Connecting to GitHub via HTTPS

A Personal Access Token is a special password that you use instead of your actual account password. When you're finished using the token, you can revoke it so that it can no longer be used. It is also possible to set an expiry time for the token. This helps to keep your account secure. Generate a Personal Access Token

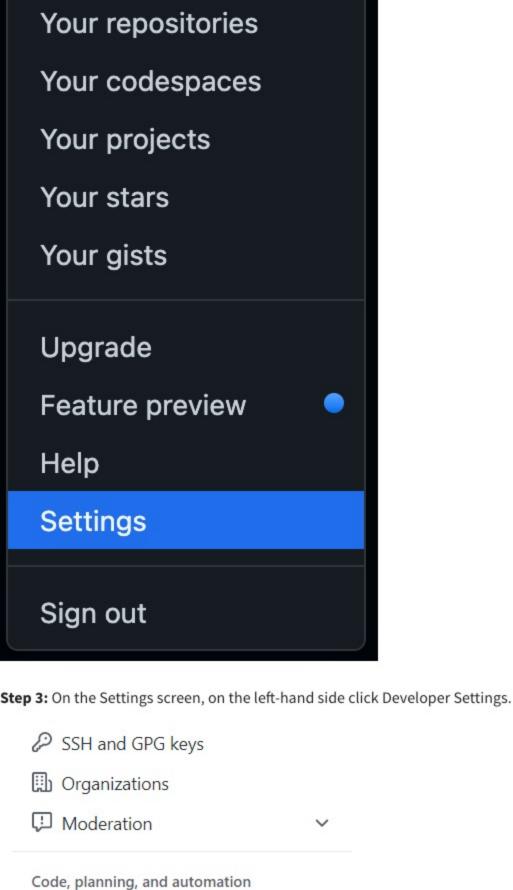
When using Github via the Coursera platform, it is required to authenticate using a Personal Access Token over HTTPS.

Step 1: Log in to Github

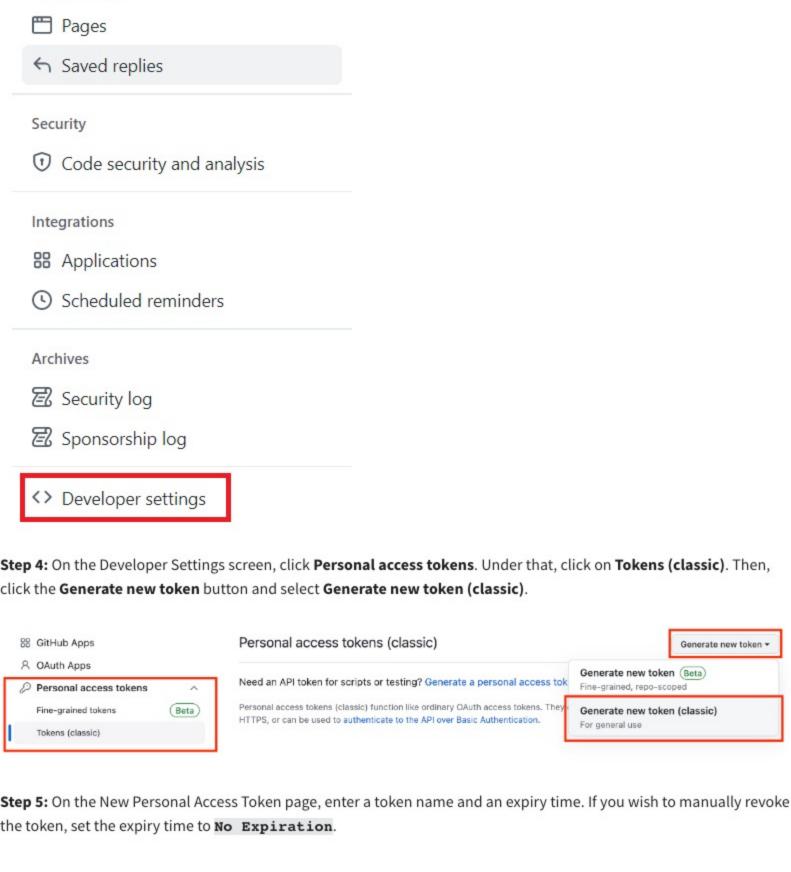
Step 2: Click on the profile icon in the top right of the screen and select Settings.

