

While you're waiting:

1. Download Visual Studio Code

- if you haven't already and don't have a text editor you're comfortable with
- all my examples going forward -- including how to compile Sass on your local machine, how to update your projects on GitHub -- will use Visual Studio Code, so I *highly* recommend using this editor for the class



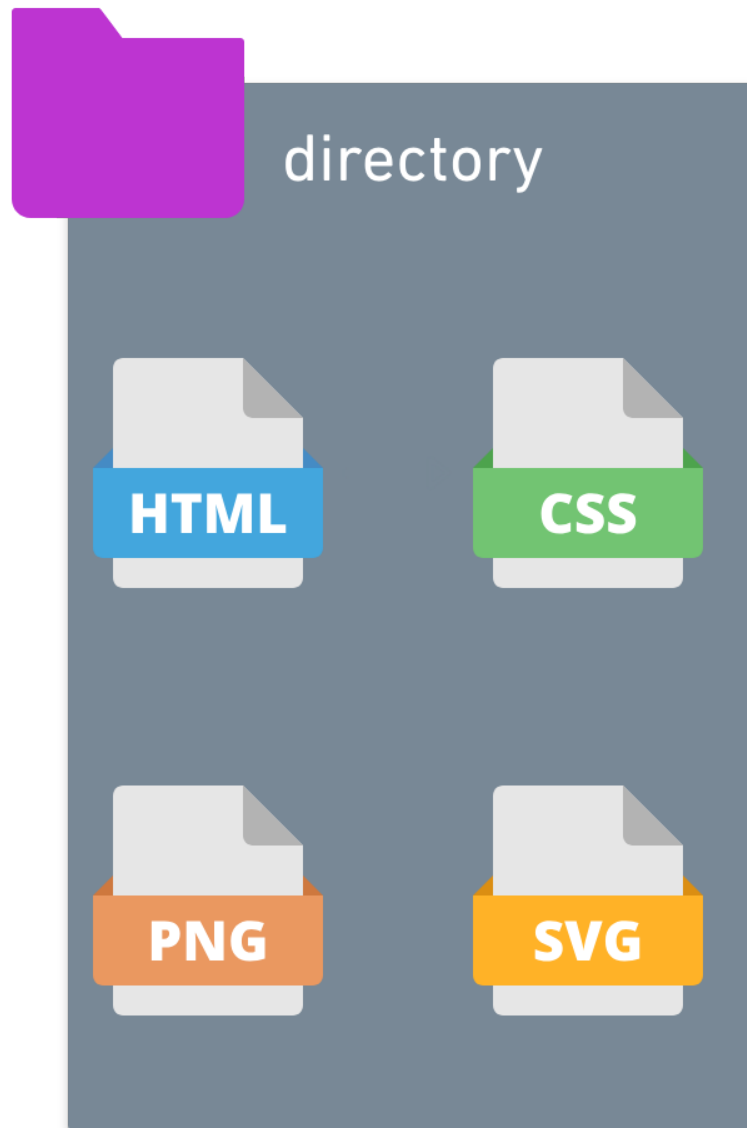
Agenda

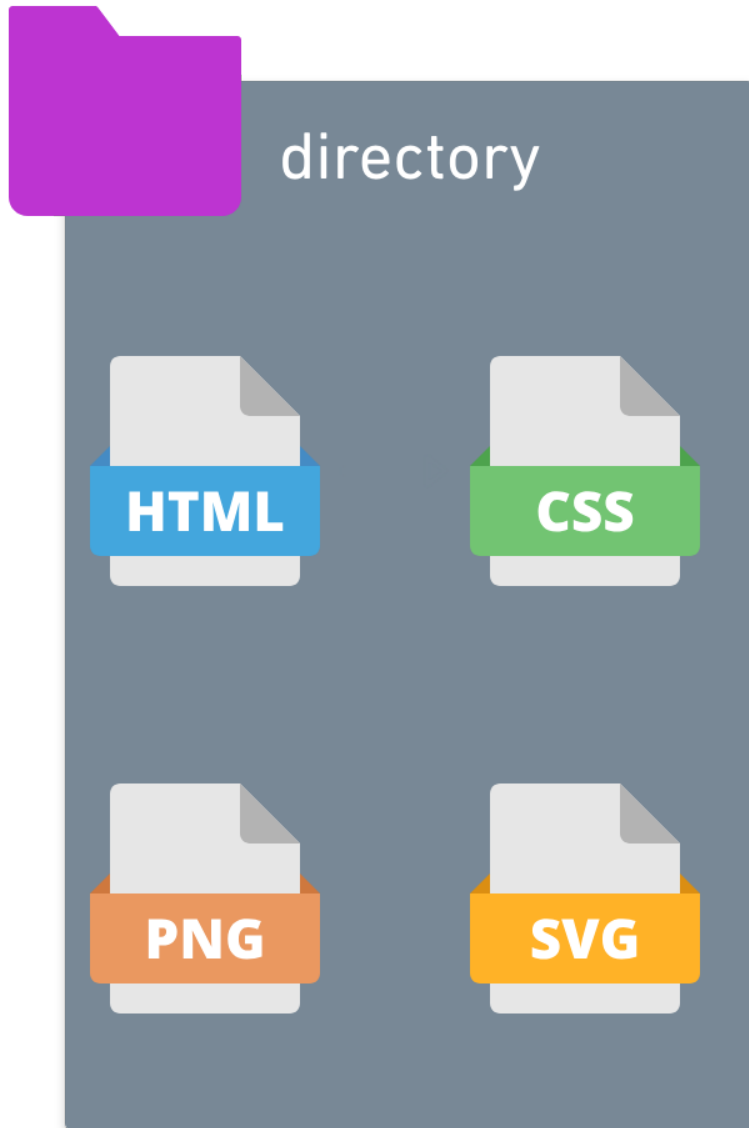
1. Understanding git
2. Getting started with git + GitHub
3. Getting started with VS Code
4. VS Code + git + GitHub

