

Commit and Push the code to GitHub

When you are done with all the changes in the project go through the below instructions to learn how to commit and push to GitHub.

Task 1 - Review changes to the repo

To review the changes that have been made to the repo, run the following commands in the terminal:

```
cd [your repo name]
git status
```

The `git status` command will display all the changes you have made to this point, similar to the following image:

The changed files will be displayed. Choose the files which you want to push to the repository.

Task 2 - Mark the changes for commit

You now need to commit the changes you've made. Before you can do that, you need to add the new and revised files to the commit:

1. Run the following commands to add the files to git tracking:

```
git add [name of revised file]
git add [name of new file]
```

2. After adding the files, run `git status` again. You will see output similar to that in the following image:

Task 3 - Commit the changes

Because this is first time you're using git on Theia, you need to set your display name and email so you can commit the changes:

1. Run the following commands, substituting your email and display name:

```
git config --global user.email rsannareddy@gmail.com
git config --global user.name "Ramesh Sannareddy"
```

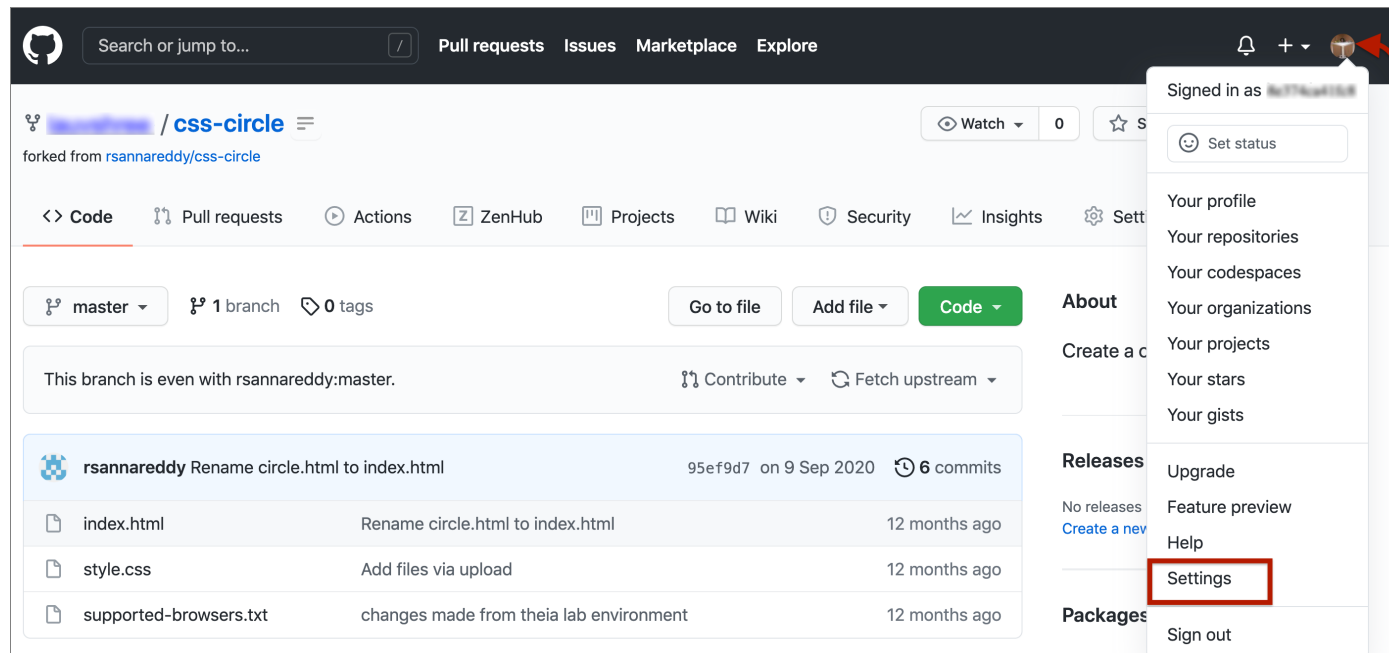
You are now able to commit the changes you've made using git.

2. Run the following command to commit the changes. You will pass a commit message using the `-m` option.

```
git commit -m 'changes made from theia lab environment'
```

Task 4 - Generate Personal Access Token

1. Verify your email address if it hasn't been verified on Github.
2. In the upper-right corner of any page, click your profile photo, then click Settings.



click Developer settings.

3. In the left sidebar,

github.com/settings/profile

Billing & plans

Security log

Security & analysis

Emails

Notifications

Scheduled reminders

SSH and GPG keys

Repositories

Packages

Organizations

Saved replies

Applications

Developer settings

Select a verified email to display

You have set your email address to private. To toggle email privacy, go to [email settings](#) and uncheck "Keep my email address private."

Bio

Tell us a little bit about yourself

You can @mention other users and organizations to link to them.

