* Super Important

* Colaboration by team members

Project (Code)
10 year old
Support & New changes
↓
Change Request
[CR]

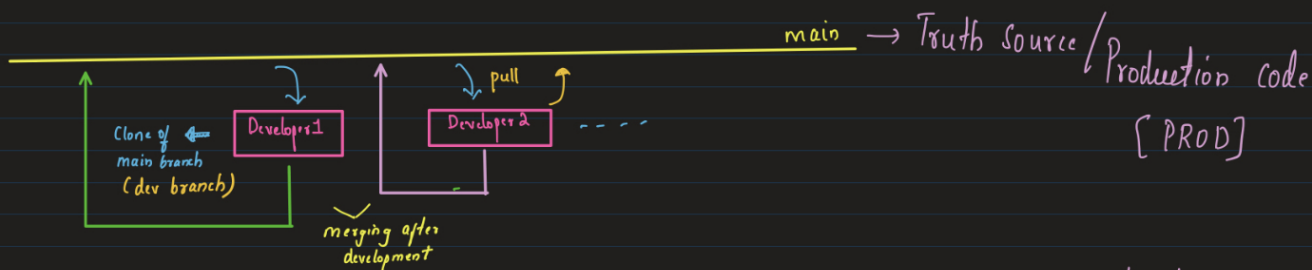Team
10 people

* Spring 1 ===⟹ 10 stories ⟶ 10 team members ∫

Conflicts, bugs
↗
* if all team members start working on main then

---

main → Truth Source / Production Code [PROD]

pull

Clone of main branch ⟸ Developer 1   Developer 2 - - - -
(dev branch)

merging after development

* If developer 2 has finished the story after developer 1 and merge in the main branch
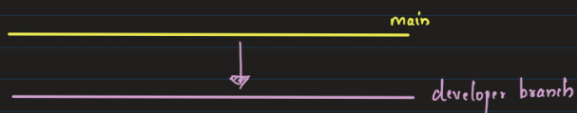merging will not happen because main branch is updated

So, after completing story, it will create a pull request (to get updated code)
①↳ resolve conflicts
②↳ after no conflicts
③↳ Push code

Usually,

_____ main
       ↓
_____ developer branch

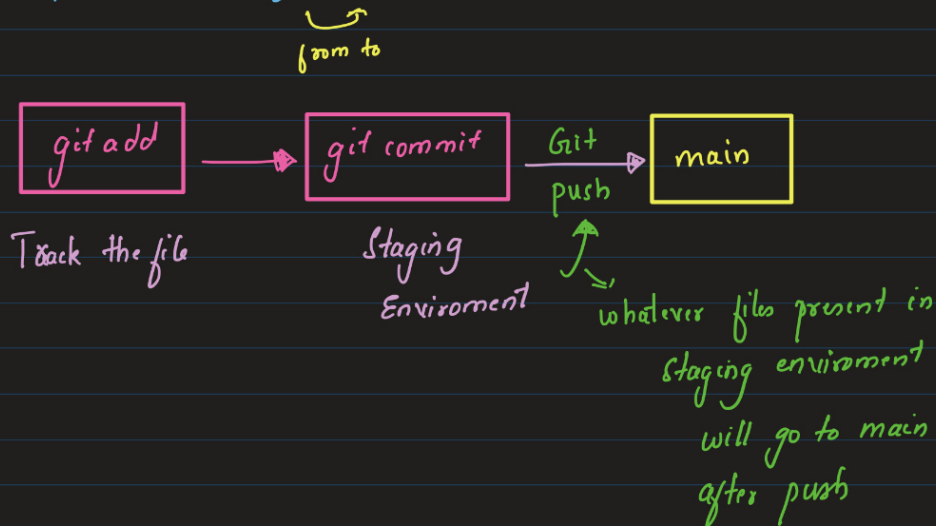* Clone main branch & create developer branch (use for development)

* Creating a repository:

① git init

② git status

③ git add . (all specific file)

④ git commit -m "___"

⑤ git branch -M main

⑥ git remote add origin <link to repository>

⑦ git push -u origin main

⤷ from to

```
┌──────────┐        ┌──────────────┐   Git    ┌──────────┐
│ git add  │ ────▶  │ git commit   │  push ⤍  │  main    │
└──────────┘        └──────────────┘          └──────────┘
```

Track the file   Staging   whatever files present in
                        Enviroment   staging enviroment
                                             will go to main
                                             after push

* Clone repository:

* git clone <http link>

* Create new branch:

  * git branch developer1

  * git branch

  * git checkout developer1 ⟶ change branch, code gets updated

  * git merge developer1 ⟶ change to main first

  * git push origin main

  * git branch -d <branch>

  * git log ⟶ check logs