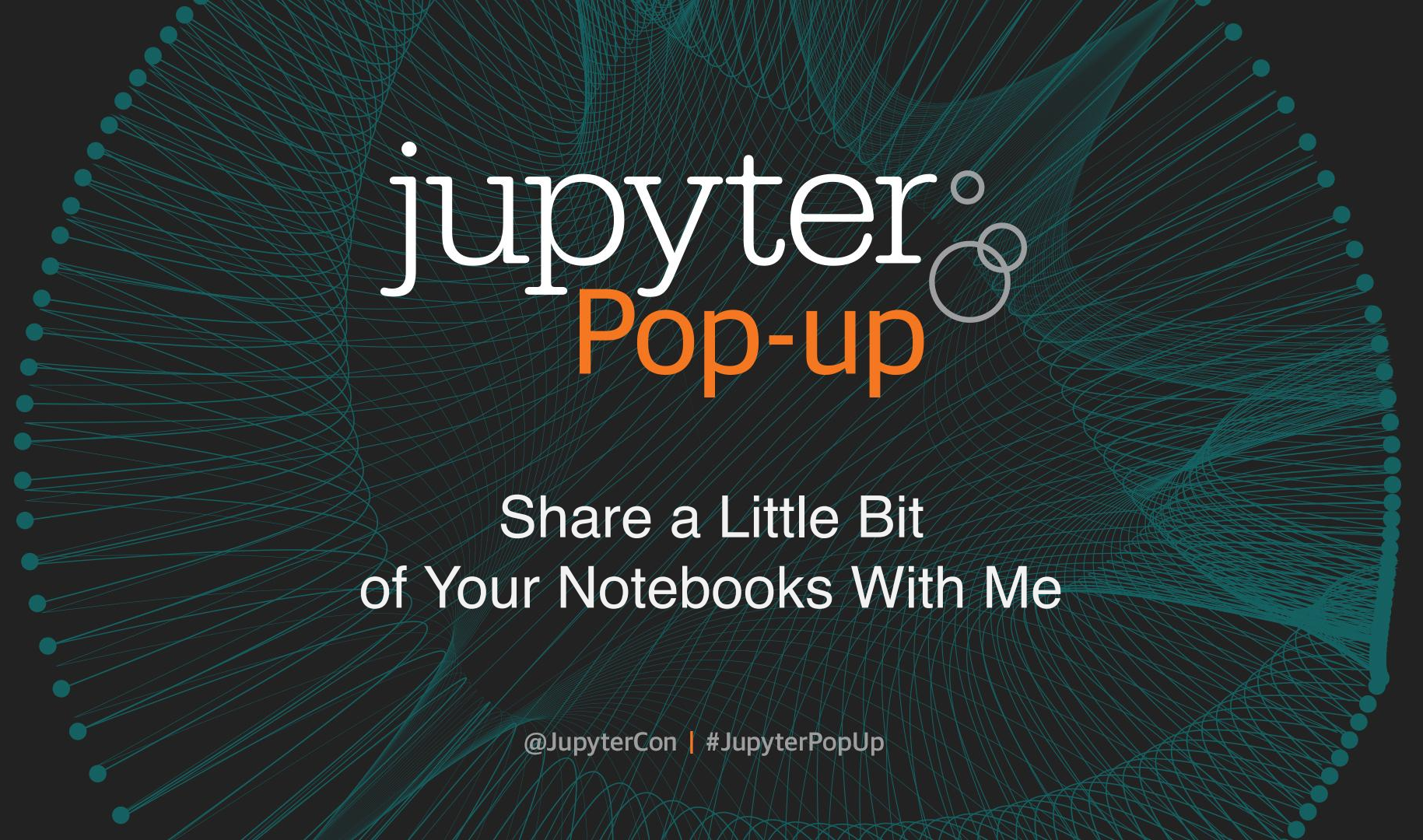
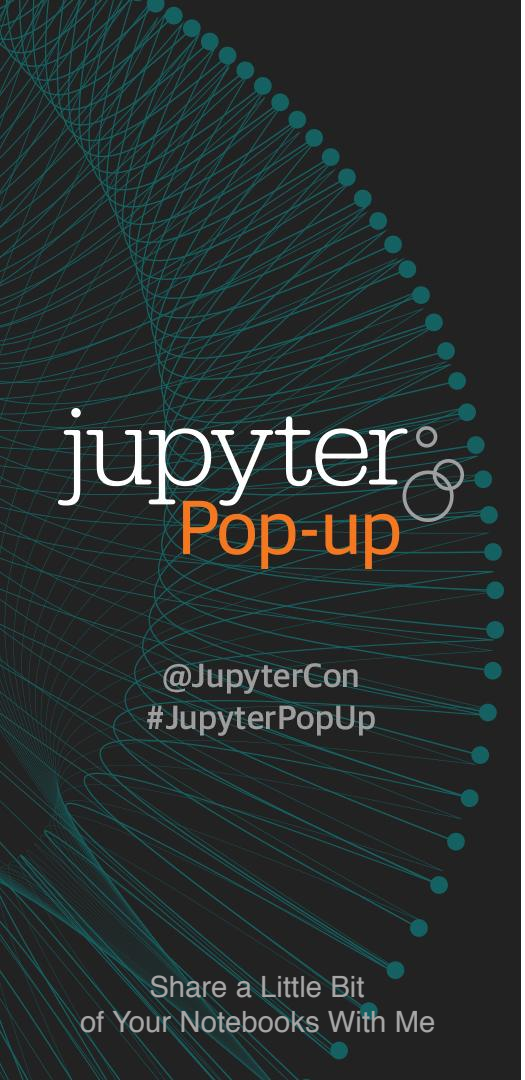

