

Introduction to GitHub tutorial: A basic visual guide

Stephanie N. DeCross

Harvard University, Department of Psychology

sdecross@g.harvard.edu | <https://github.com/sdecross>

How do I use GitHub to collaborate on a project?

Local computer

The cloud

GitHub: Parent Repo

<https://github.com/impt-person/group-project>

Local computer

The cloud

GitHub: Parent Repo

<https://github.com/impt-person/group-project>

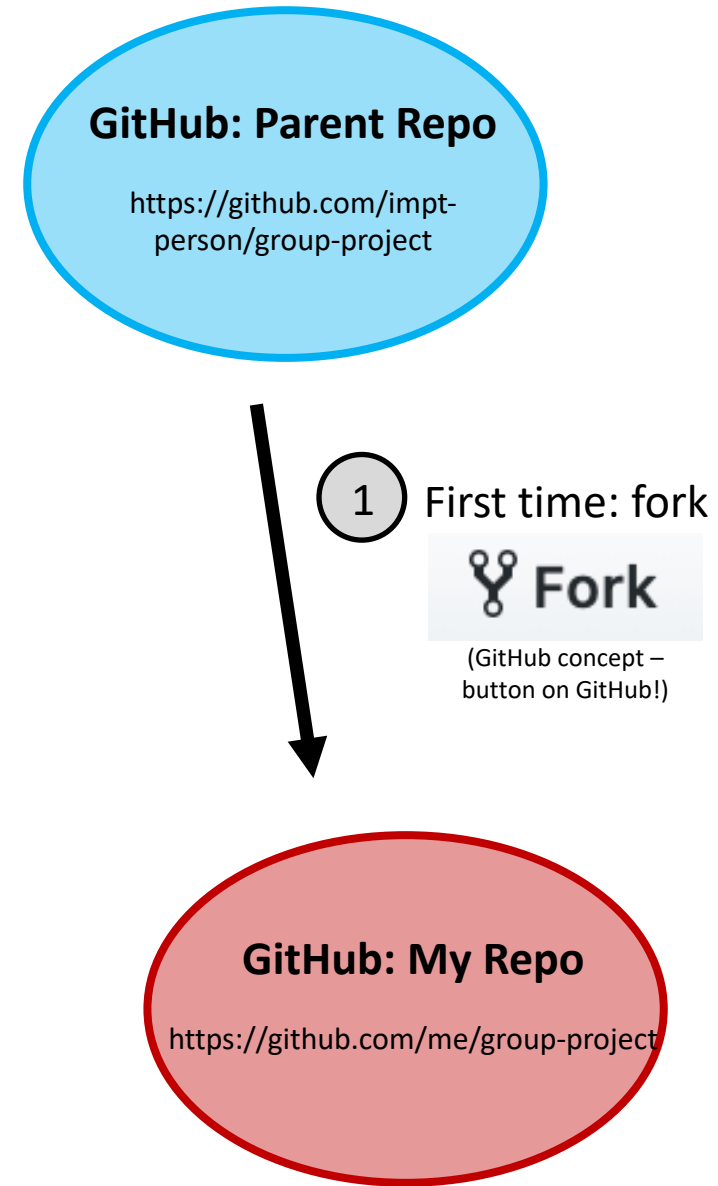
GitHub: My Repo

<https://github.com/me/group-project>

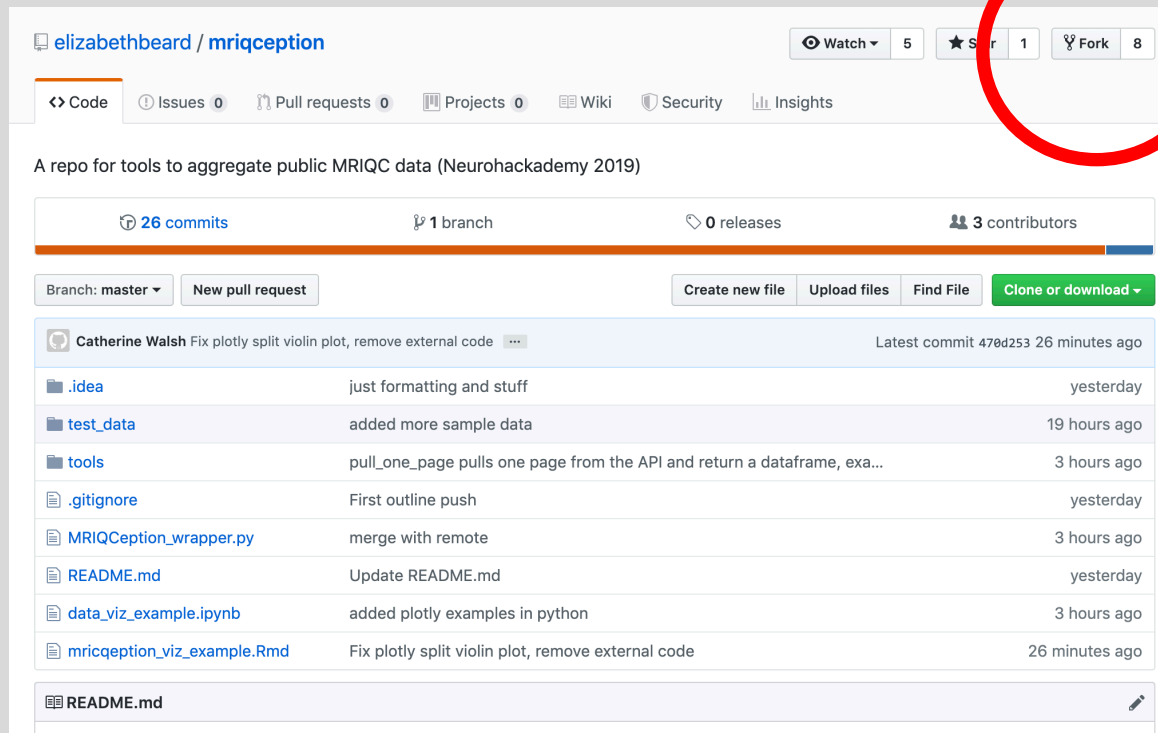
Local computer

The cloud

Local computer



The cloud



Local computer

GitHub: Parent Repo
<https://github.com/impt-person/group-project>

1 First time: fork

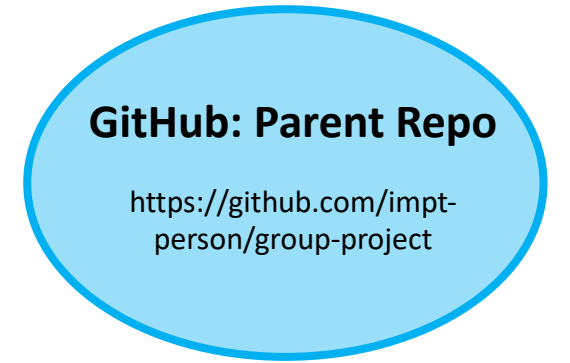
 **Fork**
(GitHub concept – button on GitHub!)

