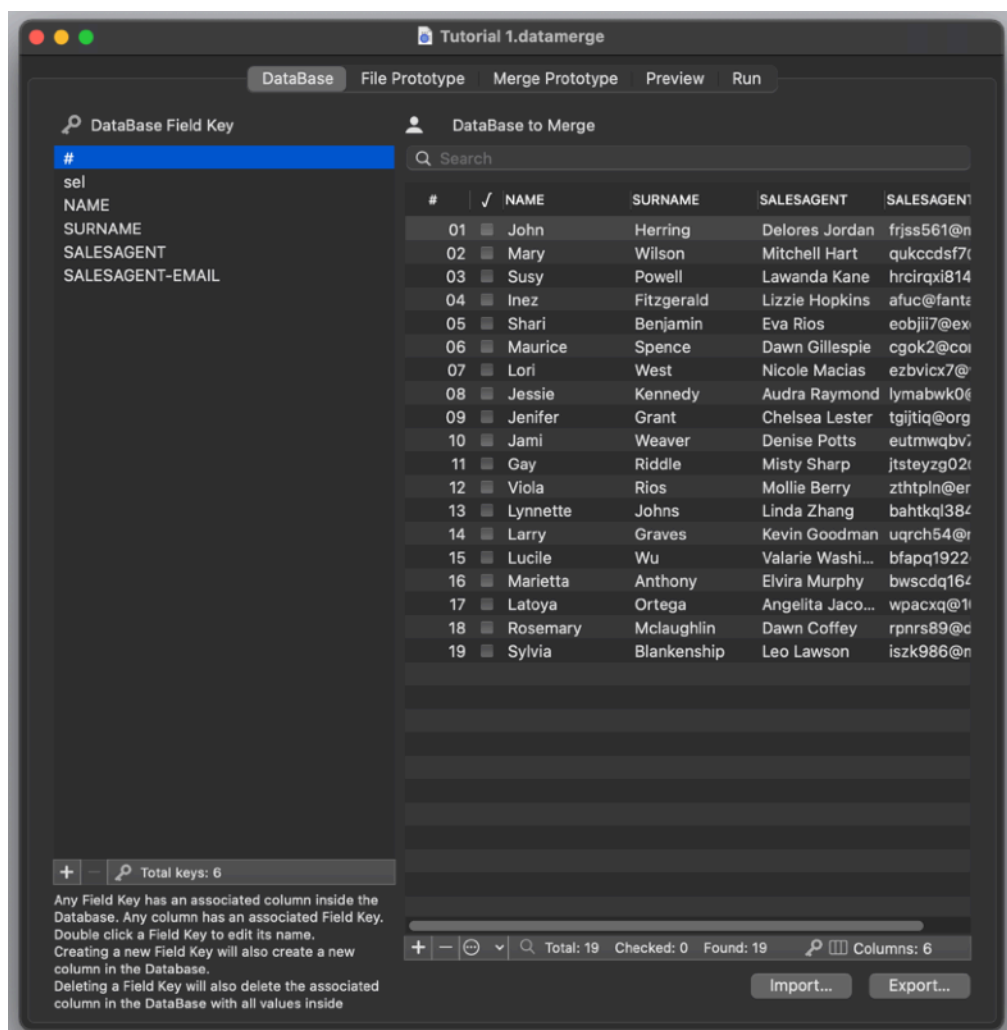
You are welcome to our site to download the 'Demo Documents for Data merge' containing the tutorial and demo documents used here.

Use the Help menu in the application to come to our web site or download the Demo Document directly using the 'Demo and Tutorials' command under the Help menu.

Download the Demo and Tutorials archive, open it and save the folder on your Hard Disk, then you can start learn how to use Data Merge.

Tutorial 1

Tutorial 1 document contains a DataBase of customers with Name, Surname, SalesAgent and SalesAgent-Email for any record

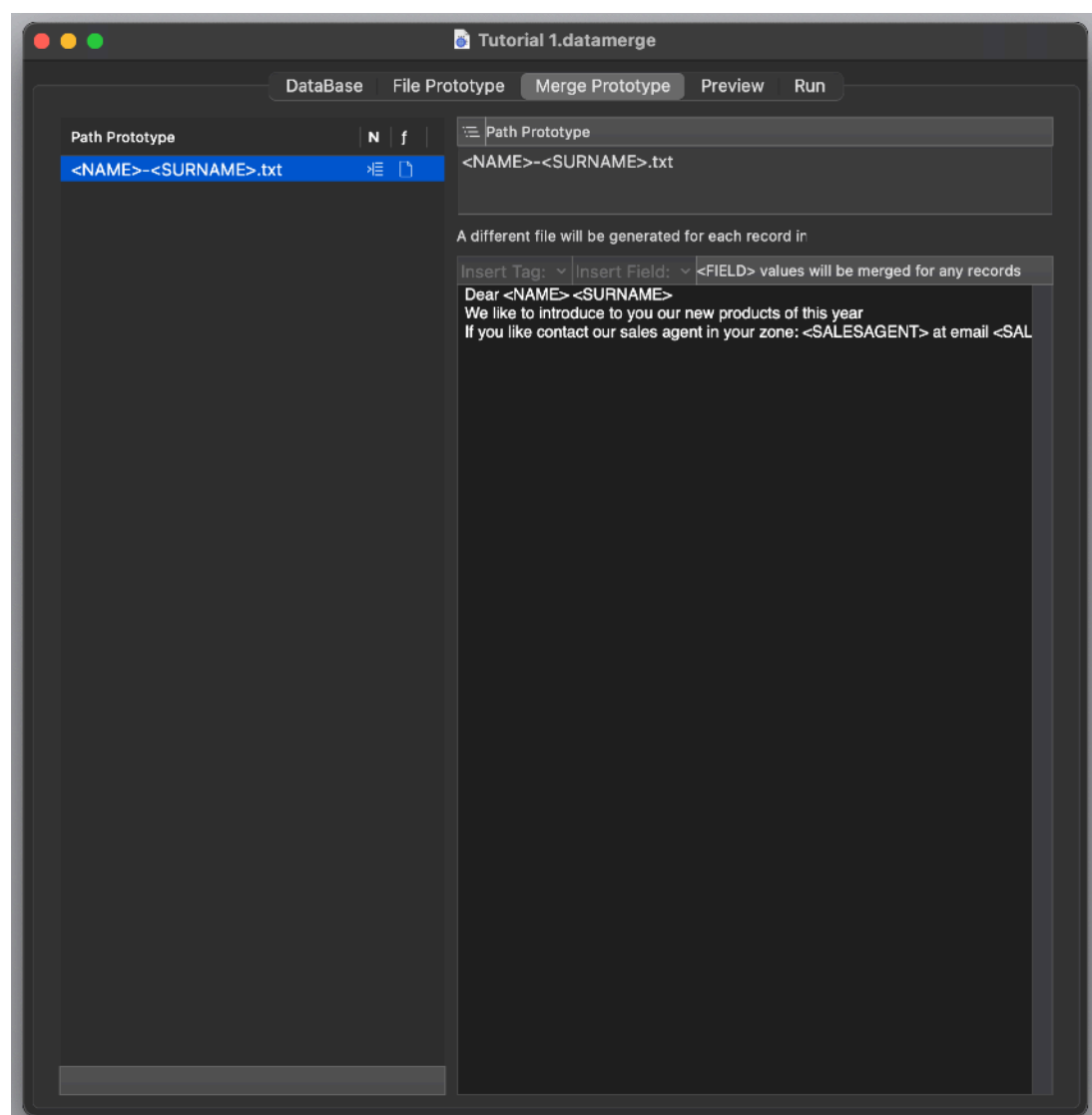


The document will create a text for each customer, inviting him to contact the sales agent at the sales agent at his own email

This is probably the most common and traditional case of data merging, creating a text file for any record using a common form

In the DataBase area you can see the data is already set
You have a total of 19 records

In the 'File prototype' area you need to set the 'Root Destination' where to save the final files obtained by the merge (we couldn't set it for you in advance)



Still in the 'File prototype' are select 'One per Record' in the secondary tab zone
You can see there is 1 Prototype Path:

<NAME>--<SURNAME>.txt

This will create a different file for any record in the DataBase

Select the prototype path in the table and then see what happens selecting various records in the table below (which mirrors the DataBase records from the DataBase section).

You can see as you change your selection in the DataBase table below, the Path preview is updated showing the merge result for any of the record

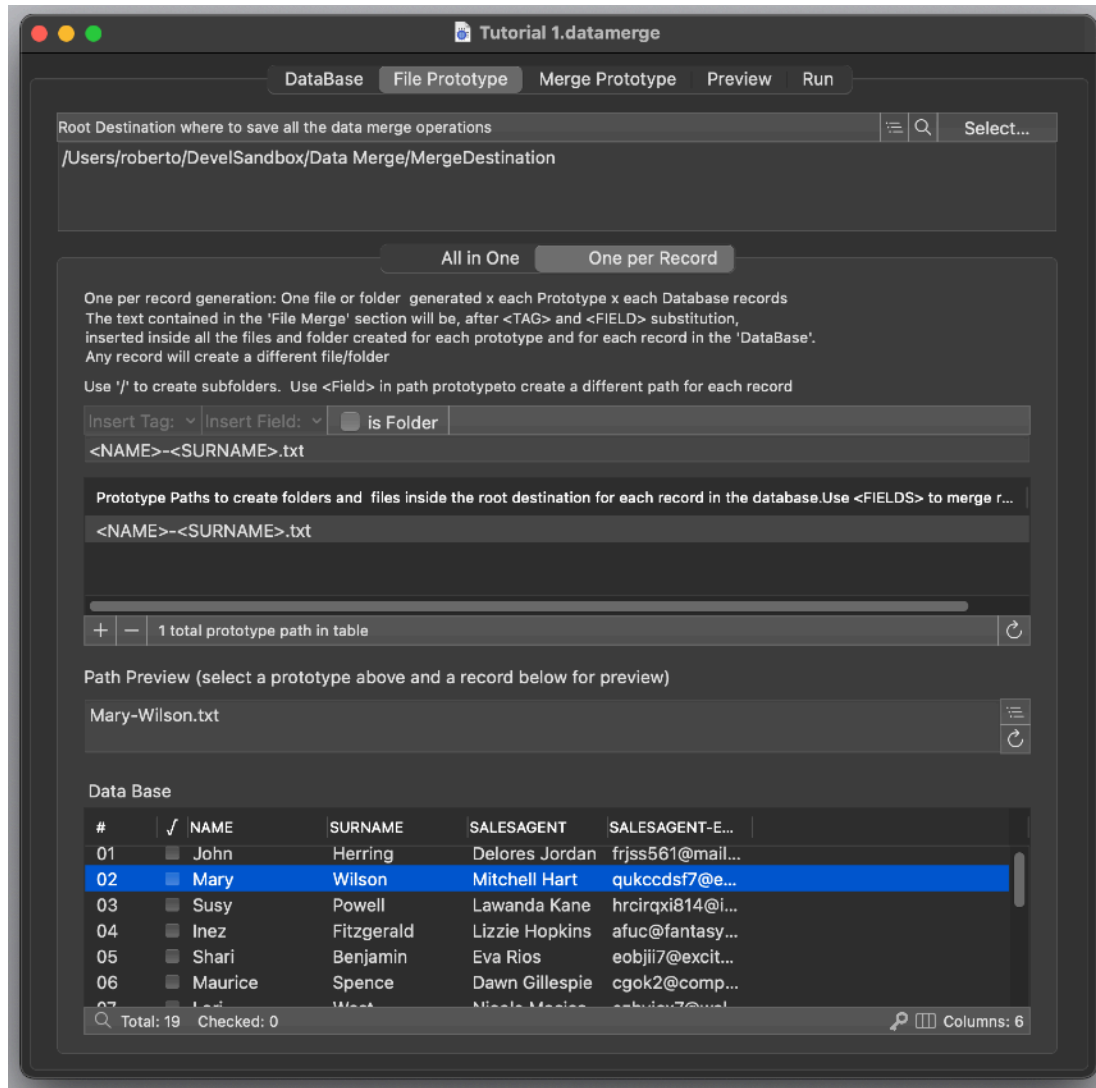
In the 'Merge prototype' area you have one entry in the table on the left, relative to the path you created in the previous area.

Select it and you can see you can visualize the form that will be used to generate the content for any of the generated file

You can see that inside the form where used the filed tags:

<NAME> <SURNAME> <SALESAGENT> and <SALESAGENT-EMAIL>

These tags will be substituted for each record with the fields value for the record

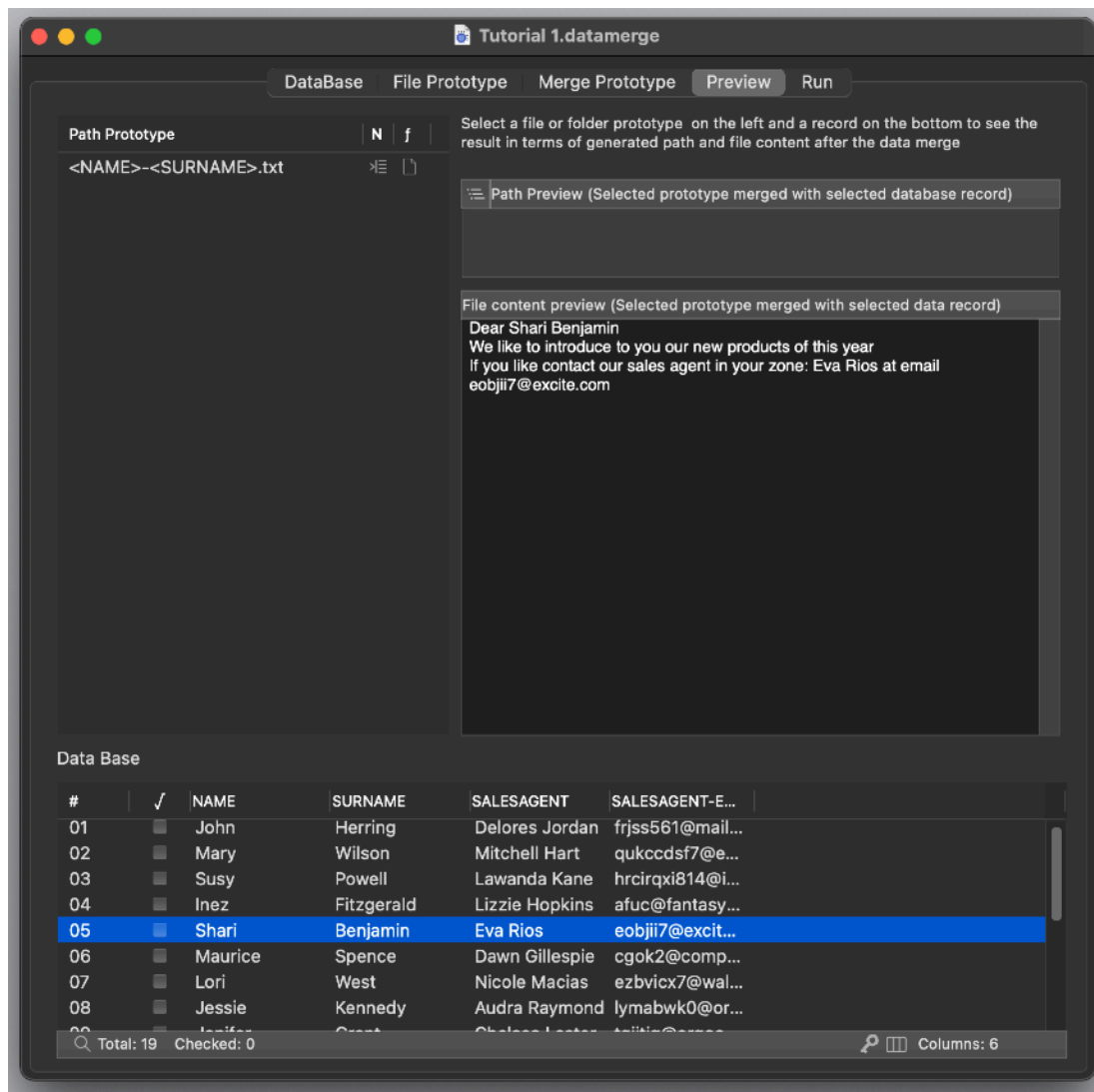


Clicking the preview area you can see that this area is very similar to the 'Merge Prototype' area with one exception:

This area is read only and there is a table in the bottom part of the area mirroring all the records of the database.

Selecting a prototype on the left and a record in the table below you can see the preview of the merge relative to that file prototype with the selected record

Moving up and down with the selection you can see the final text result for all the records



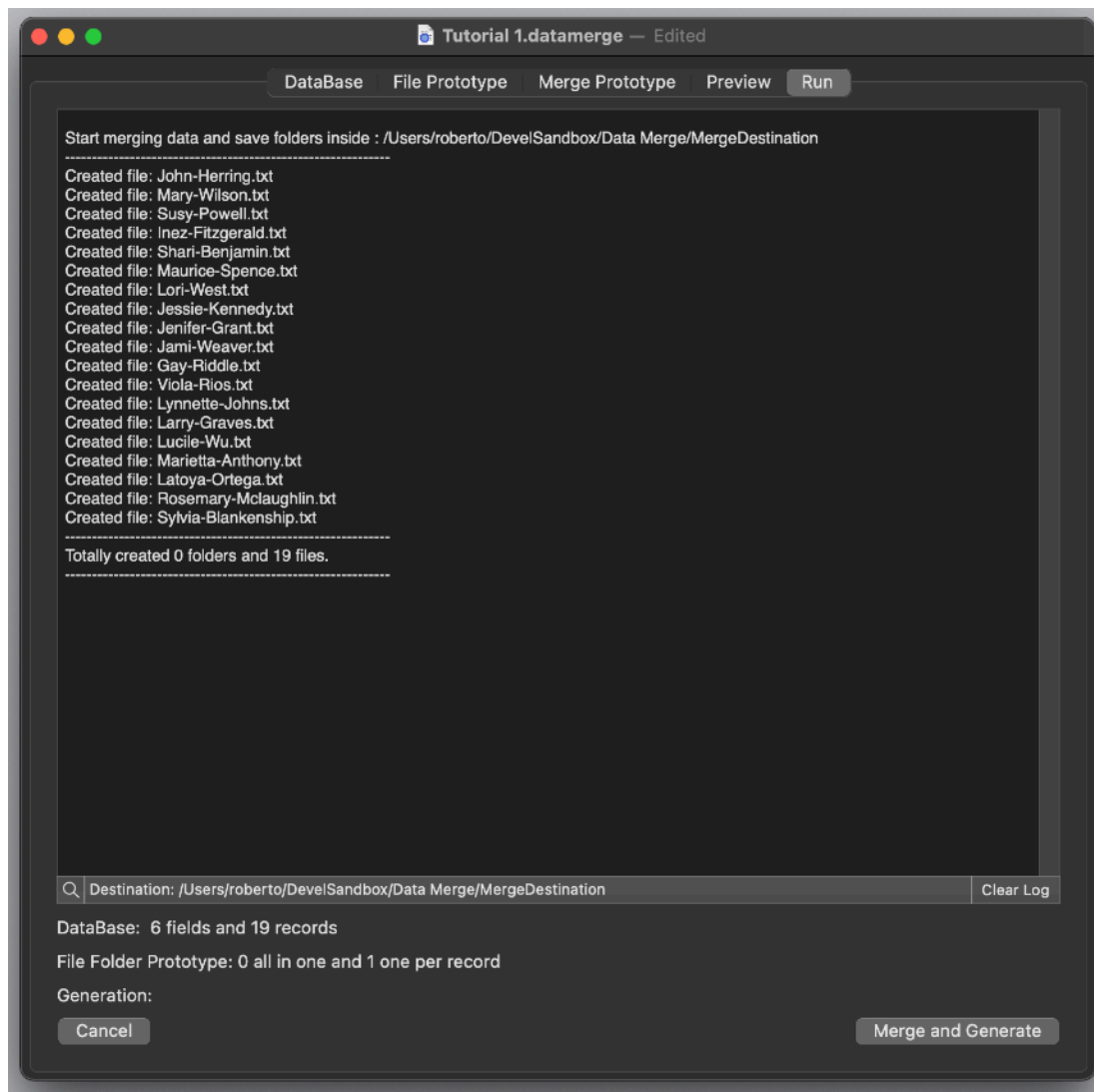
Select now to the 'Run' tab

Click the the Magnify Glass at the bottom of the log area to disclose the destination folder and check if it's empty before generate.

Remember that Data Merge will not overwrite existing files and folder, so before generating usually you need to check the destination folder is empty or at least the part you need to re-generate are non-existent
Click the 'Merge and Generate' button

You can see from the log 19 files were generated
See inside the destination folder to have a confirmation
The 19 files were generated, each one with a custom name as specified inside the 'File Prototype' area

If you look inside any file you can see the content was generated in accordance as specified inside the Merge Prototype area with all the <FIELD> tag substituted with the correspondent record value.