Understanding git



directory



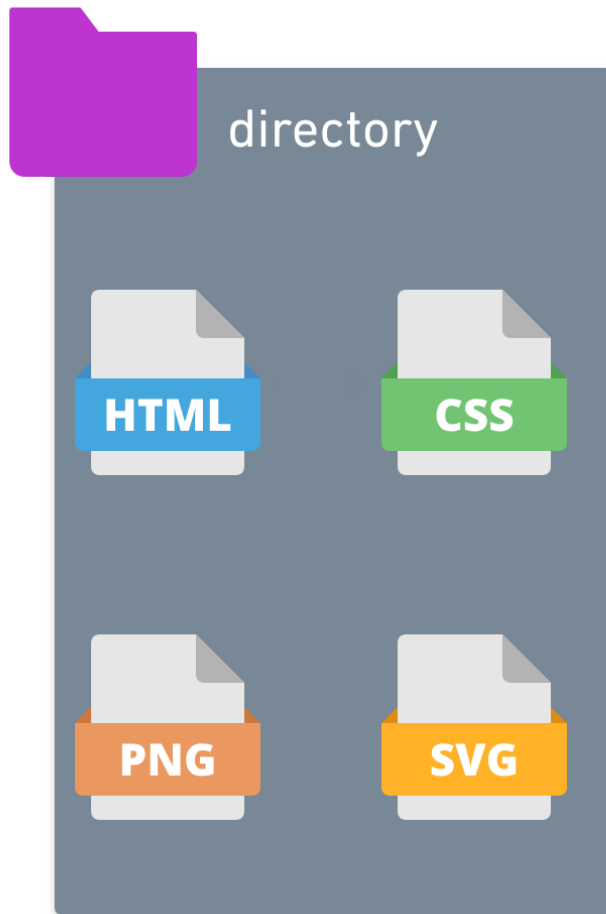


index.html

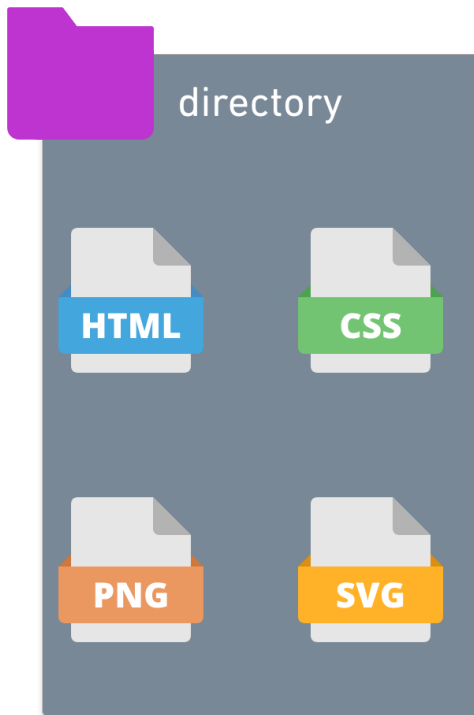


styles.css

Without version control



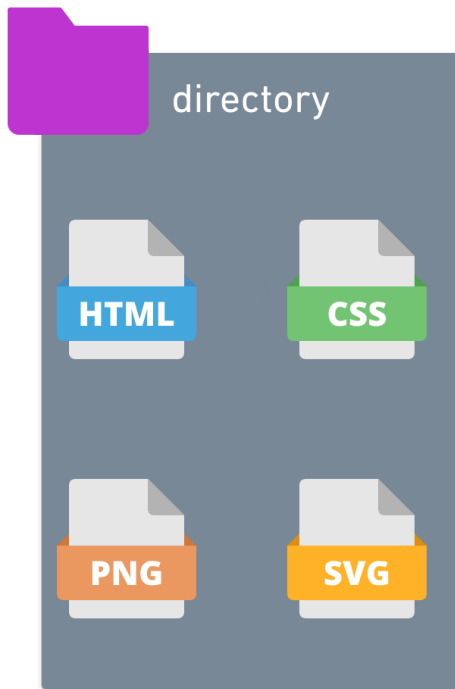
Without version control



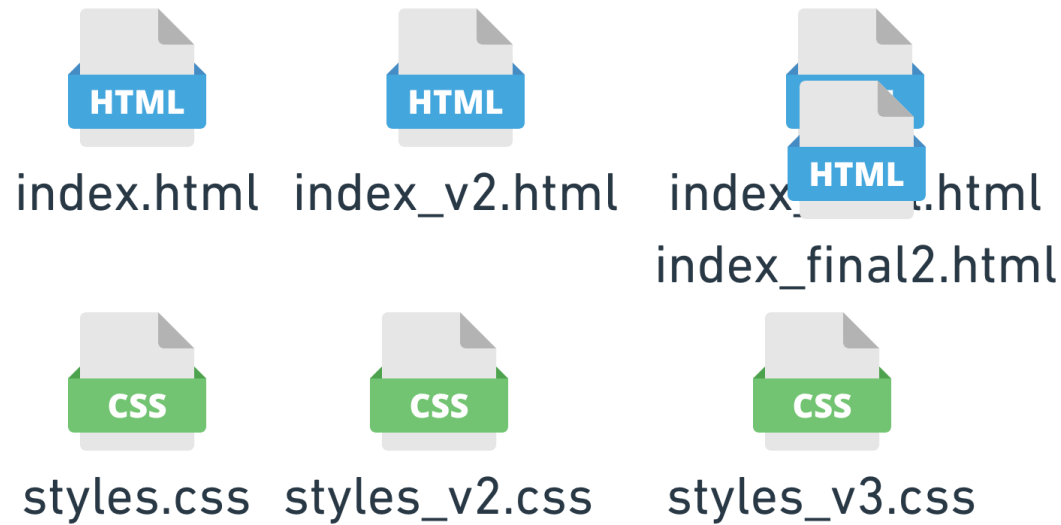
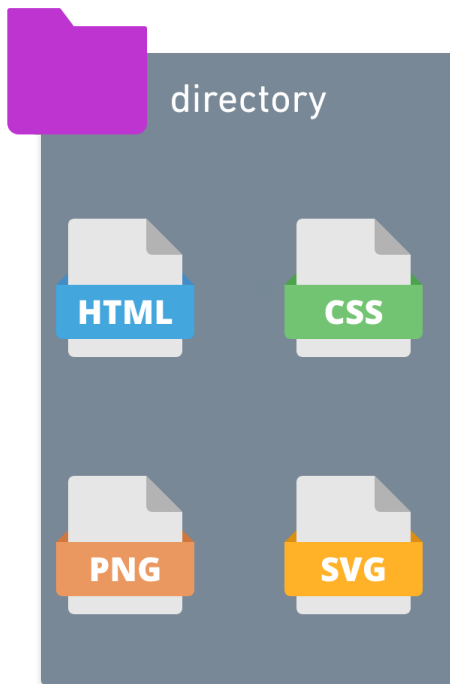
index.html index_v2.html