GitHub: My Repo
<https://github.com/me/group-project>

The cloud



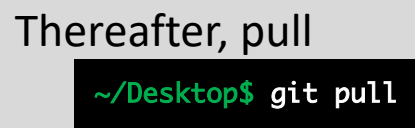
Local computer



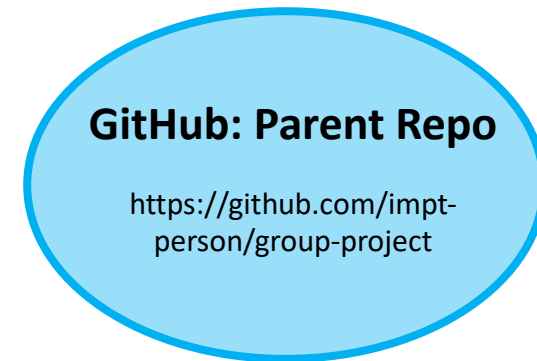
1 First time: fork



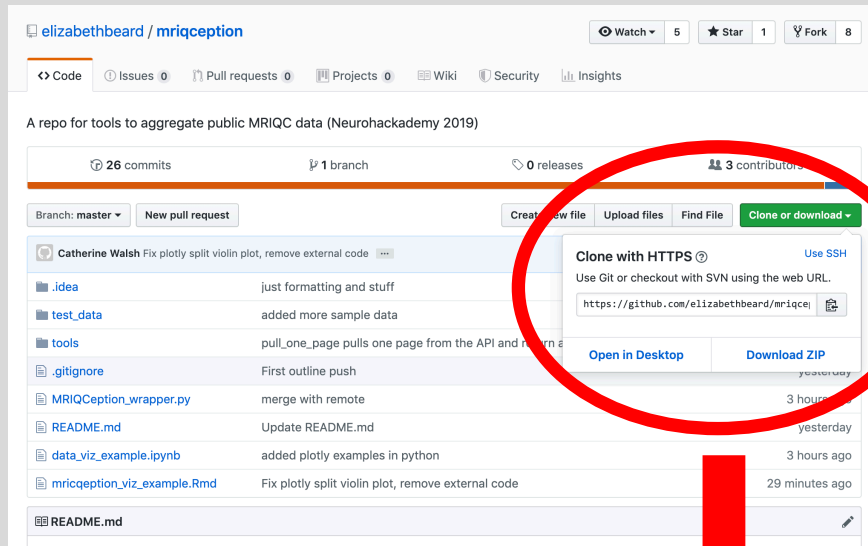
The cloud



Local computer



The cloud



My Laptop
~/Desktop/group-project

2 Clone (first time)

```
~/Desktop$ git clone  
https://github.com/impt-  
person/group-project.git
```

Thereafter, pull

```
~/Desktop$ git pull
```

Local computer

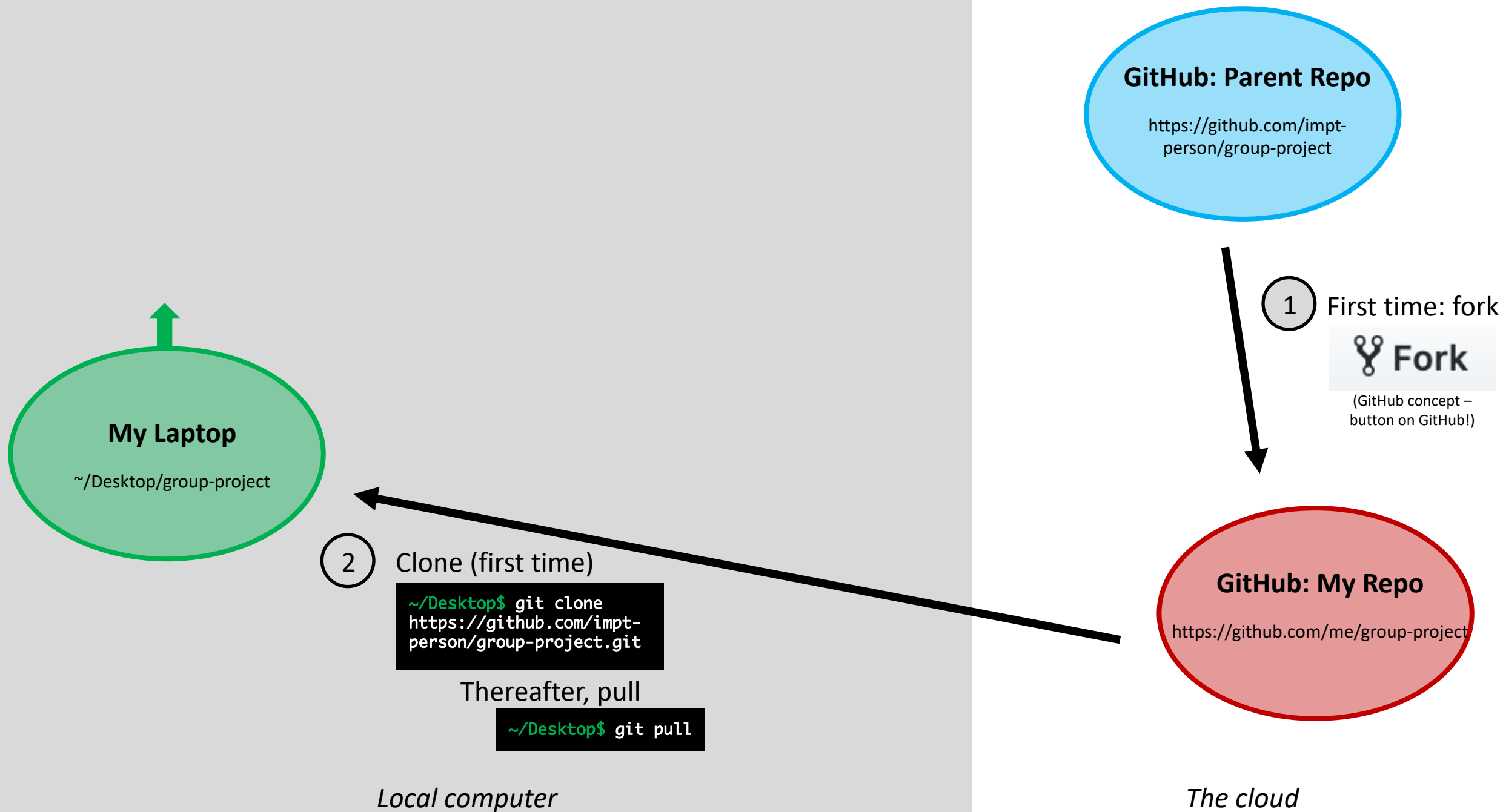
GitHub: Parent Repo
<https://github.com/impt-person/group-project>

1 First time: fork



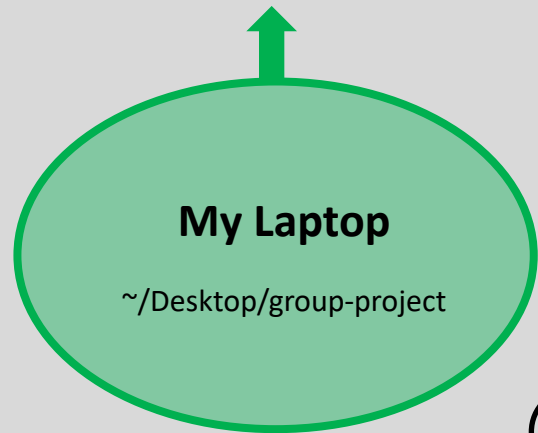
GitHub: My Repo
<https://github.com/me/group-project>

The cloud



```
1 #####
2 # LOAD LIBRARIES
3 #####
4
5 if (!require("tidyverse")) {install.packages("tidyverse"); require("tidyverse")}
6
7 #####
8 # FANTASTIC CODE
9 #####
10 load("super-cool-data.RData")
11
12
13 # data wrangling
14 for (object in seq_along(super-cool-data)){
15   super-cool-data-final[[object]] <- super-cool-data[[object]] %>%
16     gather(key = "stim", value = "response", 3:10) %>%
17     mutate(block = if_else(substr(stim, 4, 5) == "_1", 1,
18                          if_else(substr(stim, 4, 5) == "_2", 2,
```

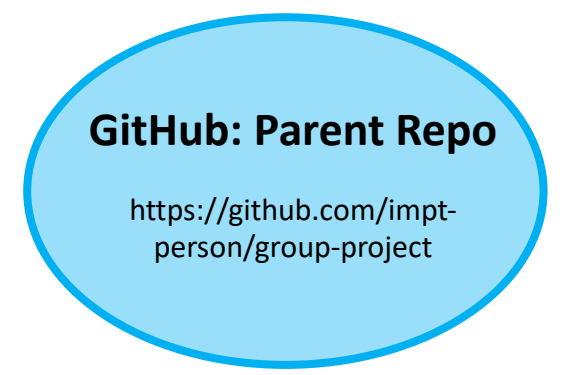
3a Work on file mycode.R and save



2 Clone (first time)
`~/Desktop$ git clone https://github.com/impt-person/group-project.git`

Thereafter, pull
`~/Desktop$ git pull`

Local computer



1 First time: fork
 Fork
(GitHub concept – button on GitHub!)

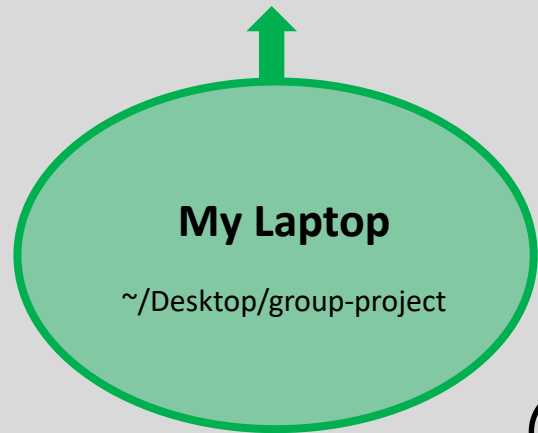


The cloud

```
1 #####
2 # LOAD LIBRARIES
3 #####
4
5 if (!require("tidyverse")) {install.packages("tidyverse"); require("tidyverse")}
6
7 #####
8 # FANTASTIC CODE
9 #####
10 load("super-cool-data.RData")
11
12
13 # data wrangling
14 for (object in seq_along(super-cool-data)){
15   super-cool-data-final[[object]] <- super-cool-data[[object]] %>%
16     gather(key = "stim", value = "response", 3:10) %>%
17     mutate(block = if_else(substr(stim, 4, 5) == "_1", 1,
18                          if_else(substr(stim, 4, 5) == "_2", 2,
```

3a Work on file mycode.R and save

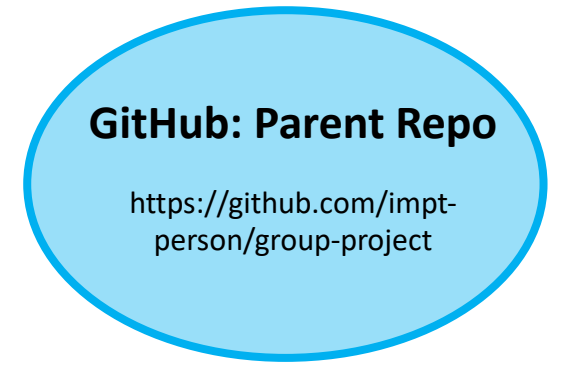
3b Add file to git
group-project\$ git add mycode.R



2 Clone (first time)
~/Desktop\$ git clone https://github.com/impt-person/group-project.git

Thereafter, pull
~/Desktop\$ git pull

Local computer



1 First time: fork
Fork
(GitHub concept – button on GitHub!)



The cloud

```
1 #####
2 # LOAD LIBRARIES
3 #####
4
5 if (!require("tidyverse")) {install.packages("tidyverse"); require("tidyverse")}
6
7 #####
8 # FANTASTIC CODE
9 #####
10
11 load("super-cool-data.RData")
12
13 # data wrangling
14 for (object in seq_along(super-cool-data)){
15   super-cool-data-final[[object]] <- super-cool-data[[object]] %>%
16     gather(key = "stim", value = "response", 3:10) %>%
17     mutate(block = if_else(substr(stim, 4, 5) == "_1", 1,
18                          if_else(substr(stim, 4, 5) == "_2", 2,
```