As example the 'John-Herring.txt' file will contain:

"Dear John Herring

We like to introduce to you our new products of this year

If you like contact our sales agent in your zone: Delores Jordan at email

frjss561@mail2web.com"

More Examples

Tutorial 2

Tutorial 2 is like Tutorial 1 adding some more concepts
You can see the database area has inside the same data records

Go to the 'File Prototype' are and specify the 'root folder' as the 'Generation' folder inside the 'Tutorial 2' folder

Please note that the file prototype area in the 'One per Record' sub area has a different path:

LETTERS/<NAME>-<SURNAME>.txt

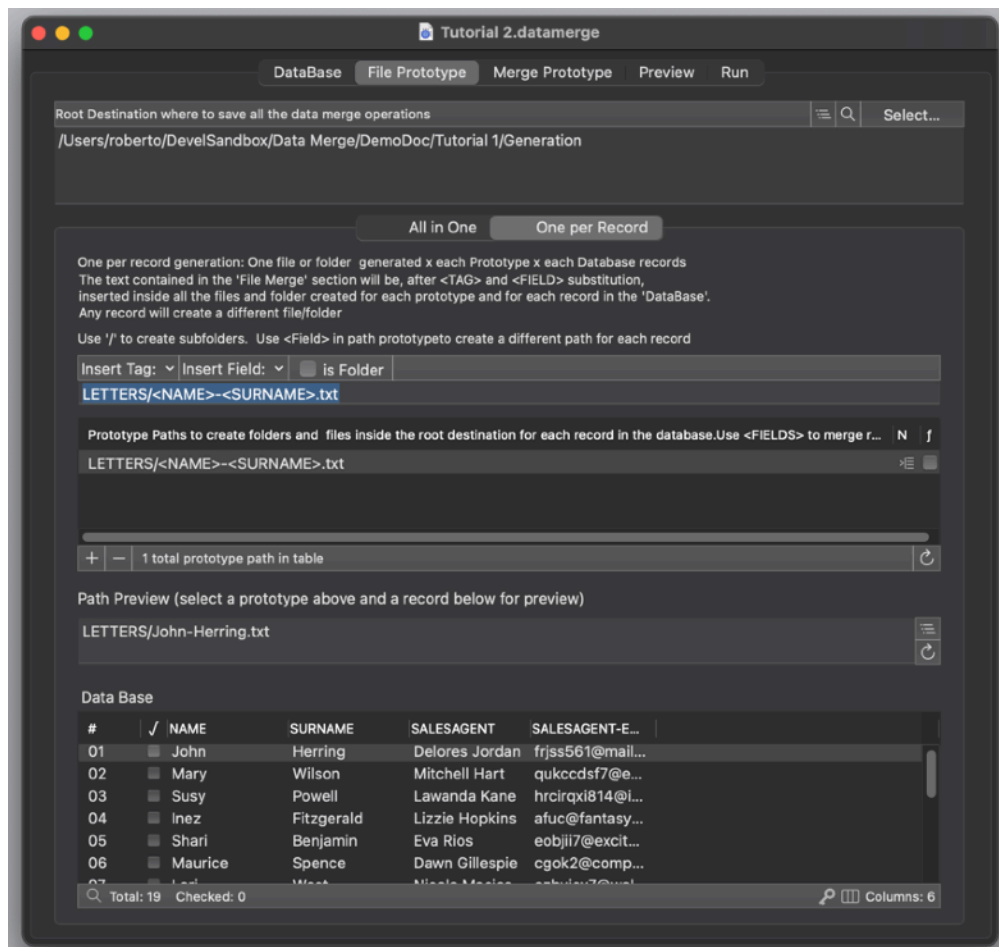
This will instruct Data merge to create all the files (one per record) inside a subfolder named 'LETTERS' inside the folder you selected as 'root folder'

You can use full complex path for any prototype made by 'constant parts' and variable parts associated to records value

So the prototype:

LETTERS/<NAME>-<SURNAME>.txt

As you can see specify a constant folder 'LETTERS' followed by something that will change for each records: '<NAME>-<SURNAME>.txt'

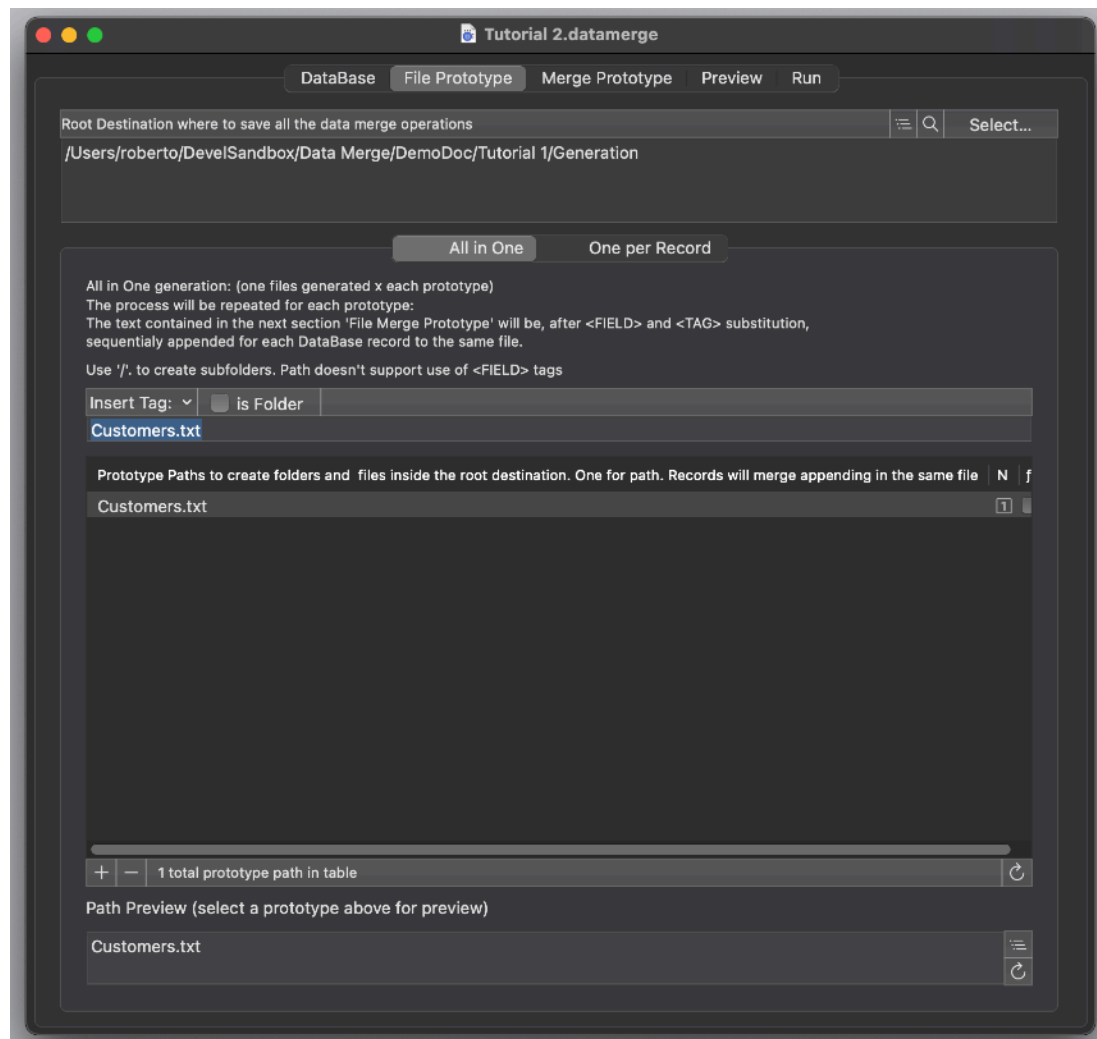


Another change respect the Tutorial 1 is the addition of another file prototype in the 'All in One' area

'Customer.txt'

As you can see this has no <FIELDS> part.

In the 'All in One' area <FIELDS> parts are not allowed, because this path is NOT generated for each records but just once for prototype



What is interesting is what an 'All in One' generated file has inside

Click the 'Merge Prototype' tab

You can see on the left table there are now 3 entries

The first is relative to the 'Customer.txt' all in one generation

The other 2 to the subfolder 'LETTERS/' and the 'LETTERS/<NAME>-<SURNAME>.txt' files that will be generated for each file

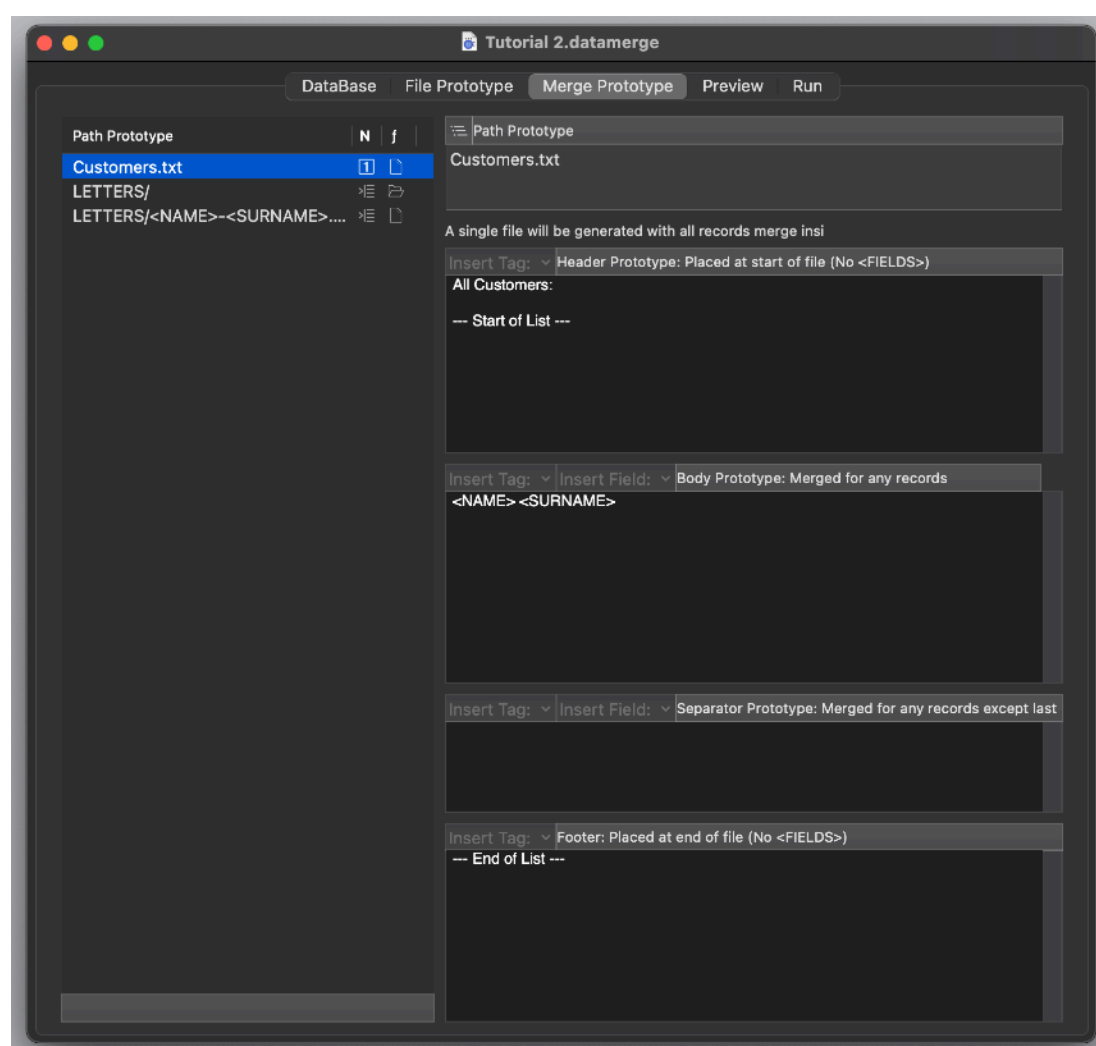
There is nothing new in the latest 2, all very similar to the previous 'Tutorial 1' except the files will be generated inside a folder named 'LETTERS'

What is new is for the content of 'Customer.txt'
Select it and see on the right 4 edit zones opens

The header prototype let you specify a header for the file (no <FIELDS> tags allowed here because it is inserted once)

The body prototype let you define a body that will be appended to the file for each record in the database. Field tags admitted here. The <FIELD> tags will be substituted for each record with the fields values

The separator Prototype is similar to the body except is not used for the last record (it only acts as separator between records). Field tags admitted here. The <FIELD> tags will be substituted for each record with the fields values



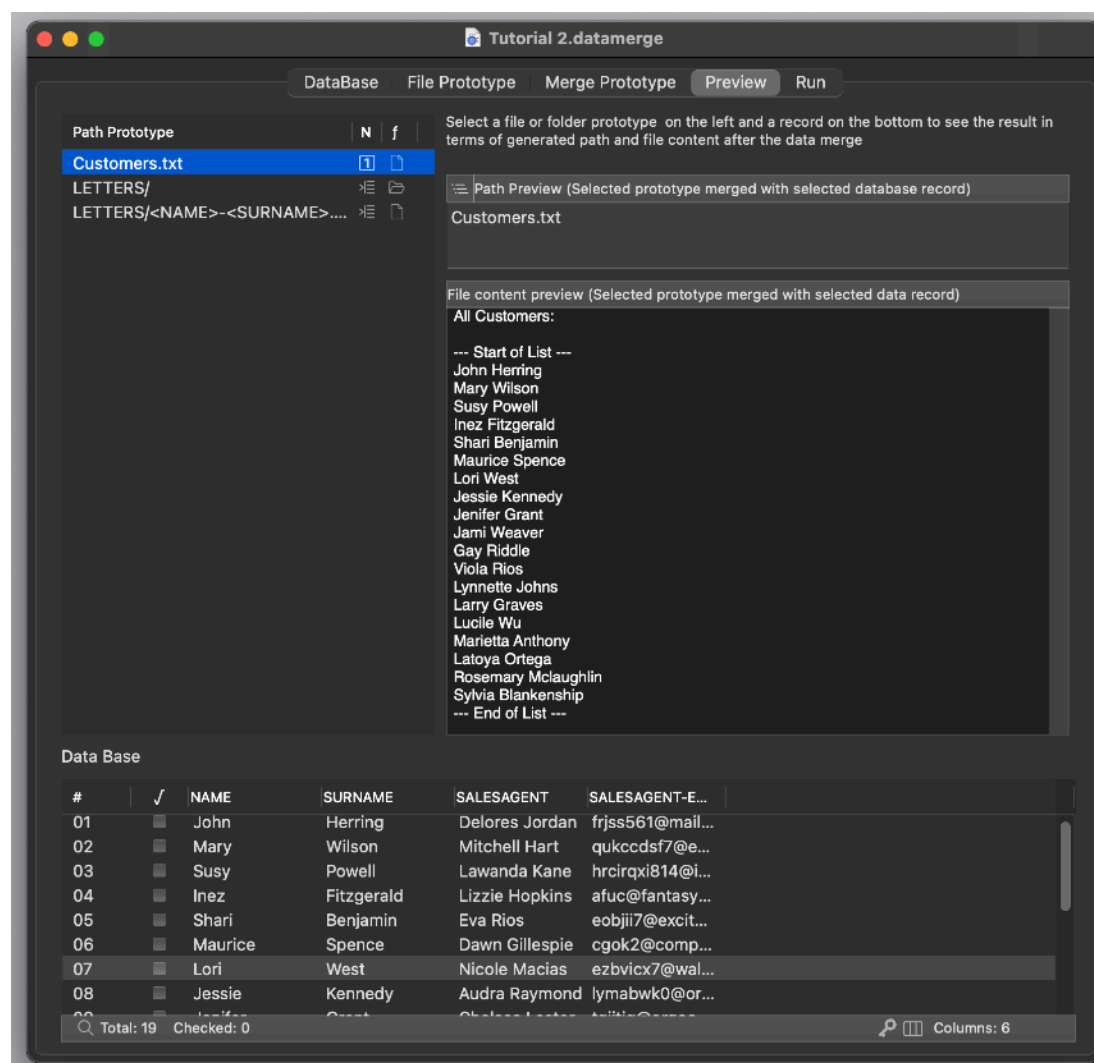
The footer prototype append at the end of the file this text (no <FIELDS> tags allowed here because it is inserted once)

Select now the Preview

For the 'One per Record' files you can see for each records you generate a different file name and content as in the previous tutorial

For the 'All in One' prototype you created, you can see you have a unique file merging all the record data inside this unique file

In detail because we had '<NAME> <SURNAME>' in the body part (which is repeated for each records, we have all the name, surname , one per row, inserted in this file



Go to the 'Run' area

Click to show the destination folder

Check it's empty

Generate.

As you can see all the one per record files are now generated inside the 'Letters' folder plus a Customers.txt file were generated using the 'all in one' method containing all the record merged as requested.

Tutorial 3

Tutorial 3 is a bit different from the previous 2

Now that you have an idea of how Data merge works we use it for a more concrete (even if basic simple) job
Creating a small basic web site showing products that we have inside our database

