Course Title: GitHub Quick Tips

Description: GitHub is the heart of many development workflows. These 30 tips and tricks help you exercise the full power of this versatile software development platform. LinkedIn staff instructor Ray Villalobos shows how to customize GitHub by building out your project homepages, turning repositories into reusable templates, adding collaborators, and enhancing security with secret keys and branch protection. Then discover how to be more efficient in the interface, from searching to creating permalinks for frequently accessed files. Next, dive into power tips for repos, including deleting and restoring branches. Finally, Ray reviews the specialized tools such as wikis, metrics, the GitHub CLI, and custom code snippets called gists. Each tip makes you a more efficient developer and GitHub contributor.


```
**********************************************
Chapter: 2. The Github Interface
**********************************************
```

```
------------------------------------------------
Video: Multi-line highlights
------------------------------------------------
```
Note Time: Note Text:

0:00:30   Click on line number to highlight that line, then hold shift and click on a line further down to highlight that block of text. Click on the three lots on upper left corner of the line to copy those lines, get a permalink = permanent link to the selected code, or reference that code in an issue


```
------------------------------------------------
Video: Access the hidden help shortcuts
------------------------------------------------
```
Note Time: Note Text:

0:01:11   On keyboard: SHIFT ? to get list of keyboard shortcuts for Github page you're on (ex. on your profile). Use the T key in a particular project to get a list of files in that project, for example


```
------------------------------------------------
Video: Compare two branches
------------------------------------------------
```
Note Time: Note Text:

0:01:12   To compare two branches in a project: to the URL

add /compare/NAME_OF_BRANCH1..NAME_OF_BRANCH2


```
-----------------------------------------------
Video: Evergreen permalinks
-----------------------------------------------
Note Time:          Note Text:
```

0:00:00   Y key on keyboard when you're on any page in Github
to create permanent link to a file when you want to point someone to a
document but ensure new commits don't change the content of the file.
This link won't change- will always take you to this version of that
file


```
************************************************
Chapter: 3. Working with Repos
************************************************
```


```
-----------------------------------------------
Video: Create empty folders
-----------------------------------------------
Note Time:          Note Text:
```

0:00:59   To commit an empty folder, then add a .gitignore
file and type placeholder text

0:00:59   To create folder in Github cite: once in
repository, CREATE FILE-> type NAME_OF_FOLDER/


```
-----------------------------------------------
Video: Organize utility templates
-----------------------------------------------
Note Time:          Note Text:
```

0:01:23   Make .github folder and add templates for issues,
pull requests, or a contributing.md file to tell users how they can
contribute to a project


```
-----------------------------------------------
Video: GitHub Flavored Markdown
-----------------------------------------------
Note Time:          Note Text:
```

0:00:53   Fence in block of code with ``` above and below
text. Next to the top ```, can write javascript to specify type of
code in the block and highlight

0:01:11   If the item in task list is checked: – [x]
name_of_task

0:01:11   Task list: – [ ] name_of_task

0:01:56   GitHub.com/gfm to see other documentations (ex.
tables)


```
************************************************
Chapter: 4. Github Tools
************************************************
```

```
------------------------------------------------
Video: GitHub Pages
------------------------------------------------
```
Note Time: Note Text:

0:01:56   Use a gh–pages branch for GitHub pages; commits
won't change the page unless you push them yourself onto the branch


```
------------------------------------------------
Video: Create a wiki
------------------------------------------------
```
Note Time: Note Text:

0:01:05   Wikis useful to create code samples tied to that
repository


```
------------------------------------------------
Video: Contribute to an open–source project
------------------------------------------------
```
Note Time: Note Text:

0:00:28   Find ways to contribute to an open–source project:
after the Github URL (ex. GitHub.com/twbs/bootstrap) add /contribute

0:01:45   Another way to look for ways to contribute, look at
issues. Filter issues with label:"help wanted"


```
------------------------------------------------
Video: Create a project kanban for your repo
------------------------------------------------
```
Note Time: Note Text:

0:00:50   Create projects (recommended with Automated kanban template) to organize a repo. Can tie cards to issues


```
*************************************************
```
Chapter: 5. External Tools
```
*************************************************
```


```
-------------------------------------------------
```
Video: What can I do with the GitHub CLI
```
-------------------------------------------------
```
Note Time: Note Text:

0:01:31   Install GitHub CLI to add gh command to your terminal and have further access to your project (show issues, pull requests, etc). To show issue list: type "gh issue list —R name_of_project"


```
-------------------------------------------------
```
Video: Start a new project with npx and degit
```
-------------------------------------------------
```
Note Time: Note Text:

0:01:15   To start new project (clone from yourself or another user's project that is not set up as a template): in terminal, make sure you're in your desktop. Type "npx degit name_of_user/ name_of_project new_name_of_project". Made a copy of this repository on your desktop, but without downloading the project history (better/ faster than cloning a project)


```
-------------------------------------------------
```
Video: Share code with gists
```
-------------------------------------------------
```
Note Time: Note Text:

0:01:15   Type gist.new in domain to create a gist (can be public or private) for sharing code. Extension of filename will tell Github how to treat the syntax color-coding. For example "sample.js" for javascript. Similar to repos, but has commenting feature. Meant for single files.