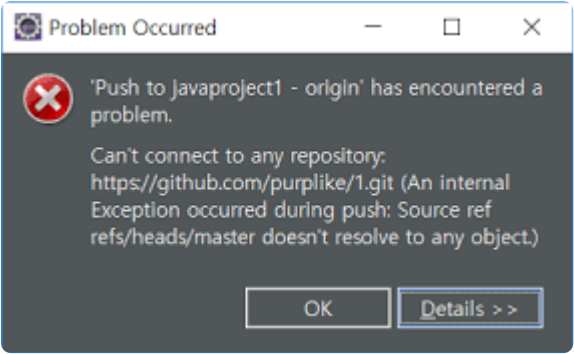


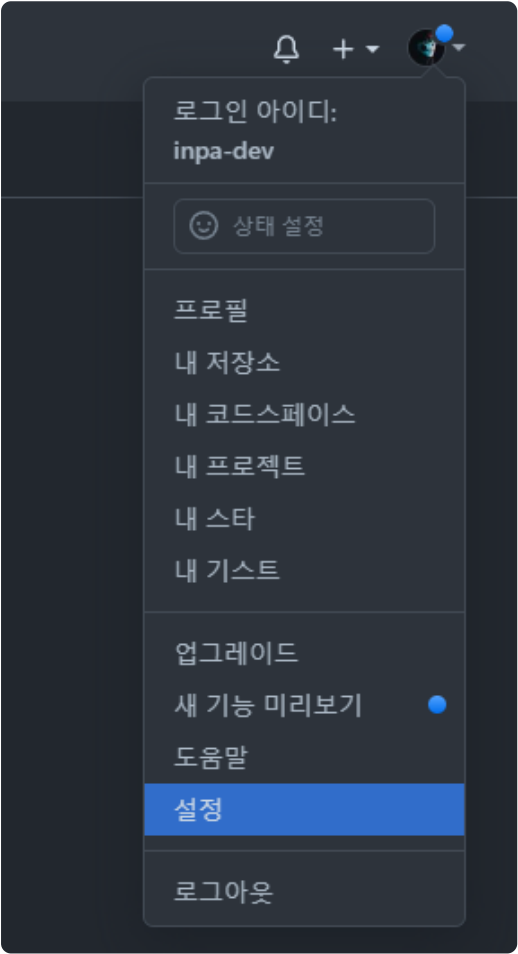
GitHub 토큰 인증

최근에 Git Hub에서 ID/PW기반의 Basic Authentication 인증을 금지하고, ID/Personal Access Token 방식의 Token Authentication 인증을 요구하고 있다. 그래서 앞으로는 소스코드를 push/clone하려고 하면, 아래와 같은 문구가 뜨면서 잘 되지 않는 상황이 발생할 수 있다.



깃헙 토큰 발급하기

1. settings -> Developer settings -> Personal access tokens



Accessibility

계정 보안

결제 및 플랜

보안 로그

보안 분석

Sponsorship log

이메일

알림

정기 재촉

SSH와 GPG키

저장소

패키지

Pages

단체

저장된 댓글

애플리케이션

개발자 설정

중재 설정

차단된 사용자

임시 소통 제한

Code review limits

Public email

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.

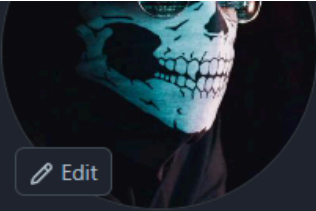
Location

All of the fields on this page are optional and can be deleted at any time, and by filling them out, you're giving us consent to share this data wherever your user profile appears. Please see our [privacy statement](#) to learn more about how we use this information.