styles.css styles_v2.css styles_v3.css

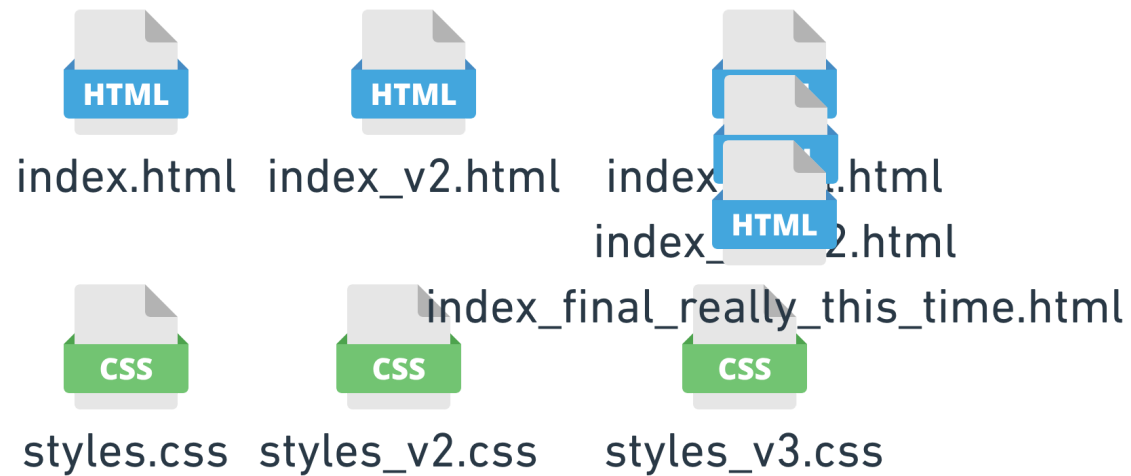
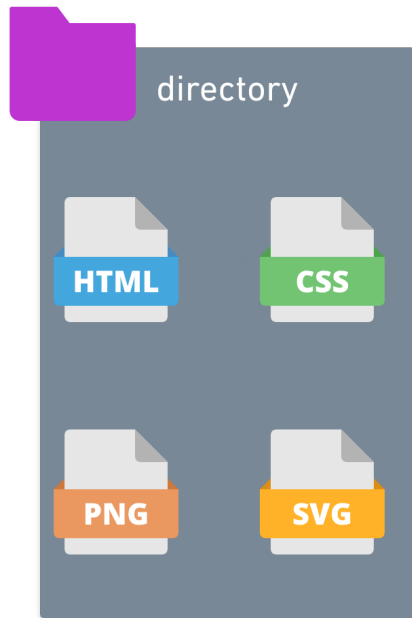
Without version control