jupyter
Pop-up
Share a Little Bit
of Your Notebooks With Me
@JupyterCon | #JupyterPopUp



Hello! 👋

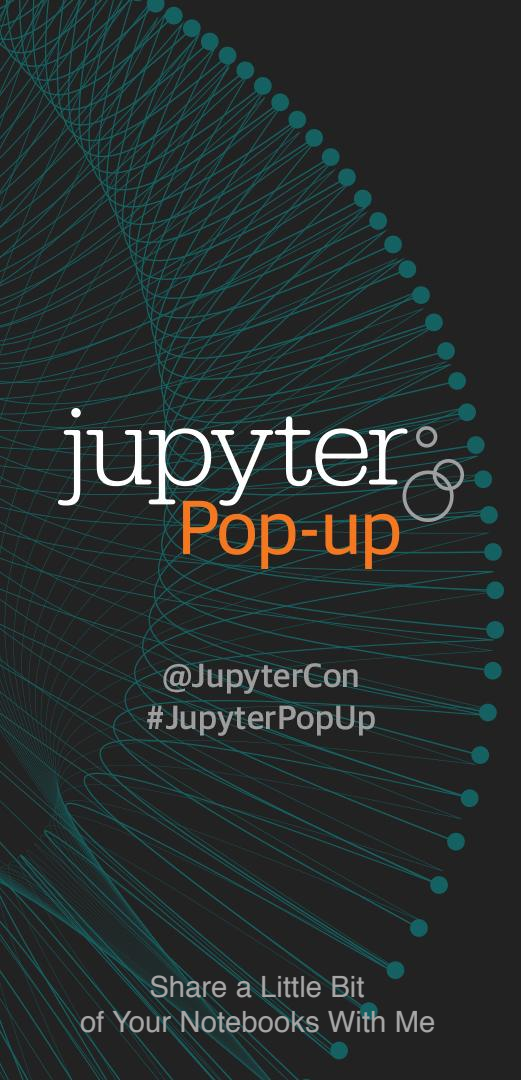
- Peter Parente (@parente)
- Jupyter Steering Council Member
- Valassis Digital employee
- Purveyor of the analytics user experience

jupyter
Pop-up

@JupyterCon
#JupyterPopUp

Share a Little Bit
of Your Notebooks With Me

That title … 🤔



Today's Topic

Tools and techniques for sharing notebooks, borne out of personal experience in open communities and private workplaces.