URL

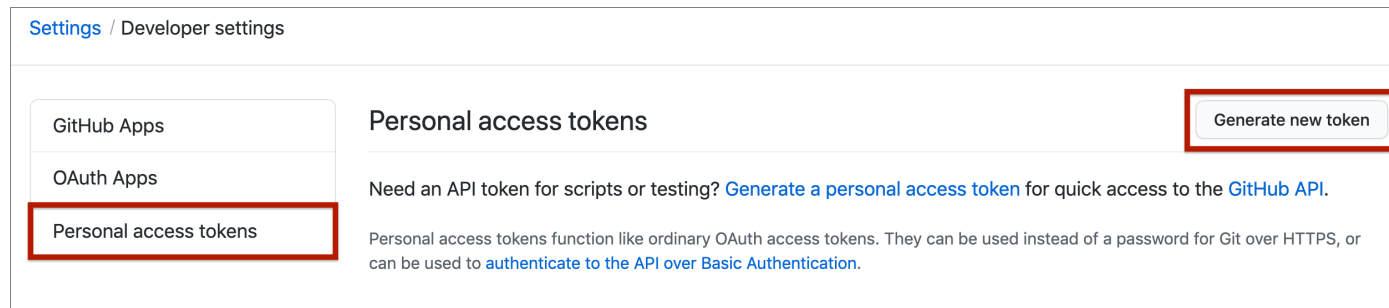
Twitter username

Company

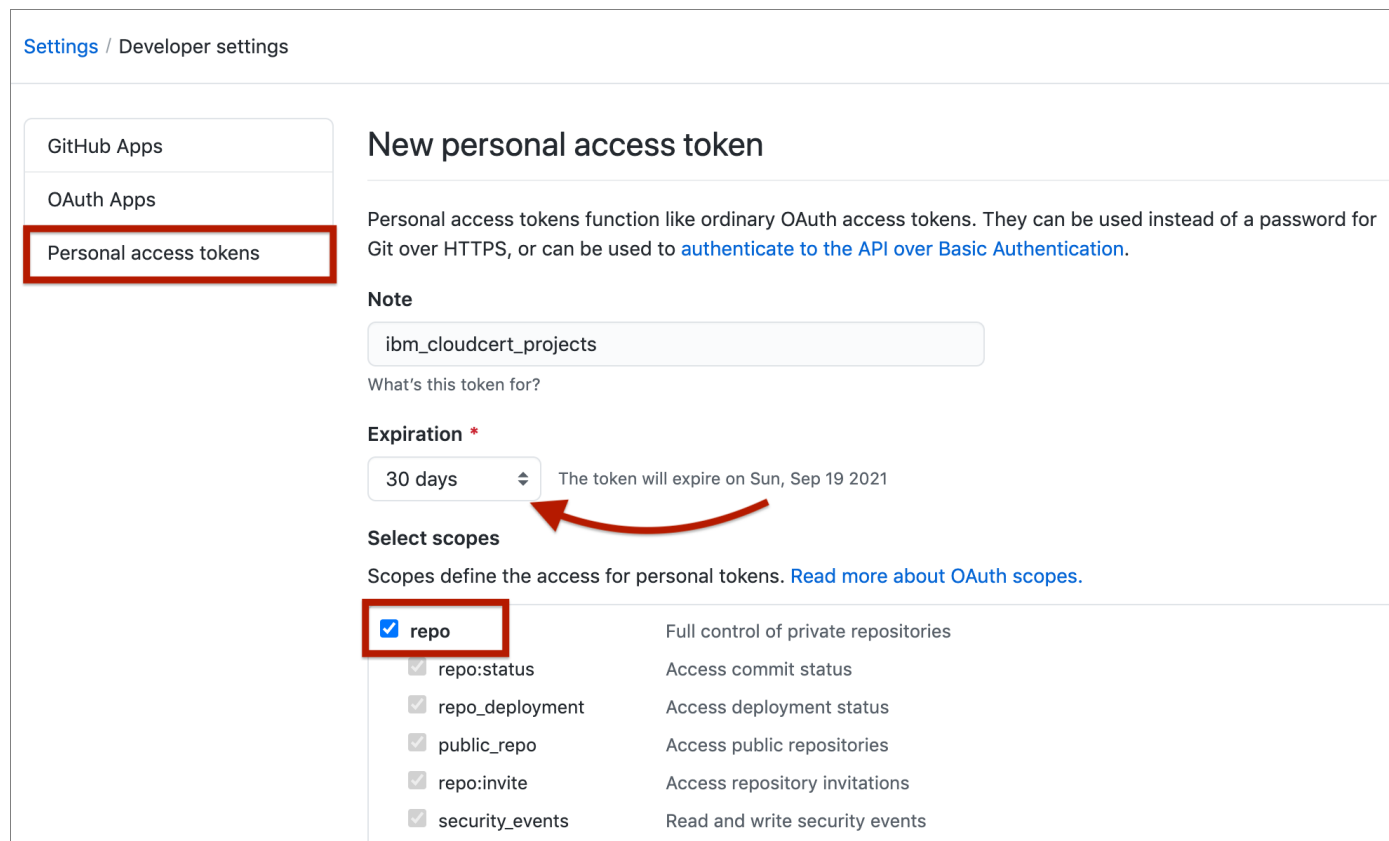
You can @mention your company's GitHub organization to link it.

