Hooks and Filters Exercises

See the hooks documentation and filters documentation

Echo out more metadata about the fake tweets (from Models demo)

- the second parameter to `metadata` is the name of one of the properties

Add content to collections pages

- use the public_collections_show hook

Add content to all pages

- use the public_body hook

Add a navigation link to the whatnot/thingy page (from ControllersAndViews demo)

- use the public_navigation_main filter
- compare how SimplePages builds the url in its admin_navigation_main filter and follow that pattern (don't worry about the 'resource' or 'privilege' stuff for now)

Models Exercises

Add 'in_reply_to_user_id' to the Tweet model

- copy and modify the sql that creates Tweets in the database
- add some fake user ids to the fake data created on install

Create a simple twitter user model

- think about how users will be related to their tweets
- create the model class
- copy and modify the sql that creates Tweets in the database
  - the `id` should be exactly the same
  - integers like identifiers will be like `tweet_id`
  - text values will be like `text`
  - modify the uninstall hook to remove your new database table when the plugin is uninstalled

Controllers And Views Exercises

Create the missing views for the Whatnot_Controller

- need views to go along with the showAction and browseAction
- pass in some fake data to display by implementing the actions

Play with different default models

- The example steals data from Items. Try it with Collections and Tags and see what happens.
  - fix things that break
  - look at the Collections and Tags models to see what other data you can display