Chapter 5

Making Tables and Importing Data into MySQL

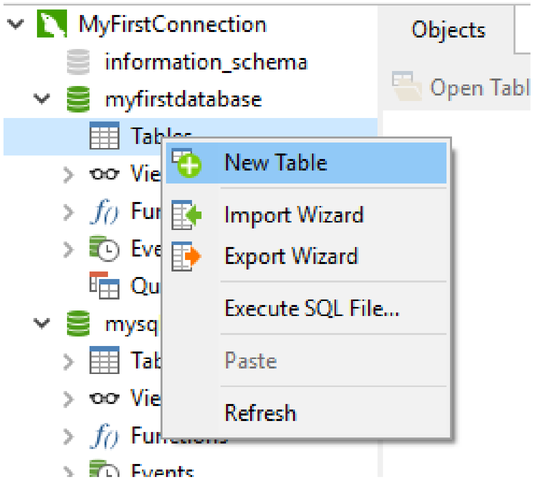
What you will learn:

How to make new tables and import data in MySQL using the Navicat and Sequel Pro front ends, and the MySQL LOAD DATA INFILE command.

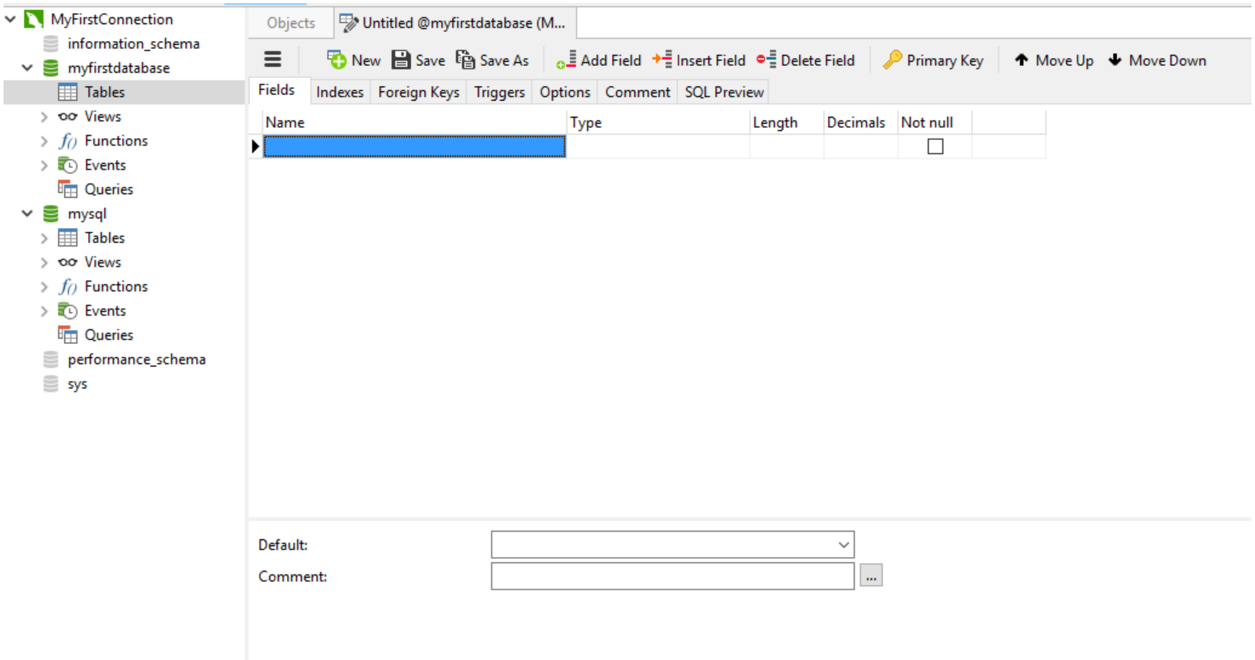
For this tutorial, we assume you have created a connection to MySQL using either Navicat or Sequel Pro. See the tutorial **Exploring Navicat and Sequel Pro** if you need to do this. For a general discussion on MySQL table creation and import, see Chapter 5 of *The Data Journalist*.

We'll start with the process of making a table to which data will be added later.

To begin, in Navicat, right click on the Tables item within the listing for your database, in the table of contents area, and choose New Table from the menu.



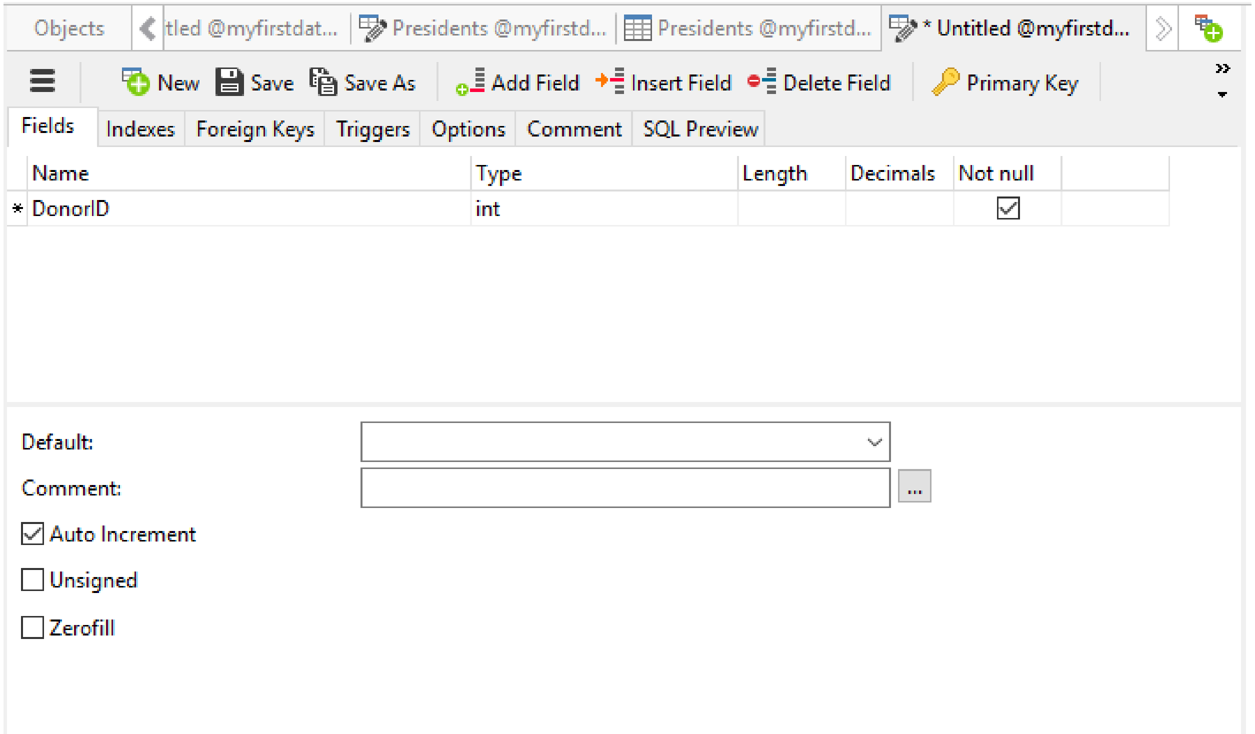
This will open the table creation dialogue. You can also click on the New Table icon.



On this screen we can enter the names for the fields in the table, set the datatypes, set options such as the number of decimal places in decimal and floating type fields, whether a field can contain null values, and the default value of a field, and set the primary key for the table, if there is to be one.

We'll work with the demonstration data on political donations to which you were introduced in Chapter 5. Delimited text files containing the three demonstration tables are available for download on this book's companion website.

First, we'll create the Donor ID field using an integer data type, make it the primary key by clicking on the Primary Key icon, and make it an auto increment field by clicking on the auto increment check box. This means the automatically be populated by a number, starting from 1 and counting upward, as each new record is added.



	Name	Type	Length	Decimals	Not null
*	DonorID	int			☑

Default:

Comment:

☑ Auto Increment

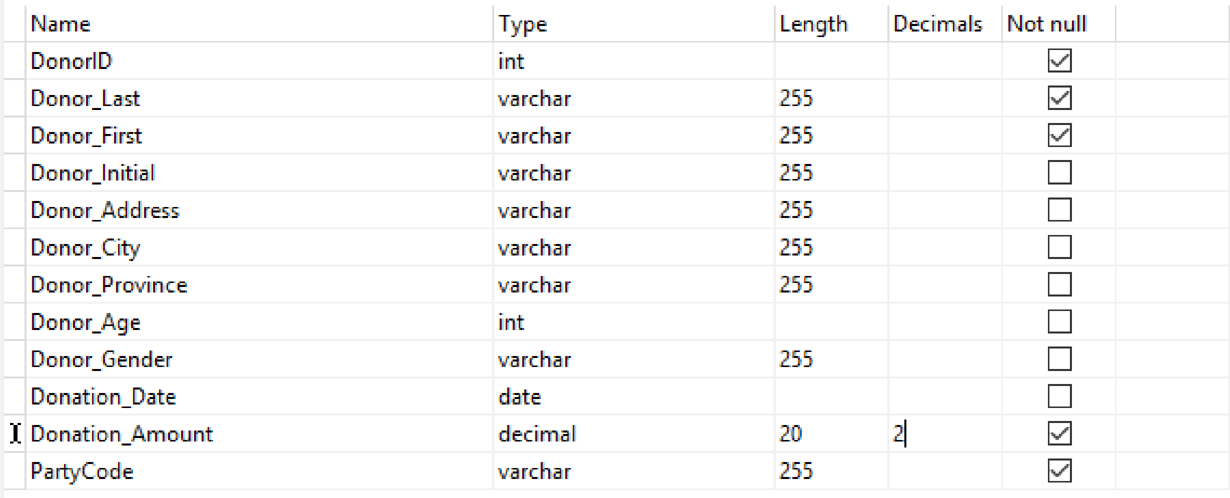
□ Unsigned

□ Zerofill

If you wish the field to be the primary key for the table, click on the Primary Key icon.

To add another field, click on the Add Field icon.

In the image below, we've added the remainder of the fields, each with a data type.



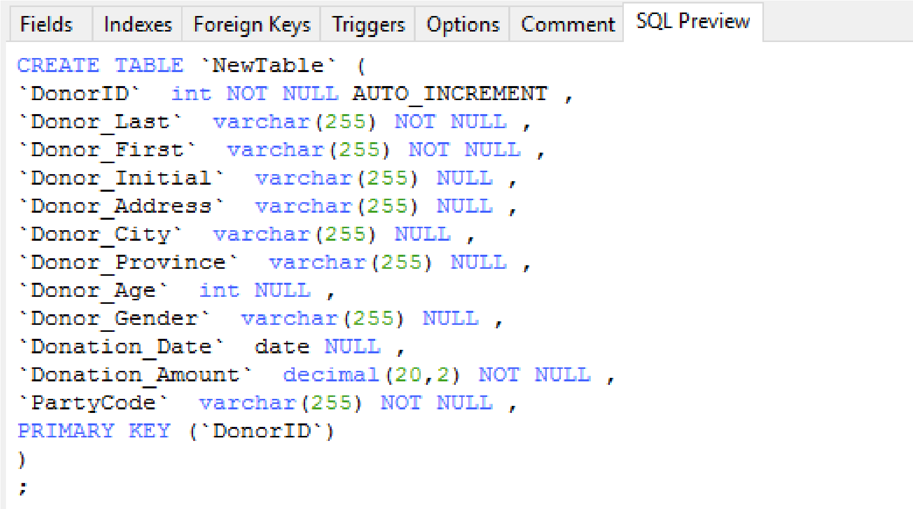
Name	Type	Length	Decimals	Not null
DonorID	int			☑
Donor_Last	varchar	255		☑
Donor_First	varchar	255		☑
Donor_Initial	varchar	255		□
Donor_Address	varchar	255		□
Donor_City	varchar	255		□
Donor_Province	varchar	255		□
Donor_Age	int			□
Donor_Gender	varchar	255		□
Donation_Date	date			□
Donation_Amount	decimal	20	2	☑
PartyCode	varchar	255		☑

Several of the text fields use the varchar data type as they will have different lengths and varchar fields take up only space taken by the data in each row. The maximum length is left at the default value of 255.