As you can see we have for each records 5 fields

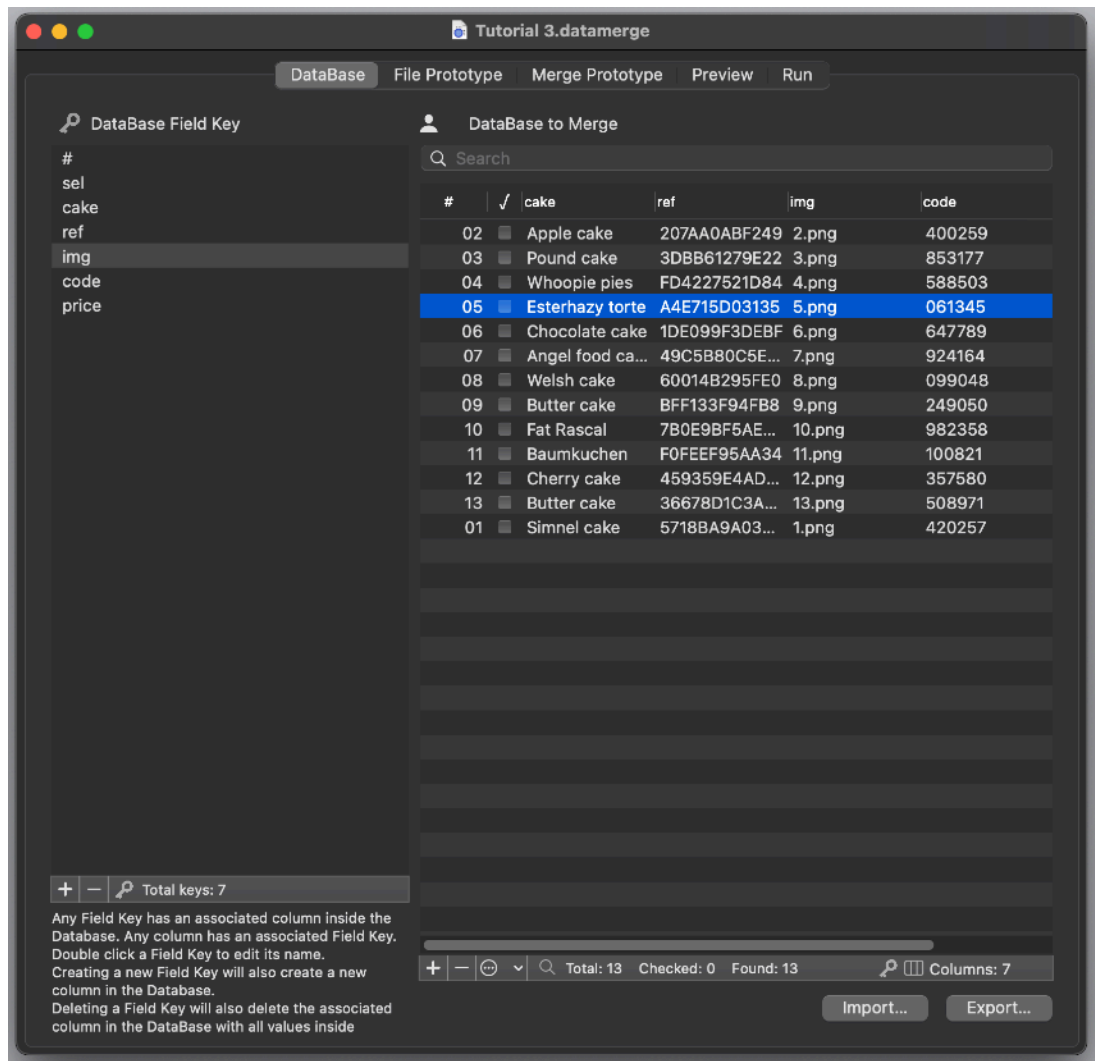
Cake

Ref

Img

Code

Price

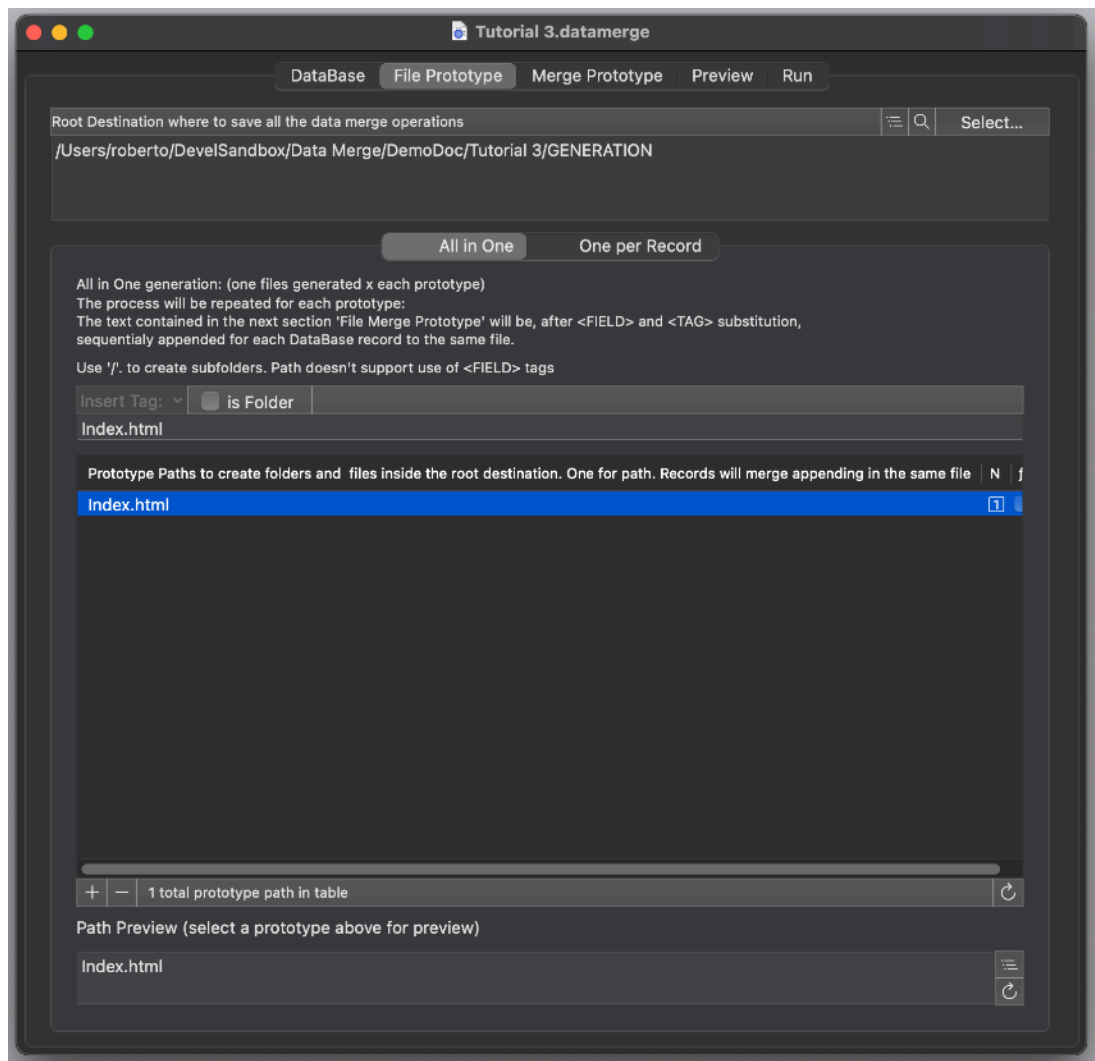


We move to the 'File Prototype' area

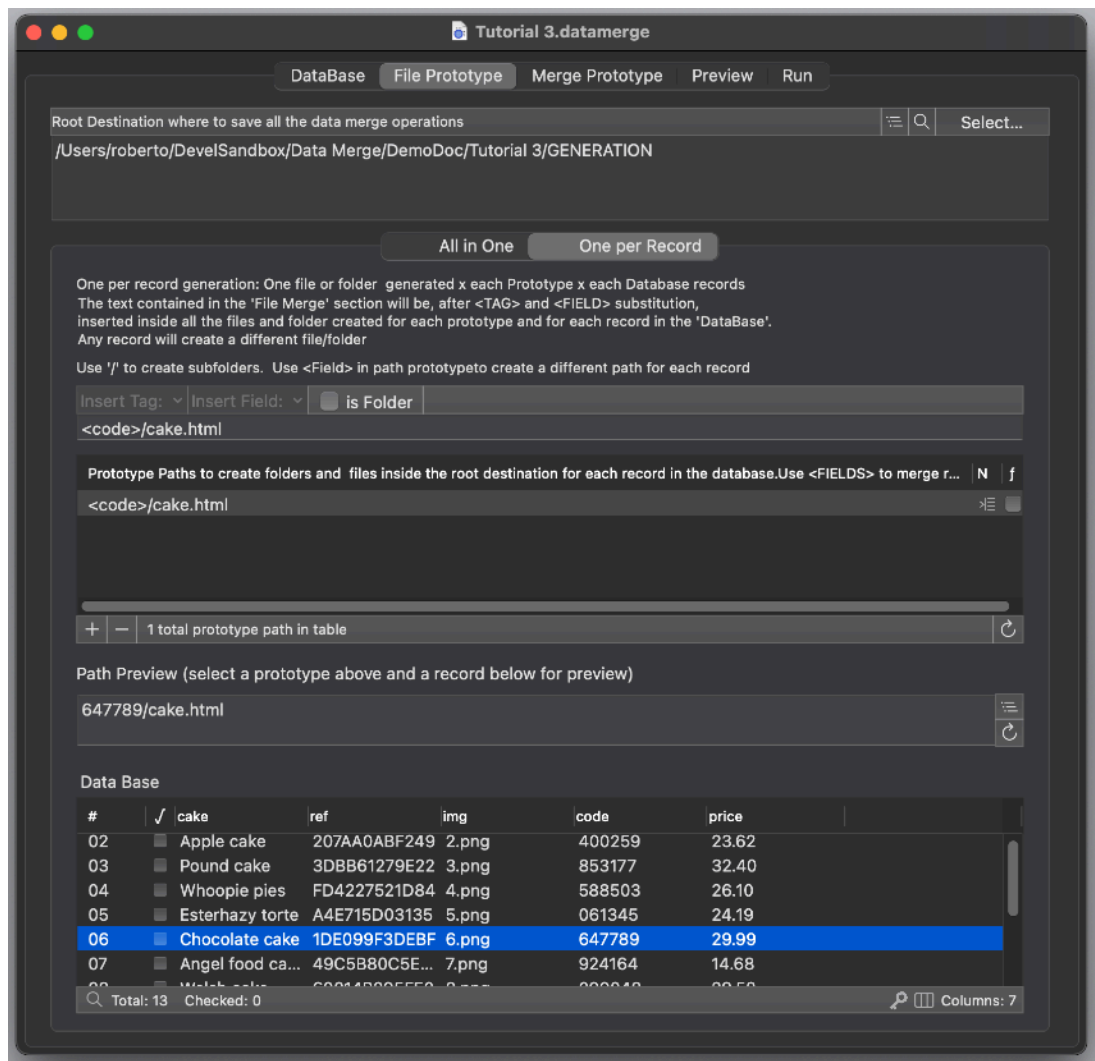
Set the root folder as usual using the a the 'Generation' folder inside the 'Tutorial 3' folder

In the 'All in one ' area we create the index.html page with a list of all products in the database

Because we want all the record inside a single file, plain clear we will use the 'All in One' method for the index.html file prototype



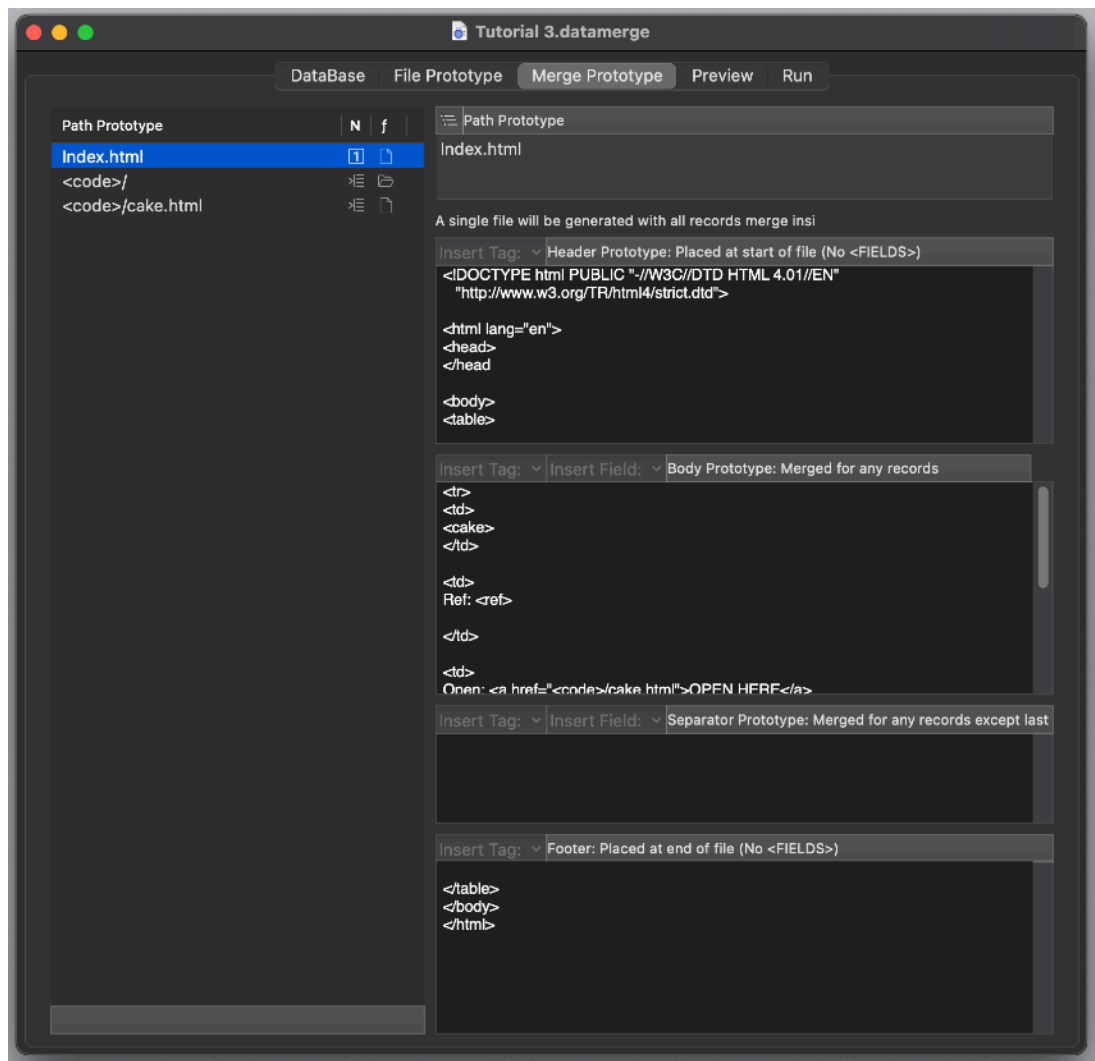
Then in the one per record area we create a file cake.html for each record and we put it inside a folder named as the code name for each product



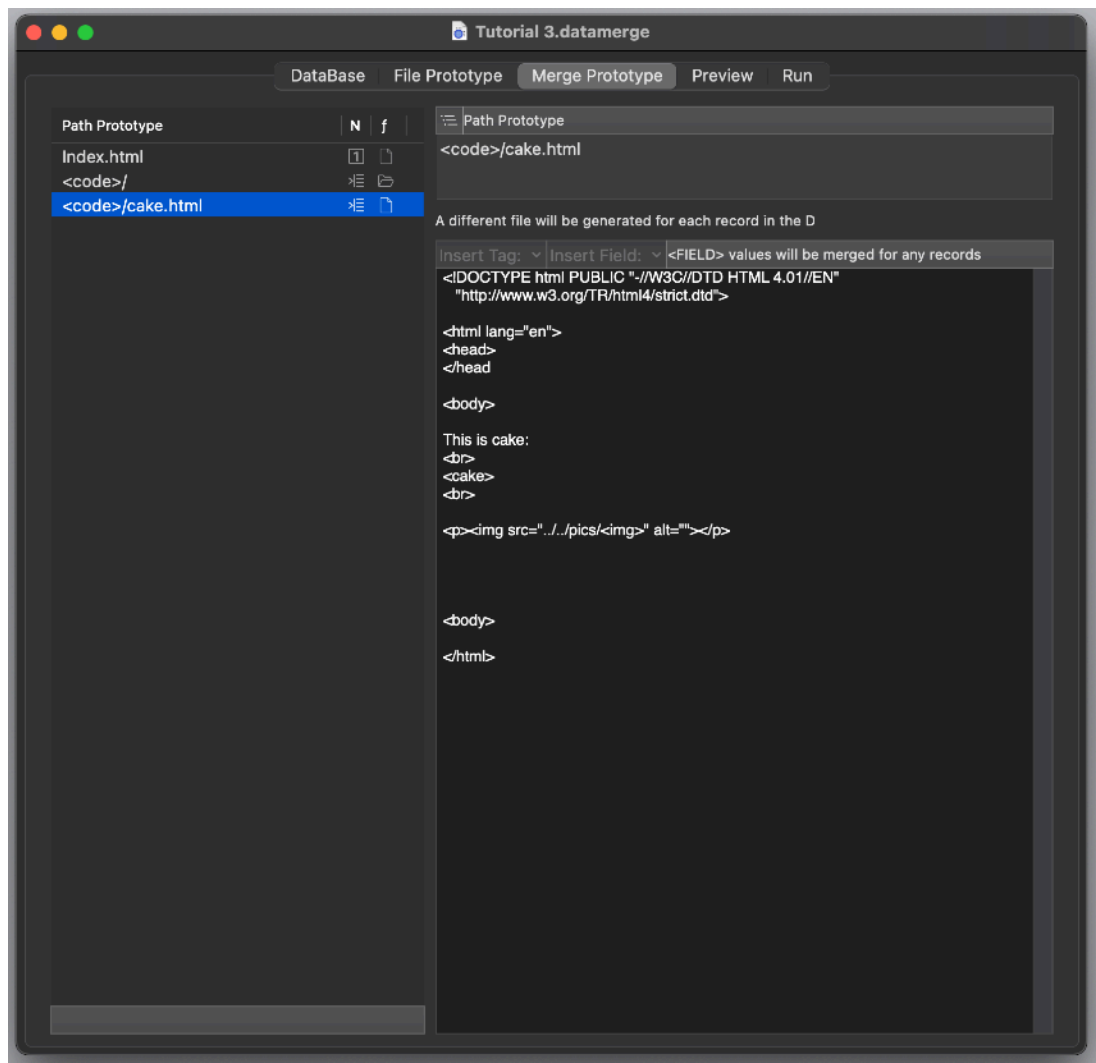
We look now at the Merge Prototype area:

For 'index.html' we use the body part to make a row inside a table for each record in the database.

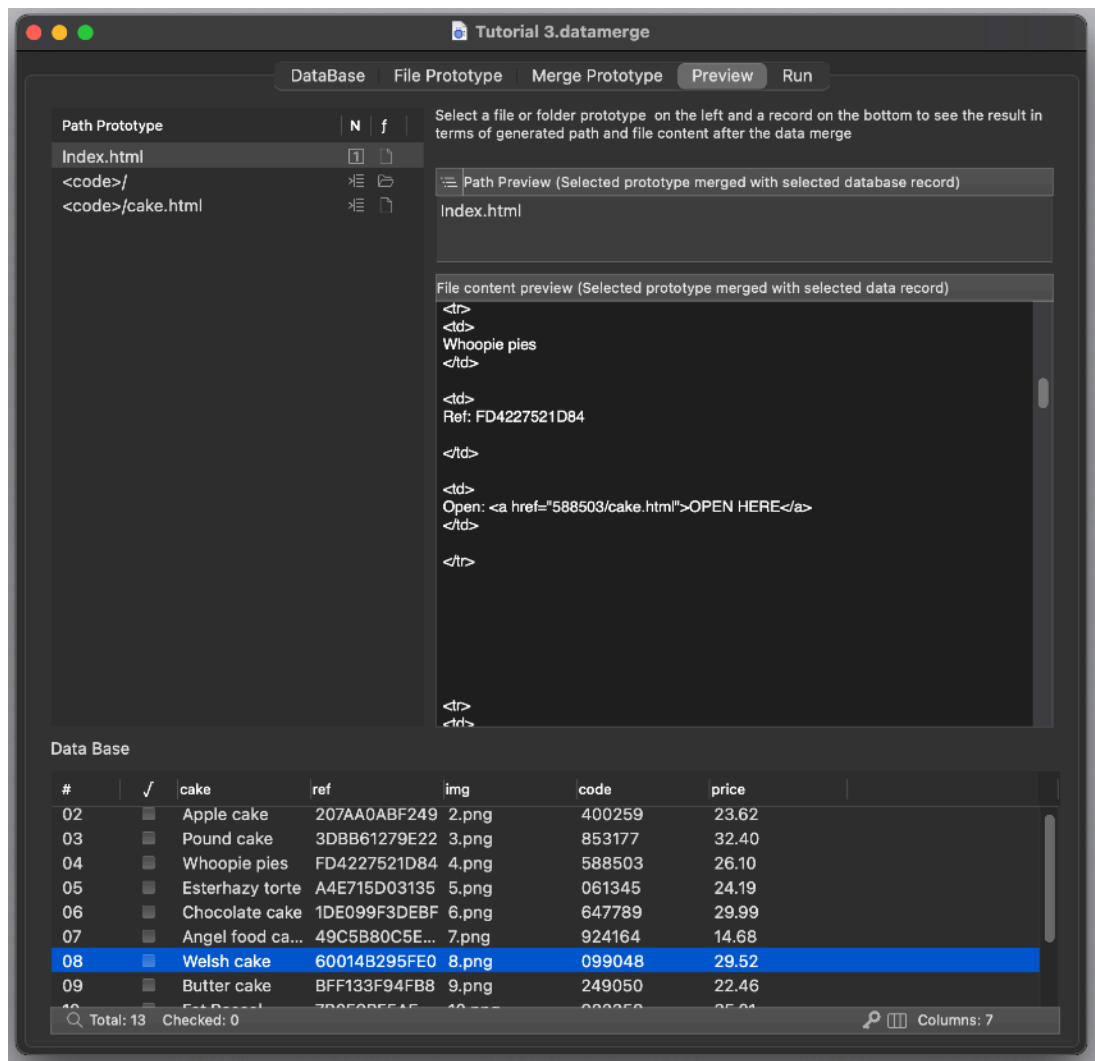
The header and the footer wrap this central code with the necessary html



For the cake.html page we generate a page for each record:



You can see the result in the 'Preview' area

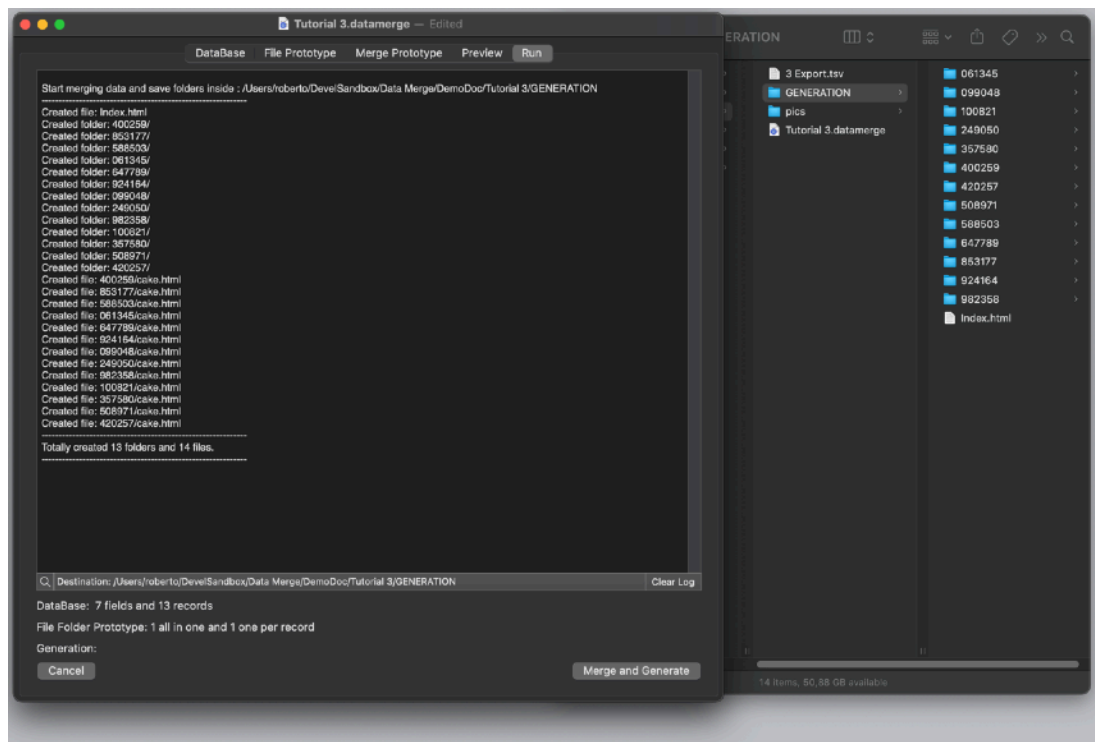


Go to the 'Run' area
 Show the destination folder
 Check it's empty
 Generate

You should obtain a full working very basic small web site with an index page and a product page for any record in the database

Open the index.html page with Safari to visualize it.

(To see also the pictures inside the web site it's important to generate inside the 'GENERATION' folder inside the 'Tutorial 3' folder because the html use the link '../pics' to access the pictures.