Update profile

Contributions

☐ Include private contributions on my profile



Edit

Settings / Developer settings

GitHub Apps

OAuth Apps

Personal access tokens

Personal access tokens

Generate new token

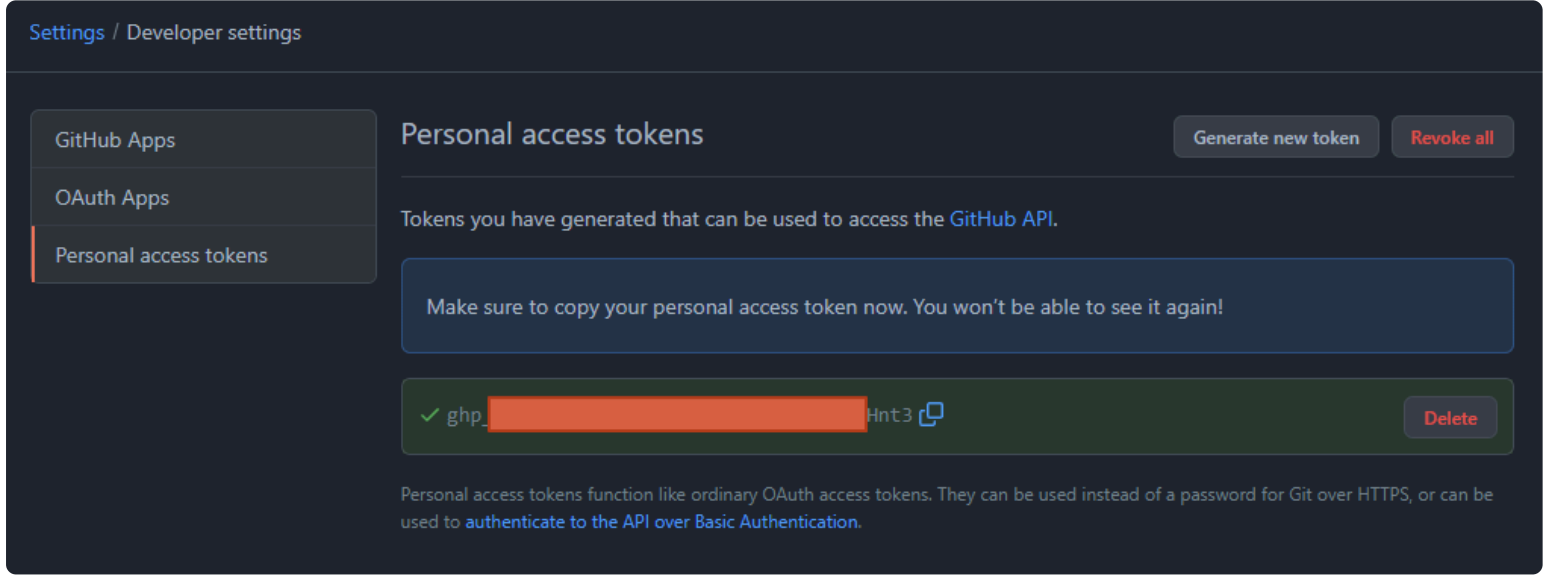
Need an API token for scripts or testing? [Generate a personal access token](#) for quick access to the [GitHub API](#).

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 [authenticate to the API over Basic Authentication](#).

2. repo, gist 체크하고 Generate token 버튼 클릭

<input checked="" type="checkbox"/> repo	Full control of private repositories
<input checked="" type="checkbox"/> repo:status	Access commit status
<input checked="" type="checkbox"/> repo_deployment	Access deployment status
<input checked="" type="checkbox"/> public_repo	Access public repositories
<input checked="" type="checkbox"/> repo:invite	Access repository invitations
<input checked="" type="checkbox"/> security_events	Read and write security events
<input type="checkbox"/> workflow	Update GitHub Action workflows
<input type="checkbox"/> write:packages	Upload packages to GitHub Package Registry
<input type="checkbox"/> read:packages	Download packages from GitHub Package Registry
<input type="checkbox"/> delete:packages	Delete packages from GitHub Package Registry
<input type="checkbox"/> admin:org	Full control of orgs and teams, read and write org projects
<input type="checkbox"/> write:org	Read and write org and team membership, read and write org projects
<input checked="" type="checkbox"/> read:org	Read org and team membership, read org projects
<input type="checkbox"/> admin:public_key	Full control of user public keys
<input type="checkbox"/> write:public_key	Write user public keys
<input type="checkbox"/> read:public_key	Read user public keys
<input type="checkbox"/> admin:repo_hook	Full control of repository hooks
<input type="checkbox"/> write:repo_hook	Write repository hooks
<input type="checkbox"/> read:repo_hook	Read repository hooks
<input type="checkbox"/> admin:org_hook	Full control of organization hooks
<input checked="" type="checkbox"/> gist	Create gists