The Donation_Date field uses the Date type as we want to enter actual dates. The Donor_Age field is set as an integer field as ages will always be whole numbers, and the Donation_Amount is set as a decimal field, to ensure the greatest precision.

Please see Chapter 5 for a full discussion of data types.

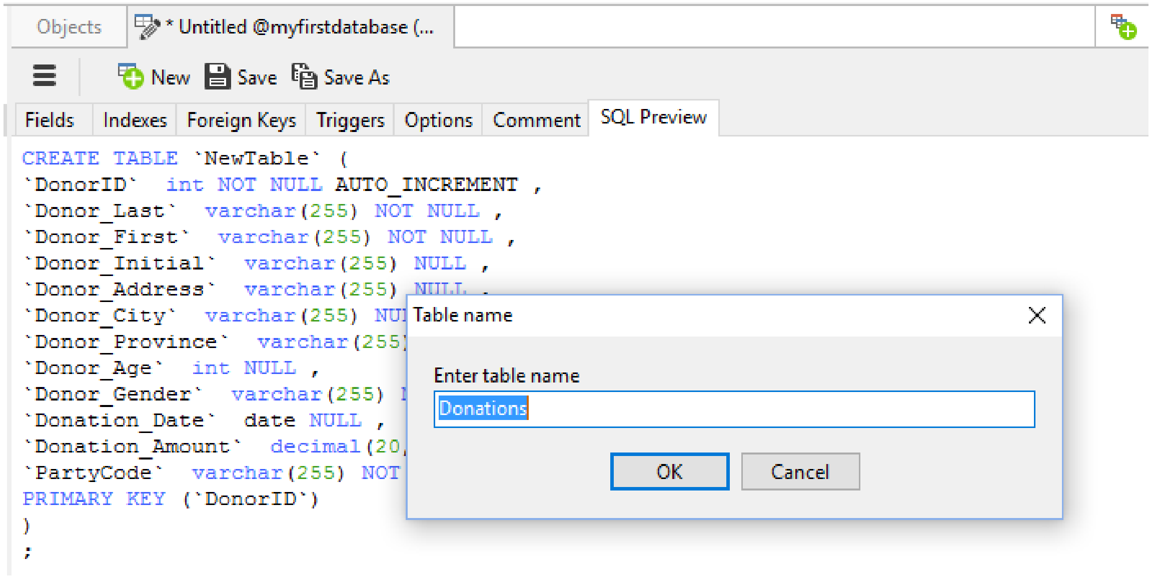
When you enter fields in this way, the front-end writes the necessary SQL table creation statement. You can see the statement in the SQL Preview tab. This is a great way to learn the syntax.



```
CREATE TABLE `NewTable` (
`DonorID`  int NOT NULL AUTO_INCREMENT ,
`Donor_Last`  varchar(255) NOT NULL ,
`Donor_First`  varchar(255) NOT NULL ,
`Donor_Initial`  varchar(255) NULL ,
`Donor_Address`  varchar(255) NULL ,
`Donor_City`  varchar(255) NULL ,
`Donor_Province`  varchar(255) NULL ,
`Donor_Age`  int NULL ,
`Donor_Gender`  varchar(255) NULL ,
`Donation_Date`  date NULL ,
`Donation_Amount`  decimal(20,2) NOT NULL ,
`PartyCode`  varchar(255) NOT NULL ,
PRIMARY KEY (`DonorID`)
)
;
```

Of course, at this point, we haven't given our table a name, so the create table syntax is using the generic NewTable name. However, we can give the table a name, and create it at the same time via the Save dialogue.

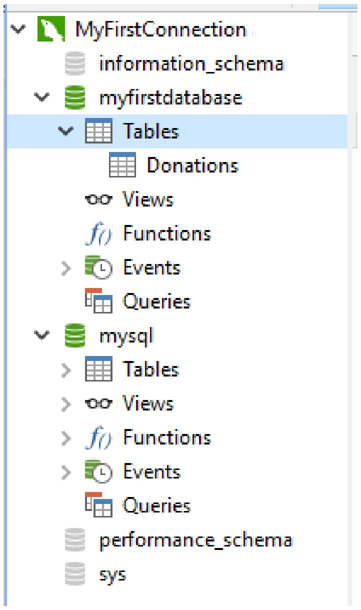
When you click Save, you will be prompted for a name.



```
CREATE TABLE `NewTable` (
`DonorID`  int NOT NULL AUTO_INCREMENT ,
`Donor_Last`  varchar(255) NOT NULL ,
`Donor_First`  varchar(255) NOT NULL ,
`Donor_Initial`  varchar(255) NULL ,
`Donor_Address`  varchar(255) NULL ,
`Donor_City`  varchar(255) NU
`Donor_Province`  varchar(255)
`Donor_Age`  int NULL ,
`Donor_Gender`  varchar(255)
`Donation_Date`  date NULL ,
`Donation_Amount`  decimal(20,
`PartyCode`  varchar(255) NOT
PRIMARY KEY (`DonorID`)
)
;
```

Enter table name: Donations

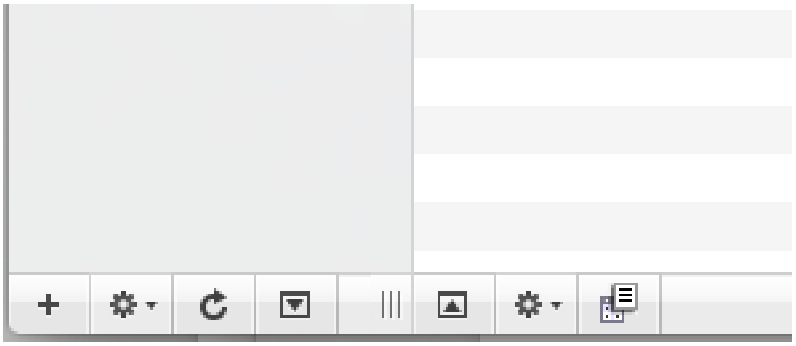
When you click OK, the table-creation statement will be run in the background, and the new table will appear in the table of contents.



If you double click on the table name, you will open it for editing. You can type data directly into the table, or use the LOAD DATA INFILE SQL command to import data directly, as we'll explore a little later in this tutorial.

The process of creating a table in Sequel Pro is similar.

In Sequel Pro, you begin the table creation process by clicking on the small plus sign in the lower left of the main user interface.



Here, you can give your table a name and choose the character encoding and collation for the table. We've left the defaults. The encoding refers to which character set is being used for characters, in this case Latin1, which handles western languages well. The collation refers to the way that characters are compared to one another when comparison operators such as =, > and <> are used. The "ci" collations are case insensitive, which means that a letter z will be considered equal to a letter Z. There are also "bin" collations that are character sensitive and would consider z and Z to be different. Generally speaking for queries by journalists, you'll want the more permissive collation.

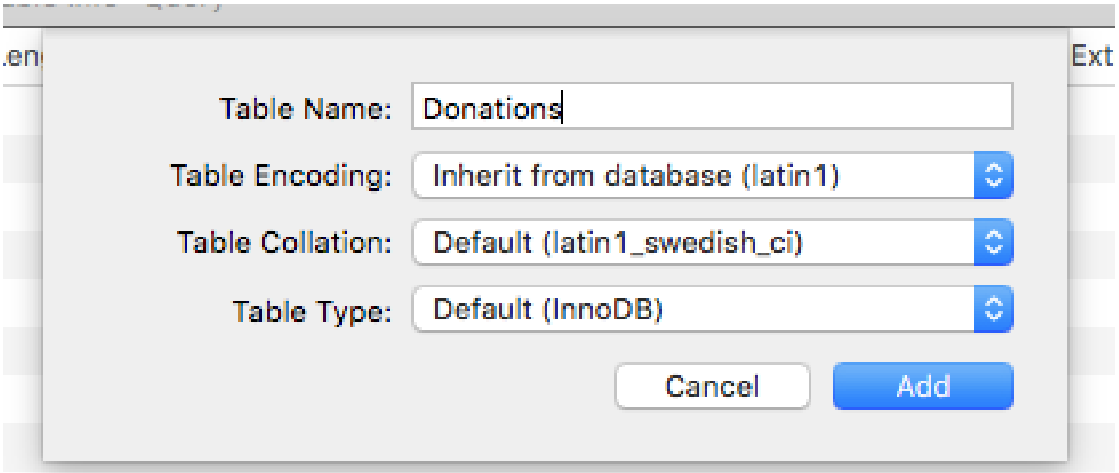


Table Name: Donations

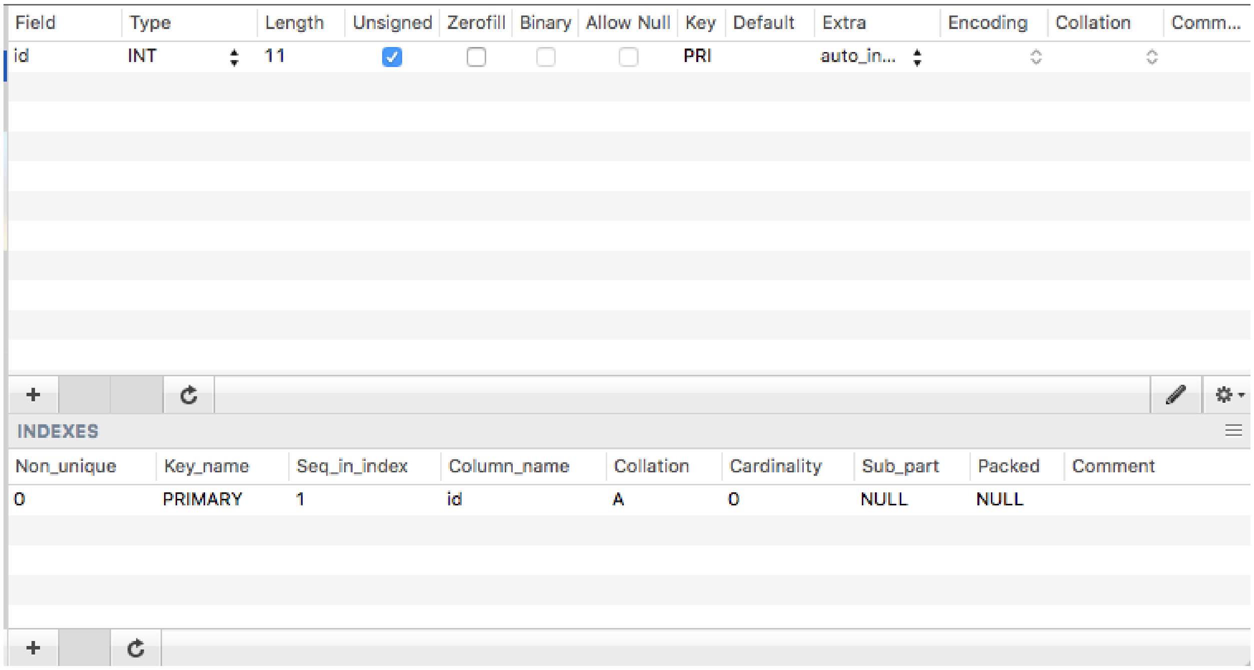
Table Encoding: Inherit from database (latin1)

Table Collation: Default (latin1_swedish_ci)

Table Type: Default (InnoDB)

Table type refers to the database engine, in this case the default MySQL InnoDB. There is usually little reason to change this, but you'll find a complete explanation of the differences between different database engines in the MySQL documentation.