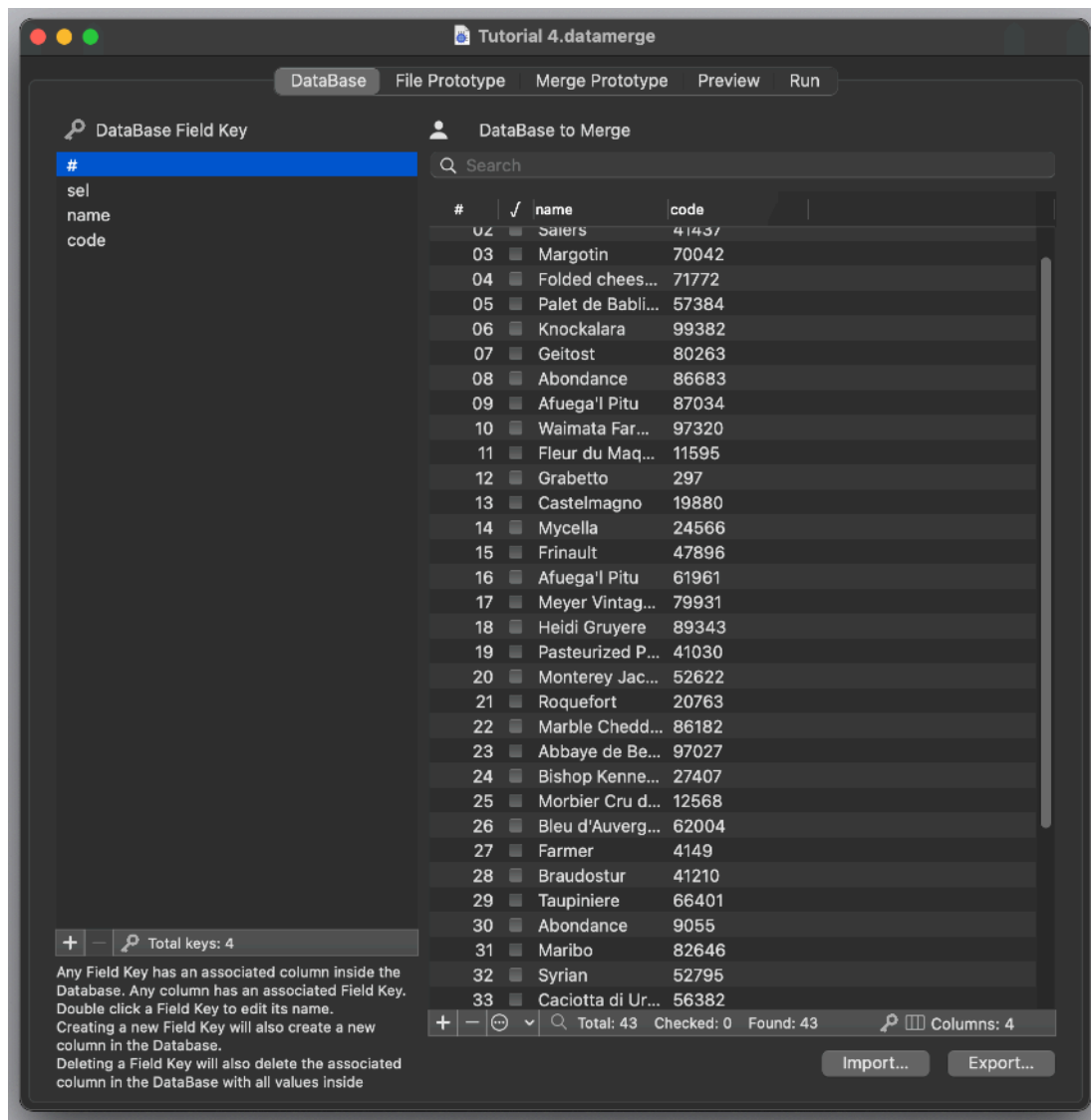


Tutorial 4

Tutorial 4 is about creating a portion of code to paste somewhere during development

It uses Objective-C as language but Data merge can generate code for any language, python, bash, swift, basic, C, C++ and so on because Data merge knows nothing about languages. It just manages text.

It uses a simple case where a long list of cases must be implemented



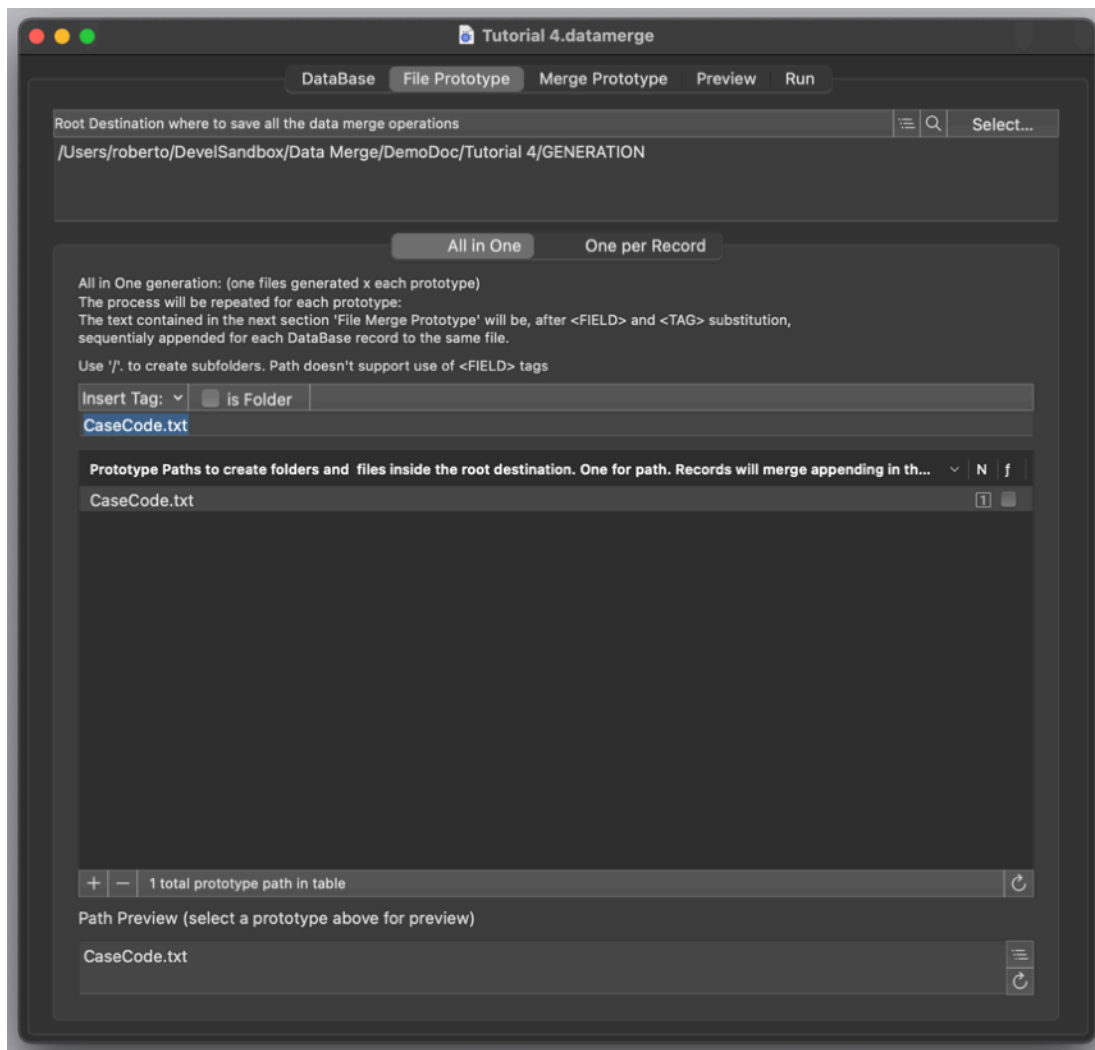
This problem can be solved using also an array at code level, or in many other ways, but the purpose of this tutorial is not teaching you how to code, but how to use Data merge in simple 'stupid' cases for learning purpose.

It's then up to you use it in complex 'smart' cases

We have in the database a long list of int and for each <code> we have to return a <name> string

Data Merge can generate the switch code for us:

In file prototype we generate a single 'All in One' file named 'CaseCode.txt'



In the merge prototype you can see we put in the header:

```
"
switch(code)
{
"
```

And in the footer:

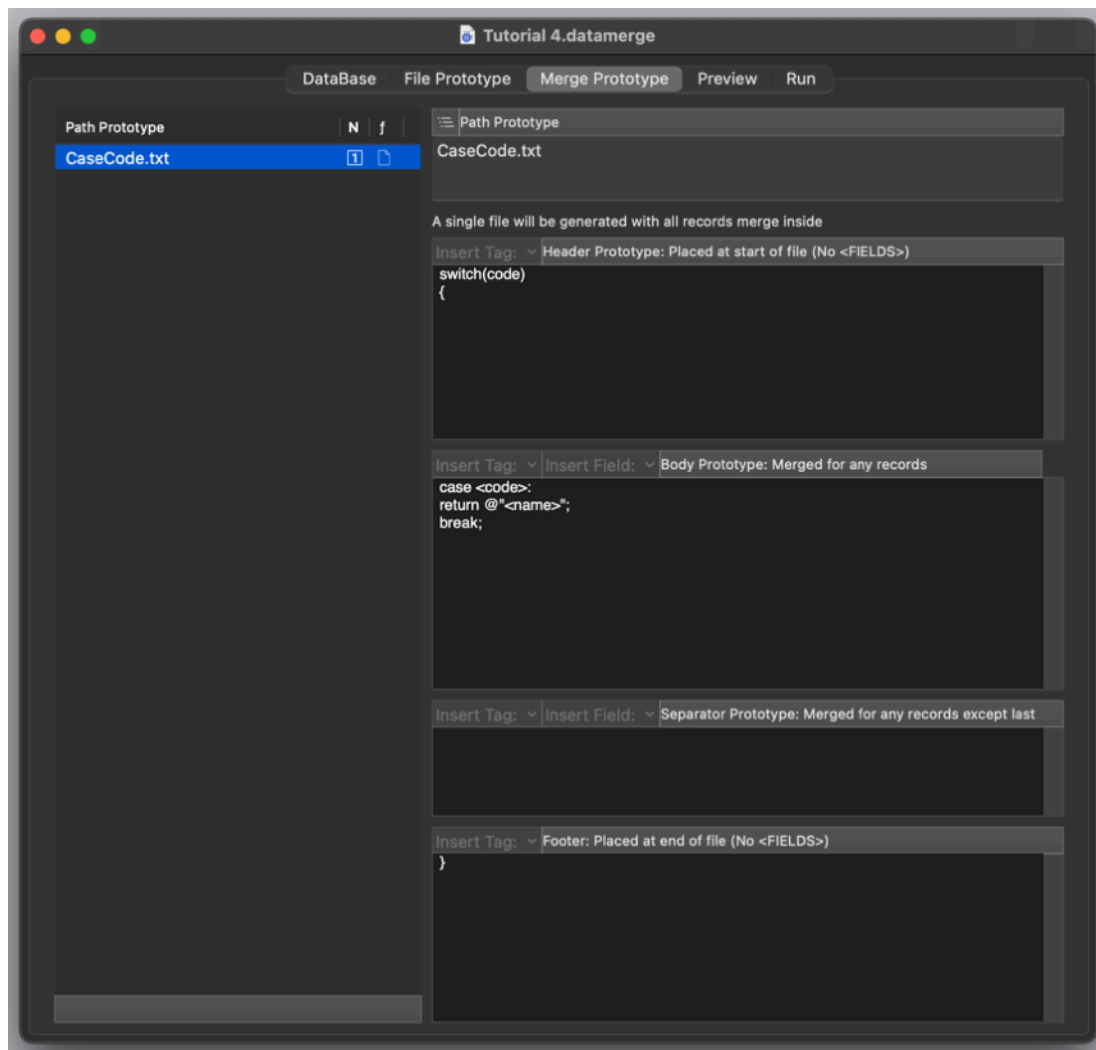
```
"
{
"
```

To open and close the switch code

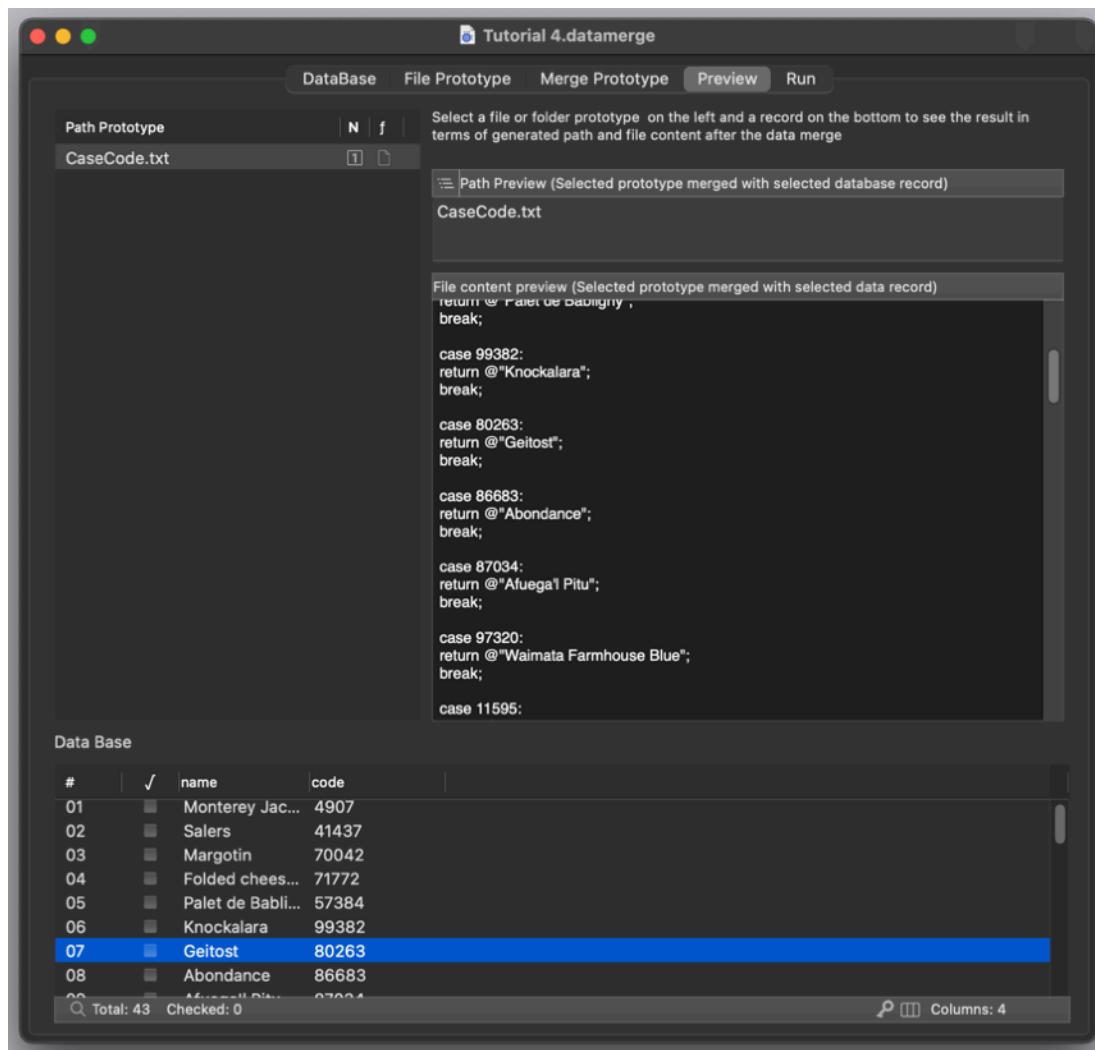
In the body we put:

```
"
case <code>:
return @"<name>";
break;
"
```

As you already now understand, this block will be repeated for each record in the Data Base and the "<code>" and the "<name>" tag will be substituted in each block for any record value of the correspondent fields named 'code' and 'name'



In the preview area you can see the result



If you need to often generate similar (or probably more complex code and structures) you can save what you need in a Data Merge document and open it when you need, import the data structure (see next paragraph on how to import) and generate the file to disk.

Probably will be easier to copy and paste the code from the preview area to your preferred code editor without saving on disk

Tutorial 5

In Tutorial 5 you need to create a folder file structure for a server

The server will look-up inside a directory for a directory with a predeterminate UUID. When it finds that directory it look-up inside the folder for the files contained inside that directory, and read the content of the 2 files named name.txt and host.txt and use that values for specified purposes.

So for example we will have to create a folder named 8DA55A1EF8959734AEC6, it will contain two files inside, one named name.txt with 'Susanne Mccoy' inside and the second file named host.txt with 'aldebaran.myhost.com' inside .

The folder name and the content of the name.txt file are provided in a text file to import.

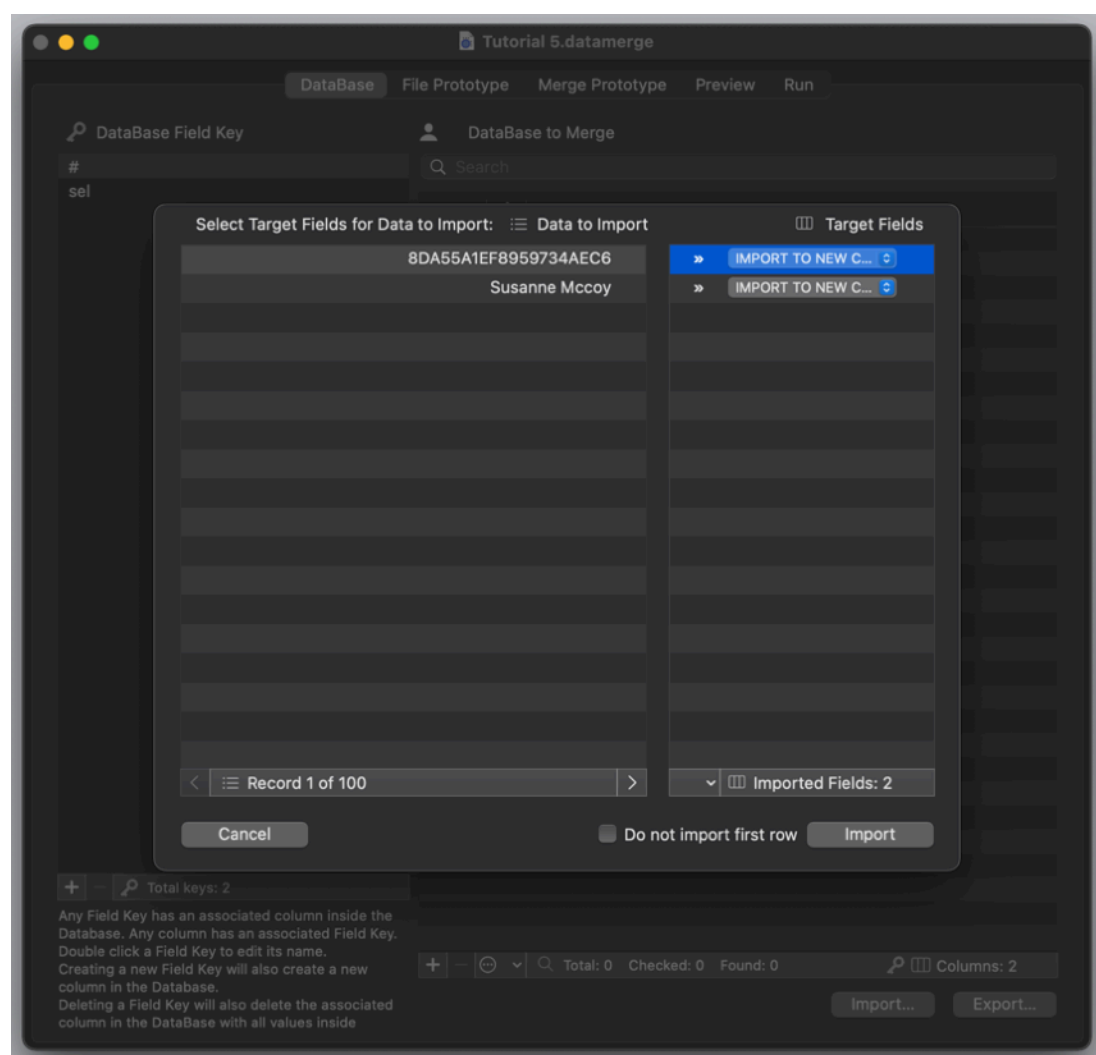
The string 'aldebaran.myhost.com' as content for the file host.txt is a constant, all the file host.txt for this generation will have the same content so no need to have it in the Data base because it's a constant value

In addition we need to generate one file containing one per row all the names and another file with one per row all the UUID

This is just an example of a complex configuration file structure we need to prepare for a server

We open the Tutorial 5 document and we can see the Data Base area is empty
We use the 'Import' Button to import the 'Import Me' file contains all our data to merge.

