Day - 3

Advance Git

* We may get problem with the upstream while merging.

```
``` js
|| code          } can be viewed as
```                 code.
```

Steps

1. select a repo

2. fork → clone → open in ur terminal → go to desired folder.

3. git remot -v

   origin   afreebegum.. (link) (fetch) // our
   orign (link) (push) // repo

   ↓

   remote name

   link to which
   we want to get/
   update code

4. git checkout -b "feat/let -to -const"
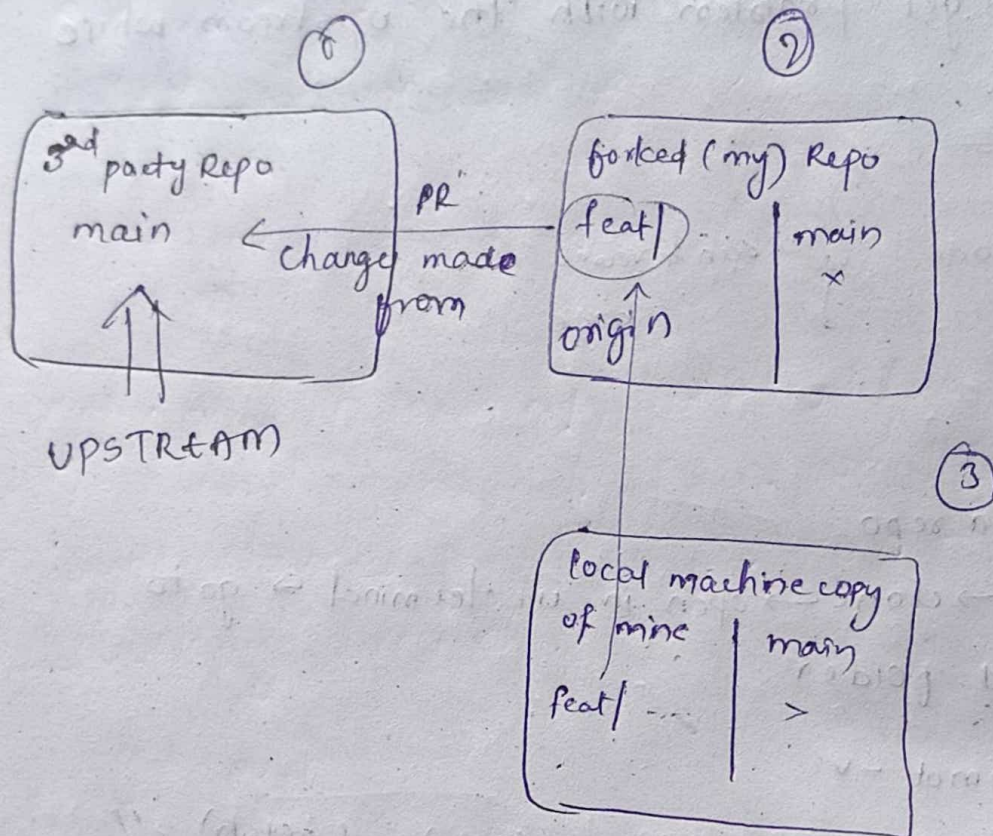
   git branch

   || branch name.

→ git add .

   (To get back all your changes use git reset.

   git commit -m "changed"

git push origin main

git remote

① ②

```
┌─────────────────┐          ┌──────────────────────┐
│ 3ʳᵈ party Repo  │    PR    │ forked (my) Repo     │
│                 │ ←─────── │ (feat)      │ main   │
│ main            │ change made│             │   x   │
│          ↑      │   from   │    ↑        │        │
│          │      │          │ origin      │        │
└──────────┼──────┘          └──────┼───────────────┘
           │                        │
        UPSTREAM)                   │        ③
                                    │
                           ┌────────┼──────────────┐
                           │ local machine copy    │
                           │ of mine    │  main     │
                           │            │           │
                           │ feat/ ...  │   >       │
                           └───────────────────────┘
```

(x git switch main (After PR is merged) x)

→ A new branch is created, from this new branch
   our changes / code is contributed / pushed from
   our local machine to forked repo.

→ Then a new PR is created, which basically
   merges code from feat/cont to main of 3ʳᵈ
   party contributed repo.

Now do

git switch main (After PR is merged)

Now in our main branch, changes are not made in our local machine.

* Now, we do bring changes made in upstream (3ʳᵈ party) to our main(forked) repo

* your local machine only identies your forked repo (e) 2 but not ① , so we need to do following

Steps:-

In terminal (reference ⬇)

1) git remote add upstream https://......//
   ⬇   (copy of main/3ʳᵈ party repo) ①
   (you can
   put
   any
   name)

   ⬆

   Through this established a link b/w ① & ③
   can check through

   git remote

   begin   _____

   origin _____   } list of all upstreams

   upstream ⟵_____
   upstream _____

                        or   ①
2) git pull upstream main / (or git fetch -all
   ⬇
   name which we have given while adding
                                          remote
   (changes will be reflected in our code)

③ git merge upstream/main ②
   git push (changes will be reflected in our forked repo ②

git pull up main ① → ③ (main → main)

git push origin main ③ — ②.(main → main)

in ③

But we don't have changes in another branch,
where we are working

so

git merge origin main — (i)

⇑

Enter this in terminal, after switching to
your working branch, if you want to
sync changes from origin main → your
working branch.

git push — (ii)

git PR — i(i)

git remote add upstream <url of repo>
git pull upstream main
git push origin main

git checkout -b "obc"
└ git g merge origin main