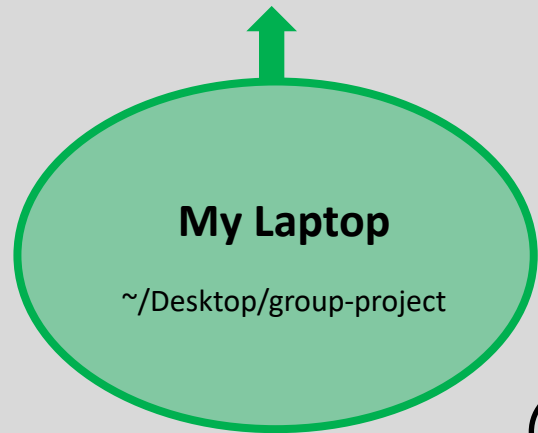
3a Work on file mycode.R and save

3b Add file to git

```
group-project$ git add mycode.R
```

3c Check status

```
group-project$ git status
```



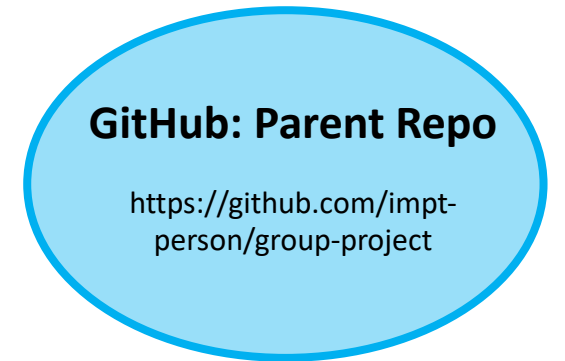
2 Clone (first time)

```
~/Desktop$ git clone https://github.com/impt-person/group-project.git
```

Thereafter, pull

```
~/Desktop$ git pull
```

Local computer



1 First time: fork



(GitHub concept – button on GitHub!)



The cloud

```
1 #####
2 # LOAD LIBRARIES
3 #####
4
5 if (!require("tidyverse")) {install.packages("tidyverse"); require("tidyverse")}
6
7 #####
8 # FANTASTIC CODE
9 #####
10
11 load("super-cool-data.RData")
12
13 # data wrangling
14 for (object in seq_along(super-cool-data)){
15   super-cool-data-final[[object]] <- super-cool-data[[object]] %>%
16     gather(key = "stim", value = "response", 3:10) %>%
17     mutate(block = if_else(substr(stim, 4, 5) == "_1", 1,
18                          if_else(substr(stim, 4, 5) == "_2", 2,
```

3a Work on file mycode.R and save

3b Add file to git

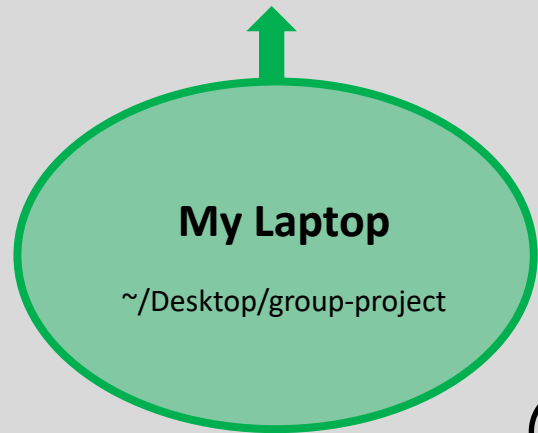
```
group-project$ git add mycode.R
```

3c Check status

```
group-project$ git status
```

3d Commit

```
group-project$ git commit
```



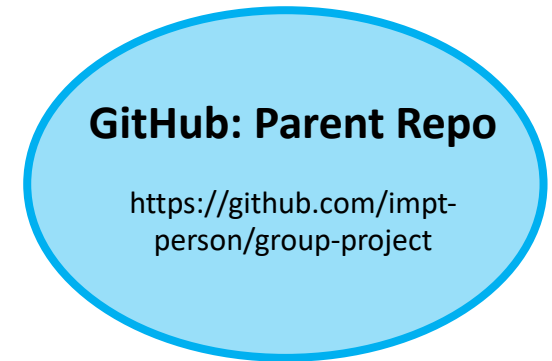
2 Clone (first time)

```
~/Desktop$ git clone https://github.com/impt-person/group-project.git
```

Thereafter, pull

```
~/Desktop$ git pull
```

Local computer



1 First time: fork



The cloud

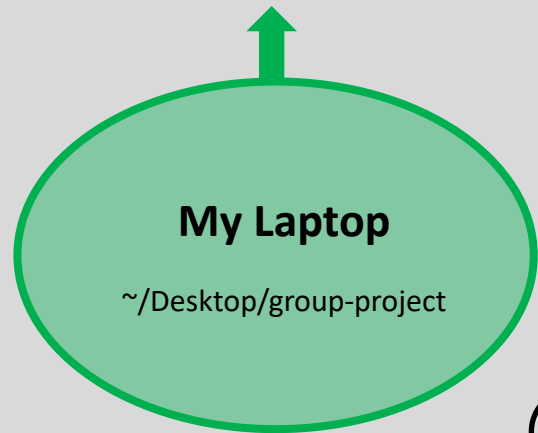
```
1 #####
2 # LOAD LIBRARIES
3 #####
4
5 if (!require("tidyverse")) {install.packages("tidyverse"); require("tidyverse")}
6
7 #####
8 # FANTASTIC CODE
9 #####
10
11 load("super-cool-data.RData")
12
13 # data wrangling
14 for (object in seq_along(super-cool-data)){
15   super-cool-data-final[[object]] <- super-cool-data[[object]] %>%
16     gather(key = "stim", value = "response", 3:10) %>%
17     mutate(block = if_else(substr(stim, 4, 5) == "_1", 1,
18                          if_else(substr(stim, 4, 5) == "_2", 2,
```

3a Work on file mycode.R and save

3b Add file to git
`group-project$ git add mycode.R`

3c Check status
`group-project$ git status`

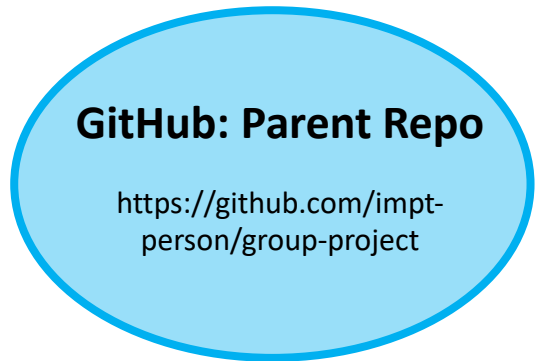
3d Commit
`group-project$ git commit`



2 Clone (first time)
`~/Desktop$ git clone https://github.com/impt-person/group-project.git`

Thereafter, pull
`~/Desktop$ git pull`

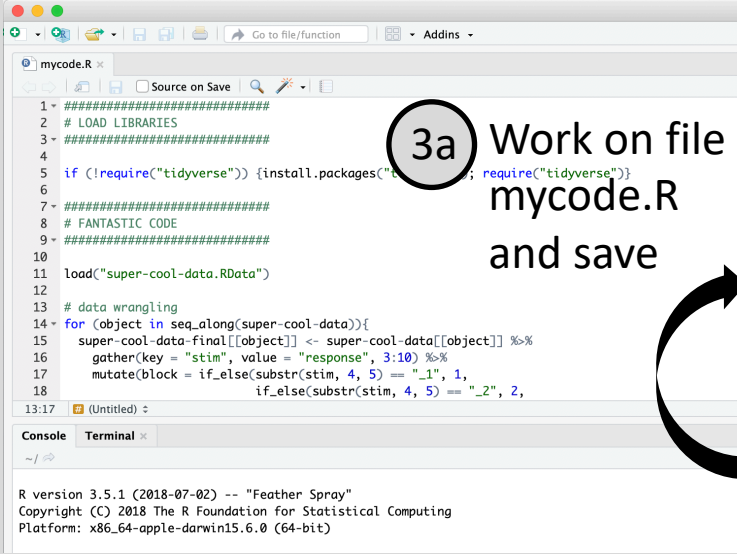
Local computer



1 First time: fork
 Fork
(GitHub concept – button on GitHub!)



The cloud

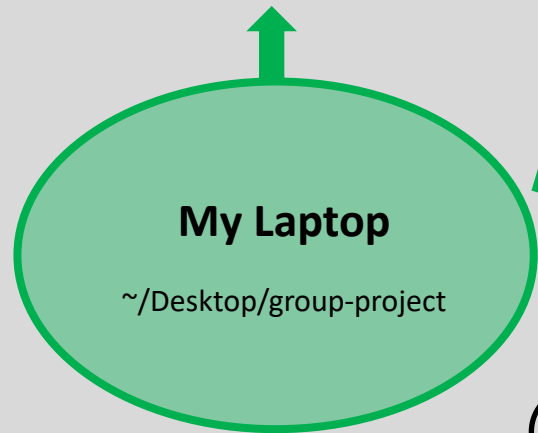


3a Work on file mycode.R and save

3b Add file to git
`group-project$ git add mycode.R`

3c Check status
`group-project$ git status`

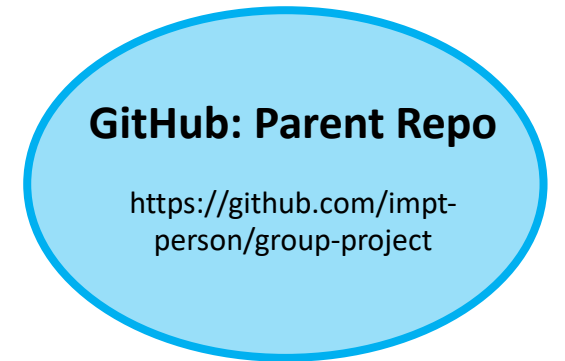
3d Commit
`group-project$ git commit`



2 Clone (first time)
`~/Desktop$ git clone https://github.com/impt-person/group-project.git`

Thereafter, pull
`~/Desktop$ git pull`

Local computer



1 First time: fork
 Fork
(GitHub concept – button on GitHub!)



The cloud

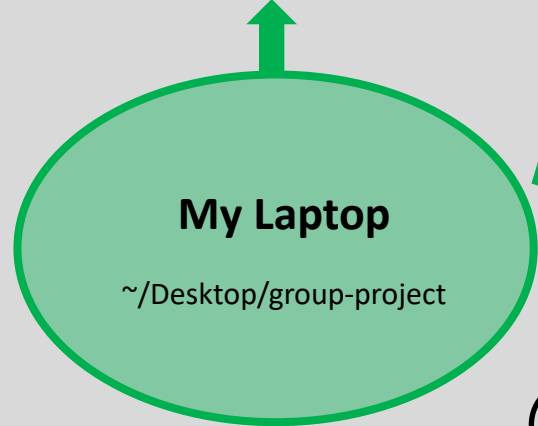
```
1 #####
2 # LOAD LIBRARIES
3 #####
4
5 if (!require("tidyverse")) {install.packages("tidyverse"); require("tidyverse")}
6
7 #####
8 # FANTASTIC CODE
9 #####
10
11 load("super-cool-data.RData")
12
13 # data wrangling
14 for (object in seq_along(super-cool-data)){
15   super-cool-data-final[[object]] <- super-cool-data[[object]] %>%
16     gather(key = "stim", value = "response", 3:10) %>%
17     mutate(block = if_else(substr(stim, 4, 5) == "_1", 1,
18                           if_else(substr(stim, 4, 5) == "_2", 2,
```

