Tabulate Documentation

Sam Wilson

Jul 10, 2018

Contents

Tabulate is a user-friendly plugin for WordPress designed to make it easy to work with relational tabular data. For more information, please read the *Introduction*.

- WordPress plugin page: https://wordpress.org/plugins/tabulate

- Support forum: https://wordpress.org/support/plugin/tabulate

- Documentation: https://tabulate.readthedocs.org

- Source code: https://github.com/tabulate/tabulate

- Issue tracker: https://github.com/tabulate/tabulate/issues

Quick-start Looking for the TL;DR? Here it is!

1. Install the plugin

2. Browse to the Tabulate screen in the Admin Area

3. Create some tables

4. Start creating or importing data

Learning more For more information, you can work through this documentation. It is written to be read in order but should be reasonably easy to dip into wherever you want.

If anything doesn't make sense, or the operation of the software doesn't match what's here, we'd be very grateful if you could raise an issue on GitHub to let us know.

We're trying to write Tabulate in a *test-first* manner, in that all new features and bug fixes are preceeded by test code that ensures their correct operation. We're also trying to *document-first*: no code, not even test cases, is written until the documentation has been created or updated to match what we're trying to make the software do. The stable version of the documentation, therefore, is complete and can be read as instructions of how to use Tabulate—the latest version, however, may at times be a bit *avant-garde* and not (yet) make any sense with respect to what you'll see on an installed site.

Contents

1 Introduction

WordPress is primarily a platform for *text*. Posts are the core of a WordPress site, and although they can be given all sorts of metadata and elaboration, there are times when one wants to work with data in a more structured and controlled way. This is where Tabulate comes in. You might think of it as the spreadsheet complement to the word-processor that WordPress is by default.

Tabulate is designed to be an easy-to-use plugin, aimed at anyone comfortable using spreadsheets. It can be used to manage any type of tabular data, and provides the power of the database for managing things like column data types, field cross-references, and searching the data in a number of ways. In addition, it provides access-control and a change-tracking mechanism.

The fundamental components of Tabulate are the *tables*. A table in Tabulate is just a standard database table, with columns of defined types and rows that contain your data. When installed, Tabulate only adds tables required for change-tracking; the first task after installation is to start creating the tables that you need for your own data.

Tabulate can be used to manage any sort of data, from the small and personal to the large and institutional. Got a book collection that you want to catalogue? Or a rain gauge that you measure daily? How about a fleet of vehicles whose service history you need to keep track of? A couple of hundred thousand environmental monitoring points all sending you data? All can be kept centrally and securely in Tabulate, and from there published to the world though your WordPress site.

Basically, Tabulate gives you access to your database, and can be used for whatever you would use a database for. It is a step beyond the usual spreadsheet-based data management scenario that is all too common... it is a step *prior* to a bespoke WordPress plugin, however (although extensive customisation is possible, and Tabulate can be thought of as a prototyping system in some ways).

2 Installing and upgrading

Installation is the same as for most WordPress plugins, but note that you can get extra features by installing other plugins, and upgrading requires some changes whenever the major version number increases (e.g. 2.x.x to 3.x.x).

2.1 Install

1. Install the plugin in the usual way by using your site's plugin manager. For Entity Relationship Diagram support, also install the TFO Graphviz plugin (`tfo-graphviz`).

2. Browse to the Tabulate overview page via the main menu in the WordPress admin area.

3. Create some tables. Alternatively, you can use a tool such as PHPmyAdmin or MySQL Workbench.

2.2 Upgrade

When upgrading, please *deactivate* and then *reactivate* the plugin. This will ensure that all required database updates are carried out (but will avoid the overhead of checking whether these are required on every Tabulate page load).

Earlier versions of Tabulate required the REST API plugin (either version 1, `json-rest-api`; or version 2, `rest-api`) as the API was moved into core WordPress. You can remove these plugins if you have them.

Tabulate can be deactivated and reactivated without losing any data; if uninstalled, it will remove everything that it's added (but you will be warned before this happens, don't worry).

None of your custom database tables are modified during upgrade, activation, deactivation, or uninstallation.

3 Using the shortcode

A Shortcode is a WordPress method of adding dynamic content to posts and pages. Tabulate provides one short code, `[tabulate]`, which can be used to add tables, lists, data-entry forms, and record-counts to your content. Its parameters (which can appear in any order) are as follows:

1. `table` — The name of the table in question. Required. No default.

2. `format` — One of `table`, `list`, `form`, `count`, or `record`. Optional. Defaults to `table`.

3. `ident` — Used for the Record format. Optional. No default.

4. `search` — Whether to display a search form for the Table format. Optional. Defaults to `false`.