jupyter
Pop-up

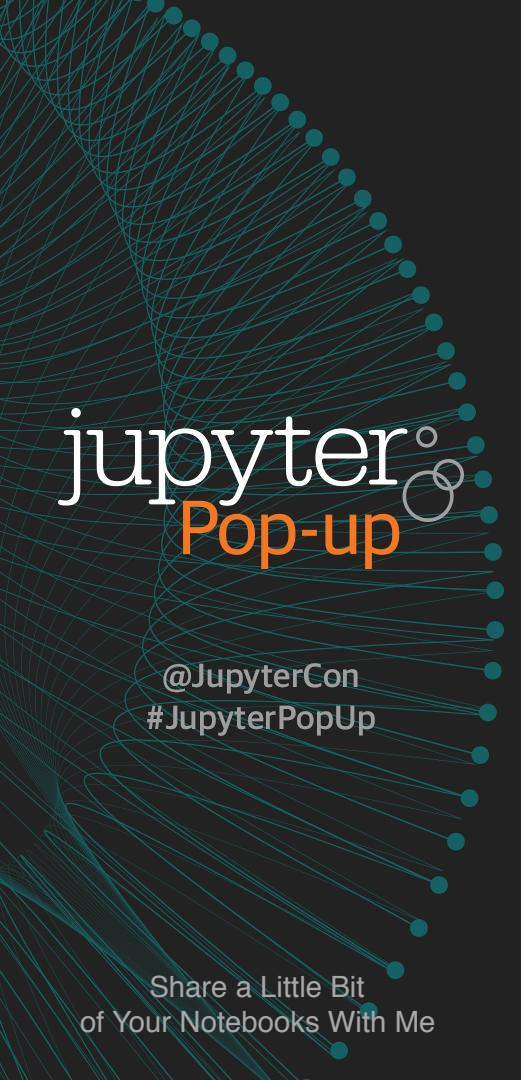
@JupyterCon
#JupyterPopUp

Share a Little Bit
of Your Notebooks With Me

Just post it on GitHub?

First, consider:

- Audience
- Reach
- Content
- Medium
- Reproducibility
- Longevity
- Privacy
- Technology