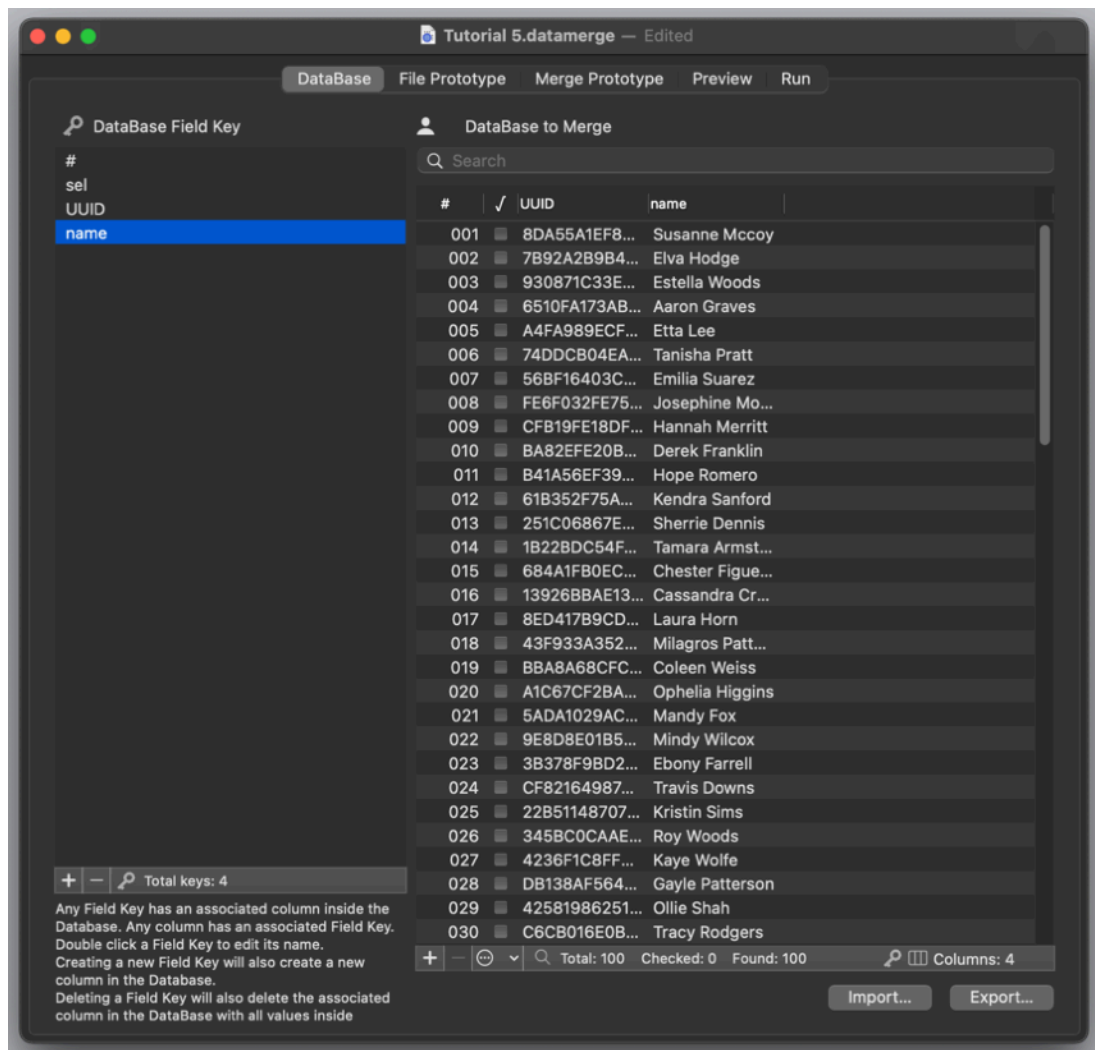
We select to import all fields in a new columns



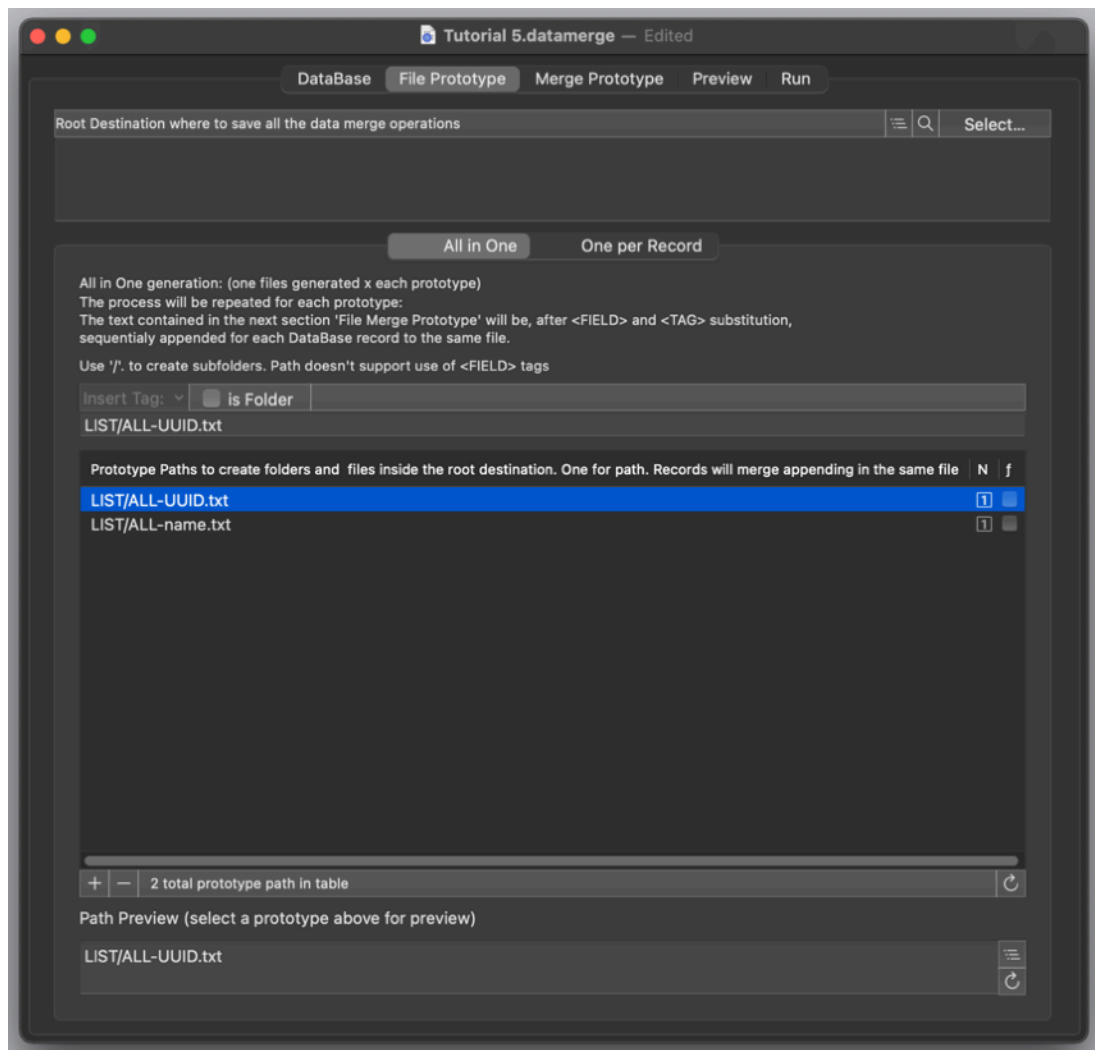
Then we import

The database is now populated

We rename the 2 key fields as 'UUID' and 'name' clicking and editing inside the 'dataBase Field Key' table

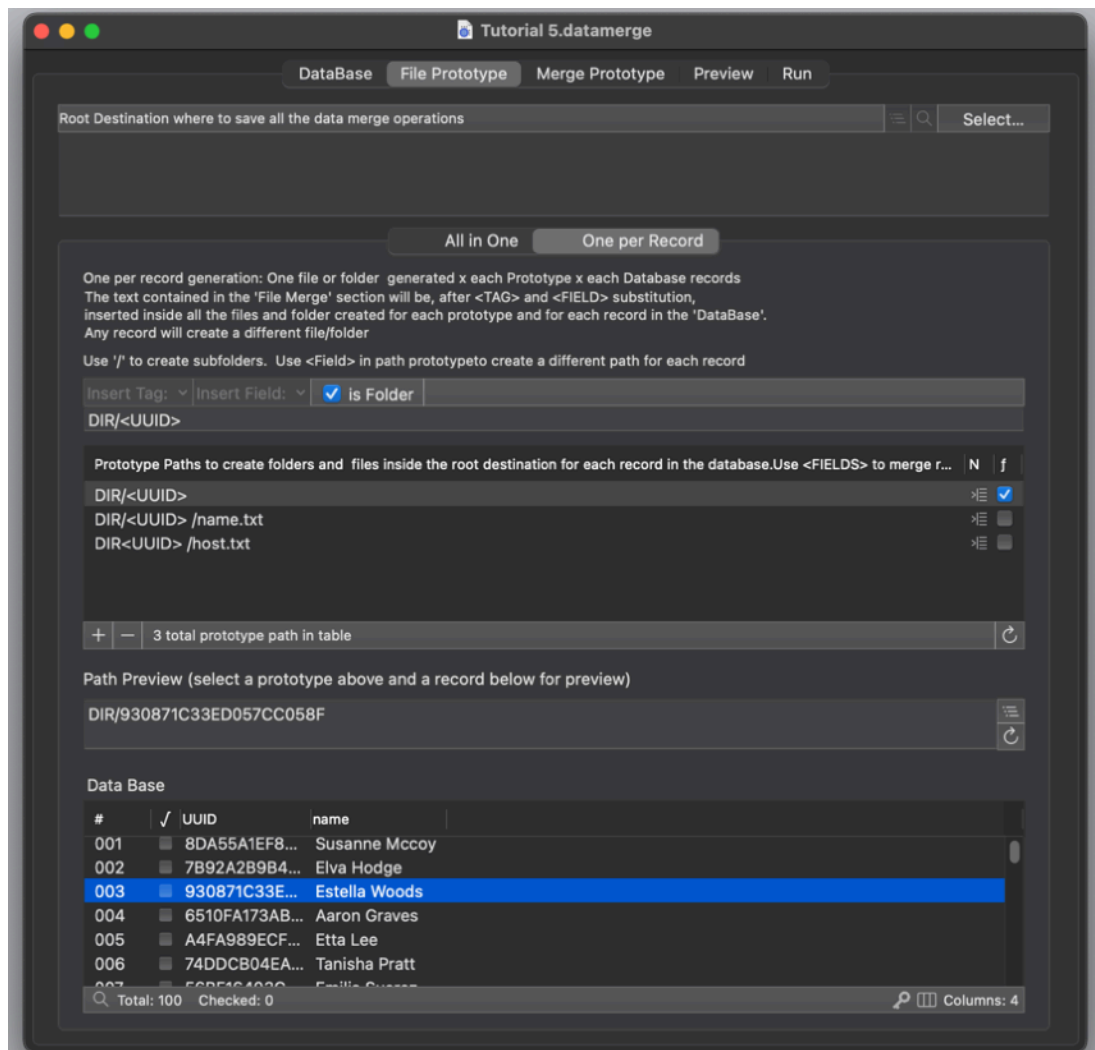


Then we move to the 'File Prototype' area
 We can see in the 'All in One' section there is already the 'LIST/ALL-UUID.txt' path prototype
 We create a new one as 'LIST/ALL-NAME.txt'



Then we move to the 'One per Record' section and create the entry:
DIR/<UUID> and set it as a folder checking the box 'is Folder'

Then we create also:
DIR/<UUID>/name.txt
and
DIR/<UUID>/host.txt



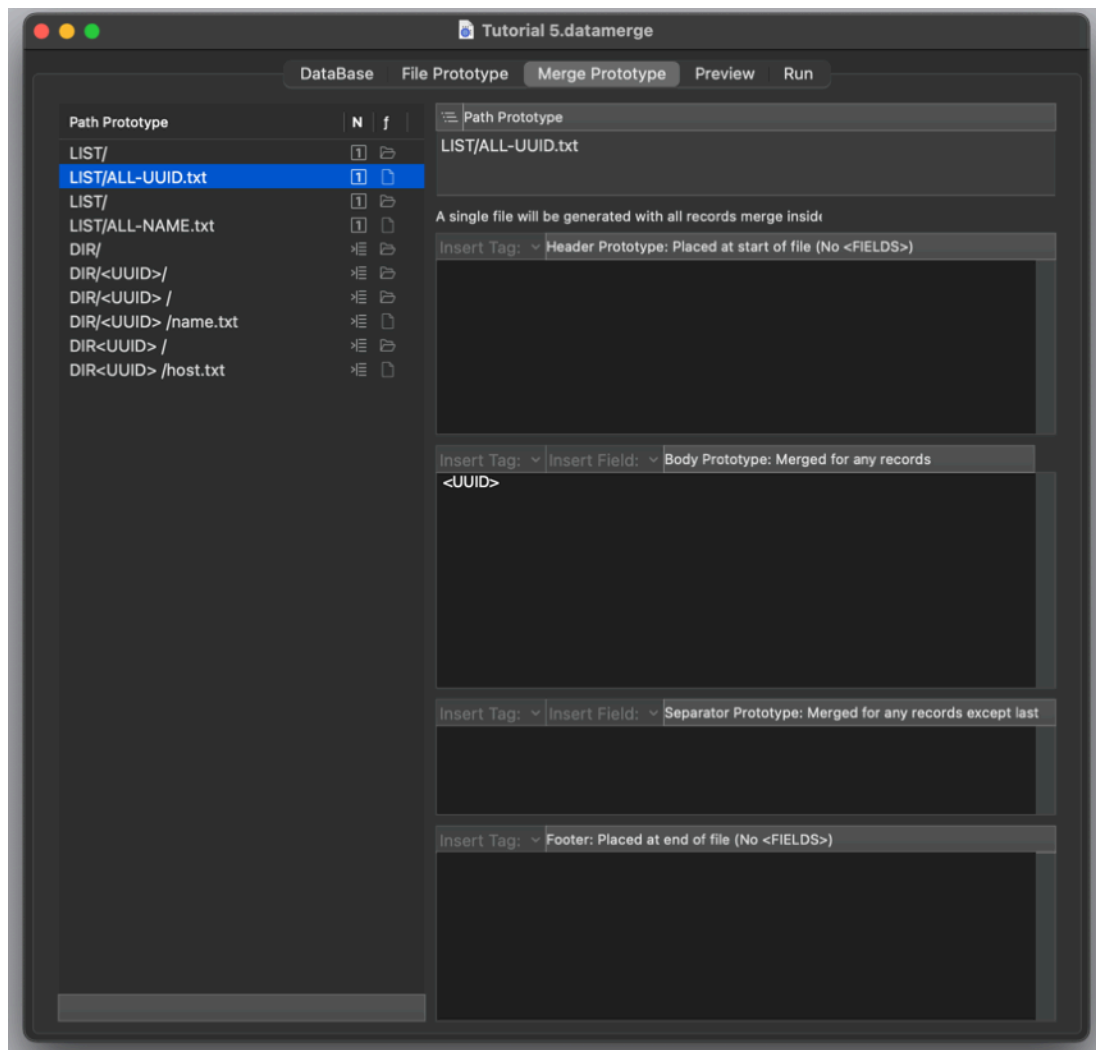
We have finished with the file prototypes

We move to the 'Merge Prototype' area now clicking the correspondent tab

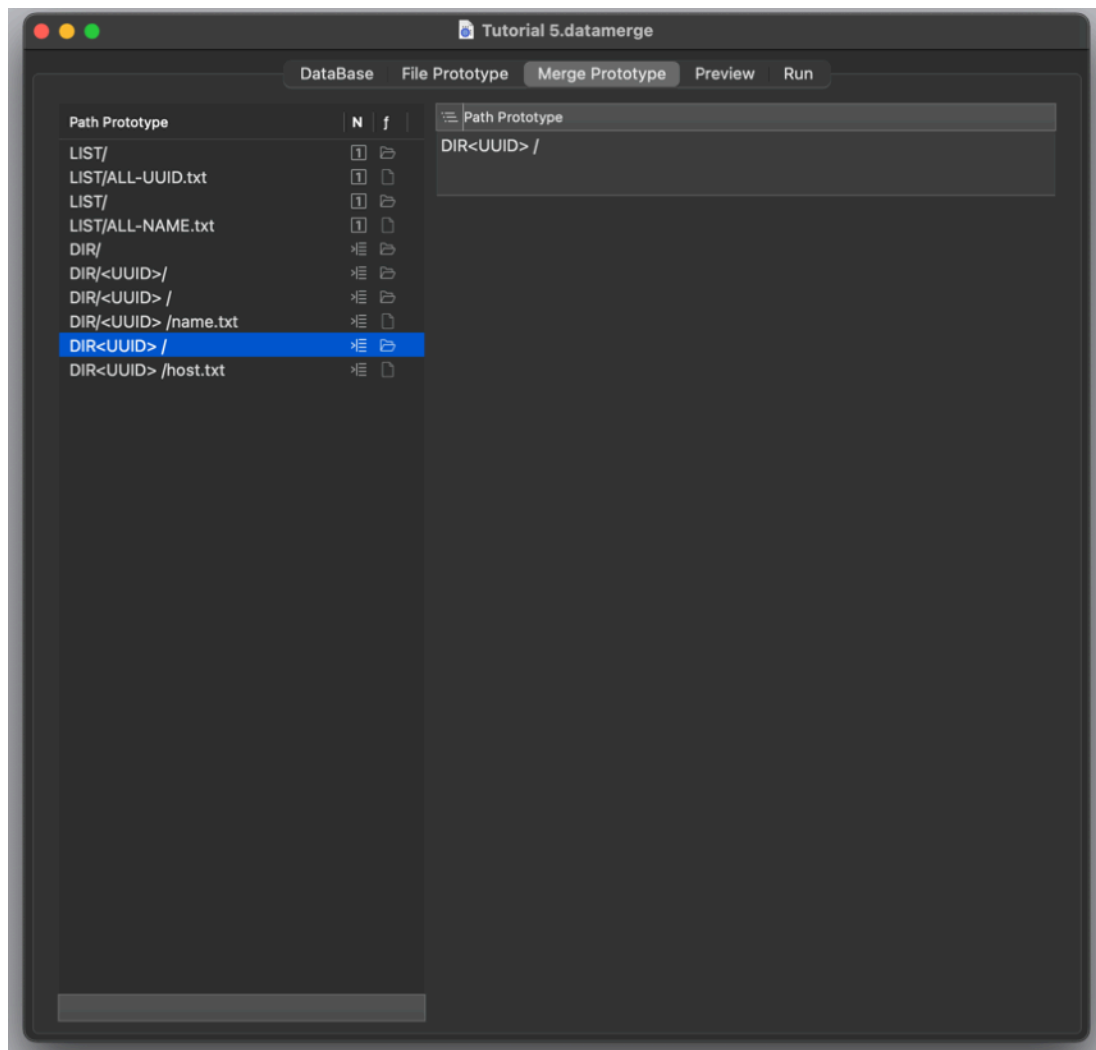
In 'LIST/ALL-UUID.txt' we put in the body:

```
"
<UUID>
"
```

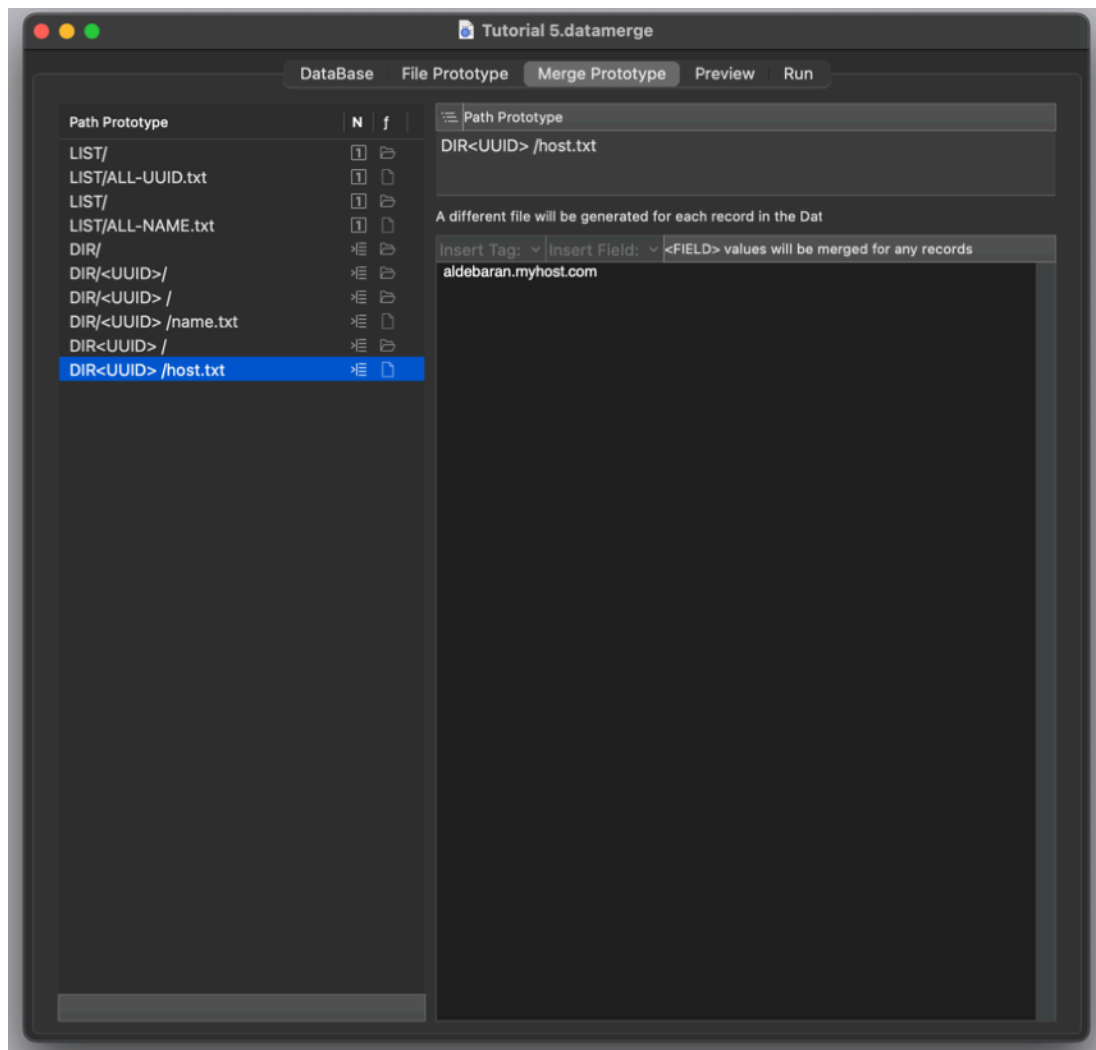
And in the separator we put a return (or all the UUID will be appended in the same row



In the LIST/ALL-NAME.txt we do the same
We put <name> in the body and 1 return in the separator text



