(/)

Push Your App to GitHub

Created by Alyson La, @realalysonla (https://www.twitter.com/realalysonla)

Things you need before you get started

Git & GitHub

- Check if Git is installed
    - In the terminal type `git --version` (1.8 or higher preferred)
- If not, download Git [here] (http://git-scm.com/downloads). Then, setup your local Git profile - In the terminal:
    - Type `git config --global user.name "your-name"`
    - Type `git config --global user.email "your-email"`

    - *To check if Git is already config-ed you can type* `git config --list`
- Create a free GitHub (https://github.com) account or login if you already have one

COACH: Talk a little about git, verison control, and open source

Push your app to GitHub using the command line

On your GitHub profile click "new repo"   give it a name (example: rails-girls), brief description, choose the "public" repo option, and click "create repository".

In the command line–make sure you `cd` into your railgirls folder–and type:

`git init`

This initializes a git repository in your project

Note: If you've already done the Heroku guide (/heroku), then you've already initialized a git repository & you can move on to the next steps.

Next check if a file called `README.rdoc` exists in your railsgirl directory:

```
dir README.rdoc
```

Choose your operating system: **Windows** | Other

If the file doesn't exist, create it by typing:

```
touch README.rdoc
```

Or if you use a Windows environment, type the following command instead:

```
type nul > README.rdoc
```

COACH: Talk a little about README.rdoc

Then type:

```
git status
```

This will list out all the files in your working directory.

COACH: Talk about some of your favorite git commands

Then type:

```
git add .
```

This adds in all of your files & changes so far to a staging area.

Then type:

```
git commit -m "first commit"
```

This commits all of your files, adding the message "first commit"

Next type:

```
git remote add origin https://github.com/username/rails-girls.git
```

Your GitHub Repository page will list the repository URL, so feel free to copy and paste from there, rather than typing it in manually. You can copy and paste the link from your GitHub repository page by clicking the clipboard icon next to the URL.

This creates a remote, or *connection*, named "origin" pointing at the GitHub repository you just created.

Then type:

```
git push -u origin master
```

This sends your commits in your "master" branch to GitHub

Congratulations your app is on GitHub! Go check it out by going to the same url you used above: https://github.com/username/rails-girls (without the .git part)

If you want to continue making changes and pushing them to GitHub you'll just need to use the following three commands:

```
git add .
```

```
git commit -m "type your commit message here"
```

```
git push origin master
```

What's next?

Be a Part of the Open Source Community

- Follow your fellow Rails Girls & coaches on GitHub
- Star or watch their projects
- Fork (https://help.github.com/articles/fork-a-repo) a repo, then clone and push changes to your fork. Share the changes with the originator by sending them a pull request (https://help.github.com/articles/using-pull-requests)!
- Create an issue on a project when you find a bug
- Explore other open source projects - search by programming languange or key word

Learn more Git

- Check out trygit.org (http://try.github.io/)
- Use a Git Cheatsheet (https://na1.salesforce.com/help/doc /en/salesforce_git_developer_cheatsheet.pdf)
- Look up Git Commands at git-scm.org (http://git-scm.com/)

Rails Girls (http://railsgirls.com) · Crafted with excitement from Helsinki, Finland · Follow us on Twitter (http://twitter.com/railsgirls) and Facebook (https://www.facebook.com/railsgirls)