Do note that if a table is not accessible to the browsing user then nothing will be displayed. Keep in mind that you can grant access to non-logged-in users to view tables if you wish (via the Grants page in the Admin Area).

When pasting example shortcode syntax from this page into the visual editor in WordPress, be sure to use the

'paste as text' toolbar button: (otherwise, you will get an error such as "The 'table' attribute must be set" even though it looks like you've set that attribute). If you are using the text editor, this problem shouldn't arise.

3.1 Table format

Example: `[tabulate table=widgets search=yes]`

The table format displays an HTML-table displaying all records from the Tabulate-table specified.

The displayed data will be paginated if there are more than a certain number of records, and the user will be able to page through the data.

If the additional parameter `search` is provided (and given any value at all; `yes` is just a convention) then a record-filtering form will be displayed.

3.2 List format

Example: `[tabulate format=list table=widgets]`

The list format displays a comma-separated list of all of the titles of the records from the table specified.

3.3 Form format

Example: `[tabulate format=form table=widgets]`

The form format displays a data-entry form to users who have been granted access to create records in the specified table.

The form operates exactly the same as the form in the Admin Area, except that after submission the user is sent back to a blank form in readiness for the next data-entry (rather than shown their saved data). A message such as "Record saved." is displayed after submission.

3.4 Count format

Example: `There are [tabulate format=count table=widgets] Widgets in our catalogue.`

The count format displays a simple integer count of the records in the given table.

This usage of the shortcode can be used inline within a sentence.

3.5 Record format

Example: `[tabulate format=form table=widgets ident=45]`

The record format displays a single record from a table.

To specify which record to display, either provide the `ident` shortcode parameter, or set a URL parameter equal to the name of the table. For example, `[tabulate table=widgets format=record]` will look for `?widgets=45` and display the record with a primary key value of `45`.

4 Reports

Reports in Tabulate are a way of combining your data with templates to produce outputs in the most flexible way possible. This means that you can display your data in non-tabular forms, such as HTML, LaTeX, or even GPX.

A report comprises basically a title, a set of source SQL statements, and an output template written in the Twig templating language.

By default, all reports are listed in a report named *Reports*, which also serves as an example for how to create other reports.

4.1 Example: Ordered HTML list

For a `widgets` table that has a `name` field, create a Report with the following template:

```
<h2>Widgets</h2>
<ol>
{% for widget in widgets %}
    <li>{{widget.name}}</li>
{% endfor %}
</ul>
```

Attach this SQL as a Report Source:

```
SELECT `name` FROM `widgets` ORDER BY `name`;
```

5 Frequently Asked Questions

5.1 Where is the documentation?

Tabulate documentation is hosted on *Read The Docs* at http://tabulate.readthedocs.org/

5.2 Where should issues be reported?

Please log all bugs, feature requests, and other issues in the GitHub issue tracker at https://github.com/tabulate/tabulate/issues

5.3 What modifications does Tabulate make to the database?

Four database tables are created, and one option, all prefixed with *tabulate_*. When Tabulate is uninstalled, all of these are deleted (but custom tables are not touched).

5.4 Is row-level access control possible?

This should be done by creating a view (of one or more tables) and granting access to that.

5.5 What reasons exist for the 'Unable to create temporary export file' error?

If you are getting an error like "Unable to create temporary export file: `C:\Windows\Temp\tabulate_5593a4c432a67.csv`" or "Unable to create temporary export file: `/tmp/tabulate_5593a4c432a67.csv`" then you should firstly turn on debug mode and re-run the export to see if you get more information in the error message, and then:

1. check that your database user has the FILE privilege.

2. make sure your MySQL server user has write-access to the directory to which the CSV files are written, and your web server user has read access to the same directory.

3. If those don't work, you can change WordPress's temporary-files' directory by creating the `wp-content/tmp/` directory and adding the following to your `wp-config.php`:

   ```
   define( 'WP_TEMP_DIR', ABSPATH . 'wp-content/tmp/' );
   ```

5.5.1 MySQL's `secure_file_priv` variable

Note that the MySQL server may be running with the `secure_file_priv` variable (you can check this with `SHOW VARIABLES LIKE "secure_file_priv"`), and in this case the web server also needs to be able to read files from that same directory. It is probably better to *disable* secure-file-priv (by setting it to an empty value) rather than setting it to a widely-readable directory (such as `/tmp`) because then you can control who can read the directories that *are* used by it.

This is a global variable that needs to be changed in MySQL's server-configuration file; it cannot be set dynamically at runtime.

5.6 Where is the developers' documentation?

For information about the development of Tabulate or integrating other plugins with it please see CONTRIBUTING.md on GitHub.