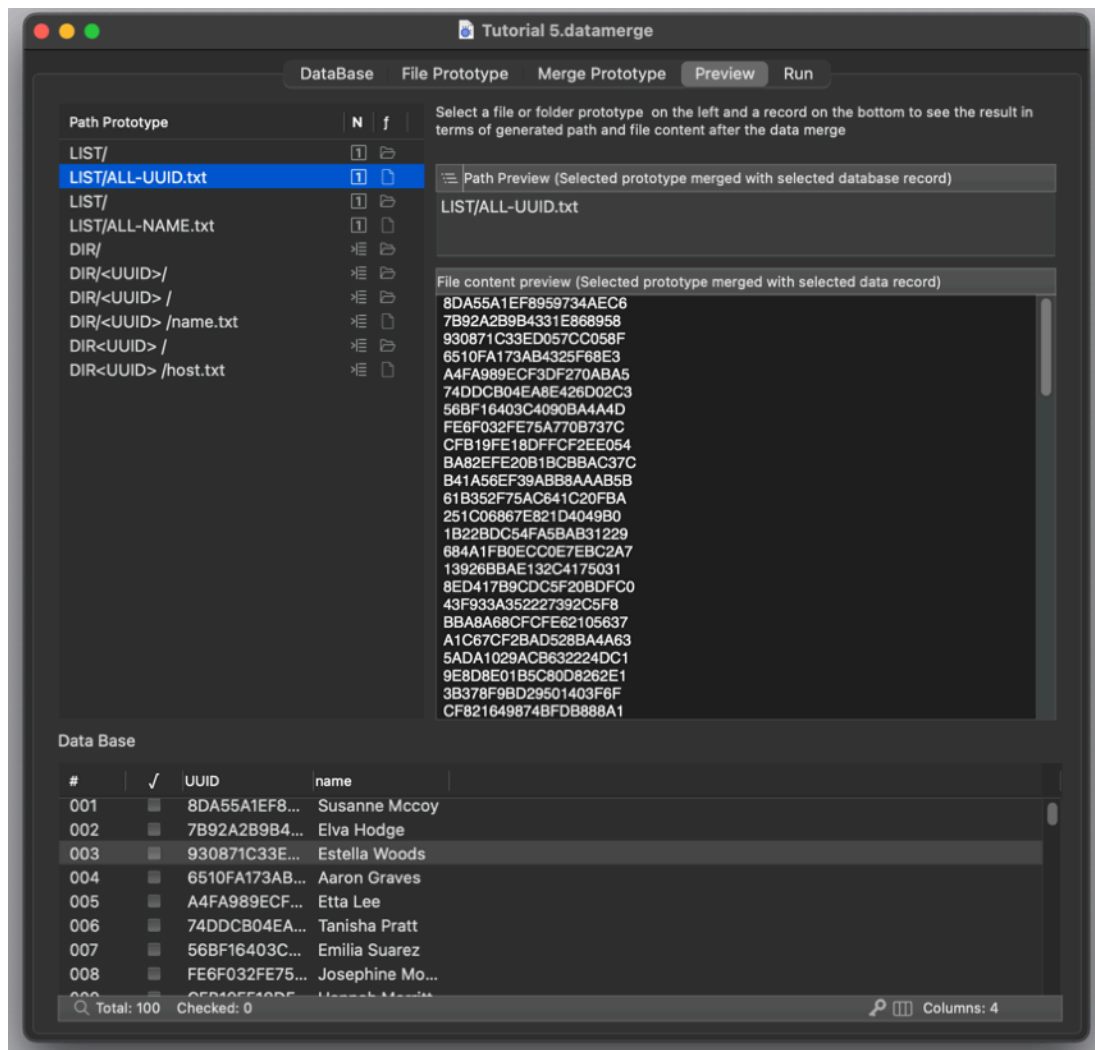
Now DIR/<UUID>/name.txt
We put <name> in the file content



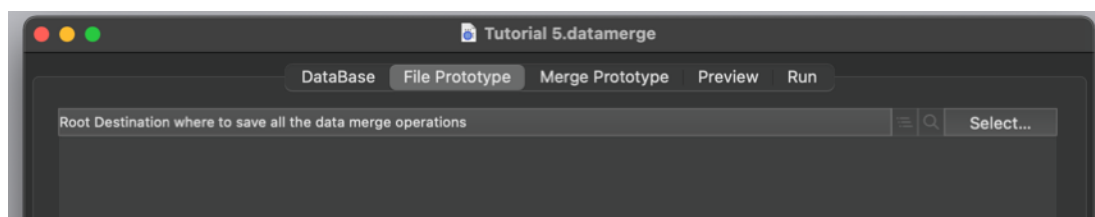
And in
DIR/<UUID>/host.txt we need always the same entry, a constant string, so we put
'aldebaran.myhost.com'

Done

Now we can go in the preview area and see the result preview

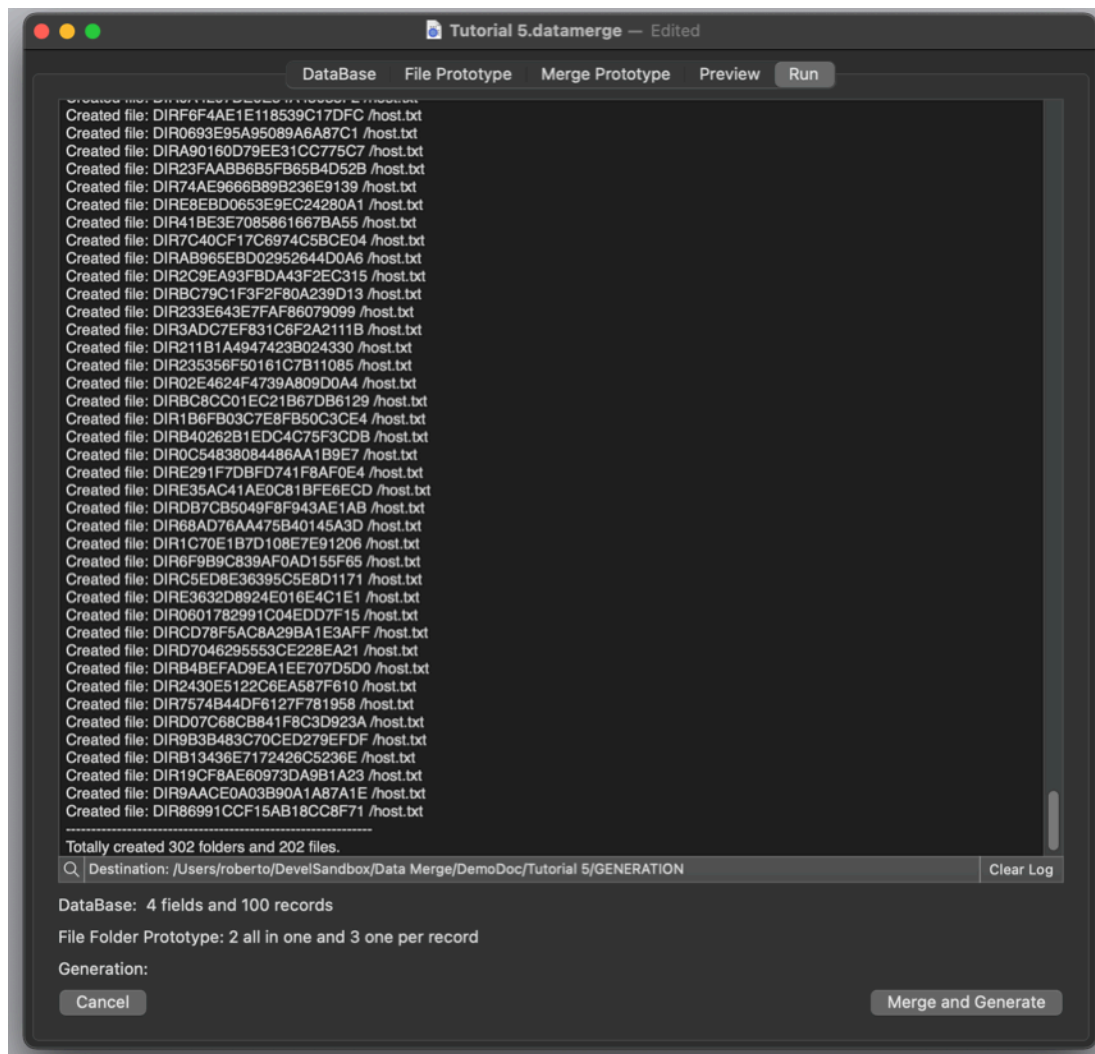


We move to the run section
 We click 'Merge and generate'
 Ops!
 We forgot to set a destination folder
 Go back to the 'File Prototype' area and set it using the 'Select...' button



We go back to 'Run area'
 We check destination is empty showing it
 We generate

Your result is now visible on disk

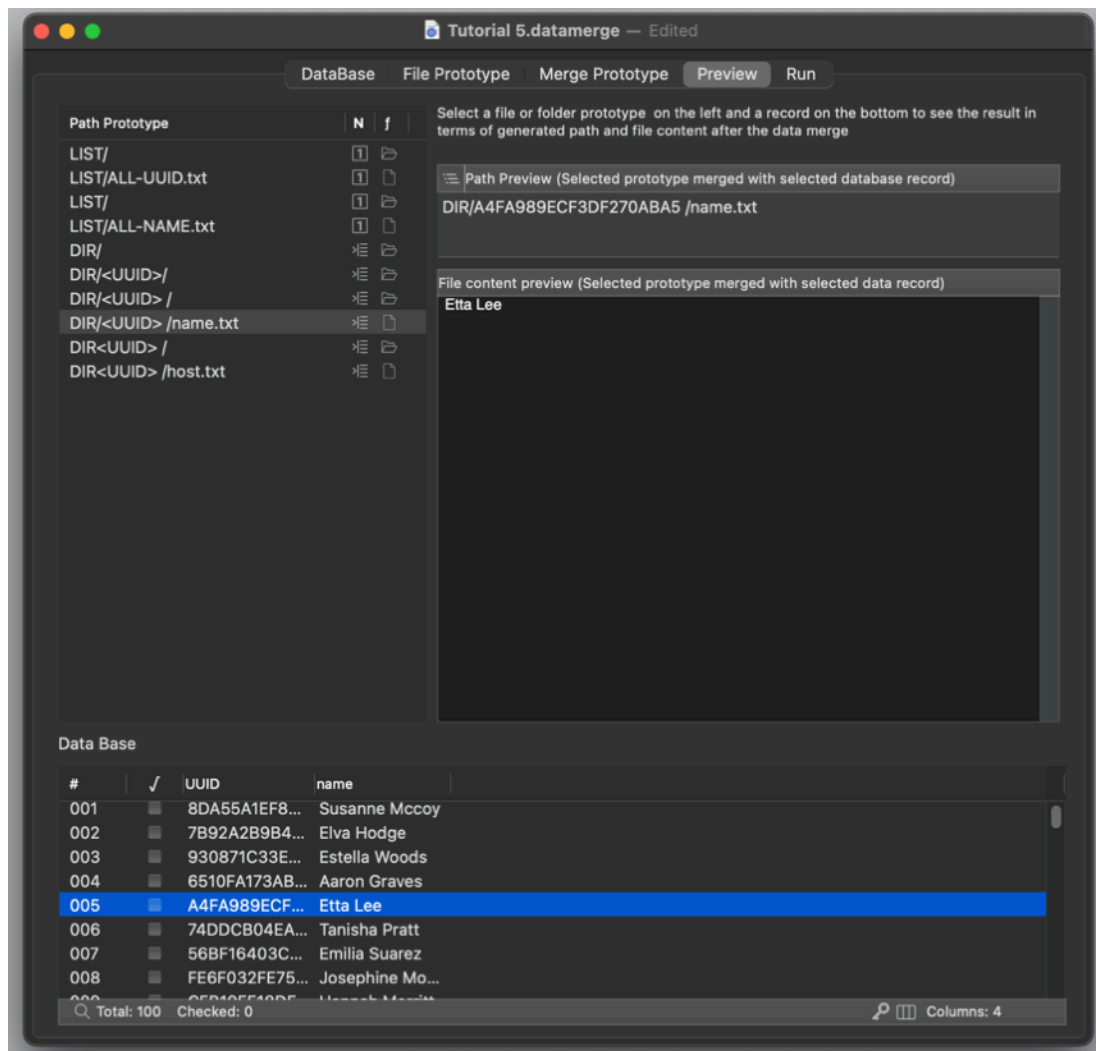


We included also the 'Tutorial 5 (done)' document completed for your reference. You can check you made the same input inside the document you prepared.

Data merge Reference

About the app

Data merge allows to merge the data inside the internal database (data can be easily imported from the most common format) to form representing the layout to give to your data to save them on disk.



You can create 'classical' merge where any records creates a different file on disk where in a text any occurrence of a field tag is replaced by the records values

Or

more complex merge where you can create folders and nested folders at any levels where folders and subfolders name and file name can have fixed or value files names and files where the file content can be different for each records

Or

also create documents with inside content relative also to all records appending to a single file the merge result of all the records using custom header and separators and footers.

The best part you can mix all this to obtain what you exactly need.

Data Merge can in some way work as a powerful text generator creating in a click complex structures of folders and text using the data inside a database and combining it with a complex structure of nested folders and files with an advanced creation of file content

The final results can be of infinite types.

You can create web pages in html and updating it with new data you import inside the database

Prepare customized text for different people with their own different data inside

Prepare text file configurations for server setup

Prepare code in any language with inside complex structure and update them updating or importing the data in the data base

Data merge is a powerful flexible tool. Your creativity is the limit

Document based

Data merge is document based application.

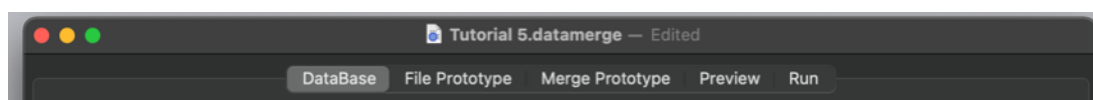
This is the right approach to let you save inside any documents a complex structure of data and form to merge.

Any document will represent a work set you can open update and then generate your merged documents

No limits to the work set you can create

Organization of a document

The main Data merge document has an upper tab bar that allow to navigate in all the five areas the work is divided.



- The 'DataBase' area list all the records to use to merge to generate files and folders. You can easily import/export here.
- The 'File Prototype' area defines the list of file and folders to create ('All in One' or 'One per Record') and name to use for each
- 'Merge Prototype' defines the text to generate and put inside files
- 'Preview' shows the name and content assigned to each folder and file merging them with the Data Base
- 'Run' let you generate and shows the log of the file/folder generation to disk

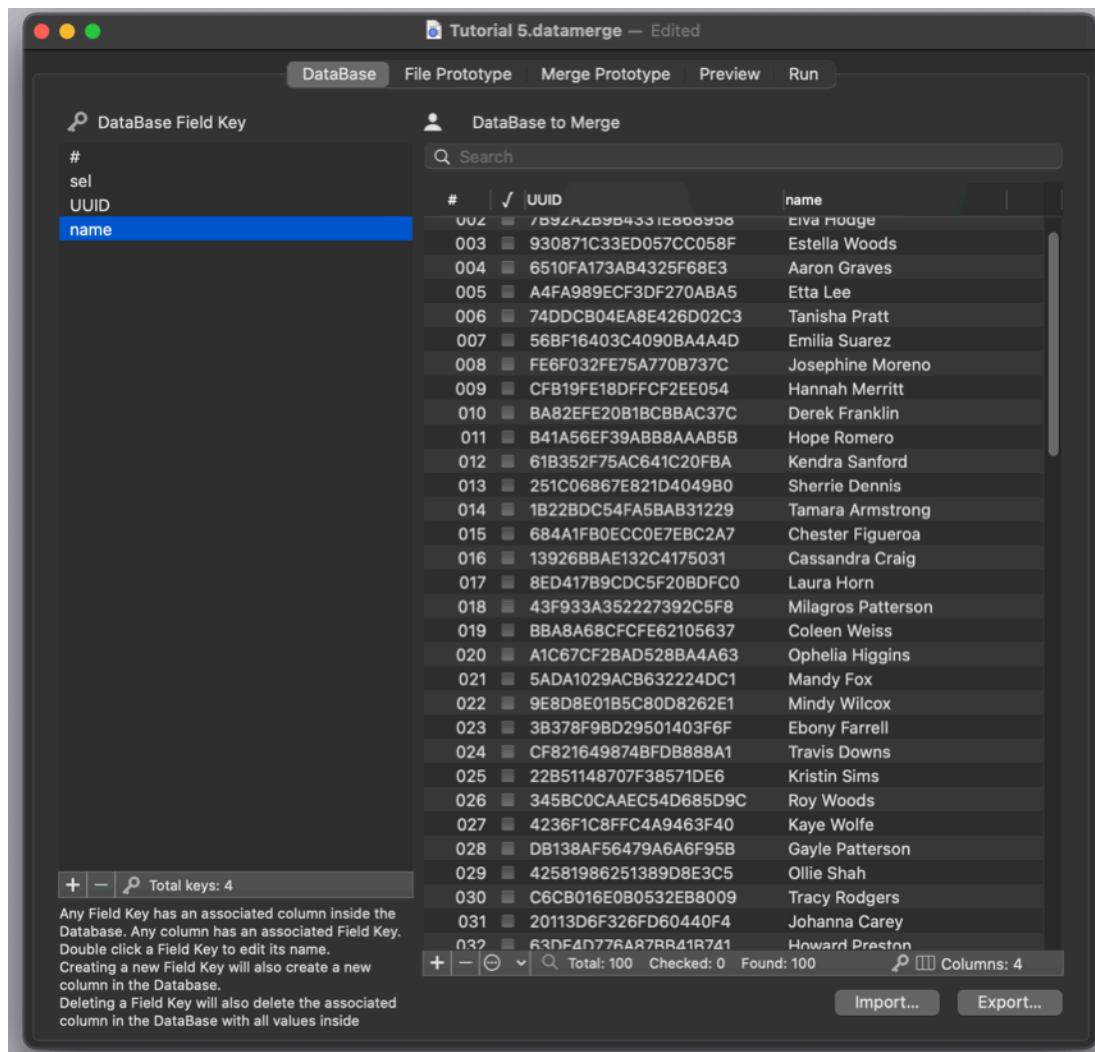
The DataBase area

The DataBase area is a table containing all the records to use in the merge and generation operations

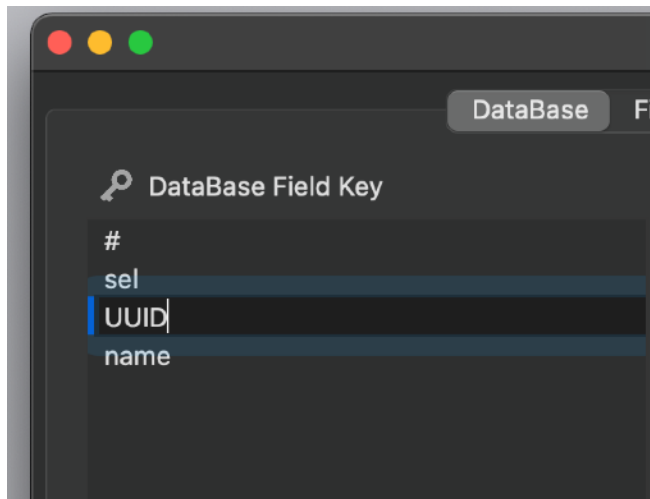
On the right we can see the table with the 'DataBase to Merge'
Each column represents a DataBase field
Each row is a DataBase record

On the left of the 'DataBase to Merge' we can see the 'DataBase Field Key' list
Here there is one key for each column in the 'DataBase to Merge' on the right

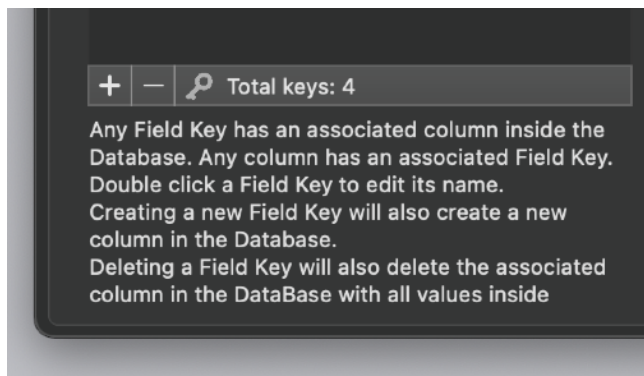
We can see that if we add one key to the 'DataBase Field Key' list, a new column with it's name is added to the 'DataBase to Merge'



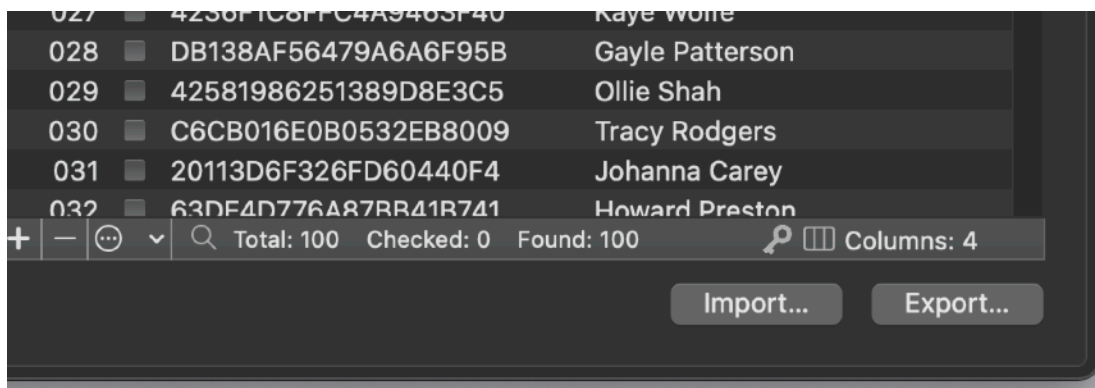
To change the name of a column of the DataBase to merge' we simply double click inside the 'DataBase Field Key' list and edit it



If we want to remove a column from the database to merge we delete the correspondent key
 So the purpose of the 'DataBase Field Key' list is simple, manage the fields (columns) of the database
 You can add and remove Keys from the buttons at the bottom of the key table



Regarding the 'DataBase to Merge' we can create manually records pressing the '+' button at the bottom of the table