Location

4. In the left sidebar, click Personal access tokens and click on `Generate Tokens`



5. Give your token a descriptive name. To give your token an expiration, select the Expiration drop-down menu, then click a default or use the calendar picker. Select the scopes, or permissions, you'd like to grant this token. To use your token to access repositories from the command line, select repo.



6. Click Generate token

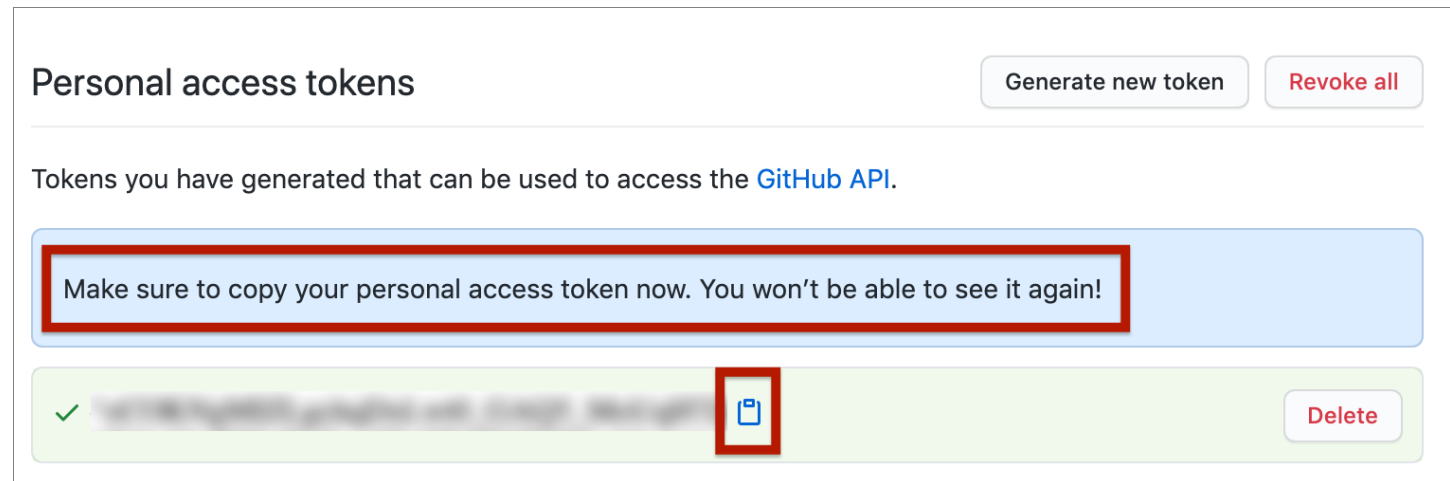
<input type="checkbox"/>	notifications	Access notifications
<input type="checkbox"/>	user	Update ALL user data
<input type="checkbox"/>	read:user	Read ALL user profile data
<input type="checkbox"/>	user:email	Access user email addresses (read-only)
<input type="checkbox"/>	user:follow	Follow and unfollow users
<input type="checkbox"/>	delete_repo	Delete repositories
<input type="checkbox"/>	write:discussion	Read and write team discussions
<input type="checkbox"/>	read:discussion	Read team discussions
<input type="checkbox"/>	admin:enterprise	Full control of enterprises
<input type="checkbox"/>	manage_billing:enterprise	Read and write enterprise billing data
<input type="checkbox"/>	read:enterprise	Read enterprise profile data
<input type="checkbox"/>	admin:pgp_key	Full control of public user GPG keys (Developer Preview)
<input type="checkbox"/>	write:pgp_key	Write public user GPG keys
<input type="checkbox"/>	read:pgp_key	Read public user GPG keys

Generate token

Cancel

and make a note of it.

2. Make sure you copy the token and keep it safe. It is not visible to you again.



Treat your tokens like passwords and keep them a secret.

Once you have a token, you can enter the Personal Access Token as password when performing Git operations.

Task 5 - Push the code to GitHub

The `git push` command will enable you to sync all the changes made locally to the GitHub web repository.

1. Run the following command with your actual HTTPS link:

```
git push [HTTPS link]
```

You will be prompted by git for your username and password.

2. Type your GitHub username and for the password, enter the personal access token you generated in the previous task. When you are authenticated, all committed changes are synced with your GitHub repository.

You can now visit the GitHub repository page and check to ensure that the revised and newly added files are in place.

Summary

Congratulations! You should now know how to access git and run commands from the command line using the Theia lab environment.

In this lab, you have learned how to:

- Commit the changes
- Push the updated files to GitHub

We encourage you to make more changes to your files using the Theia environment to become familiar with git, GitHub, and running commands from command line.

Author:

Ramesh Sannareddy

Other Contributors

Anamika Agarwal