3a Work on file mycode.R and save

3b Add file to git
`group-project$ git add mycode.R`

3c Check status
`group-project$ git status`

3d Commit
`group-project$ git commit`

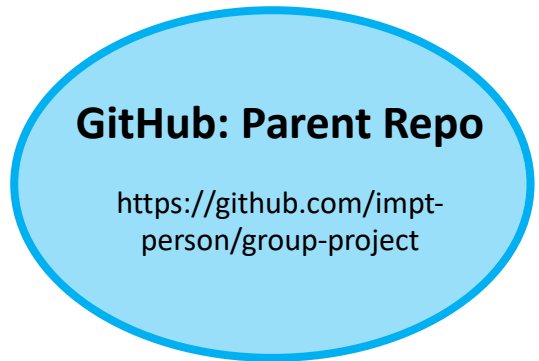


2 Clone (first time)
`~/Desktop$ git clone https://github.com/impt-person/group-project.git`

Thereafter, pull
`~/Desktop$ git pull`

Local computer

4 Push to origin (My Repo by default!)
`group-project$ git push`



1 First time: fork
 Fork
(GitHub concept – button on GitHub!)



The cloud

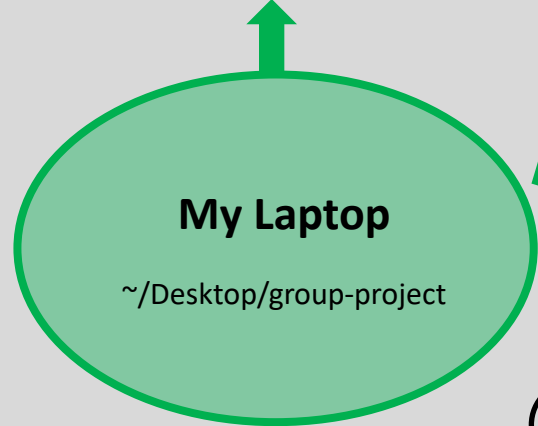
```
1 #####
2 # LOAD LIBRARIES
3 #####
4
5 if (!require("tidyverse")) {install.packages("tidyverse"); require("tidyverse")}
6
7 #####
8 # FANTASTIC CODE
9 #####
10
11 load("super-cool-data.RData")
12
13 # data wrangling
14 for (object in seq_along(super-cool-data)){
15   super-cool-data-final[[object]] <- super-cool-data[[object]] %>%
16     gather(key = "stim", value = "response", 3:10) %>%
17     mutate(block = if_else(substr(stim, 4, 5) == "_1", 1,
18                          if_else(substr(stim, 4, 5) == "_2", 2,
```

3a Work on file mycode.R and save

3b Add file to git
`group-project$ git add mycode.R`

3c Check status
`group-project$ git status`

3d Commit
`group-project$ git commit`

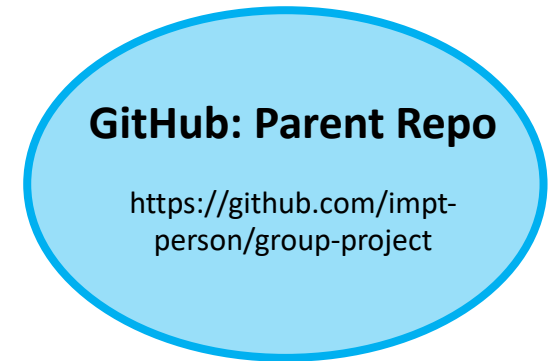


2 Clone (first time)
`~/Desktop$ git clone https://github.com/impt-person/group-project.git`

Thereafter, pull
`~/Desktop$ git pull`

Local computer

4 Push to origin (My Repo by default!)
`group-project$ git push`



1 First time: fork
 Fork
(GitHub concept – button on GitHub!)



The cloud

```

mycode.R
1 #####
2 # LOAD LIBRARIES
3 #####
4
5 if (!require("tidyverse")) {install.packages("tidyverse"); require("tidyverse")}
6
7 #####
8 # FANTASTIC CODE
9 #####
10
11 load("super-cool-data.RData")
12
13 # data wrangling
14 for (object in seq_along(super-cool-data)){
15   super-cool-data-final[[object]] <- super-cool-data[[object]] %>%
16     gather(key = "stim", value = "response", 3:10) %>%
17     mutate(block = if_else(substr(stim, 4, 5) == "_1", 1,
18                          if_else(substr(stim, 4, 5) == "_2", 2,

```

3a Work on file mycode.R and save

3b Add file to git
`group-project$ git add mycode.R`

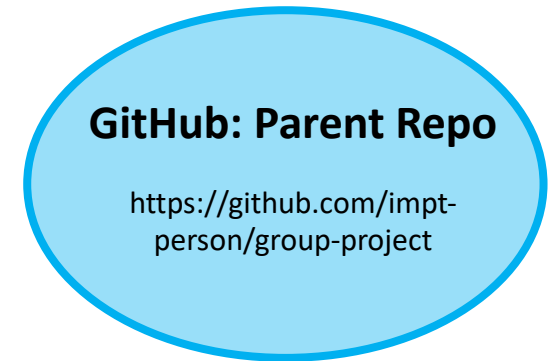
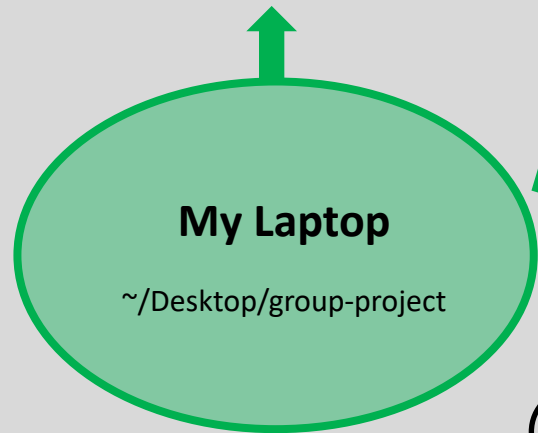
3c Check status
`group-project$ git status`

3d Commit
`group-project$ git commit`

4 Push to origin (My Repo by default!)
`group-project$ git push`

2 Clone (first time)
`~/Desktop$ git clone https://github.com/impt-person/group-project.git`

Thereafter, pull
`~/Desktop$ git pull`



5 Pull request
 New pull request
 (GitHub concept – button on GitHub!)

1 First time: fork
 Fork
 (GitHub concept – button on GitHub!)



Local computer

The cloud

```

1 #####
2 # LOAD LIBRARIES
3 #####
4
5 if (!require("tidyverse")) {install.packages("tidyverse"); require("tidyverse")}
6
7 #####
8 # FANTASTIC CODE
9 #####
10
11 load("super-cool-data.RData")
12
13 # data wrangling
14 for (object in seq_along(super-cool-data)){
15   super-cool-data-final[[object]] <- super-cool-data[[object]] %>%
16     gather(key = "stim", value = "response", 3:10) %>%
17     mutate(block = if_else(substr(stim, 4, 5) == "_1", 1,
18                          if_else(substr(stim, 4, 5) == "_2", 2,

```

3a Work on file mycode.R and save

3b Add file to git
`group-project$ git add mycode.R`

3c Check status
`group-project$ git status`

3d Commit
`group-project$ git commit`

My Laptop
 ~/Desktop/group-project

elizabethbeard / mriqcception

Code Issues 0 Pull requests 0 Projects 0 Wiki Security

A repo for tools to aggregate public MRIQC data (Neurohackademy)

commits 0 releases 3 contributors

Branch: master New pull request Create new file Upload files Find File Clone or download

Catherine	Fix plotly split violin plot, remove external code	Latest commit 470d253 1 hour ago
.idea	just formatting and stuff	yesterday
test_data	added more sample data	20 hours ago
tools	pull_one_page pulls one page from the API and return a dataframe, exa...	3 hours ago
.gitignore	First outline push	yesterday
MRIQCception_wrapper.py	merge with remote	4 hours ago
README.md	Update README.md	yesterday
data_viz_example.ipynb	added plotly examples in python	3 hours ago
mriqcception_viz_example.Rmd	Fix plotly split violin plot, remove external code	1 hour ago