When we click Add, a screen for entering the table structure will appear.



Field	Type	Length	Unsigned	Zerofill	Binary	Allow Null	Key	Default	Extra	Encoding	Collation	Comm...
id	INT	11	☑	☐	☐	☐	PRI		auto_in...			

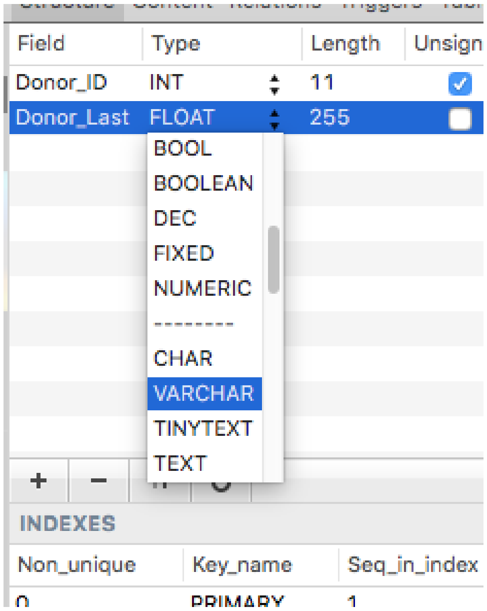
INDEXES

Non_unique	Key_name	Seq_in_index	Column_name	Collation	Cardinality	Sub_part	Packed	Comment
0	PRIMARY	1	id	A	0	NULL	NULL	

Notice that Sequel Pro automatically adds a primary key to the table, with the name id. It is an auto-incrementing, unsigned (no negative numbers) integer. We can modify this later to take the name used in our data, Donor_ID.

To add a new field, click in the + sign in the lower left of the table design area. In the image above, that is the higher of the two + signs. The one at the very bottom is for adding indexes.

Clicking the + sign adds a new line where we can add another field.



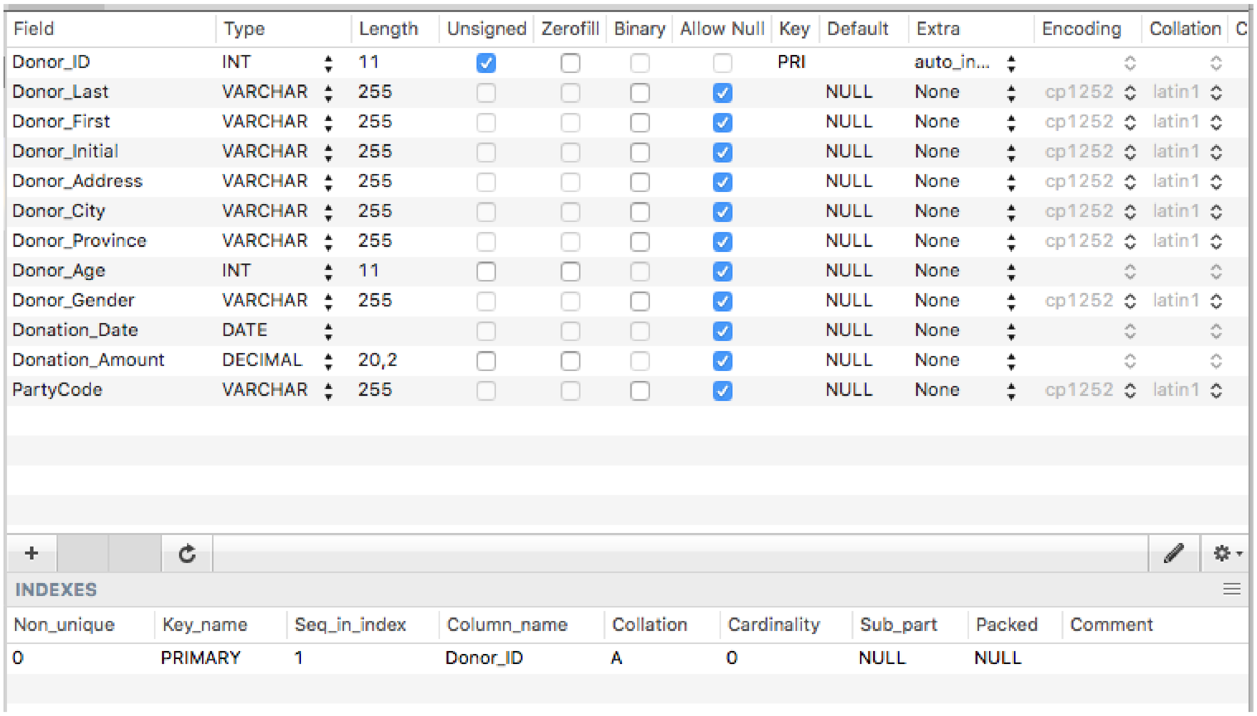
Field	Type	Length	Unsign
Donor_ID	INT	11	☑
Donor_Last	FLOAT	255	☐

INDEXES

Non_unique	Key_name	Seq_in_index
0	PRIMARY	1

Clicking on the small up and down arrow icon exposes a drop-down list of data types. We'll choose Varchar, our standard type for character fields of up to 255 characters. We'll choose a length of 255, remembering that MySQL will only store the actual number of characters used in each row, by that field.

Here is the table, filled out with the remaining fields and data types.



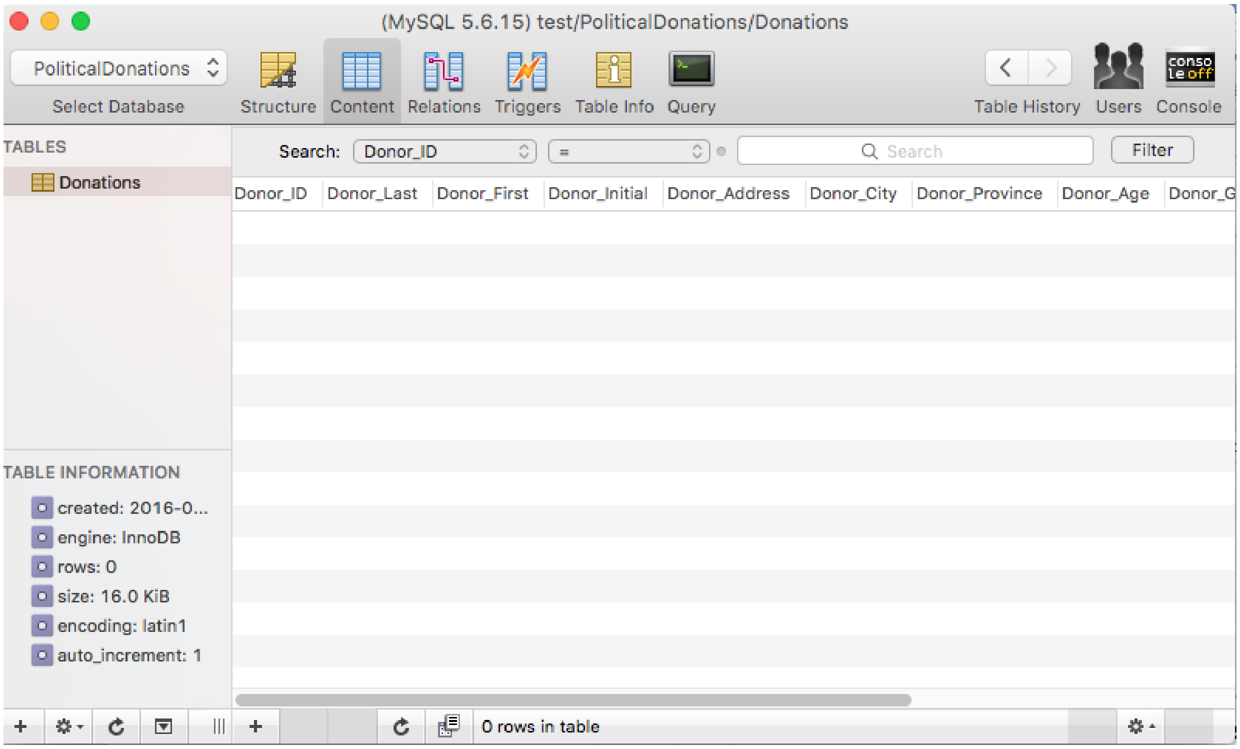
Field	Type	Length	Unsigned	Zerofill	Binary	Allow Null	Key	Default	Extra	Encoding	Collation	C
Donor_ID	INT	11	☑	☐	☐	☐	PRI		auto_in...			
Donor_Last	VARCHAR	255	☐	☐	☐	☑		NULL	None	cp1252	latin1	
Donor_First	VARCHAR	255	☐	☐	☐	☑		NULL	None	cp1252	latin1	
Donor_Initial	VARCHAR	255	☐	☐	☐	☑		NULL	None	cp1252	latin1	
Donor_Address	VARCHAR	255	☐	☐	☐	☑		NULL	None	cp1252	latin1	
Donor_City	VARCHAR	255	☐	☐	☐	☑		NULL	None	cp1252	latin1	
Donor_Province	VARCHAR	255	☐	☐	☐	☑		NULL	None	cp1252	latin1	
Donor_Age	INT	11	☐	☐	☐	☑		NULL	None			
Donor_Gender	VARCHAR	255	☐	☐	☐	☑		NULL	None	cp1252	latin1	
Donation_Date	DATE		☐	☐	☐	☑		NULL	None			
Donation_Amount	DECIMAL	20,2	☐	☐	☐	☑		NULL	None			
PartyCode	VARCHAR	255	☐	☐	☐	☑		NULL	None	cp1252	latin1	

INDEXES

Non_unique	Key_name	Seq_in_index	Column_name	Collation	Cardinality	Sub_part	Packed	Comment
0	PRIMARY	1	Donor_ID	A	0	NULL	NULL	

An argument could be made for making the Donor_Gender field a CHAR field with a length of 1. This would prevent the entry of anything more than one letter, which could help prevent mistakes.