Without version control



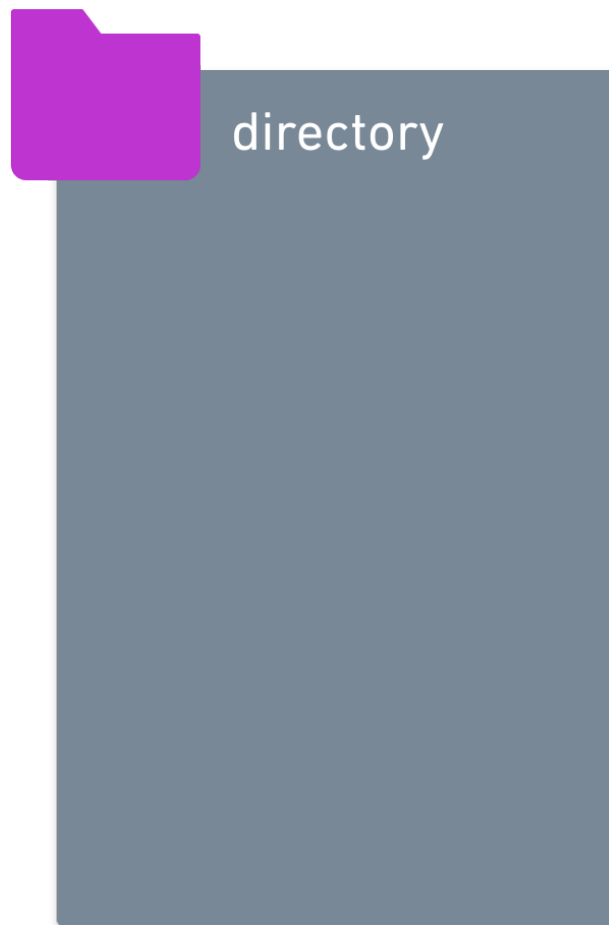
Without version control





directory

`git init`



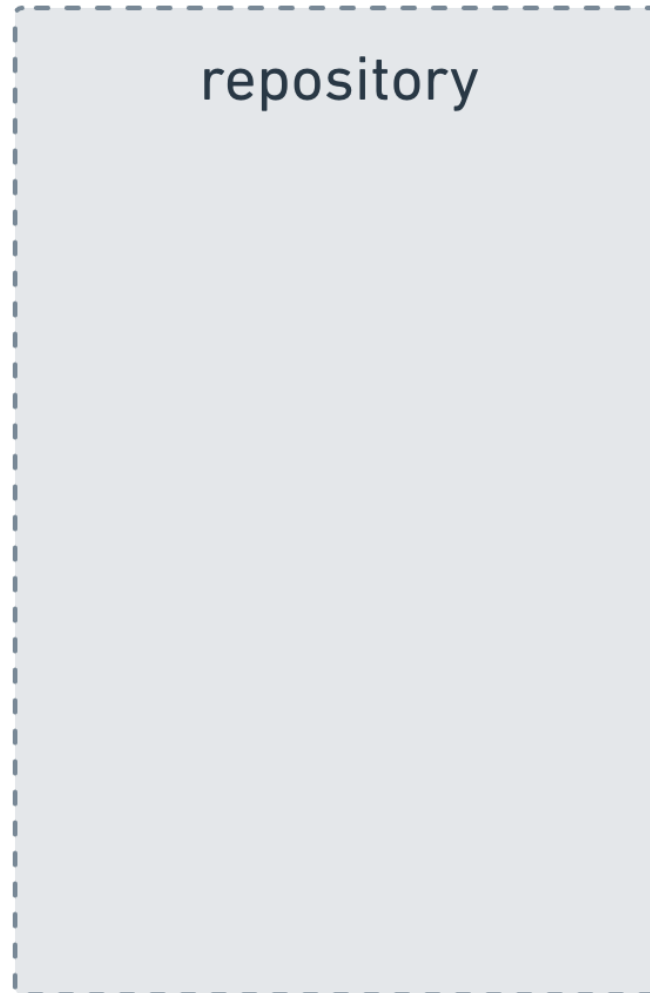
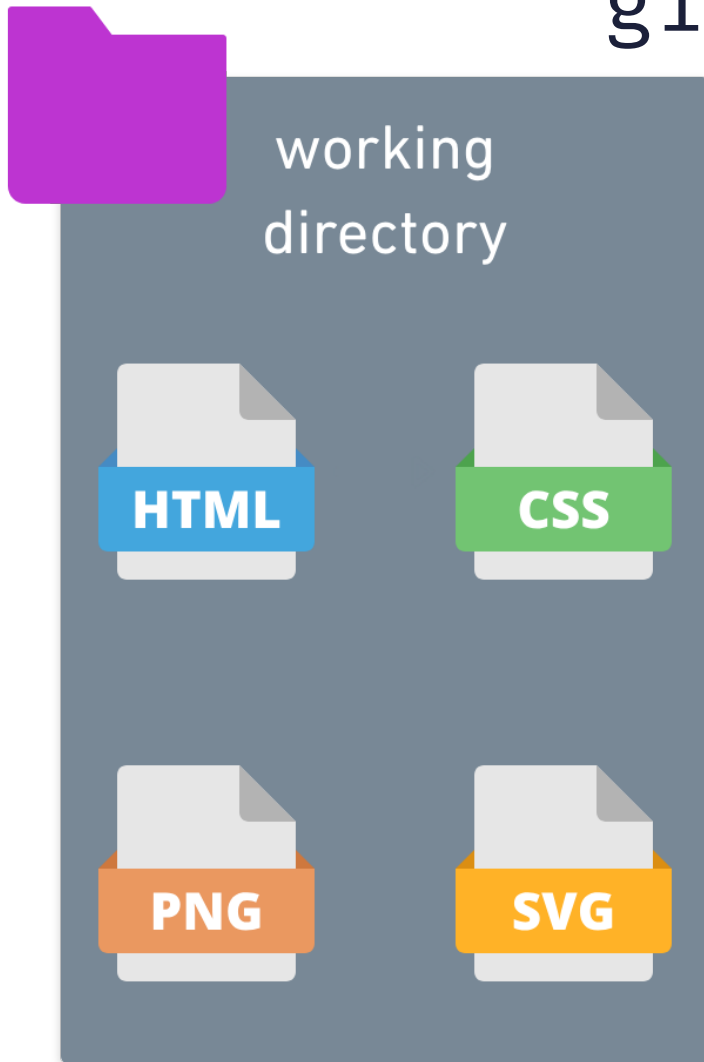
`git init`