I'll give 🖐🏼 examples to you

jupyter
Pop-up

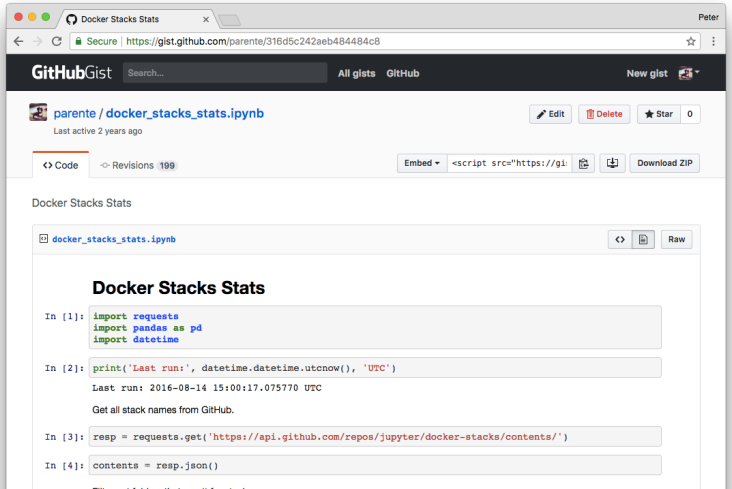
@JupyterCon
#JupyterPopUp

Share a Little Bit
of Your Notebooks With Me

Publish a one-off document

Consider a gist

Demo: Jupyter Docker Stats



jupyter
Pop-up

@JupyterCon
#JupyterPopUp

Share a Little Bit
of Your Notebooks With Me

Publish a one-off document

Consider a gist

1. Install a gist extension.
2. Enter a GitHub Personal Access token.
3. Click a button to publish.

jupyter
Pop-up

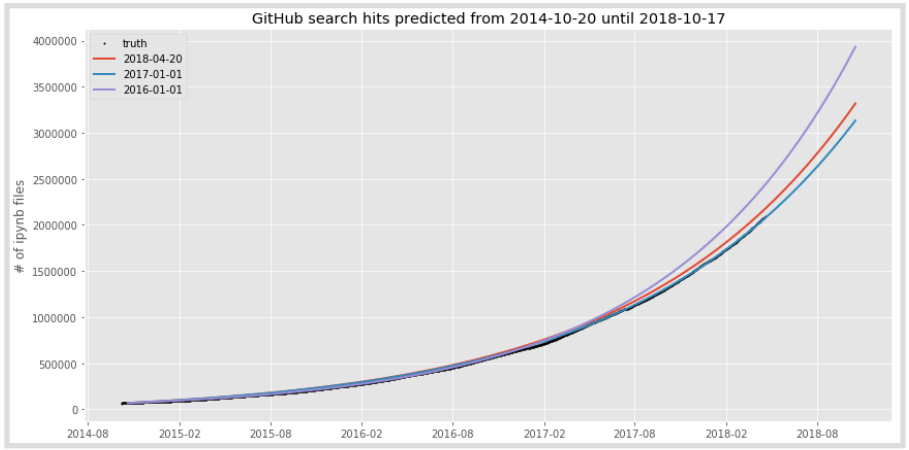
@JupyterCon
#JupyterPopUp

Share a Little Bit
of Your Notebooks With Me

Regularly update an analysis

Consider scripted notebook execution

Demo: nbestimate



jupyter
Pop-up

@JupyterCon
#JupyterPopUp

Share a Little Bit
of Your Notebooks With Me

Regularly update an analysis

Consider scripted notebook execution

1. Programmatically execute notebooks.
2. Commit and push to a git repository.
3. Automate what you can with bash, Python, nbflow, papermill, anaconda-project, etc.

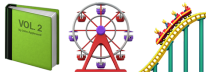
Detail: Using nbconvert to execute notebooks

```
import nbformat
from nbconvert.preprocessors import ExecutePreprocessor

with open('my_notebook.ipynb') as fh:
    nb = nbformat.read(fh, 4)

exp = ExecutePreprocessor(timeout=60)
rerun_nb, _ = exp.preprocess(nb, {})
```

jupyter Pop-up

@JupyterCon
#JupyterPopUp

Share a Little Bit
of Your Notebooks With Me



Create a learning playground

Consider Binder

Demo: Wild Wolf Watch

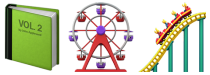
MUSEUM of LIFE + SCIENCE

Wildlands NETWORK

jupyter
Pop-up

@JupyterCon
#JupyterPopUp

Share a Little Bit
of Your Notebooks With Me



Create a learning playground

Consider Binder

1. Push a notebook and requirements to GitHub.
2. Enter your details at mybinder.org.
3. Get a link or button to share.

jupyter
Pop-up

@JupyterCon
#JupyterPopUp

Share a Little Bit
of Your Notebooks With Me

Museum Life+Science
@lifeandscience

New red wolf pups!🎉🐺The Animal Department counted three: 2 male & 1 female! All pups were noted as being in good health. On the Keeper blog: behind-the-scenes pics, their first vet check, & more! buff.ly/2qTfQjg #redwolves #newarrivals #lifeandscience #durhamnc

