Invoke scripts for Jupyter notebooks

The hub tool invoke script is located in the tool's middleware/ subdirectory. When you first create a tool, the basic invoke script provided must be edited to work with Jupyter notebook tools.

This writeup shows you how to create Jupyter tools with three different appearances: notebook, App, and Tool mode.

invoke_app and start_jupyter

To deploy a Jupyter notebook as a tool on your hub, you call the invoke_app executable, which in turn calls start_jupyter. Each have their own arguments:

arguments for start_jupyter
- d show debug (verbose) output
- t run as a Tool with no notebook controls
- A run in AppMode.

arguments for invoke_app
- C command to execute
- R Rappture version to use (normally specify none for notebook tools)
- u environment package(s) to use
- t Tool name

invoke_app: starting point

The basic invoke script for Jupyter notebooks looks like this:

```
/usr/bin/invoke_app "$@" -t TOOLNAME
-C "start_jupyter -T @tool APP.ipynb"
-r none
-u anaconda-6
```

Invoking a Jupyter tool this way gives a notebook with all its code cells displayed to the user.

Where:

- TOOLNAME is the short name of the tool
- APP is the name of the main notebook that runs the tool
- anaconda-6 is the current anaconda installation
Notice that the script uses the `-t`, `-C`, `-r`, and `-u` `invoke_app` arguments.

**start_jupyter arguments**

Control the way the notebook appears when run as a tool, using the arguments passed to the `start_jupyter` executable.

You can run a Jupyter tool in three ways:

- notebook mode, in which all code cells are displayed to the user (shown above)
- app mode, in which code cells are initially hidden but can be displayed
- tool mode, in which code cells are hidden and cannot be displayed

**for App Mode**

For a notebook tool that hides its code cells and shows only the UI and markdown elements on initial run, add the `-A` argument in the `start_jupyter` call:

```
/usr/bin/invoke_app "$@" -t TOOLNAME
-C "start_jupyter -A -T @tool APP.ipynb"
-u anaconda-6
-r none
```

The tool user can toggle the tool's "Edit App" button to show the underlying code cells, making this a great teaching/demo option.

**NOTE** that this differs from the `invoke_app -A` argument for `invoke_app`.

**for Tool Mode**

To permanently hide code cells from the user in App Mode, specify the `-A` and `-t` arguments in the `start_jupyter` call:

```
/usr/bin/invoke_app "$@" -t TOOLNAME
-C "start_jupyter -A -t -T @tool APP.ipynb"
-u anaconda-6
-r none
```

The Edit App button will not be displayed to the tool user.

**NOTE** that this differs from the `invoke_app -t` argument for `invoke_app`. 
errors

**specify no rapptune**

```
/usr/bin/invoke_app "$@" -t TOOLNAME
-C "start_jupyter -T @tool APP.ipynb"
-u anaconda-6
-r none
```

**Error:**

Running the tool's invoke script from the deb10 workspace, returns:

"could not find a rapptune installation: RAPPTURE_PATH=,"

**Fix:**

Be sure to supply the "-r none" argument in the invoke_app call, as above. No quotation marks are needed.