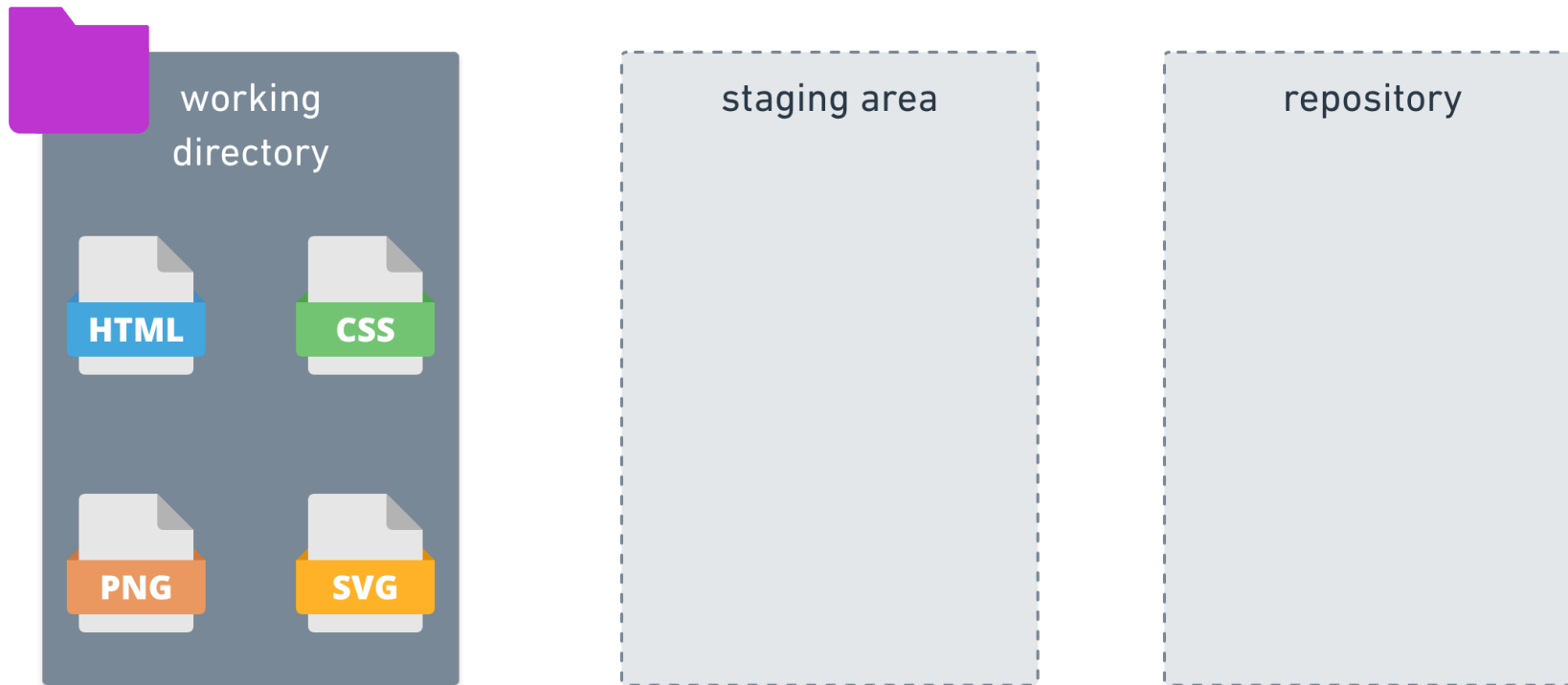
working
directory

repository

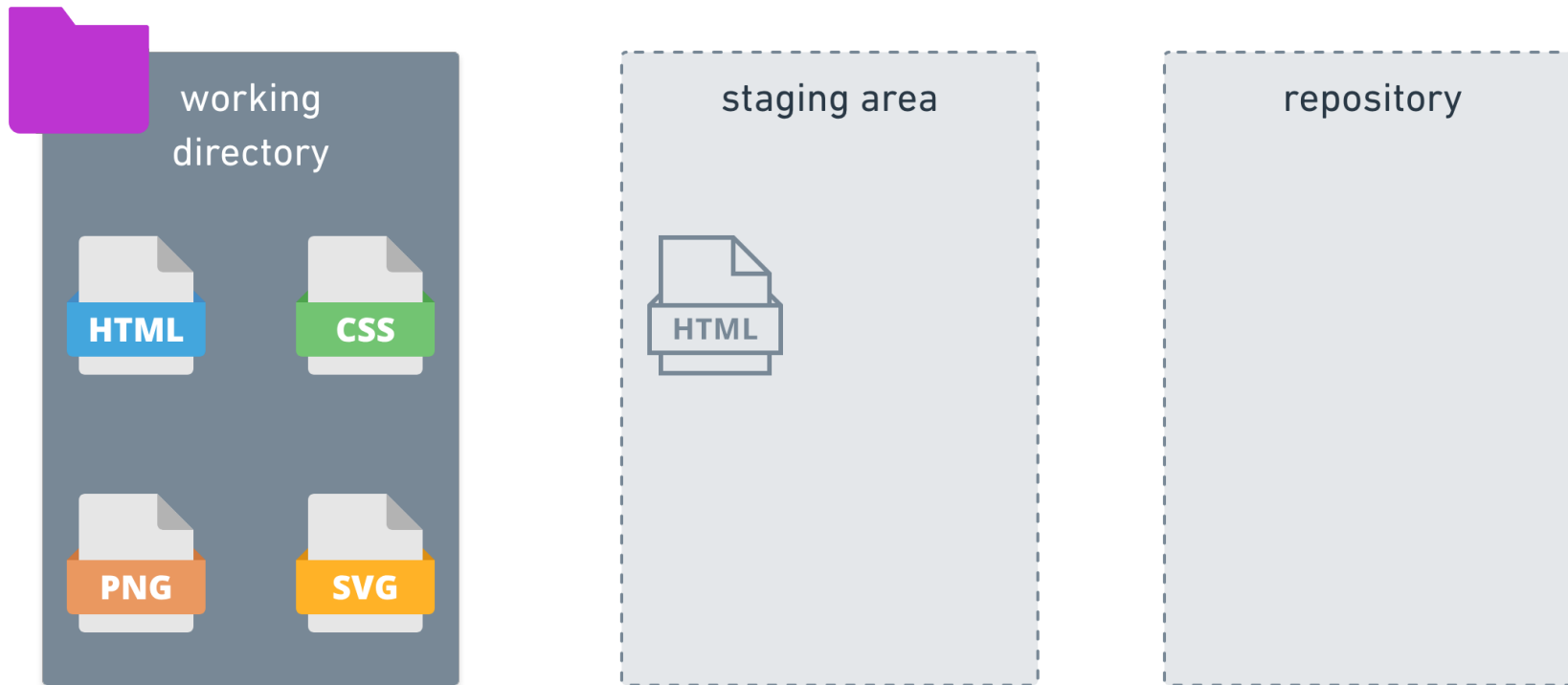
`git init`



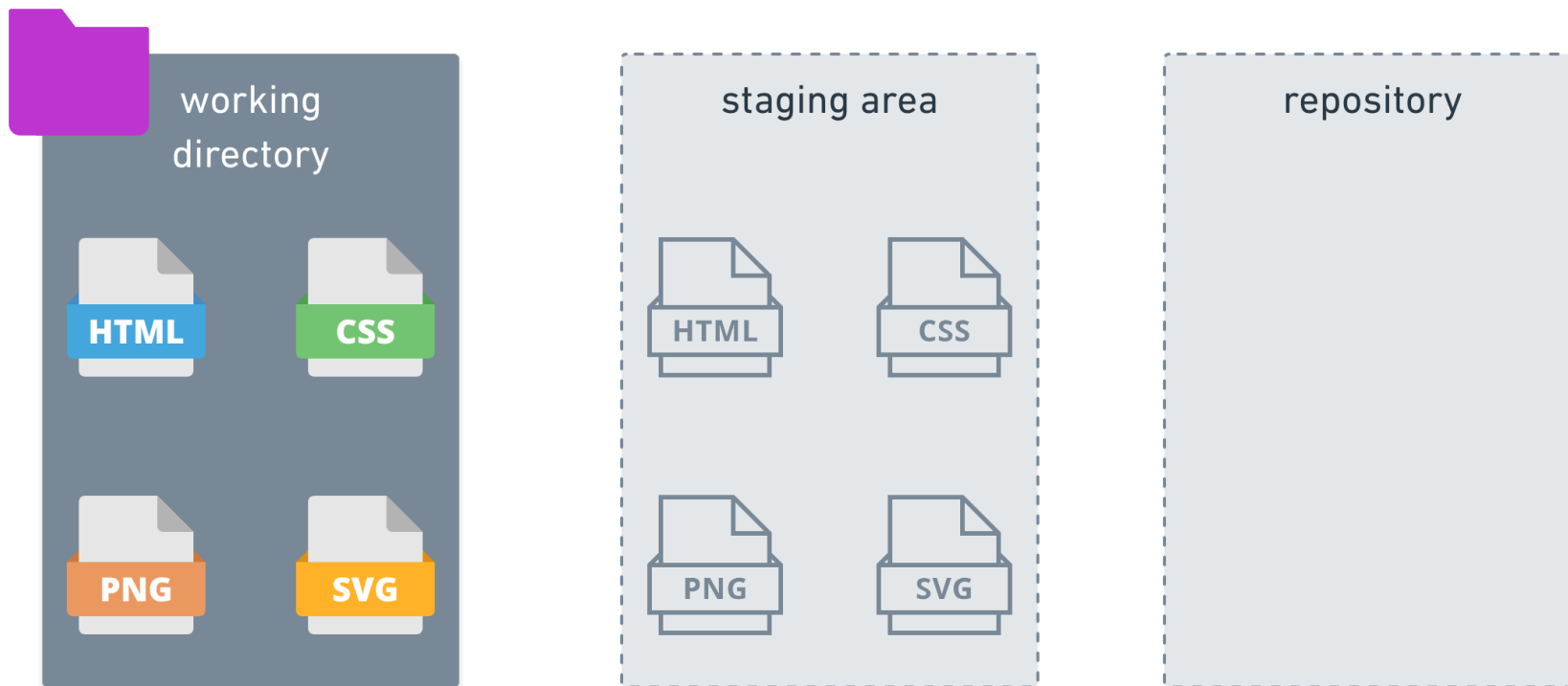
git add



git add



git add



git commit



Q&A on git workflow or concepts

Git ! = *GitHub*

Git ! = *GitHub*



Git != GitHub



Getting started with git + GitHub

- Install/verify git
- Set up your username and email
- Get/verify your SSH keys

<http://bit.ly/431-git>

Getting started with VS Code

- Interface basics
- Update some helpful settings of VS Code
- Add some useful extensions to VS Code