Local computer

GitHub: Parent Repo
<https://github.com/impt-person/group-project>

5 Pull request
 New pull request
 (GitHub concept – button on GitHub!)

1 First time: fork
 Fork
 (GitHub concept – button on GitHub!)

origin
 GitHub: My Repo
<https://github.com/me/group-project>

The cloud

```
mycode.R
1 #####
2 # LOAD LIBRARIES
3 #####
4
5 if (!require("tidyverse")) {install.packages("tidyverse"); require("tidyverse")}
6
7 #####
8 # FANTASTIC CODE
9 #####
10
11 load("super-cool-data.RData")
12
13 # data wrangling
14 for (object in seq_along(super-cool-data)){
15   super-cool-data-final[[object]] <- super-cool-data[[object]] %>%
16     gather(key = "stim", value = "response", 3:10) %>%
17     mutate(block = if_else(substr(stim, 4, 5) == "_1", 1,
18                           if_else(substr(stim, 4, 5) == "_2", 2,
```

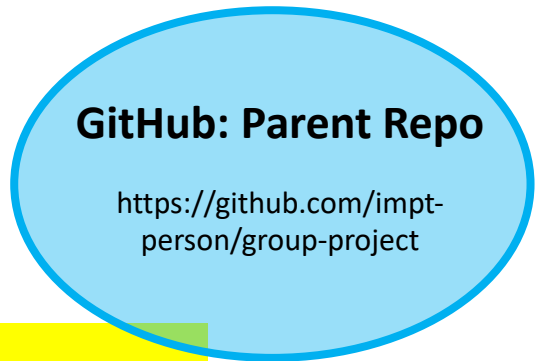
3a Work on file mycode.R and save

3b Add file to git

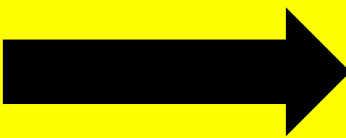
```
group-project$
git add mycode.R
```

3c Check status

```
group-project$
git status
```

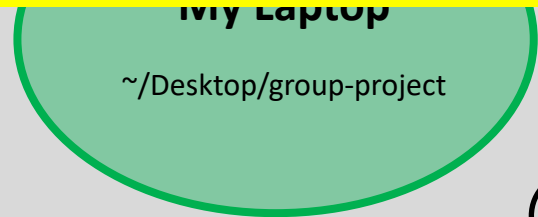


If you have direct write access to the parent repo, you can push directly to upstream (it doesn't require permission, like a pull request). This would take the place of step 5.
~/Desktop\$ git push upstream master



5 Pull request
New pull request
(GitHub concept – button on GitHub!)

1 First time: fork
Fork
(GitHub concept – button on GitHub!)



2 Clone (first time)
~/Desktop\$ git clone https://github.com/impt-person/group-project.git

Thereafter, pull
~/Desktop\$ git pull

4 Push to origin
(My Repo by default!)
group-project\$ git push



Local computer

The cloud

What if someone makes changes to the parent repo?
How do I get the latest version?

```

mycode.R
1 #####
2 # LOAD LIBRARIES
3 #####
4
5 if (!require("tidyverse")) {install.packages("tidyverse"); require("tidyverse")}
6
7 #####
8 # FANTASTIC CODE
9 #####
10
11 load("super-cool-data.RData")
12
13 # data wrangling
14 for (object in seq_along(super-cool-data)){
15   super-cool-data-final[[object]] <- super-cool-data[[object]] %>%
16     gather(key = "stim", value = "response", 3:10) %>%
17     mutate(block = if_else(substr(stim, 4, 5) == "_1", 1,
18                          if_else(substr(stim, 4, 5) == "_2", 2,

```

3a Work on file mycode.R and save

3b Add file to git
`group-project$ git add mycode.R`

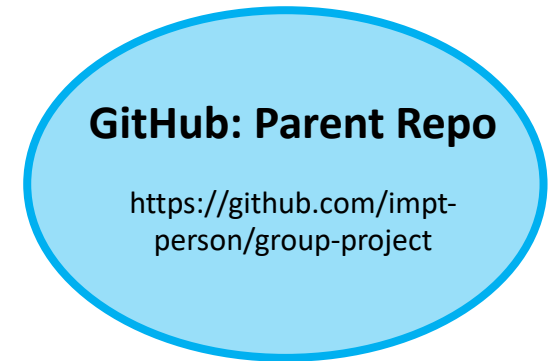
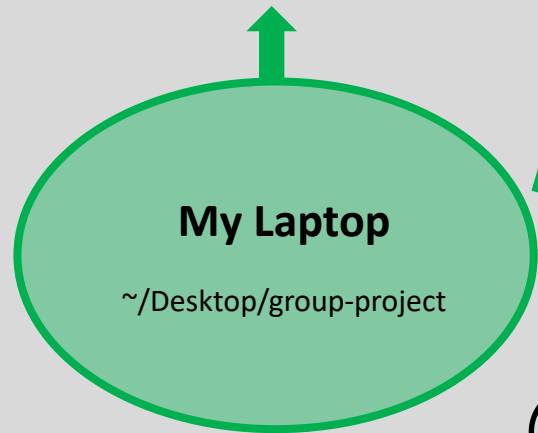
3c Check status
`group-project$ git status`

3d Commit
`group-project$ git commit`

4 Push to origin (My Repo by default!)
`group-project$ git push`

2 Clone (first time)
`~/Desktop$ git clone https://github.com/impt-person/group-project.git`

Thereafter, pull
`~/Desktop$ git pull`



5 Pull request
 New pull request
 (GitHub concept – button on GitHub!)

1 First time: fork
 Fork
 (GitHub concept – button on GitHub!)

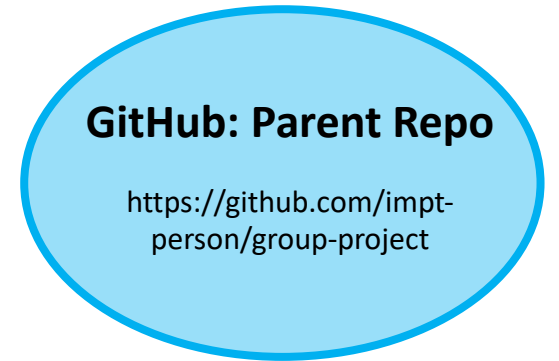


Local computer

The cloud



Local computer

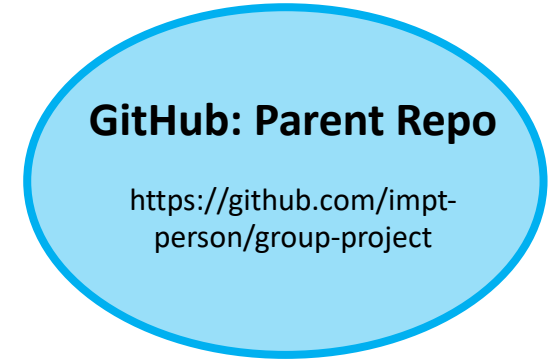


The cloud

Get updates from parent repo



Local computer

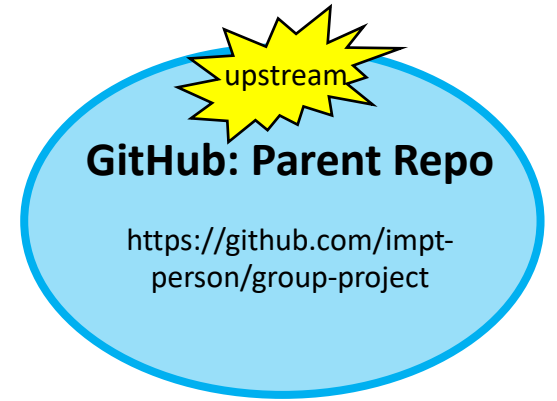


The cloud

Get updates from parent repo



Local computer



The cloud

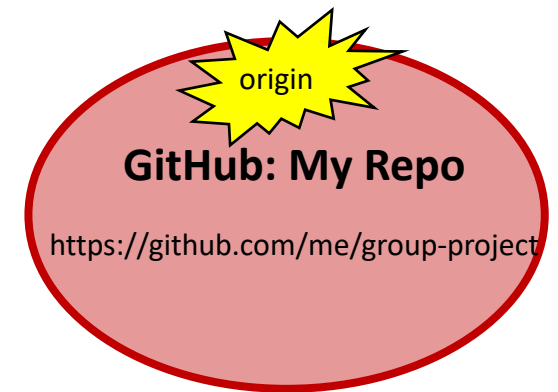
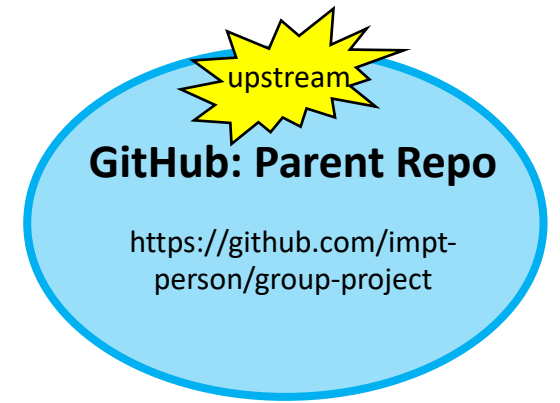
Get updates from parent repo



- 1 Link the parent repo to your local project and check that it connected (Only need to do once!)

```
group-project$ git remote add upstream  
https://github.com/impt-person/group-  
project.git  
group-project$ git remote -v
```