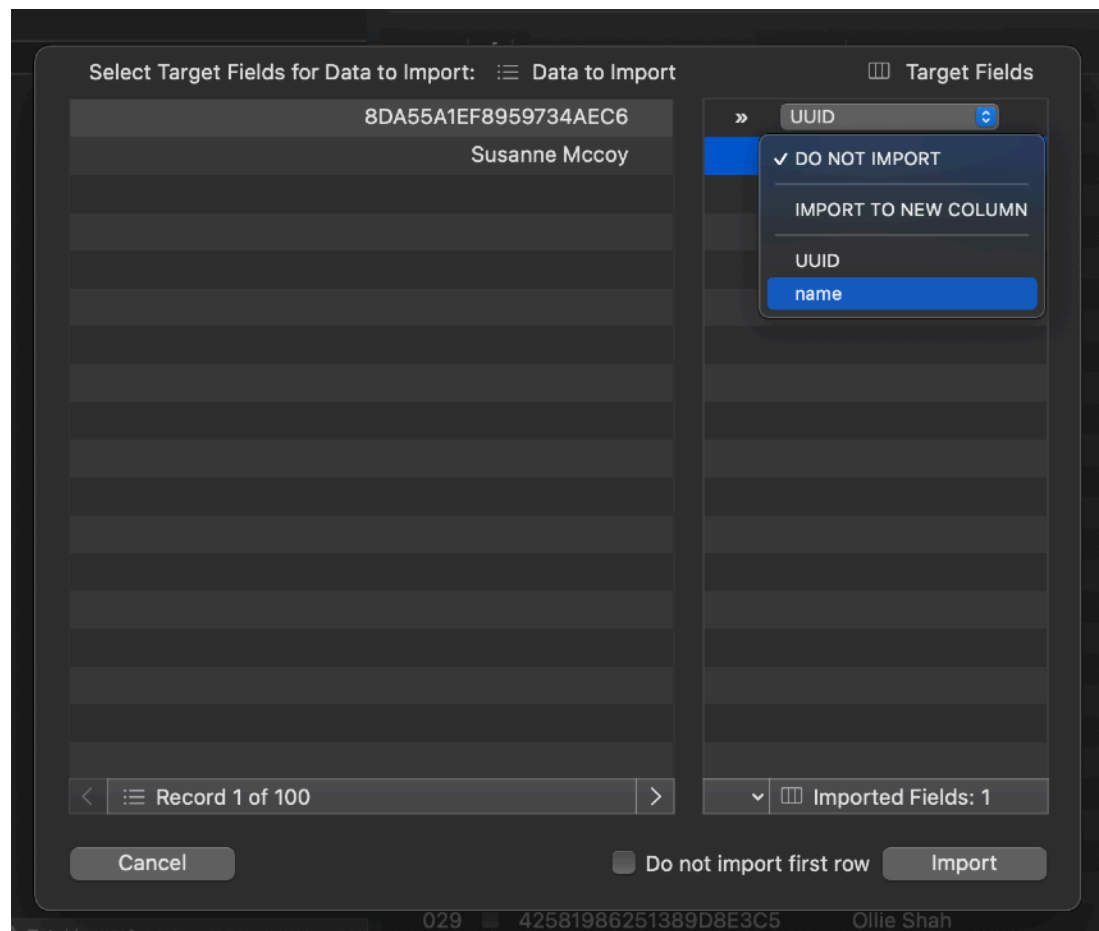
or more easily importing them

Import to the Database

The import function let you import from file on disk and import in various format, tab separated value, comma separated, semicolon separated value.

Usually it works well in auto-detect mode, detecting by itself the format and encoded method used by the selected files.

It can also be instructed manually via the options button in the 'Import' dialog
The a dialog will be opened showing you all the columns (fields) inside the files you are going to import and let you decide what to import and where
(You can import in existing fields or create new one)



Then press import and it's done

The internal 'dataBase' of Data Merge will be filled with the imported data

These data can be used for the merge generation operations

Export from the Database

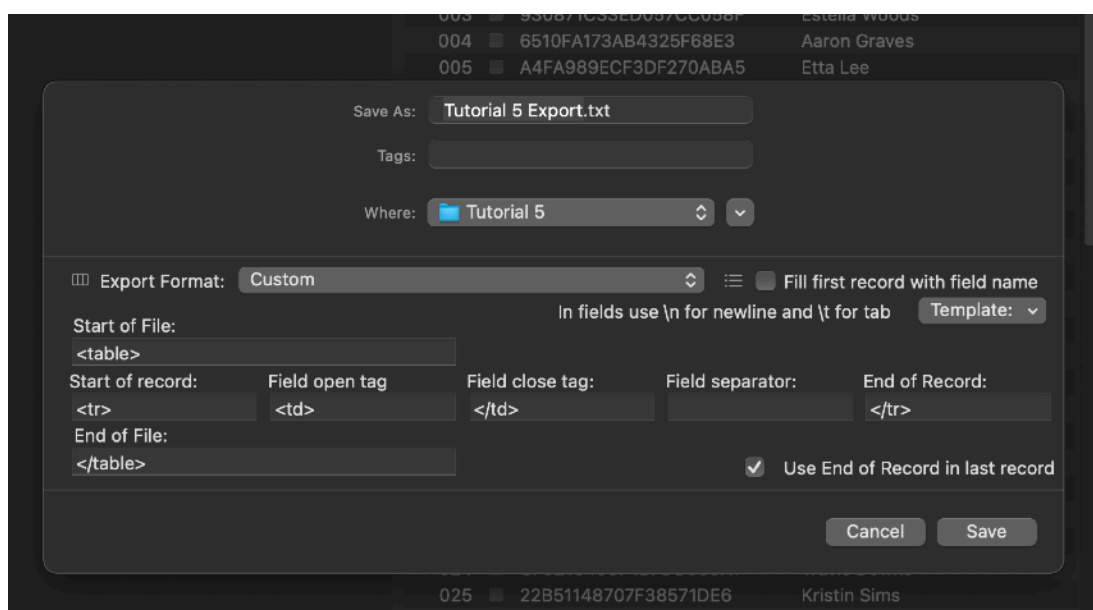
You can also export from the DataBase to an external file to use data in other application

Select export

Then the fields to export

The format to use and in case if use first row for field names

Press Save
Done



The File Prototype area

If the 'DataBase' area defines 'the data' to use in the merge operations

Now you have to define 'where to save' the merge operations

The file prototype area defines 'where to save'

We call it 'prototypes' and not 'file names' because you will define the file and folders name using 'prototypes' where file and folders name can contain also fields of the DataBase in the 'One per Record' section in case you need to create a specific file name for each record

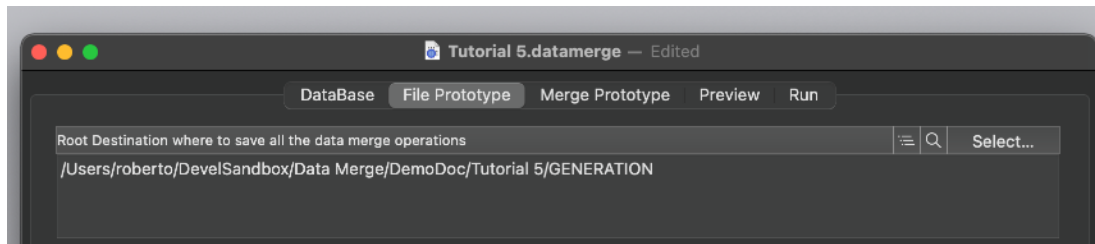
First of all we need to define a root folder where to save all the generated file and folders

Data Merge is a sandboxed macOS application. When you define a root folder inside this zone, you assign read/write right to Data Merge relative to this folder. The Data Merge document can't read write outside of this folder because macOS will not allow it

So select an empty folder you created for the purpose and you are sure this Data Merge document can't interact with other data on your Hard Disk simply because the macOS will not allow it.

So using Data merge is safe.

It will not mixup with other data on your Hard Disk



When you need to generate again and again to the same folder, remember you have to empty this folder manually before generate again.
Data Merge will not overwrite existing files and folders for security purpose.
Data Merge will not empty a folder for security purpose. You have to do it manually.

In Data merge there are 2 kinds of file prototypes:

- The 'One per Record' prototype kind
- The 'All in One' prototype kind

You can have as many as you like of them (both kind) in each document

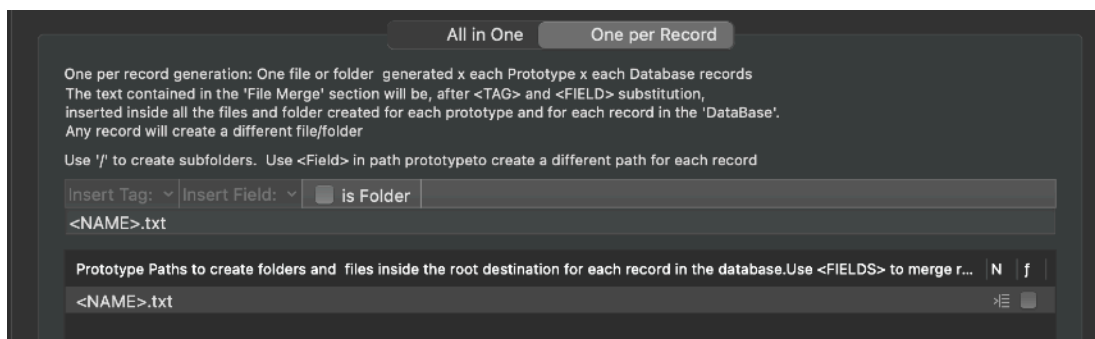
The File Prototype 'One per Record' area

In the 'One per Record' section you can define prototypes that generates a different file path for any record.

If we have, as example, a small DataBase with only 3 records with a field named "NAME"

and we use the tag <NAME> in the file name prototype creating the path:
"<NAME>.txt" in the 'One per Record' area

then, if we have the 3 record in our small database and value of the 'NAME' field is for the 3 records "John","Mary" and "Susy"



when we 'Merge and Generate'
We will obtain 3 files named "John.txt", "Mary.txt" and "Susy.txt"

More..

We can create not only specific file names but even more complex directory structures:

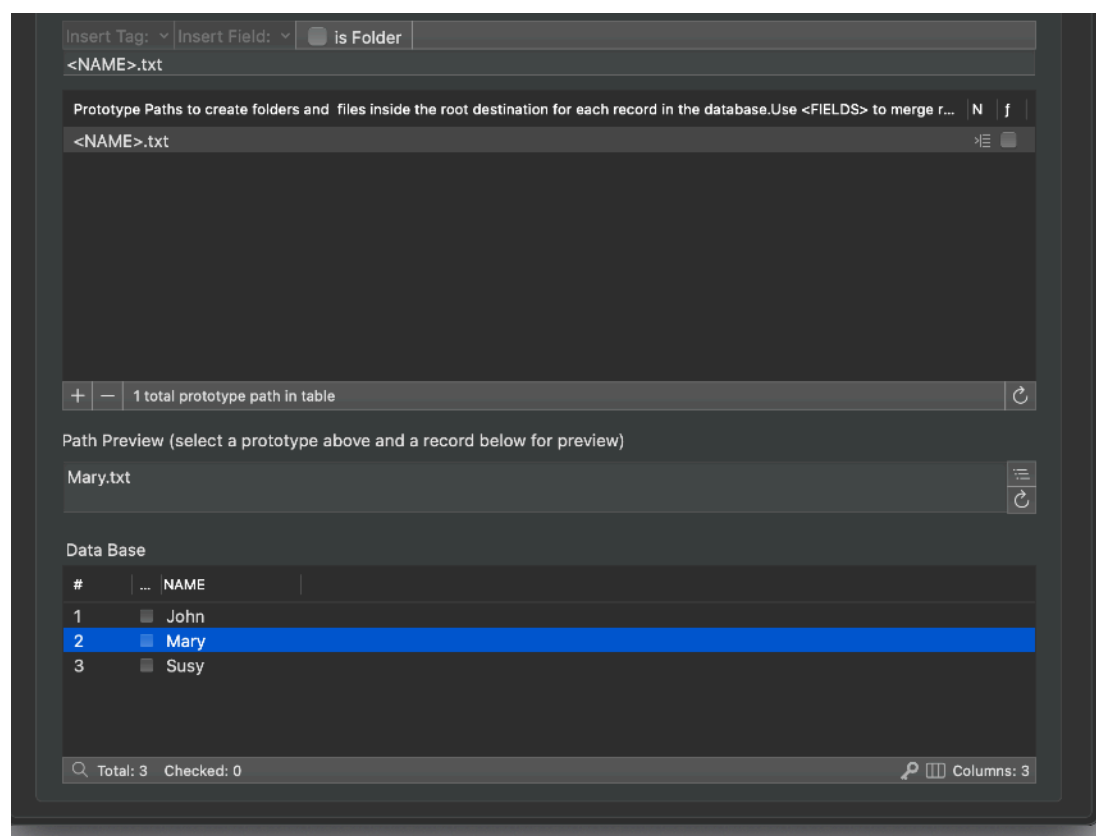
As example simply writing "<NAME>/name.txt" in the prototype will create a different directory for each record named as the 'NAME' field and inside a document named "name.txt"

There is no limit to this, you can create as many nested directories you like using fields name inside the path

What is important is to avoid collisions, if you generate an equal path for all the records, you will obtain a single file generation (file and folders are not overwritten by Data merge), so only the first matching file will be generated

When in the 'File prototype' area, at the bottom part all the record are visualized inside a mirror of the DataBase table.

If you select a record in this table and prototype in the 'Prototype Paths' you can see a preview of the name path obtained merging the prototype with the record is visualized.



If you did the things in the right way (including a file name in the path) you will see any path preview changing as you change the selected record. This can give you an idea of which files and folders you are going to generate before generating it.

You can add as many path prototype as you like, defining folders nested at any level.

No need to put a separate line for folders that have to incorporate other folders, Data MERGE will create all the necessary folders to realize the specified paths if you specify long nested path in one single prototype.

The File Prototype 'All in One' area

In the 'All in One' area a single file will be generated for all the records
All the records can merge their content in this file

This is the reason 'Field tag' can't be used to generate paths starting from this prototype.

If you are creating a 'All in One' path you can't differentiate the path for each record because only one file or folder path will be created for prototype

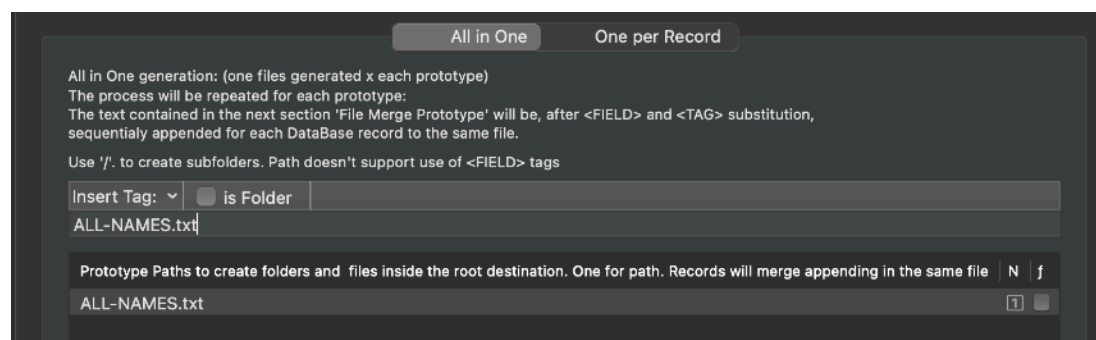
You can use only a restricted set of tags, 'Year', 'Month', 'Day' and so on
Inside this file (if it is a file) all the record will be merged sequentially using a method we will see in the successive paragraph 'Data merge'

This can be useful when you need to generate, maybe in addition to 'One per Record' also a series of files containing each one all the records, as an example a single file with a name of each record per row

So in that case we can create a single prototype named:

ALL-NAMES.txt

It will be created only once for all the record but it will have special features when it's the time to fill it with content



We will see in the 'Merge Prototype' area how to make this file contains all the data we need

The Merge Prototype area

The 'Merge Prototype' area shows on the left side all the files and folder that will be created.

It can show more entry then the one specified in the 'File prototype' area because it list also all the parent parent folders that you could have indicated in a single prototype

An icon shows if a file will be generated and filled as 'All in One' or 'One per Record'.
In the latter case you can't see the true final file name but just a prototype because the final file name is determined processing the file path prototype with the value of each records generating a different final path for each records

Selecting a path prototype on the left you see file content on the right changes

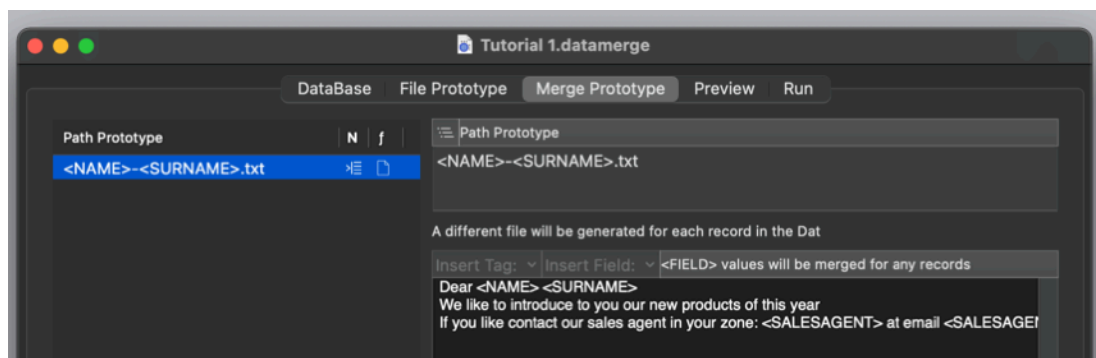
For folders there is not a lot to say, the folder path prototype is visualized
For file you have to specify what to put inside any file

'All in One' and 'One per Record' have different way to specify the content of the file

One per Record

For 'One per Record' you have a single body
Remember that if you have N numbers of records in the DataBase the document will try to generate N different file on disk per each prototype of this kind 'One per Record'
The content can be differentiated also putting <FIELD> tags inside the file content
As an example if we write inside the file:

Dear <NAME> <SURNAME>
We like to introduce to you our new products of this year
If you like contact our sales agent in your zone: <SALESAGENT> at email
<SALESAGENT-EMAIL>

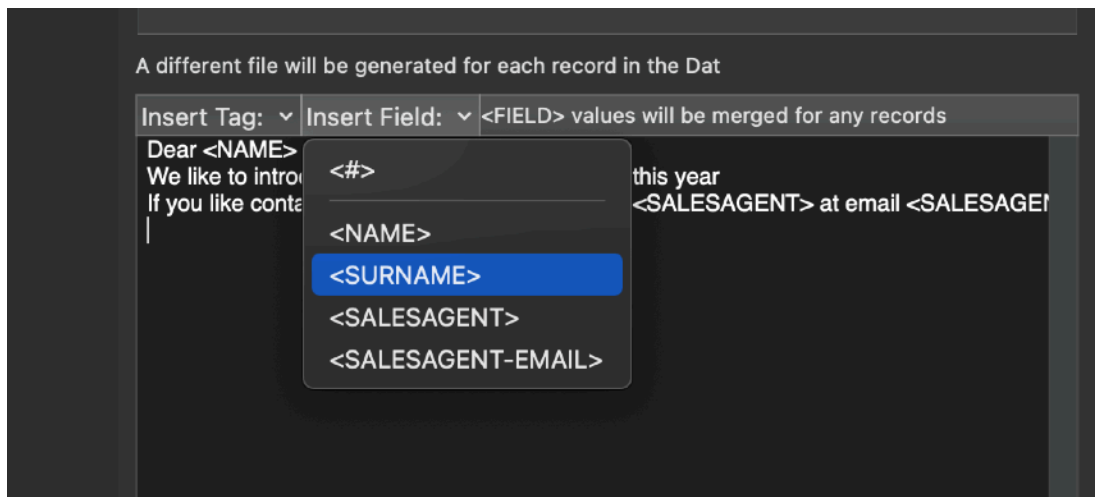


Then, inside the content prototype, the field tags:

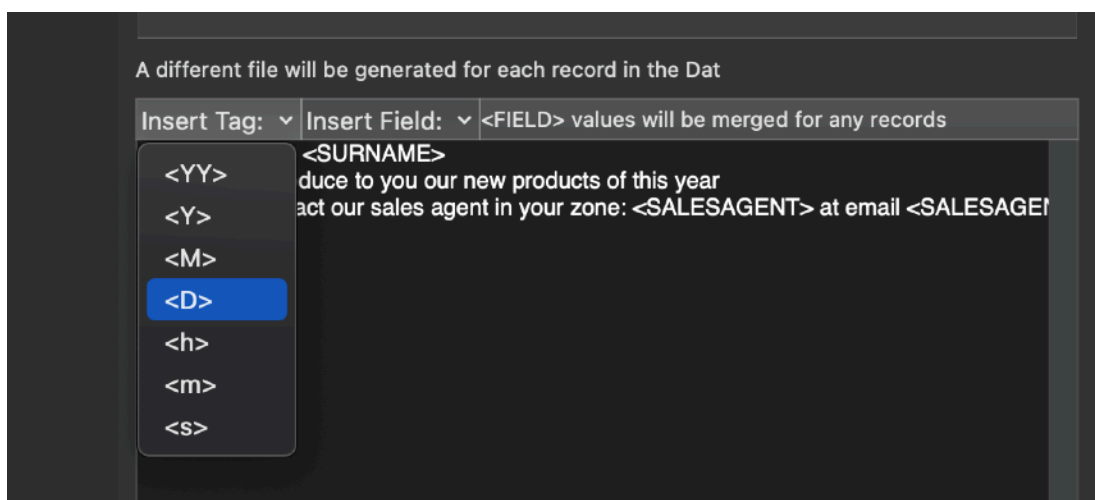
<NAME>
<SURNAME>
<SALESAGENT>
<SALESAGENT-PHONE>

Will be substituted with the fields valued for any records

A popup at the top of the text edit zone allows to insert the correct field names without mistakes



Also a popup is available to insert common fields (not database dependent)



<YY> for full year of merge date time as 2019
 <Y> for short year of merge date time as 19
 <M> as for month of merge date time as 07
 <D> as for day of month of merge date time as 24
 <h> as for hour of merge date time as 08
 <m> as for minute of merge date time as 08
 <s> as for second of merge date time as 54

These pop-up are available only when the cursor is active inside the relative fields

All in One

For 'All in One' the name is a fixed one and all the record will be merged inside a single file

You can specify

- 1)
 The header prototype : (USED ONCE)
 Just one per file at the very start and not containing any record <FIELD> value.
 It can contains only standard <TAG> values as:

<YY> for full year of merge date time as 2019
<Y> for short year of merge date time as 19
<M> as for month of merge date time as 07
<D> as for day of month of merge date time as 24
<h> as for hour of merge date time as 08
<m> as for minute of merge date time as 08
<s> as for second of merge date time as 54

2)

The body prototype: (REPEATED FOR ANY RECORDS)

It will be inserted inside the same file, each per record

It can contains both <FIELD> values and standard <TAG> value as:

<YY> for full year of merge date time as 2019
<Y> for short year of merge date time as 19
<M> as for month of merge date time as 07
<D> as for day of month of merge date time as 24
<h> as for hour of merge date time as 08
<m> as for minute of merge date time as 08
<s> as for second of merge date time as 54

3)

The separator prototype: (REPEATED FOR ANY RECORDS)

It separates each record insertion in the file

It can contains both <FIELD> values and standard <TAG> value as:

<YY> for full year of merge date time as 2019
<Y> for short year of merge date time as 19
<M> as for month of merge date time as 07
<D> as for day of month of merge date time as 24
<h> as for hour of merge date time as 08
<m> as for minute of merge date time as 08
<s> as for second of merge date time as 54

4)

The Footer prototype: (USED ONCE)

Just one per file at the very end of file and not containing any record <FIELD> value.