When you are satisfied with the table structure, click on the small, curved arrow. This will run the table creation query that Sequel Pro has quietly been writing in the background. You can now switch to the Content tab, to see the new, empty table.


(MySQL 5.6.15) test/PoliticalDonations/Donations

Donor_ID	Donor_Last	Donor_First	Donor_Initial	Donor_Address	Donor_City	Donor_Province	Donor_Age	Donor_G

TABLE INFORMATION

- created: 2016-0...
- engine: InnoDB
- rows: 0
- size: 16.0 KiB
- encoding: latin1
- auto_increment: 1

0 rows in table

To begin adding data, click on the small plus sign at the bottom of the table area (the one further to the right). You can now type in your rows of data. Or, if you prefer, you can use the import wizard or LOAD DATA INFILE to add external data.

Using LOAD DATA INFILE

LOAD DATA INFILE is a fast and powerful way to get delimited text data into a MySQL table. Its advantages are speed and, on the surface at least, simplicity.

The file we are going to import is called Donations.csv.

As with any such import, you should look at the text file in an ordinary text editor such as Notepad or TextEdit, so you can become familiar with the data structure. It is essential that the data structure of the imported data and the data structure of the table into which it are imported are identical. The number of fields must be the same, and the type of data to be imported into each field in the database table must be compatible with the data type of that field. If, as an example, a column in the CSV has alphabetical characters in it and the field is an integer field, the data either won't be imported, or the entire import will fail, depending on the SQL mode set for the database.

If you would prefer to be given warnings rather than have the import fail, you should set the SQL mode to ANSI. In a front end, or at the SQL prompt when using the SQL command line program, enter the following:

```
SET sql_mode = ‘ANSI’
```

The complete syntax for the LOAD DATA INFILE command is available in the MySQL documentation, but a typical statement will have these clauses:

```
        LOAD DATA INFILE ‘complete path to text file’
        INTO TABLE TableName
        FIELDS TERMINATED BY ‘,’
        OPTIONALLY ENCLOSED BY ‘”’
        LINES TERMINATED BY ‘\n’
        IGNORE n LINES
```

The character you enter in the FIELDS TERMINATED BY line is the delimiter (in the above column a comma) and the character(s) entered in the LINES TERMINATED BY line will depend on the operating system that created the delimited text file. Files created on Macs may have only the ‘\n’ or ‘\r\ characters, while those created by Windows machines may end with the combination ‘\r\n’ Looking at the text file in an editor that shows hidden characters can help you determine in the line-ending character. If you use the wrong one, the worst that will happen is that MySQL will try to load the entire file into one row. Try one of the other combinations if that happens.

Here is what the command would look like for the Donations data.

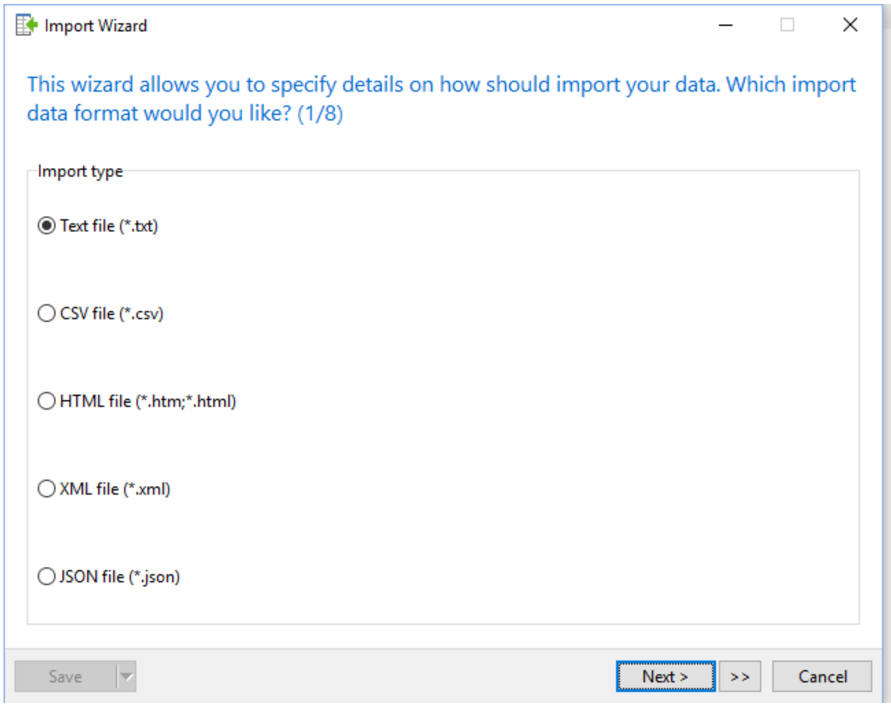
```
        LOAD DATA INFILE ‘c:/mydata/Donations.csv’
        INTO TABLE Donations
        FIELDS TERMINATED BY ‘,’
        OPTIONALLY ENCLOSED BY ‘”’
        LINES TERMINATED BY ‘\r\n’
        IGNORE 1 LINES
```

If there are problems with the import, these will be reported as warnings, if you turned off strict mode, or the first error will be reported as an error, if strict mode is on (and the import will stop). Troubleshooting issues with LOAD DATA INFILE is one of the reasons why many users prefer to use the import utility available in front ends such as Navicat or Sequel Pro.

These utilities behave in much the same way as the text import wizard in Microsoft Access. They automatically examine the structure of the text file to be imported, then create the table and import the data in one operation. If the table has already been created, they can import directly it.

For this exercise, we will presume a new table needs to be created. We'll call it Donations2 to differentiate it from the table we created by hand, but if you are using the wizard interface to create table for the first time, feel free to just call it Donations.

In Navicat, click on the large Table icon at the top of the screen, then choose Import Wizard (or right click on Tables in the table of contents, and choose Import Wizard), to bring up the first step of the wizard.



Import Wizard

This wizard allows you to specify details on how should import your data. Which import data format would you like? (1/8)

Import type

● Text file (*.txt)

○ CSV file (*.csv)

○ HTML file (*.htm;*.html)

○ XML file (*.xml)

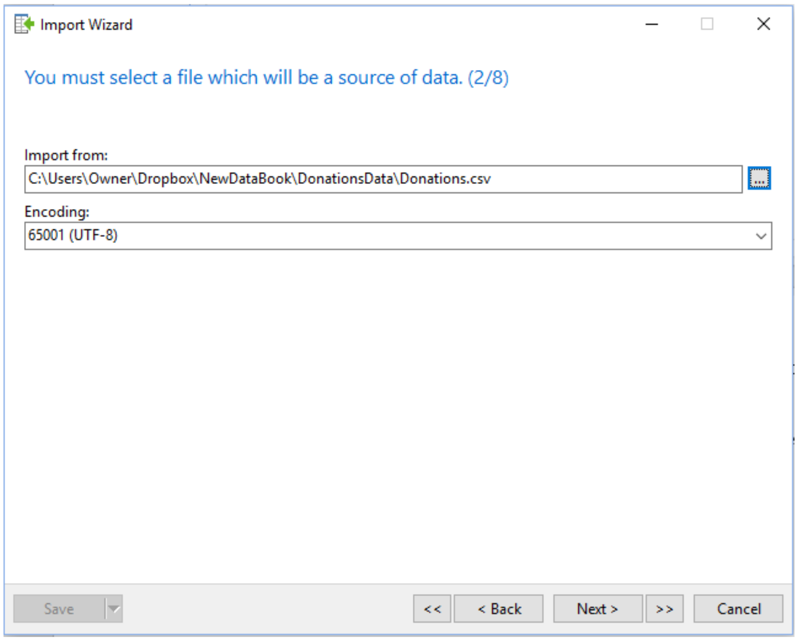
○ JSON file (*.json)

You will see the file formats that can be imported by Navicat Essentials.

For delimited text, you have the choice of Text file, which allows for importing any delimited text file, or CSV, which assumes a comma-delimited text file.

We will choose Text file, but the choices are the same for the CSV file, except that the file must be a csv.

Click Next to bring up step 2.

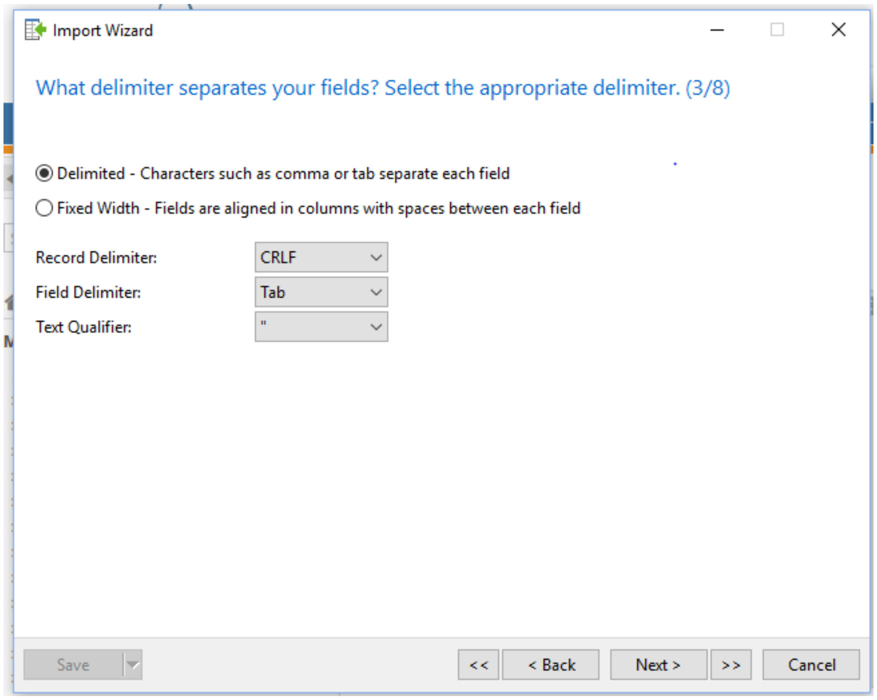


You must select a file which will be a source of data. (2/8)

Import from:
C:\Users\Owner\Dropbox\NewDataBook\DonationsData\Donations.csv

Encoding:
65001 (UTF-8)

Enter the name of the file you wish to enter, and click Next.