Local computer



The cloud

Get updates from parent repo



1 Link the parent repo to your local project and check that it connected (Only need to do once!)

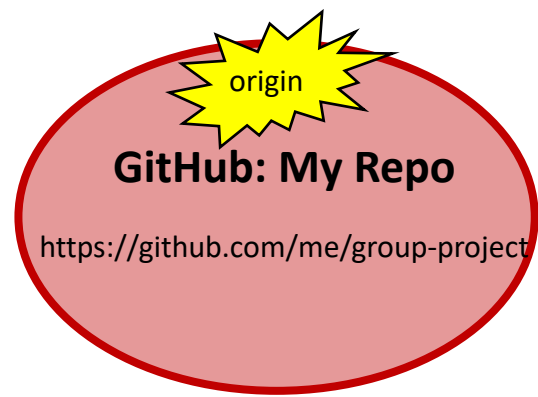
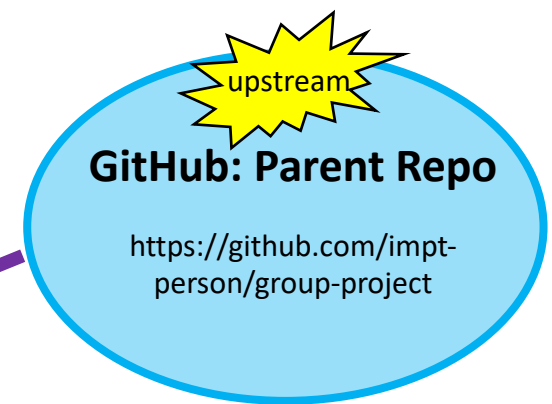
```
group-project$ git remote add upstream  
https://github.com/impt-person/group-project.git  
group-project$ git remote -v
```

Local computer

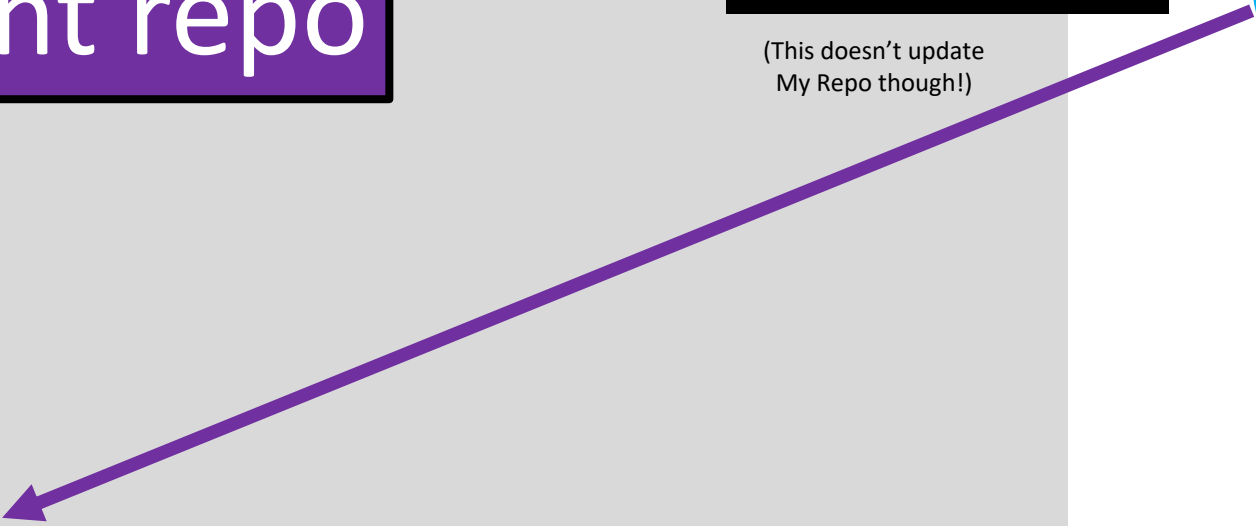
2 Pull from parent repo to local project

```
group-project$ git pull  
upstream master
```

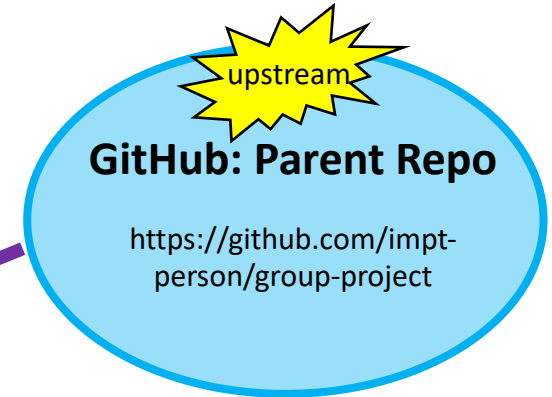
(This doesn't update My Repo though!)



The cloud



Get updates from parent repo



1 Link the parent repo to your local project and check that it connected (Only need to do once!)

```
group-project$ git remote add upstream https://github.com/impt-person/group-project.git group-project$ git remote -v
```

2 Pull from parent repo to local project

```
group-project$ git pull upstream master
```

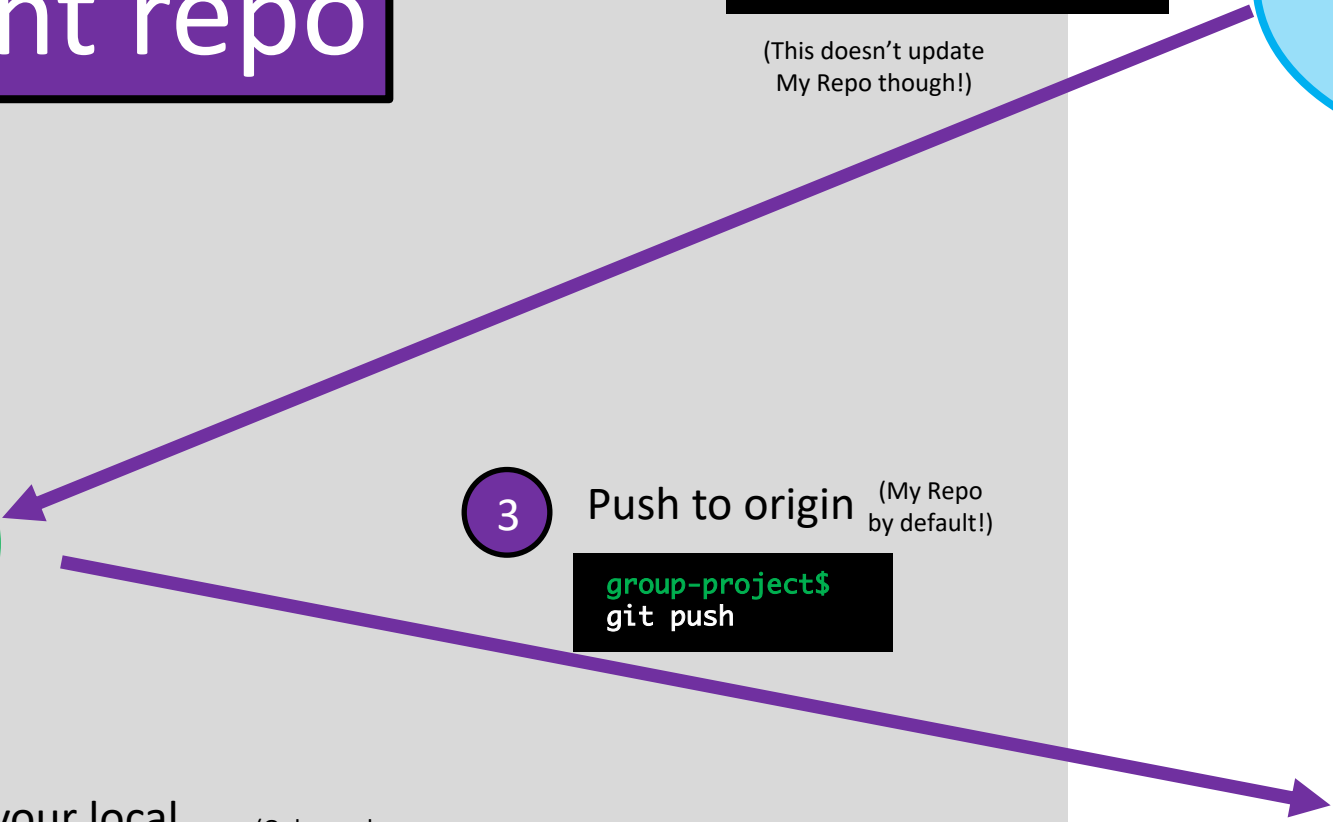
(This doesn't update My Repo though!)

3 Push to origin (My Repo by default!)

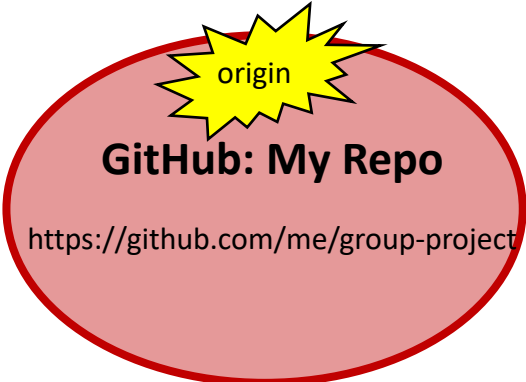
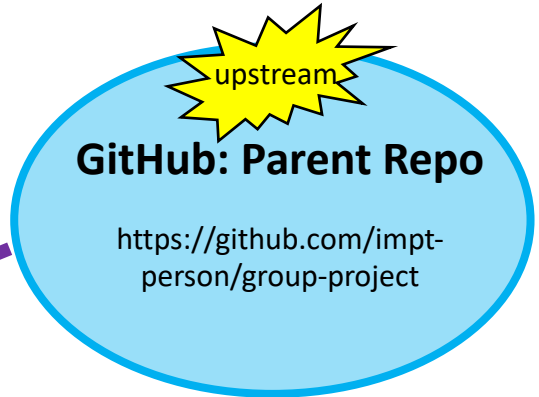
```
group-project$ git push
```

Local computer

The cloud



Get updates from parent repo



1 Link the parent repo to your local project and check that it connected (Only need to do once!)

```
group-project$ git remote add upstream https://github.com/impt-person/group-project.git group-project$ git remote -v
```

2 Pull from parent repo to local project

```
group-project$ git pull upstream master
```

(This doesn't update My Repo though!)