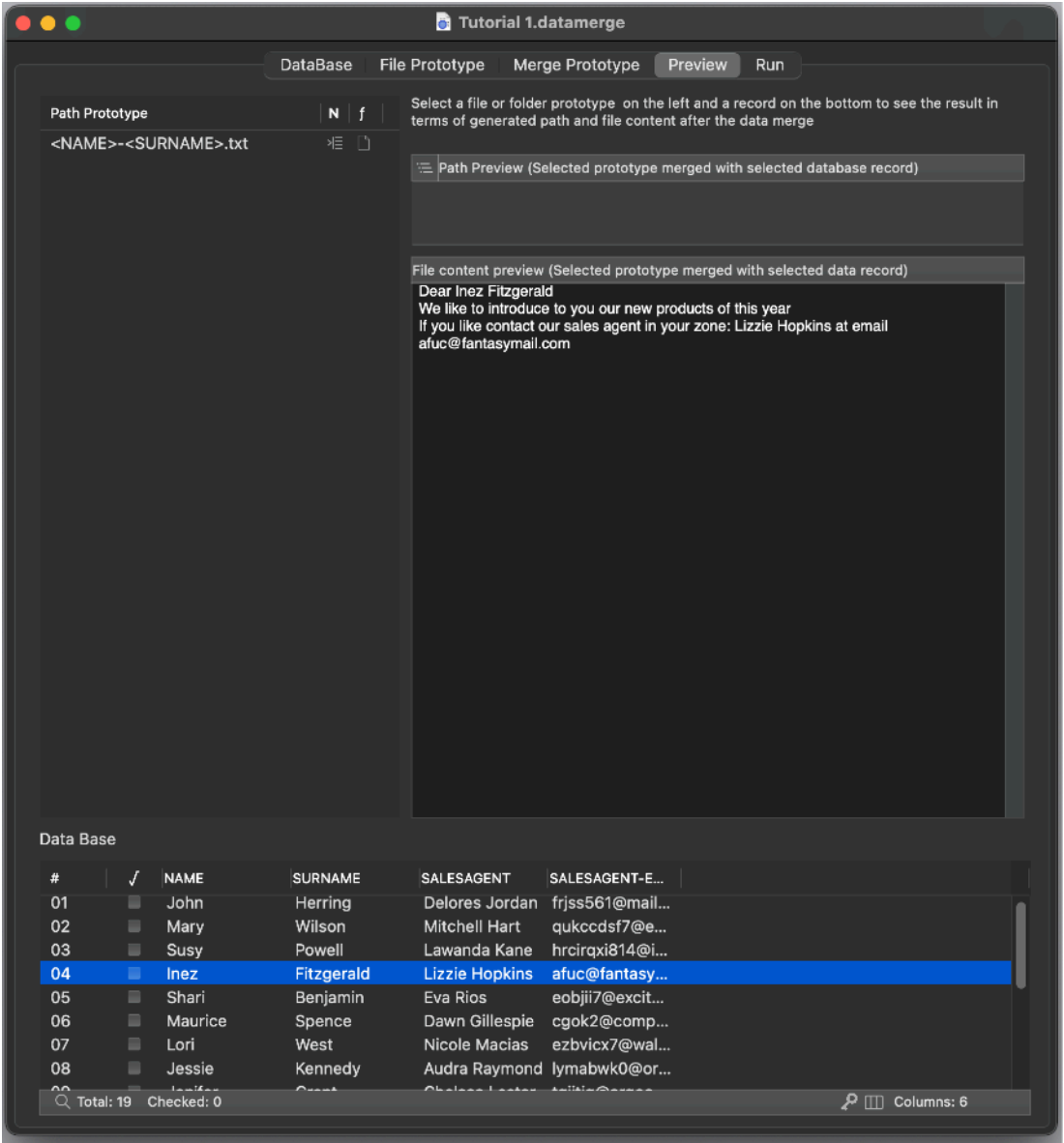
It can contains only standard <TAG> values as:

<YY> for full year of merge date time as 2019
<Y> for short year of merge date time as 19
<M> as for month of merge date time as 07
<D> as for day of month of merge date time as 24
<h> as for hour of merge date time as 08
<m> as for minute of merge date time as 08
<s> as for second of merge date time as 54

The Preview area

The preview area allows to view the final result in terms of final name of file and folders saved on disk and inner content of text files

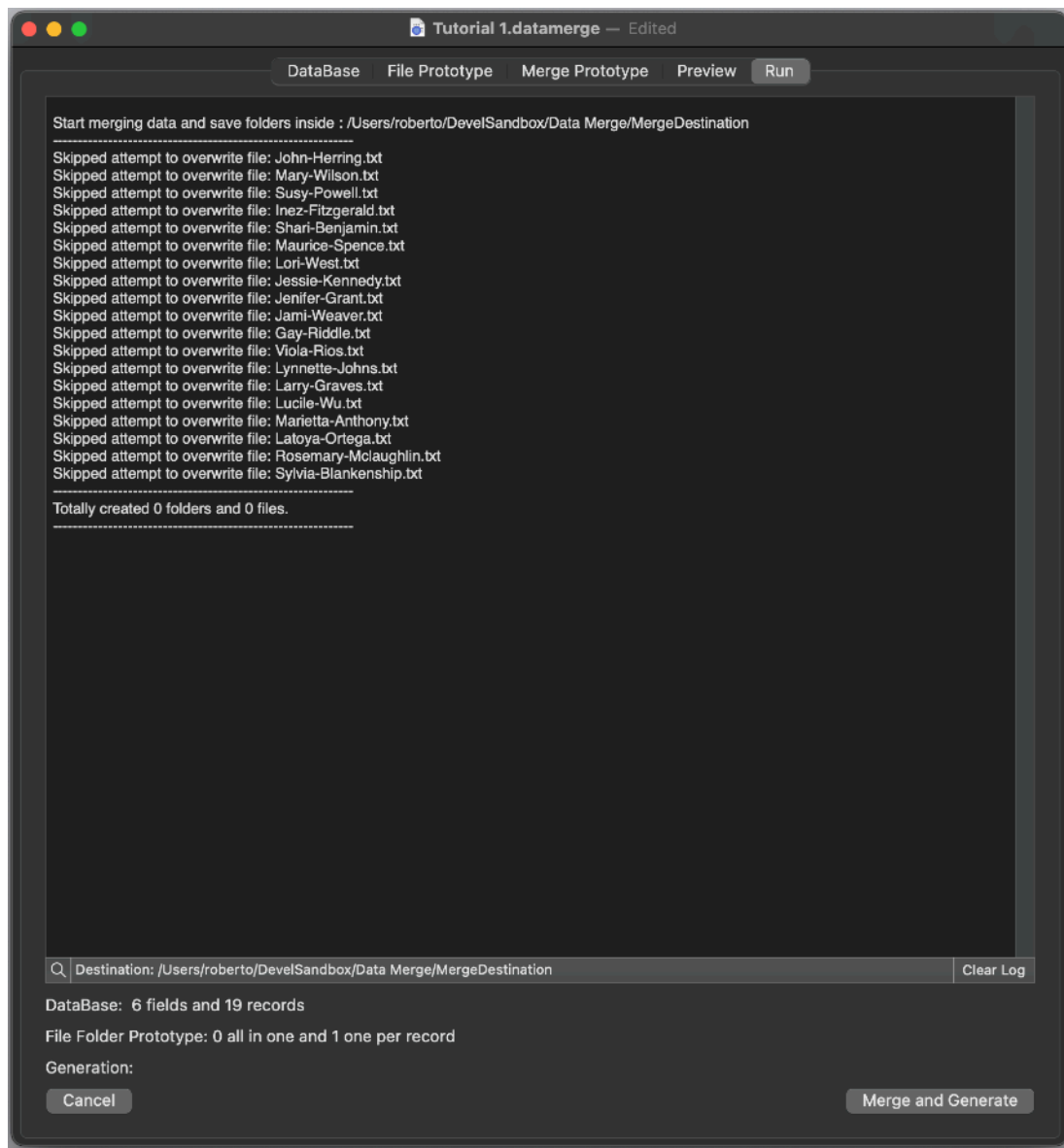
Selecting a resulting file prototype and a record (for Each per Record prototype) in the bottom DataBase table the result merge is visualized



For 'All in One' generations the result doesn't change selecting a record in the bottom table

The Run area

The Run area allows to launch the generation for the merge and save the files folders on disk, all in a single click
A log will report the list of created folders and file and in case all the error of the process



Modern MacOS features

Data merge adopts and makes use of the latest Apple technologies

They are used in the standard Apple way, so nothing new to learn if you already know how to use them.

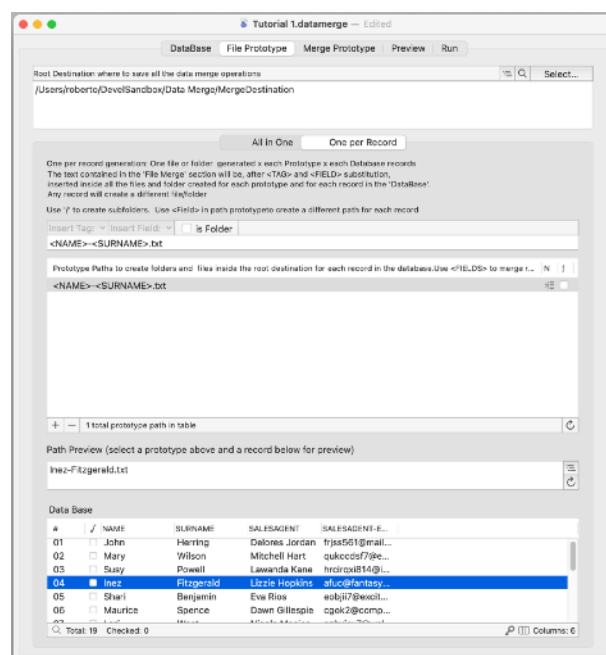
- Resume – the app will reopen at the point and state the user left it included opened documents and unsaved one
- Auto save – the app saves using the macOS autosave functionality
- Versions – the user can look at previous versions of documents and restore to any earlier version (just select Revert to Saved from the menu or the down arrow at the right of the title window, it appears only if you move the cursor over)

To use the Versions you can select 'File > Revert To > Browse All Versions' from the menu

- Full Screen - the user can switch at any time to full screen using the native macOS full screen mode and commands (use the upper right icon in the window to go full screen and bang the mouse in the upper part of the screen to re-obtain the menu and eventually clicking the standard icon to exit the full screen mode)

Dark Appearance Mode

Data merge is fully compatible with Light and Dark Appearance Mode. You can switch at any time from one mode to another.



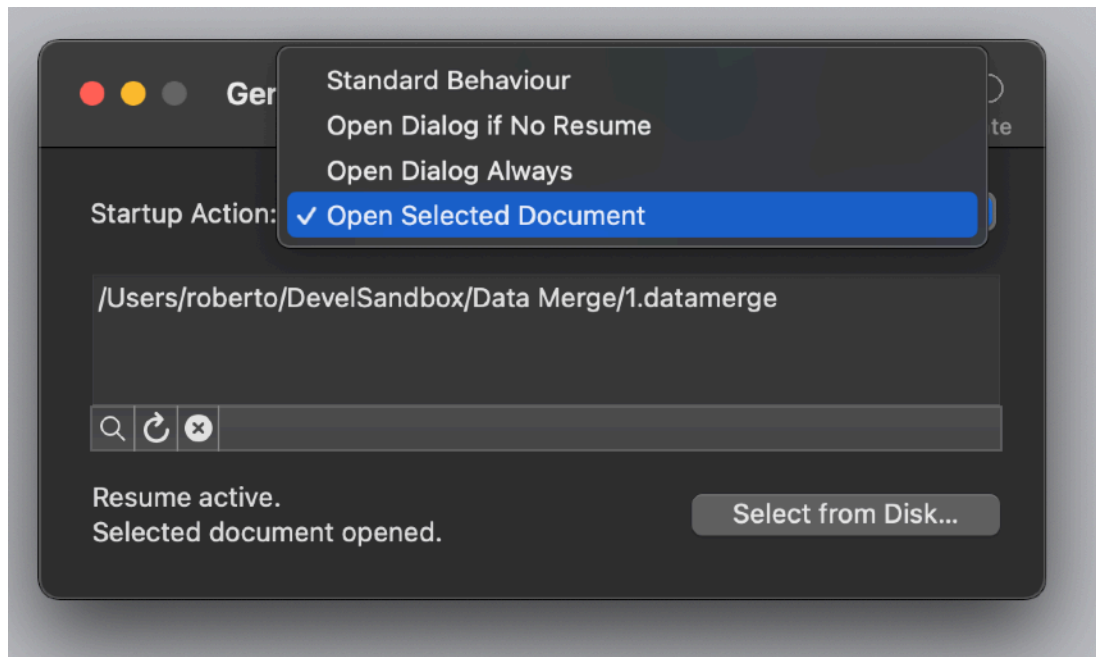
Preferences

General

Specify the action to do at startup:

- Standard behaviour

- Open Dialog if No resume
- Open Dialog Always
- Open Selected document



To select a document press the select from disk button and select it from the open dialog that will follow

At any successive relaunch Data Merge will execute the option selected.

The additional 3 buttons at the foot of the edit field let you:

- Show in the finder the selected file
- Test open the selected file as it will be done at the next application launch
- Delete the bookmark reference to the file (you will need to reselect a file)

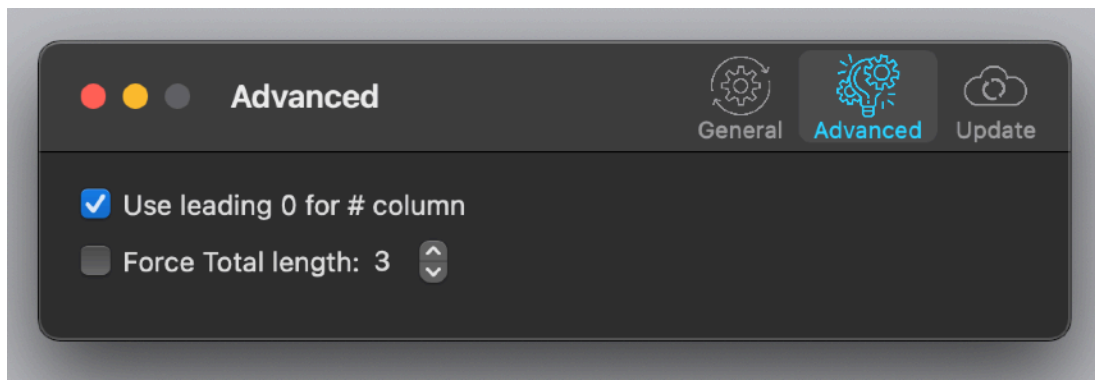
The reference to the file is taken using scooped bookmarks and they work even if you move the file on your disk, even if your file is in the trash.

To update the path display in case the file was moved, click the 'show in finder' or the 'test open' button

Advanced

These settings allow to change the format of the <#> field with is generated automatically when importing or creating new records

- Use leading 0 adapted up to the max field length it calculates
- Force total length to the length you manually insert



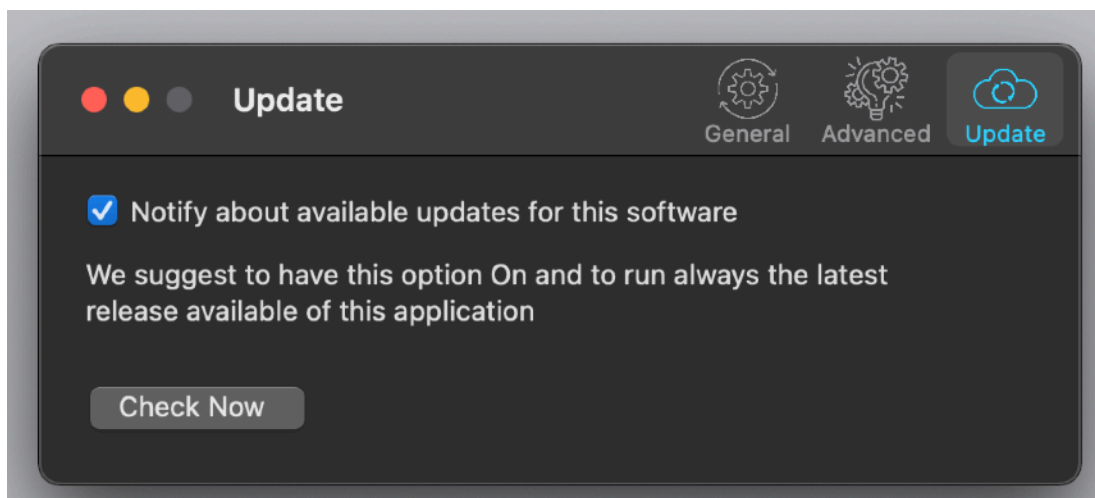
Update

*** Update section in the preferences is not available in the App Store Release (if you purchased via the App Store you don't need it, to obtain an update use the App Store Update function)**

Data merge can inform you if an update is available.
We suggest to keep this option on.

When you check if an update is available, the application ping the Tension Software server and receive an answer about the last release available, the application can in that way know if its obsolete or not.

In case the application will ask you if you like to download an upgrade.



If you download the upgrade, you need to install it.

A common error is to download an upgrade and install it maintaining the old application somewhere on the hard disk.

Then using to open the documents sometime the old application and sometime the new one. This cause some problems*. To avoid it simply install your application in the place dedicated to it, the Application folder. When you install an update on the standard Application folder, the system will ask if you want to replace the old one with the new, answering yes will install the new one replacing the old.

*If the application says the document you are trying to open was created with a newer version of the application, probably you have two different release of the application on your

Hard Disk and you are trying to open a document with the older version after having modified the document with the new one. When you receive a similar message open the about box inside the application and see if you are running the last release. In case download it, install it and USE it!

Help

Data merge provides a user guide in PDF.

Support

You can also obtain support using the 'Support Email...' command. An email will be prepared using your email client with the correct address to send to.
Yes, we answer to your emails.

Data merge is a Commercial Program

Data merge is available via the Apple App Store (fully enabled) and via our site (in an unlicensed version)

In case of the App Store release, a license is already included with your App Store purchase and you don't need to buy a license

If download from our site or on the web elsewhere, you can use our software for a test period of 10 days
After that you are required to buy a license to fully enable it and be legally authorized to continue to use our software

Licensing Data merge

If you are using a version from the Apple App Store:

a license is already included with your App Store purchase and you don't need to buy a license

If you are using a version download from websites:

you can buy a license to use our software using the 'Buy License' command in the 'License' dialog.

You will open in your browser our license web page on our web site

From there you can buy a license to use Data merge using one of the payment service we provide. It easy, fast, and secure and all most important form of payment are accepted.

We carefully selected or international reseller to be sure the buying experience for our customer will be as best as possible and without any problems.

After purchase you will receive from Tension Software a license email containing your full name and a license code

Open the License dialog with the command 'license' under the 'Data merge' menu and insert these data in the License dialog. The software will become fully licensed and fully enabled for future use.

Name and license code are remembered by the program and don't need to be re-inserted at successive launch.

In case you move on a new Mac you need to re-insert your name and license

The license is valid for a single Macintosh. You can purchase for two Mac as long as they are not used at the same time, as example a desktop and a laptop used by the same user.

About Tension Software and Ecleti

All the software asset by Tension Software is now under Ecleti

New company name but still the same owner and main developer (Roberto Panetta)

So for the users nothing changes. We provide the same software and the services.

All the user licenses are valid and assistance is provided as before to all our users.