What delimiter separates your fields? Select the appropriate delimiter. (3/8)

● Delimited - Characters such as comma or tab separate each field

○ Fixed Width - Fields are aligned in columns with spaces between each field

Record Delimiter: CRLF

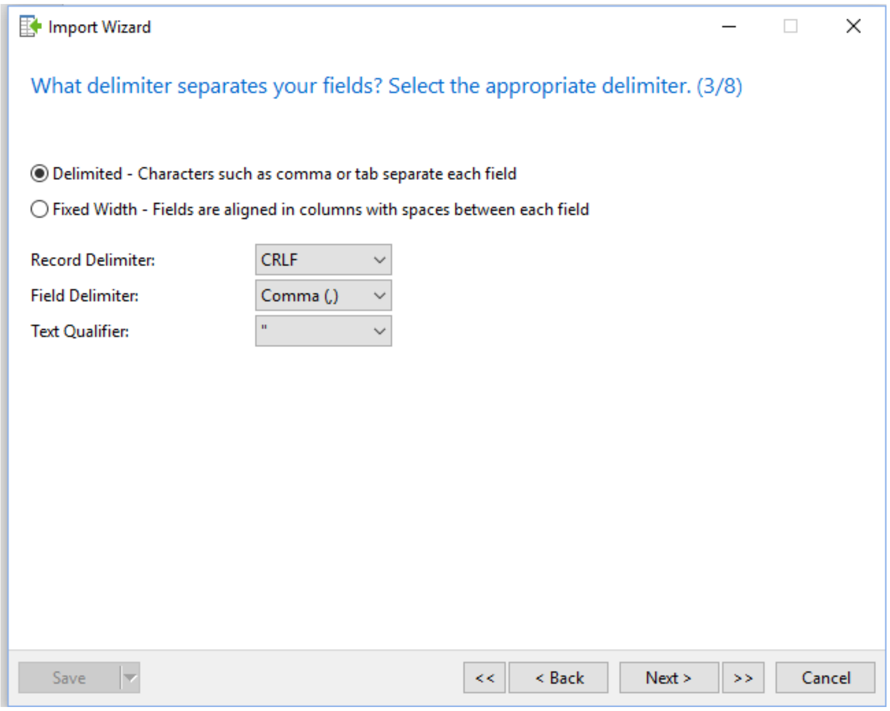
Field Delimiter: Tab

Text Qualifier: "

The record delimiter is the character that ends each record, and this will either be the carriage return/new line combination common on files created in Windows, a new line character, or a carriage return character.

The field delimiter is the character that delimits each field, in our case a comma, but also possibly a tab character, space or other symbol. The text qualifier is a character, usually a single or double quotation mark, that may appear around complete field entries. This ensures that any delimiter characters that are contained within the field itself won't be treated as field separators. The use of a text qualifier is common in csv files because it is common to have commas contained within text or numbers.

These are the completed options for the donations data.



Import Wizard

What delimiter separates your fields? Select the appropriate delimiter. (3/8)

● Delimited - Characters such as comma or tab separate each field

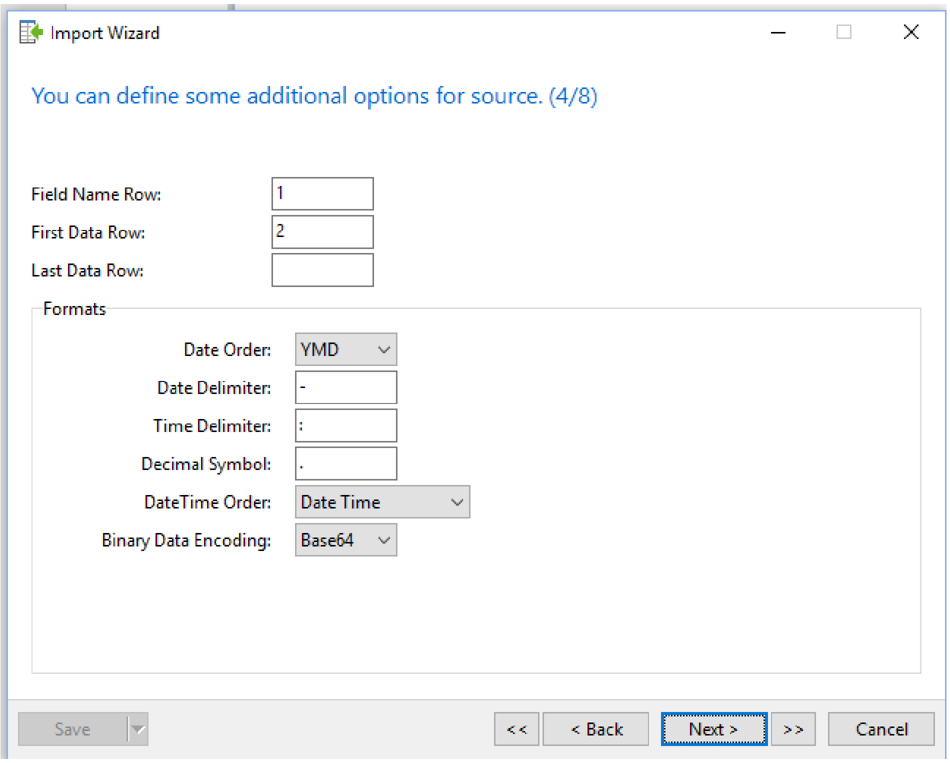
○ Fixed Width - Fields are aligned in columns with spaces between each field

Record Delimiter: CRLF

Field Delimiter: Comma (,)

Text Qualifier: "

Click Next for more options.



You can define some additional options for source. (4/8)

Field Name Row: 1
First Data Row: 2
Last Data Row:

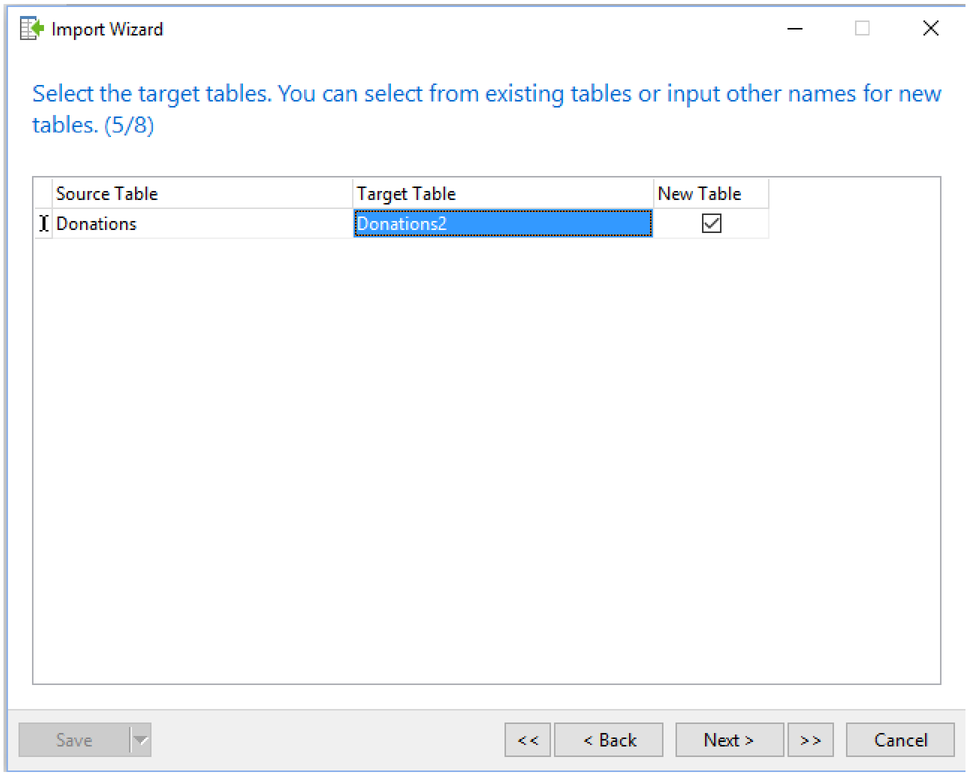
Formats

Date Order: YMD
Date Delimiter: -
Time Delimiter: :
Decimal Symbol: .
DateTime Order: Date Time
Binary Data Encoding: Base64

If your examination of the text file showed that it had field names in the first two, leave the top value as is. Otherwise, change to the appropriate row, or leave it blank if there are no field names present. The second box allows you to indicate where the first row of actual data appears.

The remaining options allow you to specify how data is arranged in the text file, such as the order of elements in dates, the symbol used between date elements, and the symbol used for decimals.

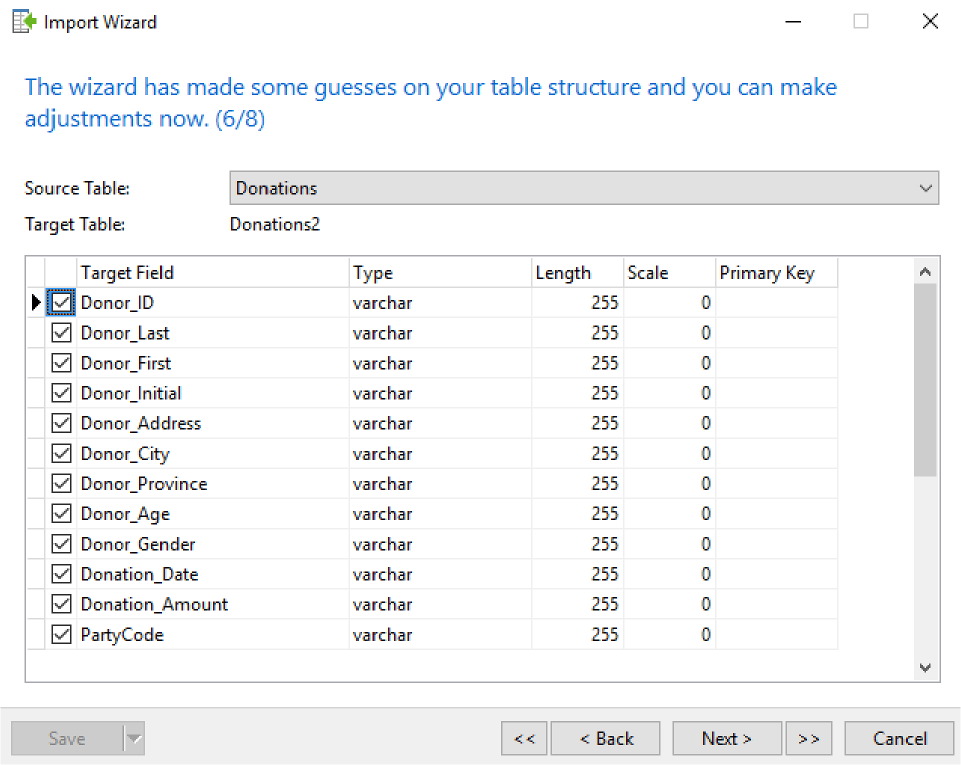
When you are finished making your choices, click Next.