그럼 토큰이 발급된다. 반드시 메모장이나 어딘가에 저장해 놓자. 새로 고침하면, 토큰을 다시 볼수 없기 때문이다. 만일 토큰을 잃어 버렸다면 다시 생성해야 된다.



발급 받았다면, 이제부터는 GitHub에 push 등의 작업을 수행할 때, PW를 입력하라고 뜨면, 방금 생성한 Personal Access Token을 입력하면 된다.

깃헙 토큰 등록하기

아래 코드처럼 글로벌 유저랑 유저 이메일을 세팅한다.

BASH

```
$ git config --global user.name "aaaa"
$ git config --global user.email "aaaa@bbb.com"
```

그리고 github.com에서 clone, pull 할때 이메일과 패스워드를 물어보는데 이제부터는 패스워드에 기존 github.com 패스워드가 아닌 아까 만든 토큰을 복붙 하면 된다. 그런데 토큰은 외우기 힘들기 때문에 매번 복사해서 붙여넣기 해야 하는데, 이게 귀찮을 수 있다.

그래서 cache 설정을 이용해서 자신의 토큰을 해당 git repository에 저장할 수 있는데, 아래 명령어를 입력하면 다음번에 넣는 유저이메일과 패스워드(토큰)을 캐싱하게 된다.

BASH

```
$ git config --global credential.helper cache
```

한번 위와 같이 캐싱하게 되면 더이상 패스워드(토큰)을 물어보는 일은 없을 것이다.
만일 캐시를 지우고자 할때는 다음 명령어를 입력한다

BASH

```
$ git config --global --unset credential.helper
```

깃헙 토큰 저장소에 영구 적용

우리가 만든 토큰을 기존에 있던 git 리포지터리에 아예 저장하여 영구 적용하는 방법을 알아 보자.
먼저 본인의 git repository 폴더에 들어간다. 그러면 .git이란 폴더가 있을 것이다. (안보이면, 폴더 숨김 항목 보기 설정)
들어가보면 config 란 파일이 있다.

hooks	2021-11-26 오후 3:53	파일 폴더	
info	2021-11-26 오후 3:53	파일 폴더	
objects	2021-11-26 오후 3:53	파일 폴더	
refs	2021-11-26 오후 3:53	파일 폴더	
config	2021-11-26 오후 3:55	파일	1KB
description	2021-11-26 오후 3:53	파일	1KB
HEAD	2021-11-26 오후 3:53	파일	1KB

이 config란 파일을 편집기로 엽니다.

SHELL

```
[core]
  repositoryformatversion = 0
  filemode = false
  bare = false
  logallrefupdates = true
  symlinks = false
  ignorecase = true
[remote "origin"]
  url = https://github.com/inpa-dev/Anki-Vector-mp3.git
  fetch = +refs/heads/*:refs/remotes/origin/*
```

우리가 필요한거는 [remote "origin"] 부분에 있는 url 부분이다. url을 아래와 같은 형식으로 바꾸면 된다.

SHELL

```
url = https://<user-id>:<token>@github.com/cpro95/myvimrc.git
```

이제 본인의 git repository에서 git push 명령어를 이용해 보시면 github 에 push 가 알아서 잘 되는걸 볼수 있을 것이다.