10:00 AM - Apr 22, 2018

♡ 44 💬 17 people are talking about this

twitter.com/lifeandscience/status/988054922355793921

jupyter Pop-up

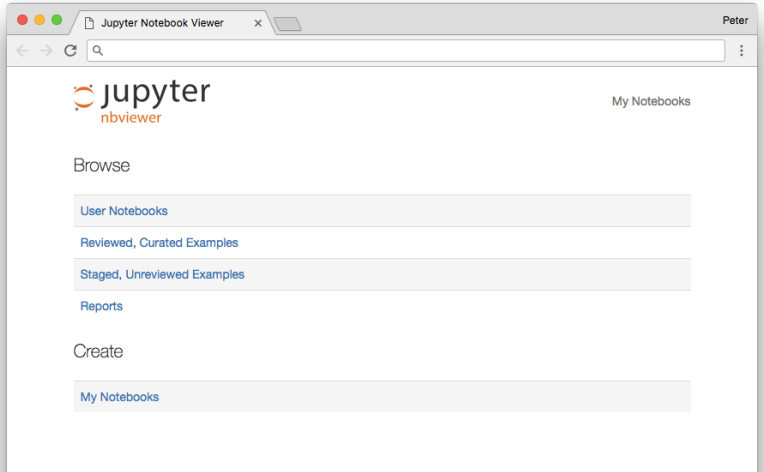
@JupyterCon
#JupyterPopUp

Share a Little Bit
of Your Notebooks With Me

👩🏾‍💻 📓 👨🏼‍💻 Access team work-in-progress

Consider running a local notebook viewer

Demo: nbviewer in JupyterHub



jupyter
Pop-up

@JupyterCon
#JupyterPopUp

Share a Little Bit
of Your Notebooks With Me

👩🏾‍💻 📓 👨🏼‍💻 Access team work-in-progress

Consider running a local notebook viewer

1. Install nbviewer or commuter.
2. Point it to user directories or S3 buckets.
3. Secure it (e.g., as a JupyterHub service).

Detail: nbviewer as a JupyterHub service

```
c.JupyterHub.services = [
    {
      'name': 'nbviewer',
      'url': 'http://127.0.0.1:9000',
      'cwd': '/home',
      'command': ['python', '-m', 'nbviewer', '--localfiles']
    }
]
```