3 Push to origin (My Repo by default!)

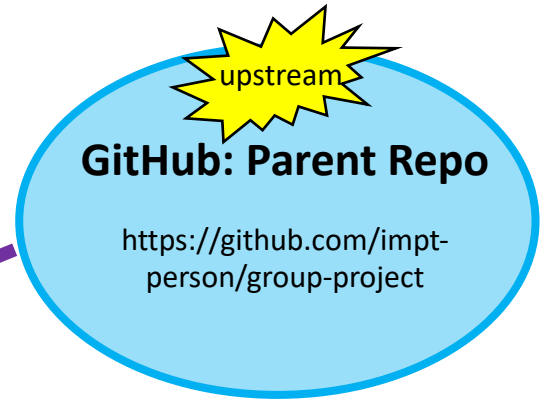
```
group-project$ git push
```

Parent repo, local project, and my repo are now all updated.

Local computer

The cloud

Get updates from parent repo



1 Link the parent repo to your local project and check that it connected

```
group-project$ git remote add upstream https://github.com/impt-person/group-project.git  
group-project$ git remote -v
```

(Only need to do once!)

2 Pull from parent repo to local project

```
group-project$ git pull upstream master
```

(This doesn't update My Repo though!)

3 Push to origin (My Repo by default!)

```
group-project$ git push
```

Parent repo, local project, and my repo are now all updated.

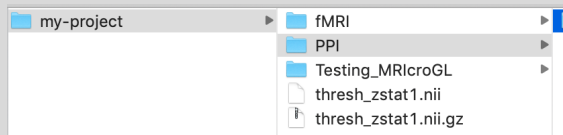
Pull from parent repo & push to origin (steps 2-3) at least once a day to stay updated!

Local computer

The cloud

How do I get my folder on my computer onto GitHub?
(How do I go the other way
and go FROM my computer TO GitHub?)

(Directory to be git-ified)



My Laptop

~/Desktop/my-project

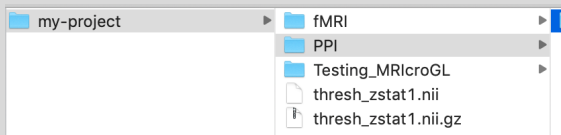
GitHub: My Account

FROM my computer TO GitHub

Local computer

The cloud

(Directory to be git-ified)



My Laptop

~/Desktop/my-project

1

On GitHub, click the plus button to add an empty new repository, name it, and note the URL.

GitHub: My Repo

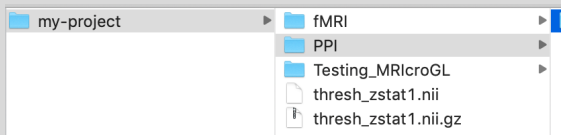
<https://github.com/me/repo-project>

FROM my computer TO GitHub

Local computer

The cloud

(Directory to be git-ified)



My Laptop

~/Desktop/repo-project

My Laptop

~/Desktop/my-project

2 Clone repo to desktop

```
~/Desktop$ git clone https://github.com/me/repo-project.git
```

1

On GitHub, click the plus button to add an empty new repository, name it, and note the URL.

GitHub: My Repo

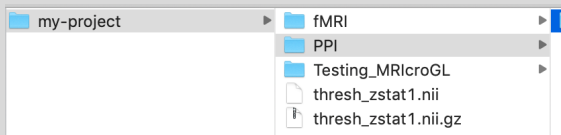
<https://github.com/me/repo-project>

FROM my computer TO GitHub

Local computer

The cloud

(Directory to be git-ified)



3 Drag-and-drop (or `$ mv`) contents of my-project into repo-project. Repo-project is already initialized and already has the origin linked, since it was cloned.

1 On GitHub, click the plus button to add an empty new repository, name it, and note the URL.



2 Clone repo to desktop

```
~/Desktop$ git clone https://github.com/me/repo-project.git
```

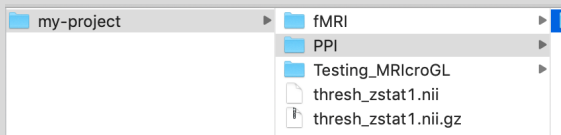


FROM my computer TO GitHub

Local computer

The cloud

(Directory to be git-ified)

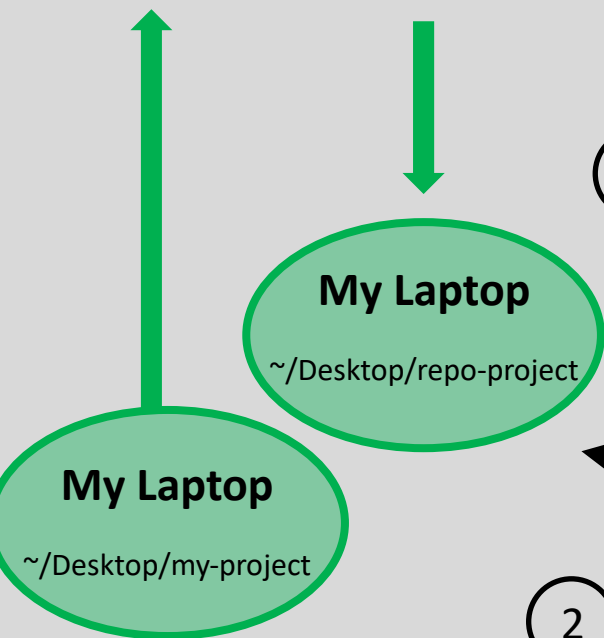


3 Drag-and-drop (or `$ mv`) contents of my-project into repo-project. Repo-project is already initialized and already has the origin linked, since it was cloned.

4 Add and commit the contents.

```
~/Desktop/repo-project$ git add .  
~/Desktop/repo-project$ git commit -m "First commit"
```

1 On GitHub, click the plus button to add an empty new repository, name it, and note the URL.



2 Clone repo to desktop

```
~/Desktop$ git clone https://github.com/me/repo-project.git
```

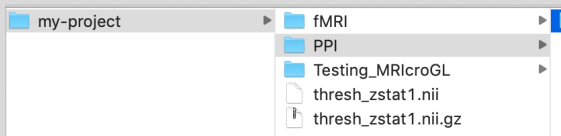


FROM my computer TO GitHub

Local computer

The cloud

(Directory to be git-ified)



3 Drag-and-drop (or `$ mv`) contents of my-project into repo-project. Repo-project is already initialized and already has the origin linked, since it was cloned.

4 Add and commit the contents.

```
~/Desktop/repo-project$ git add .  
~/Desktop/repo-project$ git commit -m "First commit"
```

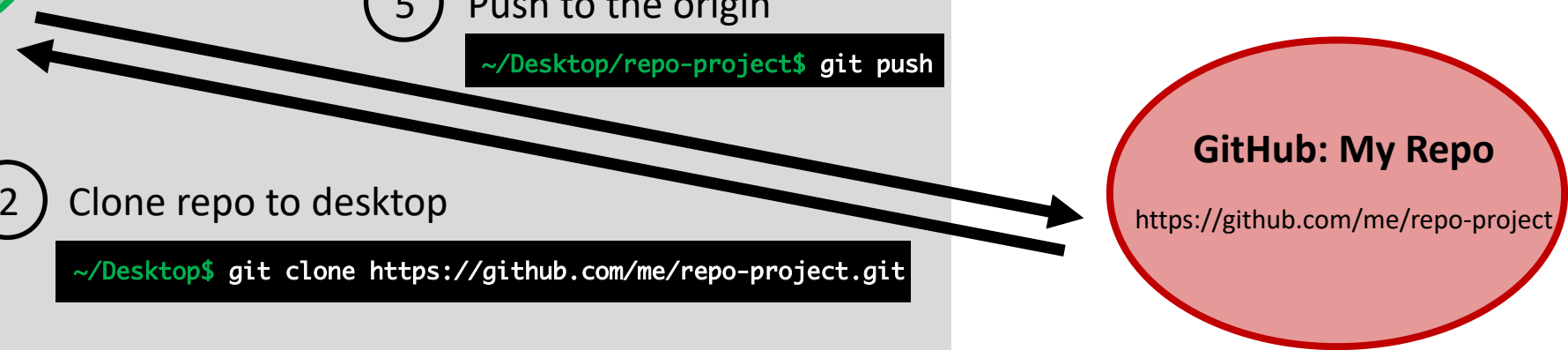
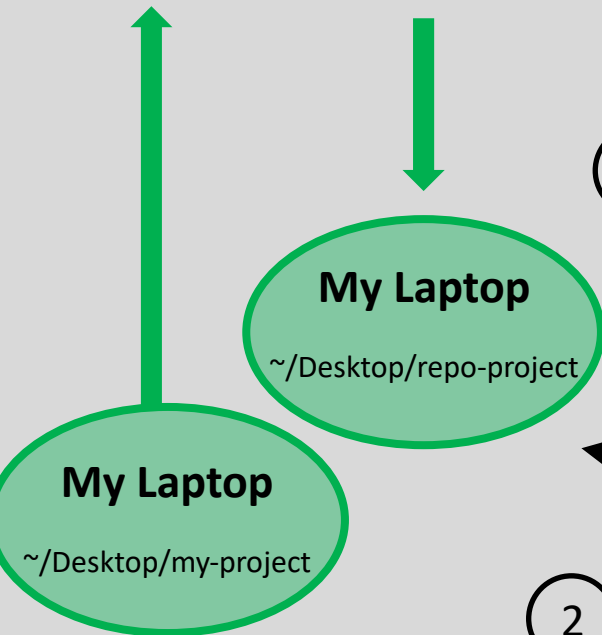
5 Push to the origin

```
~/Desktop/repo-project$ git push
```

2 Clone repo to desktop

```
~/Desktop$ git clone https://github.com/me/repo-project.git
```

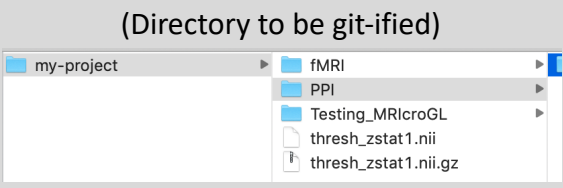
1 On GitHub, click the plus button to add an empty new repository, name it, and note the URL.