We now need to set up our Personal Access Token.

Your profile



Repositories Packages



Personal access tokens (classic) function like ordinary OAuth access tokens. They can be used instead of a password for Git over HTTPS, or can be used to authenticate to the API over Basic Authentication.

GitHub strongly recommends that you set an expiration date for your token to help keep your information secure.

Full control of private repositories

Access commit status

Access deployment status

Access public repositories

Access repository invitations

Read and write security events

Update GitHub Action workflows

Personal access tokens (classic) function like ordinary OAuth access tokens. They can be used instead of a password

Generate new token *

Generate new token (Beta)

Generate new token (classic)

Fine-grained, repo-scoped

For general use

Select scopes Scopes define the access for personal tokens. Read more about OAuth scopes.

The token will never expire!

New personal access token (classic)

Note

coursera

Expiration *

What's this token for?

No expiration \$

Learn more

repo

repo:status

public_repo

repo:invite

workflow

Learn more

Select scopes

repo

repo:status

public_repo

repo:invite

workflow

security_events

write:ssh_signing_key

read:ssh_signing_key

repo_deployment

repo_deployment

security_events

Step 6: Under scopes, select repo.

New personal access token (classic)

for Git over HTTPS, or can be used to authenticate to the API over Basic Authentication.

Scopes define the access for personal tokens. Read more about OAuth scopes.

Note coursera What's this token for? Expiration * No expiration \$ The token will never expire!

GitHub strongly recommends that you set an expiration date for your token to help keep your information secure.

Full control of private repositories

Access commit status

Access deployment status

Access public repositories Access repository invitations

Read and write security events

Update GitHub Action workflows

Step 7: Scroll to the end of the page and click Generate token. admin:ssh_signing_key Full control of public user SSH signing keys

Write public user SSH signing keys

Read public user SSH signing keys

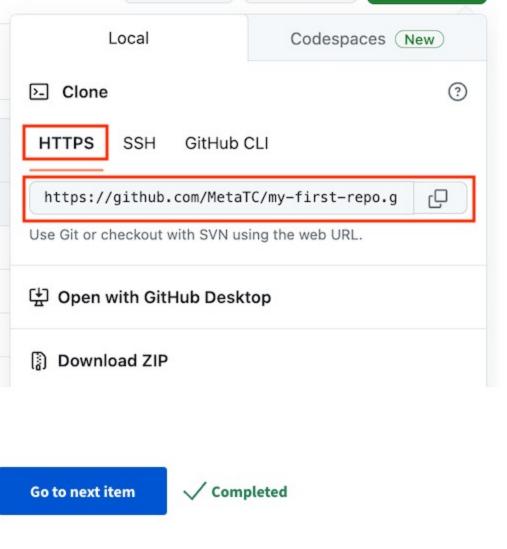
```
Generate token
                         Cancel
Step 8: The token is now generated. Make sure to copy and keep note of the token as it will be hidden when you leave
the page. This token can now be used when connecting to a repository over HTTPS.
  Personal access tokens
                                                                                     Generate new token
                                                                                                          Revoke all
  Tokens you have generated that can be used to access the GitHub API.
    Make sure to copy your personal access token now. You won't be able to see it again!
```

Delete

then just use the HTTPS address for the Git repository itself. Go to file Add file ▼ <> Code -

When accessing a repository and using HTTPS authentication, make sure you have access/permission to connect, and

Personal access tokens function like ordinary OAuth access tokens. They can be used instead of a password for Git over HTTPS, or can be used to



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🗗

Note: If you lose the token, you can delete the old token and create a new one.

authenticate to the API over Basic Authentication.

Accessing Repositories

Report an issue

Dislike

🖒 Like