**GitHub Actions**

### 🚀 What is ****GitHub Actions****?

**GitHub Actions** is a **CI/CD (Continuous Integration and Continuous Deployment)** tool provided by GitHub that lets you **automate** tasks in your software development workflow — **directly in your GitHub repository**.

Create an account on GitHub:- <https://github.com/>

Create new repository :- Terraform-code

Organization requirement is to create a storage account code on GitHub (remote repository) then what will be the steps:-

1. We will access the ticket.
2. Clone the repo from remote to local (Git clone <https://github.com/getsumantra/Terraform-code.git>)
3. Create a feature branch
4. Write the storage account code in feature branch.
5. Git status
6. Git add **.**
7. Git commit -m “storage account code”
8. Git push
9. Raise pull request – feature branch to main and put reviewer

### ⚔️ What is a ****Git Conflict****?

A **Git conflict** happens when **two or more people** (or branches) have made **changes to the same part of the same file**, and Git doesn't know **which change to keep** during a **merge or rebase**.

Steps for pushing a storage account code from local to remote repository.

* Create a remote repository and copy its hyperlink
* Open vs code and clone it using (git clone https://github.com/getsumantra/Terraform-code.git)
* Create a feature branch using (git branch feature/storage)
* To check which branch is selected (git branch)
* To switch or directly create feature branch (git checkout -b feateure/storage)
* Create a storage account code inside VScode
* Add. gitignore file and add all file which you won’t push like (.terraform, lock.hcl, .exe file)
* Git status
* Git add main.tf (to add one file in staging area)
* Git add **.** (to copy complete file in staging area)
* Git commit -m “add\_storage\_account”
* Git push (but due to applied policies on main branch code could note be pushed on the main)
* git push --set-upstream origin feature/storage
* Now code has been pushed to repository on feature branch select this and raise a PR
* Raise PR (pull request) and add reviewer
* After the approval of reviewer code will be merged from the feature branch to main branch.

Steps for pushing a todoapp infra code from local to remote repository by raising a Pull Request…

* Create a remote repository and copy its hyperlink
* Open vs code and clone it using (git clone https://github.com/getsumantra/Terraform-code.git)
* Create a feature branch using (git branch feature/Jira-101-new-resources)
* To check which branch is selected (git branch)
* To switch or directly create feature branch (git checkout -b feature/ Jira-101-new-resources)
* Create a VM code using modules approach code inside VScode
* Add. gitignore file and add all file which you won’t push like (. terraform, lock.hcl, .exe file)
* Git status

Now we need to setup a policy to remote repository so that any can’t add the code directly to our main branch

* <https://github.com/getsumantra/testing> go to repository first where you want to set policy
* Repo >> Setting >> collaborator >> add people >> add person email id which will be approve all PR
* Repo >> setting >> Branches >> add classic branch protection rule
* Check on box :- Require a pull request before merging >> add at least 1 approver
* Go to VS code and do it
* Git add **.**
* Git commit -m “code added”
* Git push (but due to applied policies on main branch code could note be pushed on the main)
* git push --set-upstream origin feature/storage
* Now code has been pushed to repository on feature branch select this and raise a PR
* Raise PR (pull request) and add reviewer
* After the approval of reviewer code will be merged from the feature branch to main branch.

Now policy has become set lets and code pushed in feature branch now raise a Pull Request (PR):

* [getsumantra](https://github.com/getsumantra) / [testing](https://github.com/getsumantra/testing) (go to repo) >> pull request >> New pull request >> add one approver >> add comment and submit
* Email has sent to approver email id open that link and check and approve the required changes

Now the code has become merged to the main branch and changes will be shown here