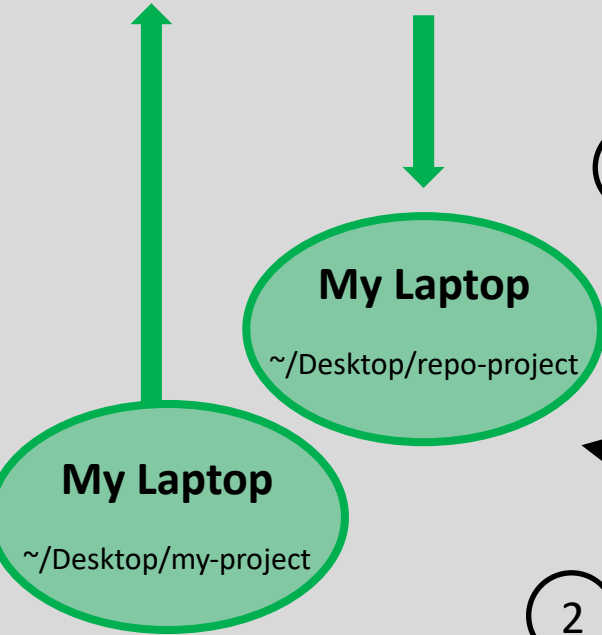
FROM my computer TO GitHub

Local computer

The cloud



(Directory to be git-ified)



You can delete the empty "my-project" locally. Work more locally (in repo-project) and save file, then repeat steps 4-5. Rinse & repeat.

3 Drag-and-drop (or `$ mv`) contents of my-project into repo-project. Repo-project is already initialized and already has the origin linked, since it was cloned.

4 Add and commit the contents.

```
~/Desktop/repo-project$ git add .
~/Desktop/repo-project$ git commit -m "First commit"
```

5 Push to the origin

```
~/Desktop/repo-project$ git push
```

2 Clone repo to desktop

```
~/Desktop$ git clone https://github.com/me/repo-project.git
```

1 On GitHub, click the plus button to add an empty new repository, name it, and note the URL.



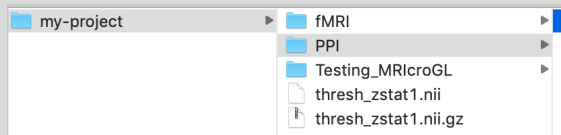
FROM my computer TO GitHub

Local computer

The cloud

How do I get my folder on my computer onto GitHub?
(How do I go the other way
and go FROM my computer TO GitHub?)
... another way.

(Directory to be git-ified)

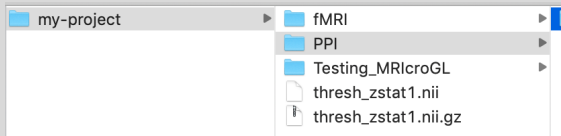


FROM my computer TO GitHub

Local computer

The cloud

(Directory to be git-ified)



1

On GitHub, click the plus button to add an empty new repository, name it, and note the URL.

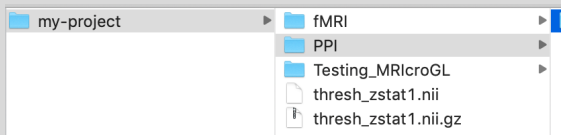


FROM my computer TO GitHub

Local computer

The cloud

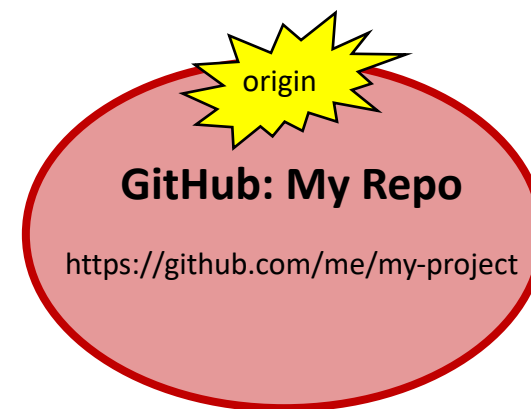
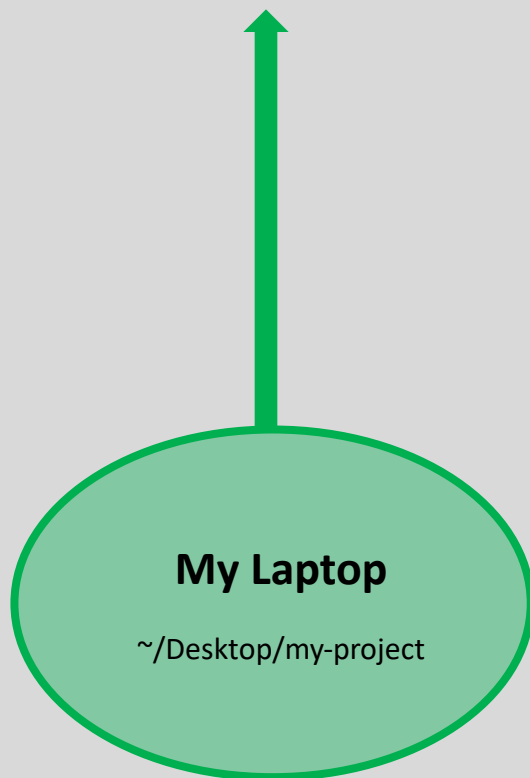
(Directory to be git-ified)



- 2 Initialize the directory, add it, and commit it. Then add the origin with the URL. Check that it's been added. (Only need to init, add origin, and verify origin once!)

```
~/Desktop$ cd my-project
~/Desktop/my-project$ git init
~/Desktop/my-project$ git add .
~/Desktop/my-project$ git commit -m "First commit"
~/Desktop/my-project$ git remote add origin
https://github.com/me/my-project.git
~/Desktop/my-project$ git remote -v
```

- 1 On GitHub, click the plus button to add an empty new repository, name it, and note the URL.

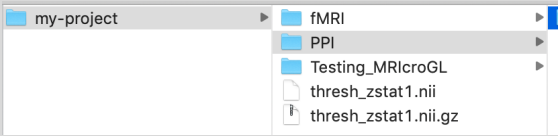


FROM my computer TO GitHub

Local computer

The cloud

(Directory to be git-ified)



- ② Initialize the directory, add it, and commit it. Then add the origin with the URL. Check that it's been added. (Only need to init, add origin, and verify origin once!)

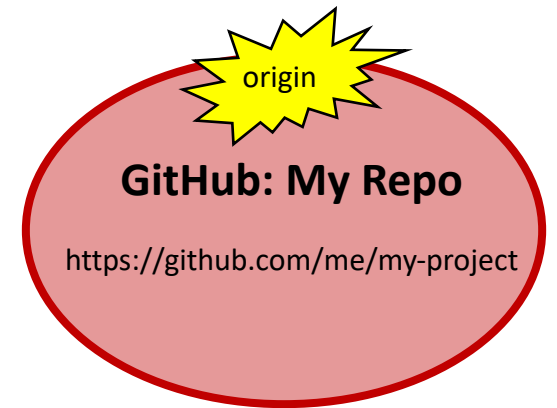
```
~/Desktop$ cd my-project
~/Desktop/my-project$ git init
~/Desktop/my-project$ git add .
~/Desktop/my-project$ git commit -m "First commit"
~/Desktop/my-project$ git remote add origin
https://github.com/me/my-project.git
~/Desktop/my-project$ git remote -v
```

- ① On GitHub, click the plus button to add an empty new repository, name it, and note the URL.



- ③ Push to the origin

```
~/Desktop/my-project$ git push origin master
```

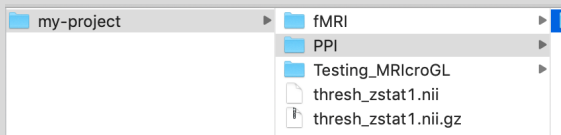


FROM my computer TO GitHub

Local computer

The cloud

(Directory to be git-ified)



- 2 Initialize the directory, add it, and commit it. Then add the origin with the URL. Check that it's been added. (Only need to init, add origin, and verify origin once!)

```
~/Desktop$ cd my-project
~/Desktop/my-project$ git init
~/Desktop/my-project$ git add .
~/Desktop/my-project$ git commit -m "First commit"
~/Desktop/my-project$ git remote add origin
https://github.com/me/my-project.git
~/Desktop/my-project$ git remote -v
```

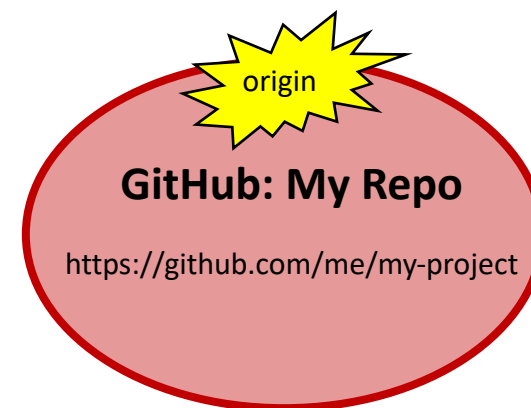
- 1 On GitHub, click the plus button to add an empty new repository, name it, and note the URL.



Work more locally and save file, then repeat yellow lines only in step 2 to add and commit, then repeat yellow part of step 3. Rinse & repeat.

- 3 Push to the origin

```
~/Desktop/my-project$ git push origin master
```



FROM my computer TO GitHub

Local computer

The cloud