Extensions

1. **Beautify** by HookyQR
2. **Live Server** by Ritwick Dey
3. **Live Sass Compiler** by Ritwick Dey
4. **Intellisense for CSS class names in HTML** by Zignd
5. **Path Intellisense** by Christian Kohler
6. **scss-scan** by aaronphy
7. **minify** by HookyQR
8. **Auto Rename Tag** by Jun Han
9. **Bracket Pair Colorizer** by CoenraadS

Extensions

1. **Beautify** by HookyQR
2. **Live Server** by Ritwick Dey
3. **Live Sass Compiler** by Ritwick Dey
4. **Intellisense for CSS class names in HTML** by Zignd
5. **Path Intellisense** by Christian Kohler
6. **scss-scan** by aaronphy
7. **minify** by HookyQR
8. **Auto Rename Tag** by Jun Han
9. **Bracket Pair Colorizer** by CoenraadS

- What does this extension do?
- What problem does it solve?
- How is it helpful?

VS Code Q&A

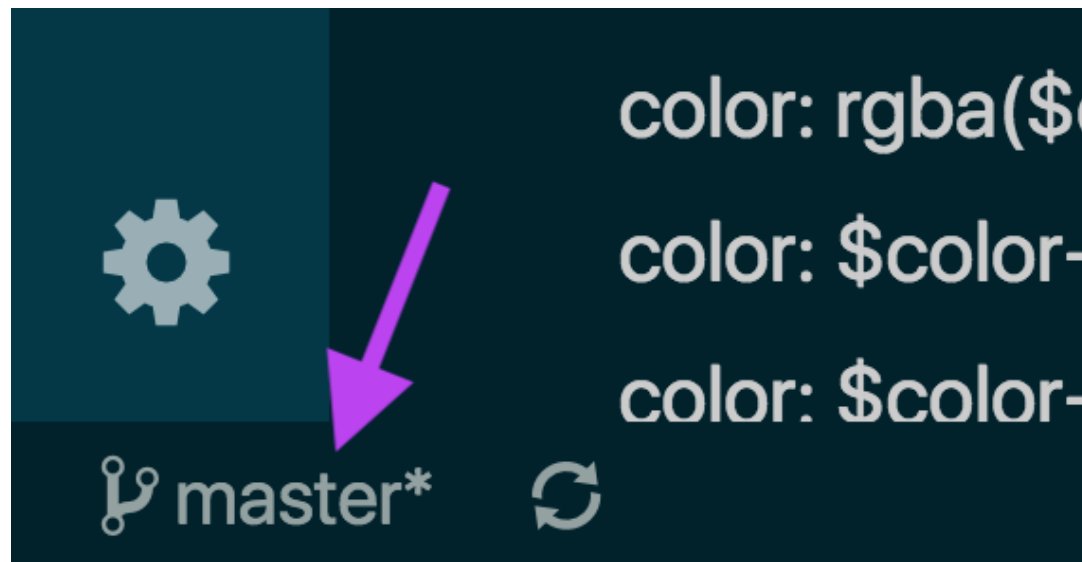
VS Code + git

Git branches

`git branch`

Git branches

git branch



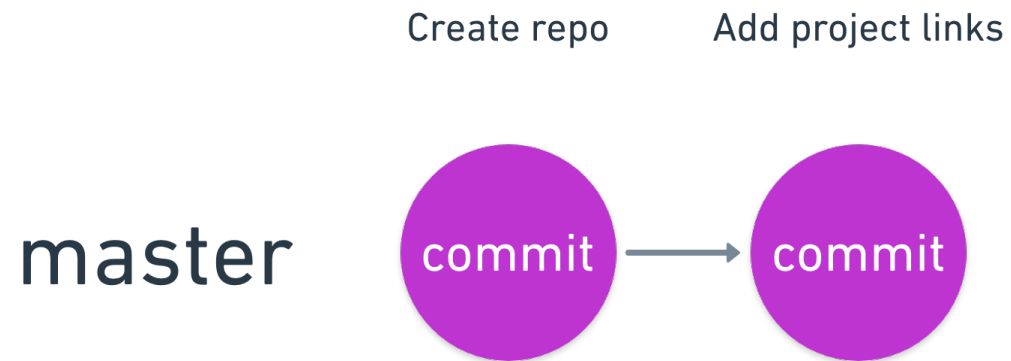
Git branches

Create repo

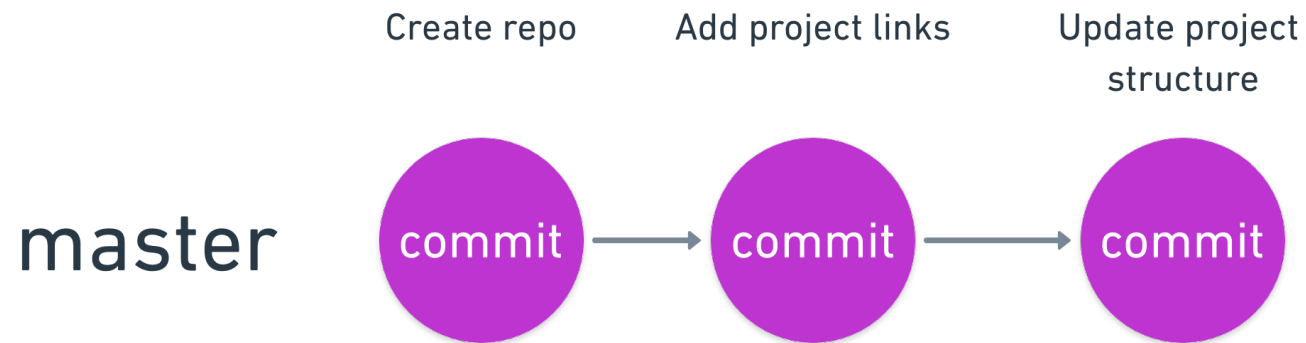
master



Git branches



Git branches



Git branches

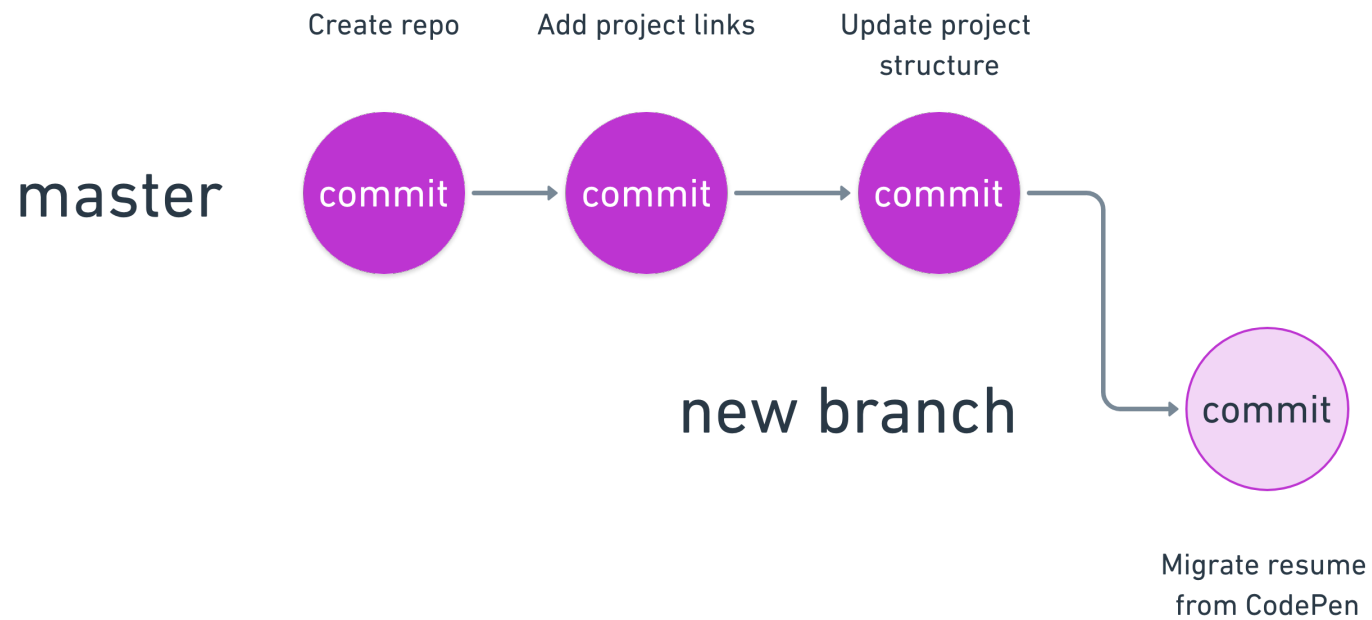
Git branches

```
git branch {new-branch-name}
```

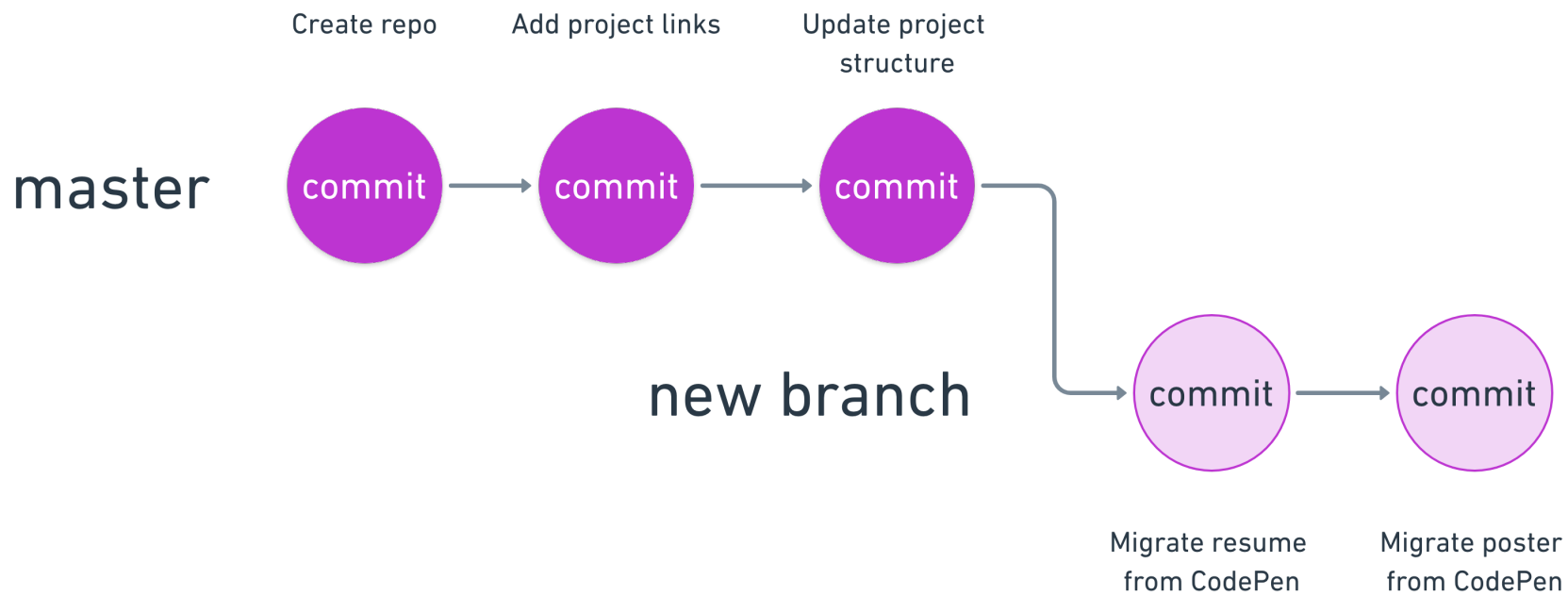
Git branches

```
git branch {new-branch-name}  
git checkout {new-branch-name}
```

