

Jupyter as Your GitHub Console

Moshe Zadka – <https://cobordism.com>

2020

Health and Safety

Stay safe
Stay healthy
Wear masks

The World in 2020

...never mind

The World in 2019

- ▶ GitHub
- ▶ Microservices

The World in 2019

- ▶ GitHub
- ▶ Microservices
- ▶ Microrepositories

GitHub Automation

- ▶ Great API!
- ▶ Great UI!

GitHub Automation

- ▶ Great API!
- ▶ Great UI!
- ▶ But different

GitHub Automation

- ▶ Not optional: repos people+
- ▶ Dangerous
- ▶ Exploratory

Jupyter

Origin: data science

Features:

- ▶ Exploration
- ▶ Sharing

Browser Interaction

Universal UI

Notebooks

Artifacts of exploration session

Notebooks

Artifacts of exploration session
Share or export

github3.py

```
$ pip install github3.py
Collecting github3.py
...
Successfully installed ... github3.py-1.3.0 ...

import github3
```

Using GitHub3

```
tokenfile = os.path.expanduser("~/.github-token")
with open(tokenfile) as fpin:
    token = fpin.read().strip()
ghclient = github3.GitHub(token=token)

ghclient.organization("too-many-repos").name

'Demo Organization for Jupyter Talk'
```

Listing Repositories

```
organization = ghclient.organization(  
    "too-many-repos"  
)  
  
for repository in organization.repositories():  
    print(repository.name)  
  
project-1  
project-2  
project-3
```

Getting Content

```
project_1 = ghclient.repository(  
    "too-many-repos",  
    "project-1",  
)  
  
contents = project_1.file_contents("README.md")  
data = contents.decoded.decode("utf-8")  
print(data.splitlines()[0])  
  
# project-1
```


Summarizing Licenses

```
for repository in organization.repositories():  
    contents = repository.file_contents("LICENSE")  
    data = contents.decoded.decode("utf-8")  
    title = data.splitlines()[0].strip()  
    print(repository.name, title)
```

```
project-1 MIT License  
project-2 MIT License  
project-3 Apache License
```

Creating a Branch

```
root = next(project_1.commits())  
  
'1c6fc1633ae2345b63bd5002195bd22efe479861 '  
  
branch = project_1.create_branch_ref(  
    "change-readme",  
    sha=root.sha  
)
```

Changing a File

```
readme = project_1.file_contents(  
    "README.md",  
    ref=branch.ref  
)  
  
current = readme.decoded.decode("utf-8")  
current += "\nUpdated in November, 2020"  
  
readme.update(  
    "Updating thing",  
    content=current.encode("utf-8"),  
    branch=branch.ref,  
)  
  
{ 'content': <Contents [README.md]>,  
  'commit': <Commit [2314674db5f05bcd5dd68a20daca682  
'Last-Modified': '',  
  'ETag': 'W/"f8521eebc27a2546b8799d75e7c7e5c3bf62bd
```

Creating a PR

```
project_1.create_pull(  
    "Update README",  
    "main",  
    "change-readme",  
)
```

<Short Pull Request [#1]>

Verify Manually

smaller_image

Update README #1

Open

moshez wants to merge 1 commit into `main` from `change-readme`



Conversation 0



Commits 1



Checks 0



Files changed 1

Changes from all commits ▾ File filter... ▾ Jump to... ▾ ⚙ ▾

3 README.md

... -1 +1,2

1 - # project-1

1 + # project-1

2 + Updated in November, 2020

Figure: png

Loop

```
for project in organization.repositories():
    if project.name == "project-1":
        continue # skip
    root = next(project.commits())
    branch = project.create_branch_ref("change-readme")
    readme = project.file_contents("README.md", ref=branch)
    current = readme.decoded.decode("utf-8")
    current += "\nUpdated in November, 2020"
    readme.update("Updating thing", content=current)
    project.create_pull("Update README", "main", "c
```

UI vs API

Writing new code? Use UI

UI vs API

Writing new code? Use UI
Need consistency? API!

Everything via Source Control

PR/Issue templates

Actions

README

License

...

Sometimes You F Up

To mess up one repository is human, to destroy all of them takes a computer...and a human

Sometimes You F Up

To mess up one repository is human, to destroy all of them takes a computer...and a human

Attaching notebooks to tickets

Repeatable and documented

Conclusion

If you are using GitHub the way it was meant to be used...

Conclusion

If you are using GitHub the way it was meant to be used...
have a plan for automating changes.