




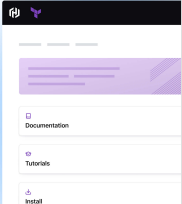
Terraform 설치/설정

Install | Terraform | HashiCorp Developer

Explore Terraform product documentation, tutorials, and examples.

 https://developer.hashicorp.com/terraform/downloads?product_intent=terraform



Explore Terraform product documentation, tutorials, and examples.
[Learn More →](#)



terraform alias 지정

```
$ vim ~/.zshrc
# 밑에 추가
alias "tf"="terraform"

$ source ~/.zshrc

또는

$ vim ~/.bashrc
# 밑에 추가
alias "tf"="terraform"

$ source ~/.bashrc
```

Terraform 명령어

명령	설명
terraform -help	terraform 사용법
terraform init	테라폼을 수행하기 위한 공급자의 플러그인들을 초기 설정하는 명령어
terraform fmt	terraform 표준 스타일로 파일 구성 포매팅
terraform validate	파일의 구성이 유효한지 문법 검사

명령	설명
terraform plan	테라폼을 통해 실제로 생성되고 변경되는 내역을 보여준다. 더하기(+) : 생성 빼기(-) : 제거 물결(~) : 변경
terraform apply terraform apply -auto-approve	실제 작업을 진행, 변경 내역을 반영하려면 yes 입력 yes를 자동 실행
terraform output	출력 파일
terraform state list	주어진 주소와 일치하는 상태 파일의 모든 리소스 확인
terraform graph	Graph를 통해 종속성 및 생성할 리소스에 대한 관계를 시각적으로 볼 수 있는 기능
terraform destroy terraform destroy -auto-approve	인프라 삭제

git코드 관리

```
$ git init
$ git add main.tf
$ git commit -m "Initial commit"
```

.gitignore 파일은 git에 올리지 않아야 할 파일은 선택한다.

```
$ vim .gitignore
.terraform
*.tfstate
*.tfstate.backup

$ git add .gitignore
$ git commit -m "Add a .gitignore file"

# 로컬 리포지토리 구성
$ git remote add origin git@github.com:<YSENAME>/<REPO_NAME>.git

$ git push origin main
$ git pull origin main
```