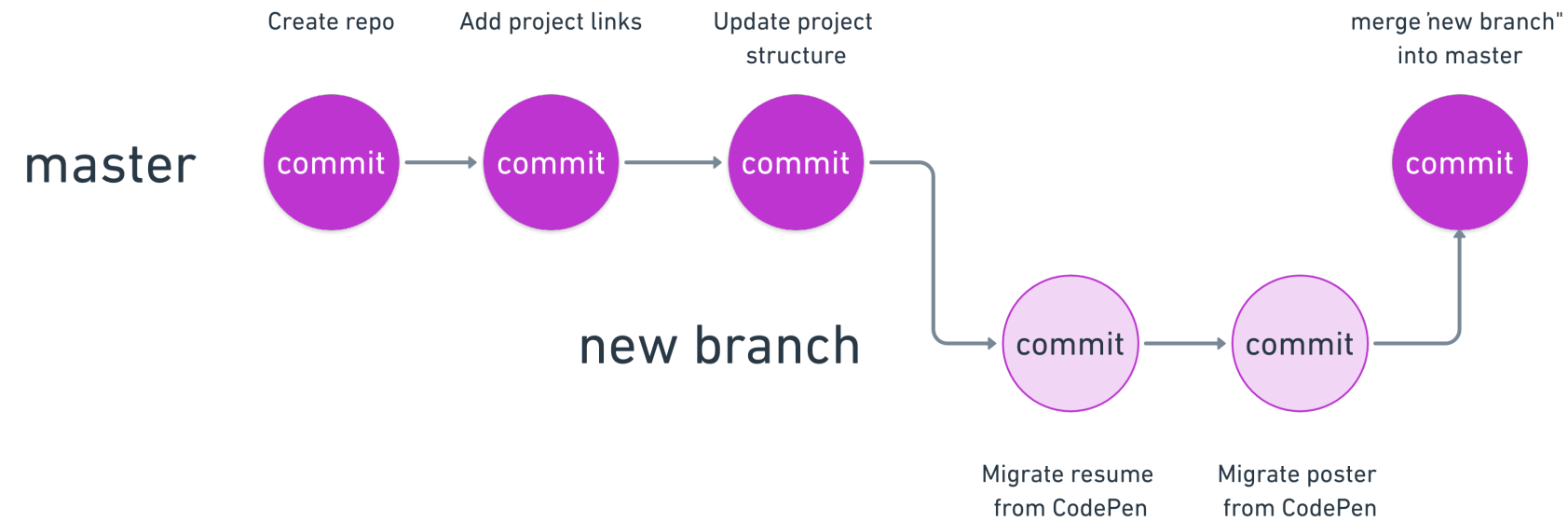
Git branches



Git branches



Git branches



Migrating from CodePen to VS Code

Migrating from CodePen to VS Code

```
git branch migrate-from-codepen
```

```
git checkout migrate-from-codepen
```

Migrating from CodePen to VS Code

```
git branch migrate-from-codepen
```

```
git checkout migrate-from-codepen
```

1. Copy over most recent version of resume
 - resume/index.html
 - resume/styles/styles.scss

Migrating from CodePen to VS Code

- Copy over most recent version of resume
 - `resume/index.html`
 - `resume/styles/styles.scss`

1. "Go Live" to view in browser
2. "Watch Sass" to compile Sass

Migrating from CodePen to VS Code

- Copy over most recent version of resume
 - `resume/index.html`
 - `resume/styles/styles.scss`
- "Go Live" to view in browser
- "Watch Sass" to compile Sass

1. Stage your changes
2. Make a new commit

Migrating from CodePen to VS Code

- Copy over most recent version of resume
 - `resume/index.html`
 - `resume/styles/styles.scss`
- "Go Live" to view in browser
- "Watch Sass" to compile Sass
- Stage your changes
- Make a new commit

1. Repeat the process for your poster project

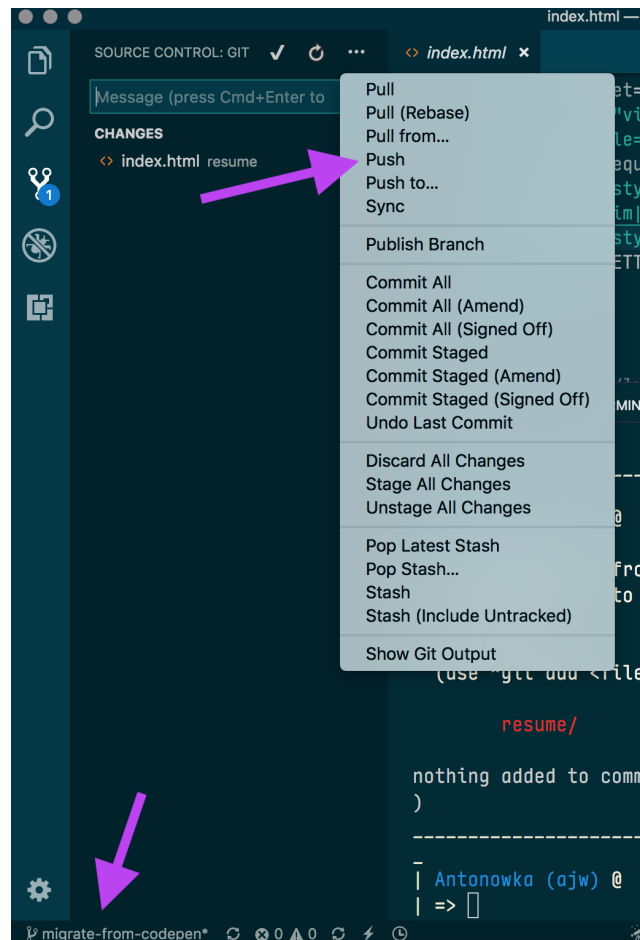
- `copy`
- `verify`
- `then commit`

Migrating from CodePen to VS Code

```
git push origin migrate-from-codepen
```

Migrating from CodePen to VS Code

`git push origin migrate-from-codepen`



Q&A for branches & workflow

Best practices for commits

A commit represents work that is

- Modular
- Connected

A commit represents work that is

- Modular
 - Connected
1. Add web fonts
 2. Add JS for flyout menu
 3. Make header responsive
 4. Fix typo

Tasks as commits

Tasks as commits

1. Add web fonts
2. Add JS for flyout menu
3. Make header responsive
4. Fix typo

Best practices for commit messages

- Be useful
- Be brief (~50 char)

Best practices for commit messages

- Be useful
- Be brief (~50 char)
- Start with a verb in the present tense, like giving a command:
 - Update body font to Roboto
 - Fix typo in nav
 - Note tense: not updates or fixed

Best practices for commit messages

- Be useful
- Be brief (~50 char)
- Start with a verb in the present tense, like giving a command:
 - Update body font to Roboto
 - Fix typo in nav
 - Note tense: not updates or fixed
- Doesn't need to mention which file you worked on

Commits Q&A