# Developing Extensions for JupyterLab

#### Meet the instructors



Piyush Jain

AWS

Piyush is a software engineer working on JupyterLab



IBM
Alex is a software
engineer working on
Elyra and JupyterLab

Alex Bozarth



Martha Cryan
IBM

Martha is a software
engineer working on
Elyra and JupyterLab

# **Exploring Extensions**

- What are extensions?
- Examples
  - DrawIO
  - Latex
  - Git
  - Spellchecker
  - Themes
- Installing prebuilt vs source







#### Anatomy of an Extension

Extensions, plugins and widgets

#### Code Walkthrough...

- We will be using an example created from https://github.com/jupyterlab/extension-cookiecutter-ts
- The example is loosely based on the in-depth example found in the jupyterlab documentation: <a href="https://jupyterlab.readthedocs.io/en/stable/extension/extension/extension">https://jupyterlab.readthedocs.io/en/stable/extension/extension/extension tutorial.html</a>

#### Code Exercise

- 1. Open examples/tutorial extension dir in your IDE
- 2. Try adding a toolbar button that refreshes the image using the following hints:
  - The toolbar can be accessed from MainAreaWidget.toolbar
  - A ToolbarButton class can be found in @jupyterlab/apputils
- 3. An example answer can be found on the next slide if you get stuck

#### Code Exercise Example Answer

```
const button = new ToolbarButton({
    icon: refreshIcon,
    onClick: () => widget.load_image()
});
main.toolbar.addItem('refresh', button);
```

## Debugging JupyterLab Extensions

- When is debugging useful
- Setting up for debugging
- Launching JuyterLab for debugging
- Setting breakpoints
- Other ways to debug

## When is debugging useful

- To find errors in code
- Investigating unexpected results
- Understanding the code path
- Learning internals of other extensions

## How to know something has gone wrong

- UI elements are missing
- Errors appear in the server log
- Errors appear in the browser console

#### Debugging in Visual Studio Code

- Instructions are in <u>DEBUGGING.md</u>
- Install the cookiecutter package pip install cookiecutter
- Use the debug-config-cookiecutter cookiecutter ../debug-config-cookiecutter
- Install the debug dependencies
   jlpm install
- Build the extensions jlpm build

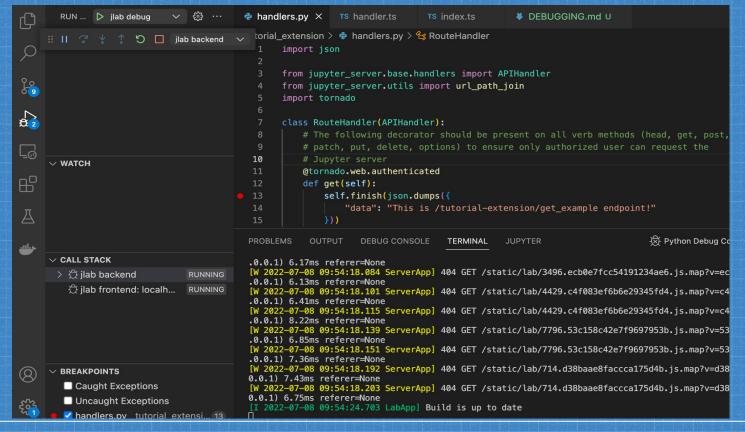
## Setting breakpoints

```
RUN ... | jlab debug
                            v 👸 ...
                                          handlers.py X
TS handler.ts
                                                                            TS index.ts

■ DEBUGGING.md U

     ∨ VARIABLES
                                          tutorial_extension > ♦ handlers.py > ♀ RouteHandler > ♀ get
                                                 import json
Q
                                                 from jupyter_server.base.handlers import APIHandler
from jupyter_server.utils import url_path_join
                                                 import tornado
₽
                                                 class RouteHandler(APIHandler):
                                                     # The following decorator should be present on all verb methods (head, get, post,
                                                     # patch, put, delete, options) to ensure only authorized user can request the
@tornado.web.authenticated
                                                     def get(self):
     ∨ WATCH
                                                         self.finish(json.dumps(
                                                             "data": "This is /tutorial-extension/get example endpoint!"
                                                         }))
                                                 def setup_handlers(web_app):
                                                     host_pattern = ".*$"
                                                     base_url = web_app.settings["base_url"]
     ∨ CALL STACK
                                                     route_pattern = url_path_join(base_url, "tutorial-extension", "get_example")
                                                     handlers = [(route_pattern, RouteHandler)]
                                                     web_app.add_handlers(host_pattern, handlers)
(8)
```