Select the target tables. You can select from existing tables or input other names for new tables. (5/8)

Source Table	Target Table	New Table
Donations	Donations2	☑

Here, we choose whether to import the data into an existing table, or a new one. We've renamed the Target Table as Donations2



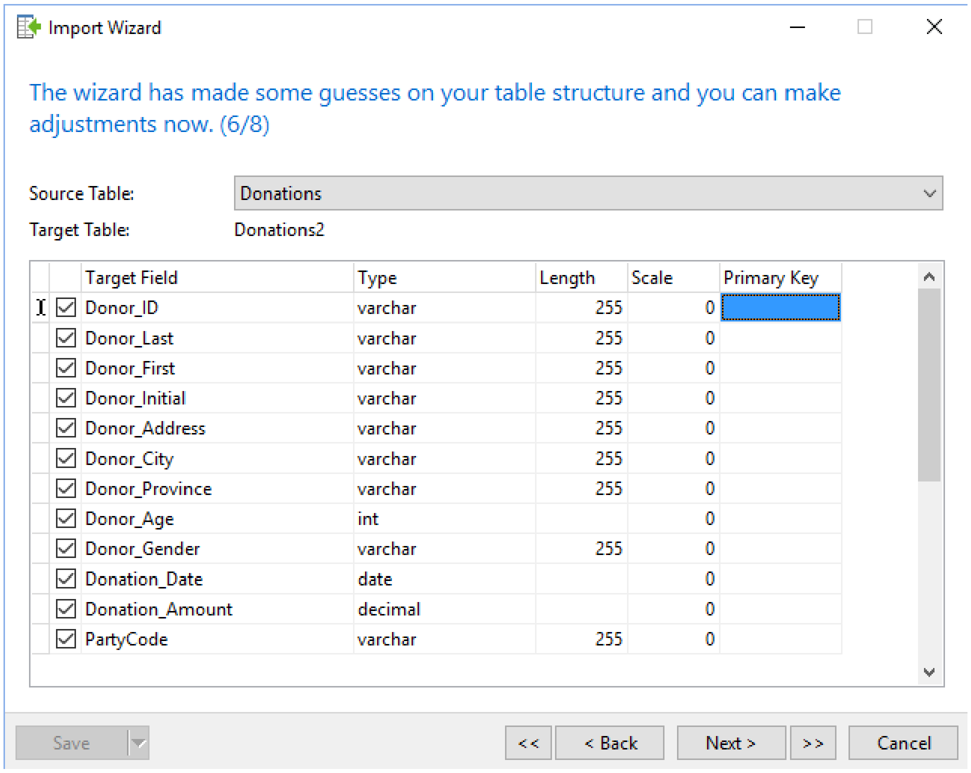
The wizard has made some guesses on your table structure and you can make adjustments now. (6/8)

Source Table: Donations

Target Table: Donations2

	Target Field	Type	Length	Scale	Primary Key
☑	Donor_ID	varchar	255	0	
☑	Donor_Last	varchar	255	0	
☑	Donor_First	varchar	255	0	
☑	Donor_Initial	varchar	255	0	
☑	Donor_Address	varchar	255	0	
☑	Donor_City	varchar	255	0	
☑	Donor_Province	varchar	255	0	
☑	Donor_Age	varchar	255	0	
☑	Donor_Gender	varchar	255	0	
☑	Donation_Date	varchar	255	0	
☑	Donation_Amount	varchar	255	0	
☑	PartyCode	varchar	255	0	

In our case, we can see that Navicat has made every field a character field. We can now change the fields that should be of other data types.



Import Wizard

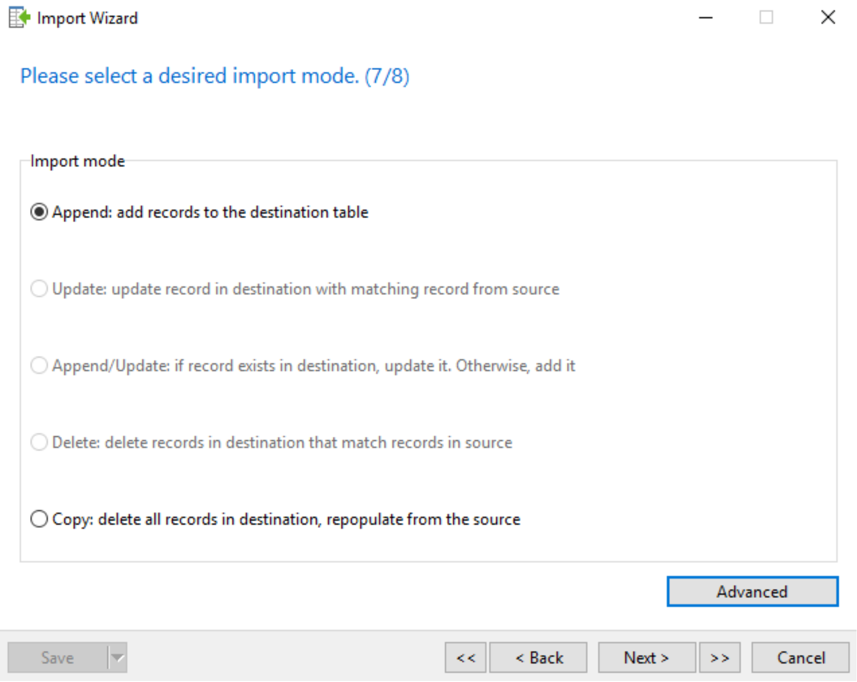
The wizard has made some guesses on your table structure and you can make adjustments now. (6/8)

Source Table: Donations

Target Table: Donations2

	Target Field	Type	Length	Scale	Primary Key
☑	Donor_ID	varchar	255	0	
☑	Donor_Last	varchar	255	0	
☑	Donor_First	varchar	255	0	
☑	Donor_Initial	varchar	255	0	
☑	Donor_Address	varchar	255	0	
☑	Donor_City	varchar	255	0	
☑	Donor_Province	varchar	255	0	
☑	Donor_Age	int		0	
☑	Donor_Gender	varchar	255	0	
☑	Donation_Date	date		0	
☑	Donation_Amount	decimal		0	
☑	PartyCode	varchar	255	0	

Click Next.



Import Wizard

Please select a desired import mode. (7/8)

Import mode

● Append: add records to the destination table

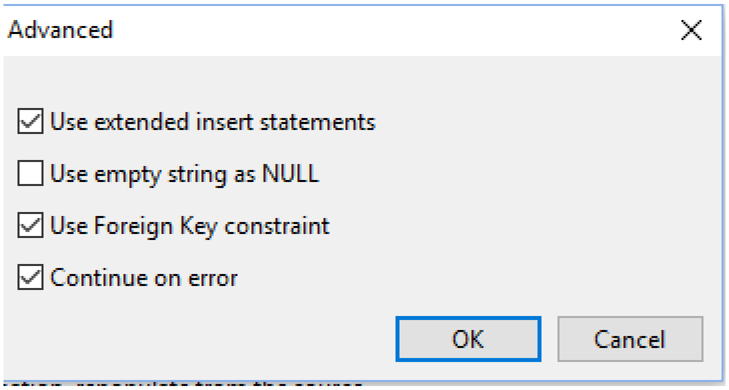
○ Update: update record in destination with matching record from source

○ Append/Update: if record exists in destination, update it. Otherwise, add it

○ Delete: delete records in destination that match records in source

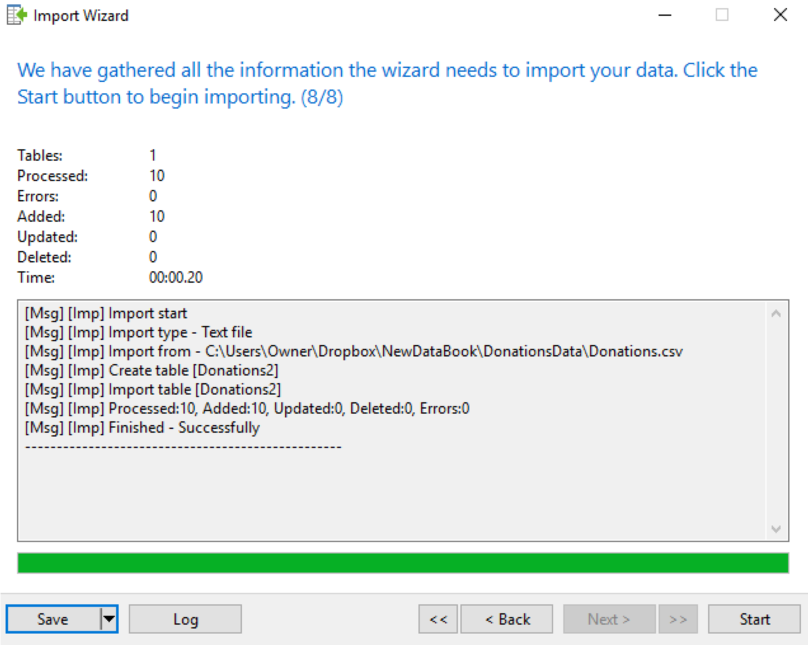
○ Copy: delete all records in destination, repopulate from the source

On the next screen, you will typically leave the options as is, though in the Advanced options you can change what is inserted if the import wizard encounters NULL (empty) values, and whether the import will continue if there is an error. Generally, it is better to continue on errors, then examine any warnings produced to see if critical data is missing.



☑ Use extended insert statements
☐ Use empty string as NULL
☑ Use Foreign Key constraint
☑ Continue on error

Click OK, then Next, and finally Start to begin the import process.



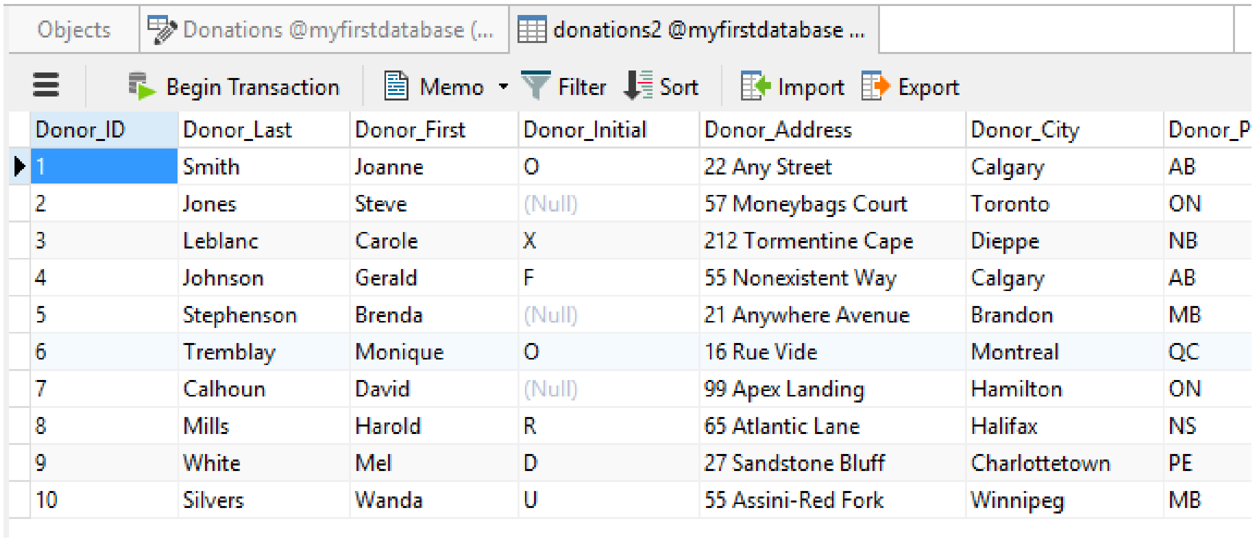
We have gathered all the information the wizard needs to import your data. Click the Start button to begin importing. (8/8)

```
Tables:      1
Processed:   10
Errors:      0
Added:       10
Updated:     0
Deleted:     0
Time:        00:00.20
```

```
[Msg] [Imp] Import start
[Msg] [Imp] Import type - Text file
[Msg] [Imp] Import from - C:\Users\Owner\Dropbox\NewDataBook\DonationsData\Donations.csv
[Msg] [Imp] Create table [Donations2]
[Msg] [Imp] Import table [Donations2]
[Msg] [Imp] Processed:10, Added:10, Updated:0, Deleted:0, Errors:0
[Msg] [Imp] Finished - Successfully
--------------------------------------------------
```

The final screen reports on the import, and whether there were any errors. If there are errors, they are recorded in a log file, so you can see what went wrong and make changes to the import settings or the source text file before trying again.