jupyter
Pop-up

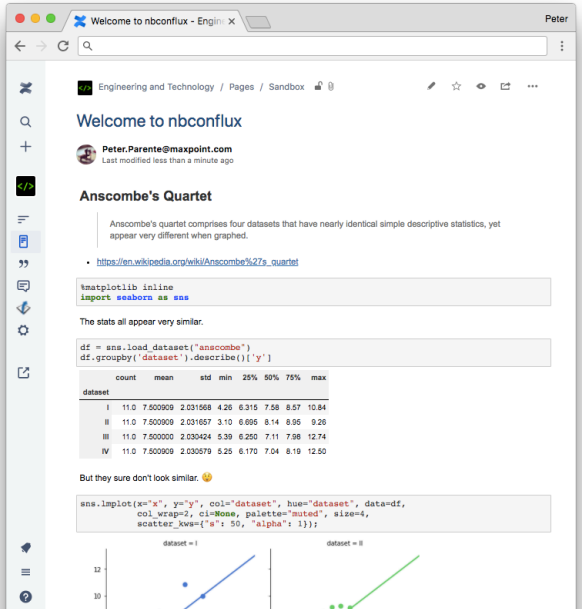
@JupyterCon
#JupyterPopUp

Share a Little Bit
of Your Notebooks With Me

Share with a broader org

Consider knowledge base integration

Demo: nbconflux



jupyter
Pop-up

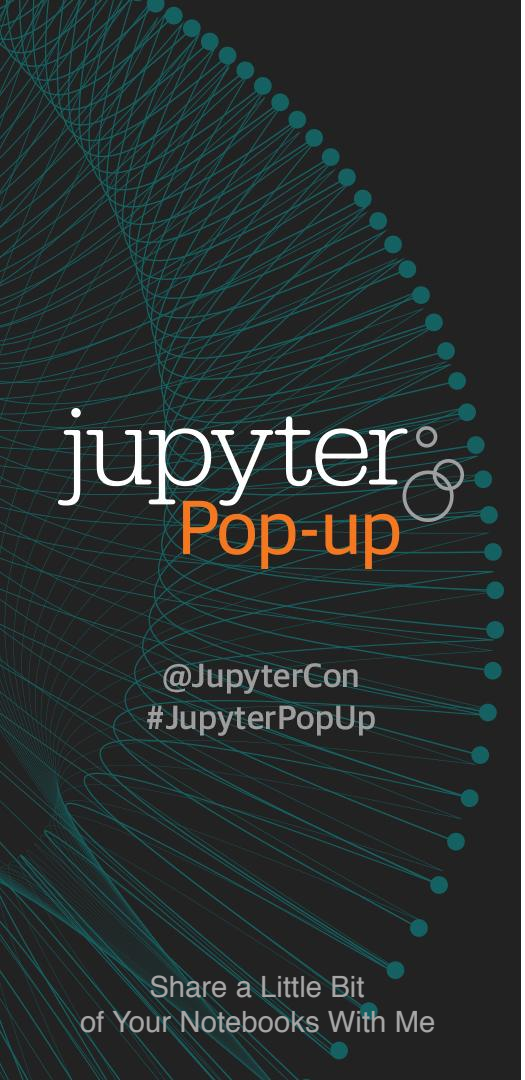
@JupyterCon
#JupyterPopUp

Share a Little Bit
of Your Notebooks With Me

👩🏼‍💻📕📘📗👨🏻‍💼👩🏽‍💼 Share with a broader org

Consider knowledge base integration

1. Use nbconvert preprocessors and exporters.
2. Package conversion scripts and APIs.
3. Create extensions for one-click publishing.

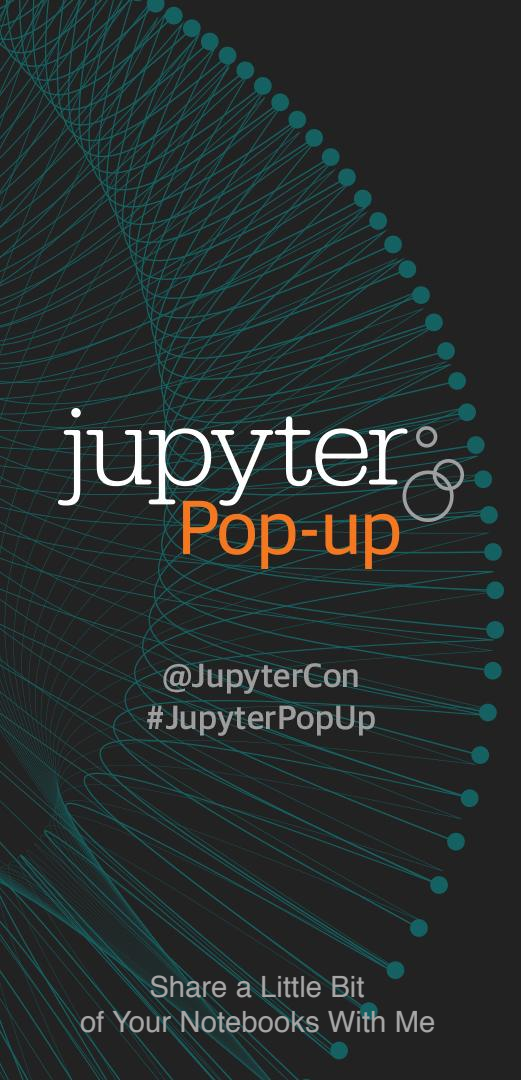


Come along!

- Jupyter notebooks are JSON files
- They have a formal schema
- nbconvert uses Python + jinja to transform them
- You can use whatever tools / languages you like
  - e.g., @nteract/notebook-preview is JS+React

Detail: A simple, valid notebook

```
import nbformat
simple = {
    "nbformat": 4,
    "nbformat_minor": 4,
    "metadata": {
        "kernelspec": {
            "name": "python3",
            "display_name": "Python 3"
        }
    },
    "cells": []
}
nbformat.validate(simple, version=4)
```



Going home

Remember:

- Many tools exist to help you share notebooks
- It's OK to use them as-is or extend them to fit your workflow
- Consider audience, longevity, privacy, reproducibility, etc. in your selection