# Launching JupyterLab for debugging



#### Debugging Front End Extension

```
handlers.pv
                                                                                 TS widget.ts
                                                                                                                    (i) README.md
       RUN ... > ilab debug
                                                                TS handler.ts
                                                                                                   TS index.ts X
                                              c > TS index.ts > [∅] plugin > 🕥 activate
                                                             console.log(data);
          > app: W {_started: true, _plugi...
                                                           .catch(reason => {
          > launcher: E {stateChanged: e, ...
console.error(
          > settingRegistry: g {schema: {.....
                                                               `The tutorial extension server extension appears to be missing.\n${reason}
            this: undefined
       Clasura ( Hiblinday is)

✓ WATCH

                                                      papp.commands.oaddCommand('tutorial:open', {
                                           48
// code to run when this command is executed
                                                          execute: () => {
                                                             const widget = new TutorialWidget();
                                                             const main = new MainAreaWidget({ content: widget });
                                                             main.title.label = 'Tutorial Widget';
                                                             main.title.icon = jupyterIcon;

✓ CALL STACK

                                                             main.title.caption = widget.title.label;
               _run_code
                           runpv.pv 87:1
               _run_module_as_main run...
                                                             app.shell.add(main, 'main');
          > ThreadPoolExecutor-... PAUSED

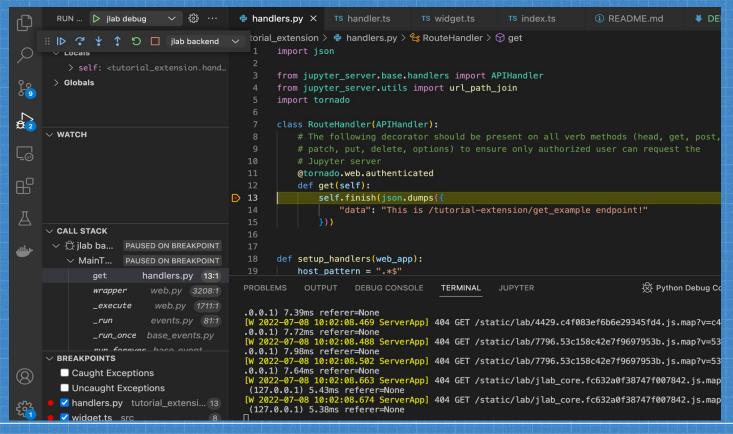
∨ ☆ ilab fro... PAUSED ON BREAKPOINT

                                             PROBLEMS
                                                                                                                              → Pvthon Debug Co
                                                                    DEBUG CONSOLE
                                                                                      TERMINAL
            activate
                        src/index.ts 48:5
                                              .0.0.1) 6.72ms referer=None
            <anonymous> localhost:999...
                                              [W 2022-07-08 10:03:19.862 ServerApp] 404 GET /static/lab/4429.c4f083ef6b6e29345fd4.js.map?v=c4
      > LOADED SCRIPTS
                                              .0.0.1) 5.70ms referer=None
                                              [W 2022-07-08 10:03:19.878 ServerApp] 404 GET /static/lab/7796.53c158c42e7f9697953b.js.map?v=53

∨ BREAKPOINTS

                                              .0.0.1) 7.19ms referer=None
        ■ Caught Exceptions
                                              [W 2022-07-08 10:03:19.893 ServerApp] 404 GET /static/lab/7796.53c158c42e7f9697953b.js.map?v=53
        Uncaught Exceptions
                                              .0.0.1) 6.21ms referer=None
                                              [W 2022-07-08 10:03:20.066 ServerApp] 404 GET /static/lab/jlab_core.fc632a0f38747f007842.js.map
        ✓ handlers.py tutorial extensi... 13
                                              (127.0.0.1) 6.90ms referer=None
                                              [W 2022-07-08 10:03:20.077 ServerApp] 404 GET /static/lab/jlab_core.fc632a0f38747f007842.js.map
         ✓ index.ts src
                               1 × 48
                                               (127.0.0.1) 6.37ms referer=None
        BROWSER BREAKPOINTS
```

#### Debugging Server Extension



#### Other ways to debug

- Front end extension
  - Use the browser directly to debug
- Server extensions
  - Python's command line debugger (pdb)
    import pdb; pdb.set trace()
  - PIPython pdb, a better alternative to
    pdb (pip install ipdb)
    import ipdb; ipdb.set trace()

#### Working on Your Own Extension

- Jupyter Server extension
- Theme extension
- Whatever you wanted to start on or pick from <u>here</u>