You’re done. The table is imported into MySQL and you can query it.


Objects
Donations @myfirstdatabase (...
donations2 @myfirstdatabase ...
Begin Transaction
Memo
Filter
Sort
Import
Export

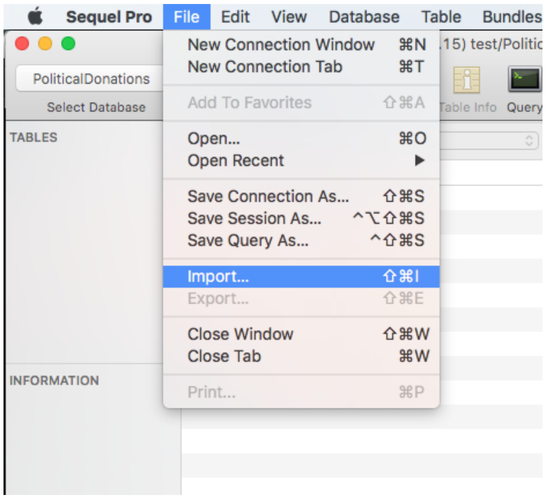
Donor_ID	Donor_Last	Donor_First	Donor_Initial	Donor_Address	Donor_City	Donor_P
1	Smith	Joanne	O	22 Any Street	Calgary	AB
2	Jones	Steve	(Null)	57 Moneybags Court	Toronto	ON
3	Leblanc	Carole	X	212 Tormentine Cape	Dieppe	NB
4	Johnson	Gerald	F	55 Nonexistent Way	Calgary	AB
5	Stephenson	Brenda	(Null)	21 Anywhere Avenue	Brandon	MB
6	Tremblay	Monique	O	16 Rue Vide	Montreal	QC
7	Calhoun	David	(Null)	99 Apex Landing	Hamilton	ON
8	Mills	Harold	R	65 Atlantic Lane	Halifax	NS
9	White	Mel	D	27 Sandstone Bluff	Charlottetown	PE
10	Silvers	Wanda	U	55 Assini-Red Fork	Winnipeg	MB

Using the Sequel Pro import wizard

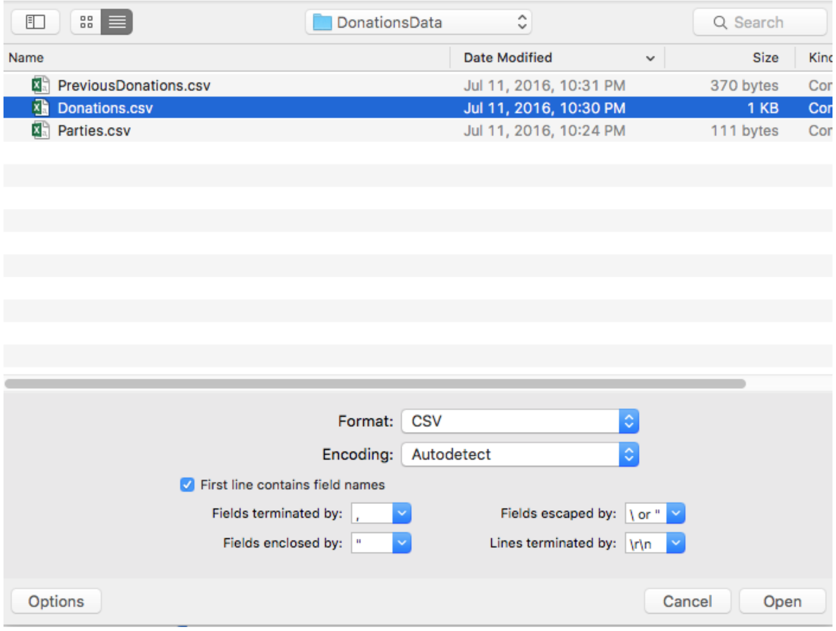
The import process in Sequel Pro is similar to that in Navicat.

Begin by connecting to the MySQL Server and selecting a database. If you are unsure how, see the tutorial **Exploring Navicat and Sequel Pro**

To begin your file import, Choose Import from the File menu.



On the screen that opens, navigate to the folder containing the file you wish to import, in this case Donations.csv On the same page, you set the various attributes of the file you wish to import, including whether the fieldnames are in the first line, the field and line terminators, whether there is a text qualifier (Fields enclosed by), and how any quotation marks within fields are escaped.



Name	Date Modified	Size	Kinc
PreviousDonations.csv	Jul 11, 2016, 10:31 PM	370 bytes	Cor
Donations.csv	Jul 11, 2016, 10:30 PM	1 KB	Cor
Parties.csv	Jul 11, 2016, 10:24 PM	111 bytes	Cor

Format: CSV

Encoding: Autodetect

☑ First line contains field names

Fields terminated by: ,

Fields escaped by: \ or "

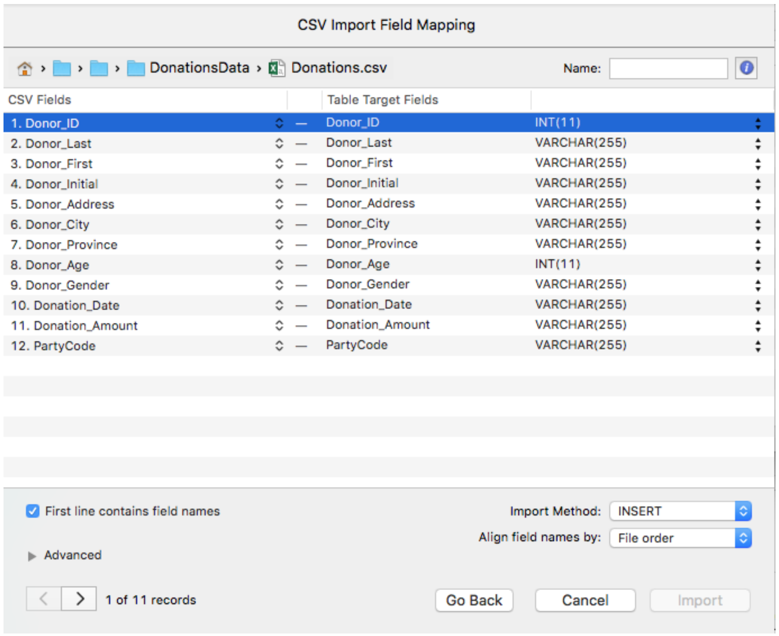
Fields enclosed by: "

Lines terminated by: \r\n

In our case, the file is a CSV file, which means fields are terminated by commas. The file was originally created on a Windows machine, so the file uses the windows \r\n terminator for lines, the field qualifier is a double quotation mark, and quotation marks can be escaped either by quotation marks or the \ slash character.

Click on Options to toggle these settings from visible to hidden and vice versa.

When you are done, click Open. Sequel Pro will provide its best guess at the data type for each field.



CSV Fields		Table Target Fields	
1. Donor_ID	—	Donor_ID	INT(11)
2. Donor_Last	—	Donor_Last	VARCHAR(255)
3. Donor_First	—	Donor_First	VARCHAR(255)
4. Donor_Initial	—	Donor_Initial	VARCHAR(255)
5. Donor_Address	—	Donor_Address	VARCHAR(255)
6. Donor_City	—	Donor_City	VARCHAR(255)
7. Donor_Province	—	Donor_Province	VARCHAR(255)
8. Donor_Age	—	Donor_Age	INT(11)
9. Donor_Gender	—	Donor_Gender	VARCHAR(255)
10. Donation_Date	—	Donation_Date	VARCHAR(255)
11. Donation_Amount	—	Donation_Amount	VARCHAR(255)
12. PartyCode	—	PartyCode	VARCHAR(255)

☑ First line contains field names

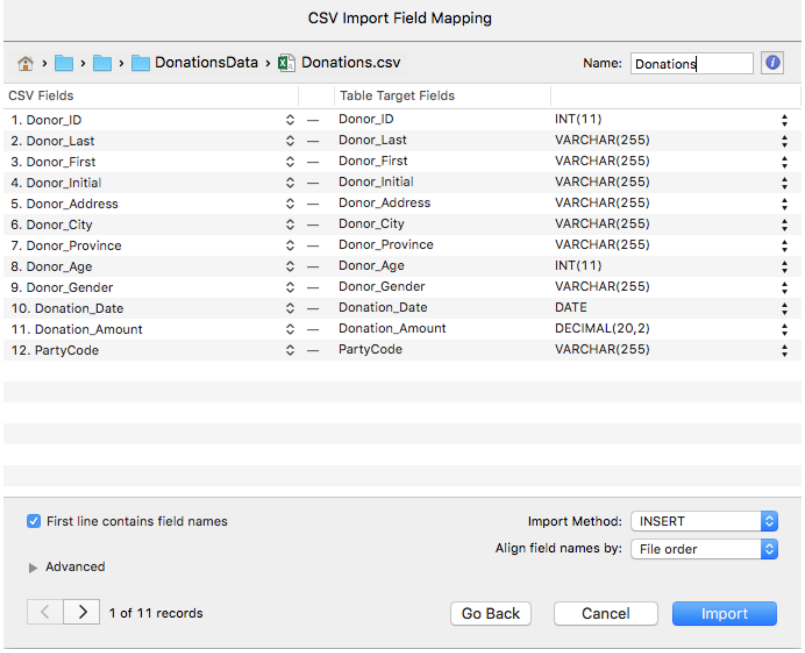
Import Method: INSERT

Align field names by: File order

As you can see, it has interpreted the Donor_ID as an integer, as well as the donor age, but is suggesting that the Donation_Date and the Donation_Amount both be varchar text fields when they ought to be date and decimal fields respectively. The remainder of the fields have been correctly assessed as text fields and the Varchar data type suggested. Again, this is the most flexible data type for fields containing up to 255 characters of text data.

We'll change the Donation_Date field to a date field, from the dropdown list, and because Sequel Pro doesn't offer decimal as a data type in the drop down, we'll double click on the data type beside the Donation_Amount field and replace VARCHAR(255) with DECIMAL(20,2), or a decimal field with a length of 20 and two decimal places. You can manually change any data type this way.

Be sure also to give the table a name as we have here.



CSV Import Field Mapping

DonationsData › Donations.csv

Name: Donations

CSV Fields		Table Target Fields	
1. Donor_ID	—	Donor_ID	INT(11)
2. Donor_Last	—	Donor_Last	VARCHAR(255)
3. Donor_First	—	Donor_First	VARCHAR(255)
4. Donor_Initial	—	Donor_Initial	VARCHAR(255)
5. Donor_Address	—	Donor_Address	VARCHAR(255)
6. Donor_City	—	Donor_City	VARCHAR(255)
7. Donor_Province	—	Donor_Province	VARCHAR(255)
8. Donor_Age	—	Donor_Age	INT(11)
9. Donor_Gender	—	Donor_Gender	VARCHAR(255)
10. Donation_Date	—	Donation_Date	DATE
11. Donation_Amount	—	Donation_Amount	DECIMAL(20,2)
12. PartyCode	—	PartyCode	VARCHAR(255)

☑ First line contains field names

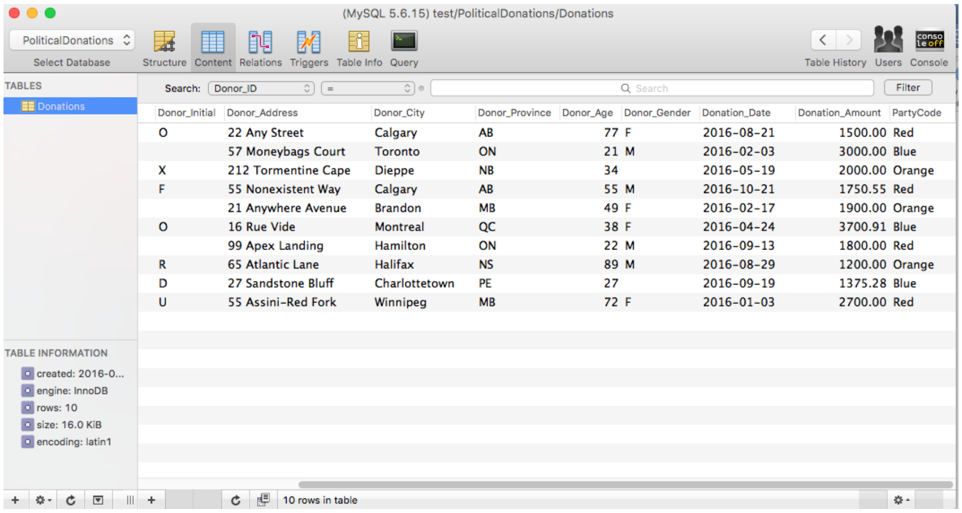
Import Method: INSERT

Align field names by: File order

1 of 11 records

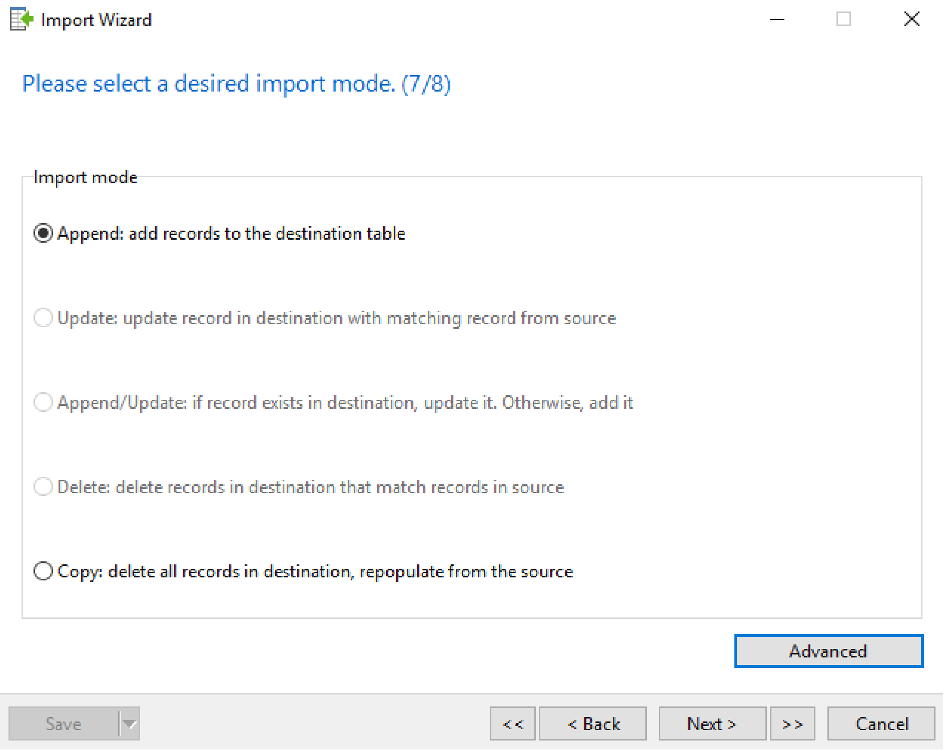
Click Import when you are done setting the data types for all of the fields and naming the table.

Barring fatal errors, your data will import as seen below.



Donor_Initial	Donor_Address	Donor_City	Donor_Province	Donor_Age	Donor_Gender	Donation_Date	Donation_Amount	PartyCode
O	22 Any Street	Calgary	AB	77	F	2016-08-21	1500.00	Red
	57 Moneybags Court	Toronto	ON	21	M	2016-02-03	3000.00	Blue
X	212 Tormentine Cape	Dieppe	NB	34		2016-05-19	2000.00	Orange
F	55 Nonexistent Way	Calgary	AB	55	M	2016-10-21	1750.55	Red
	21 Anywhere Avenue	Brandon	MB	49	F	2016-02-17	1900.00	Orange
O	16 Rue Vide	Montreal	QC	38	F	2016-04-24	3700.91	Blue
	99 Apex Landing	Hamilton	ON	22	M	2016-09-13	1800.00	Red
R	65 Atlantic Lane	Halifax	NS	89	M	2016-08-29	1200.00	Orange
D	27 Sandstone Bluff	Charlottetown	PE	27		2016-09-19	1375.28	Blue
U	55 Assini-Red Fork	Winnipeg	MB	72	F	2016-01-03	2700.00	Red

You are ready to start querying the data, or to import additional tables. If there are already tables in the database, Sequel Pro will give you the option of adding the new data to an existing table, or creating a new one with a new name.



Please select a desired import mode. (7/8)

Import mode

● Append: add records to the destination table

○ Update: update record in destination with matching record from source

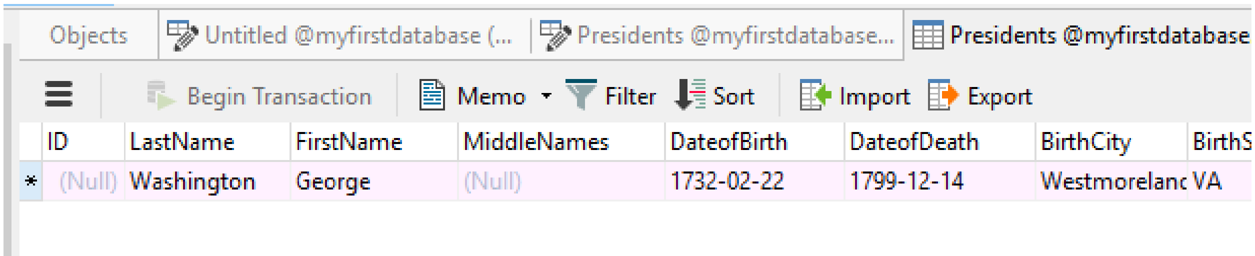
○ Append/Update: if record exists in destination, update it. Otherwise, add it

○ Delete: delete records in destination that match records in source

○ Copy: delete all records in destination, repopulate from the source

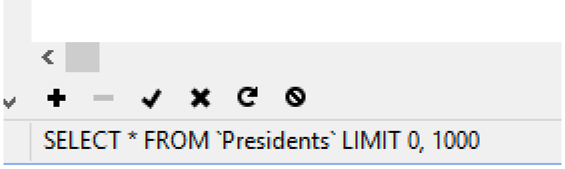
It will be imported into a new table. Click Next to see how Navicat has interpreted the data and the choices it is suggesting for the table structure.

As you enter each value, hit the TAB key to move right to the next field. Those fields that were set as not allowing NULL values will require a value to be entered.


Objects
Untitled @myfirstdatabase (...
Presidents @myfirstdatabase...
Presidents @myfirstdatabase
Begin Transaction
Memo
Filter
Sort
Import
Export

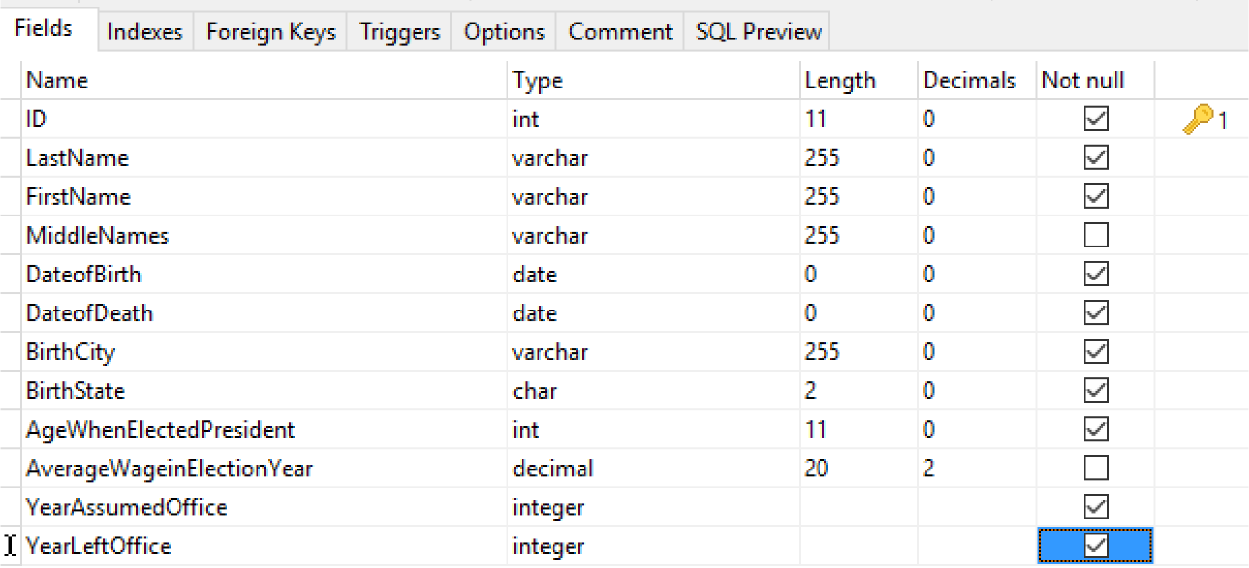
ID	LastName	FirstName	MiddleNames	DateofBirth	DateofDeath	BirthCity	BirthS
(Null)	Washington	George	(Null)	1732-02-22	1799-12-14	Westmorelanc	VA

When you have finished adding the first record, you can click on the checkmark at the bottom of the screen to commit the record to the table. Clicking on the + sign allows you to add a new record. Continue until you have finished adding all of the records.



```
SELECT * FROM `Presidents` LIMIT 0, 1000
```

If, at any point, you realize that you need to add additional fields, right click on the table name and choose Design Table. Let's say we wanted to add fields for the years of the start and end of each president's term(s) in office. Click on Add Field and you can add a new field as you did when you first created the table.



Name	Type	Length	Decimals	Not null	
ID	int	11	0	☑	🔑1
LastName	varchar	255	0	☑	
FirstName	varchar	255	0	☑	
MiddleNames	varchar	255	0	□	
DateofBirth	date	0	0	☑	
DateofDeath	date	0	0	☑	
BirthCity	varchar	255	0	☑	
BirthState	char	2	0	☑	
AgeWhenElectedPresident	int	11	0	☑	
AverageWageinElectionYear	decimal	20	2	□	
YearAssumedOffice	integer			☑	
YearLeftOffice	integer			☑	

Save the table, and you can add the additional information.
Continue until you have completed adding all of your data.

Creating tables and adding data with the import wizard

If you are importing external data from a CSV file