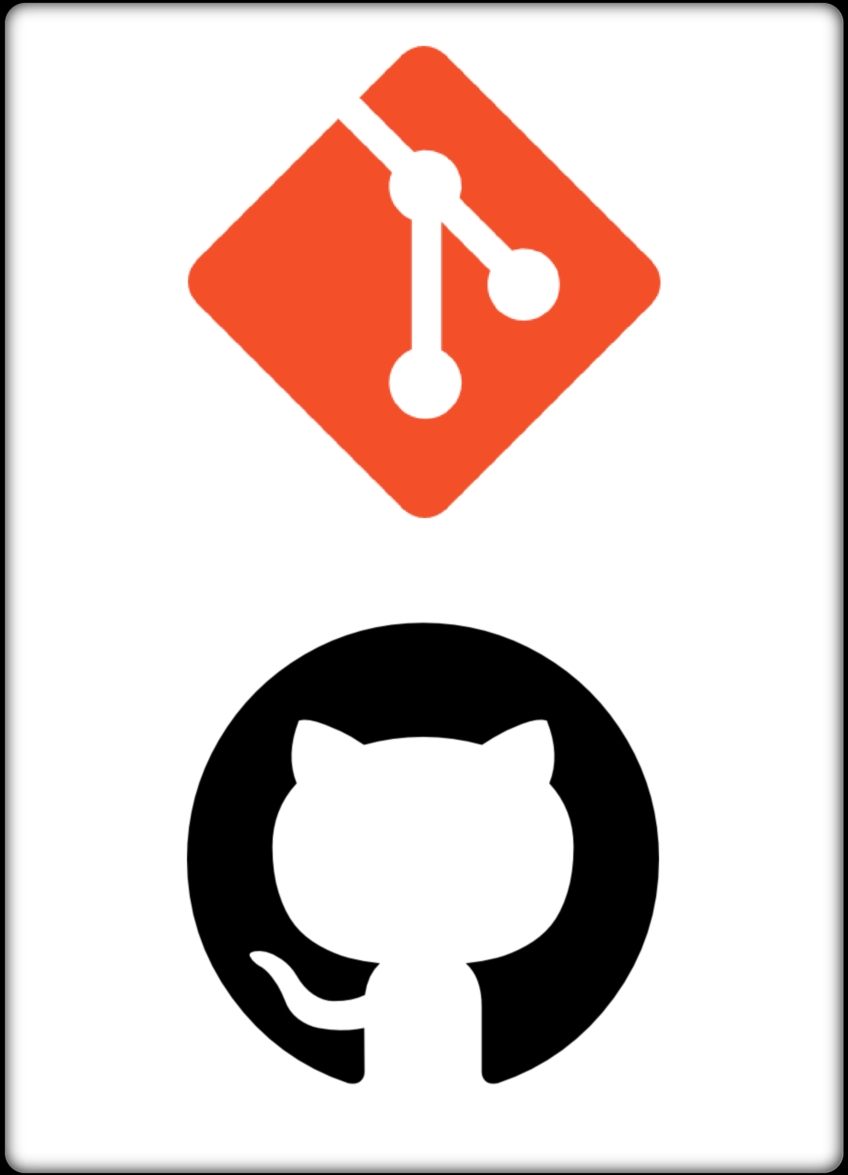


VERSION CONTROL USING GIT

Instructor:

Zaheer ul Hussain Sani

DISCUSSION POINTS

- What is **Git**?
- Why **Git** is Important?
- How **Git** Works?
- Basic **Git Commands**
- Installing and Launching **Git Bash**
- Creating and Publishing **Git Repository**
- Use of **Sourcetree** Software for **Git**

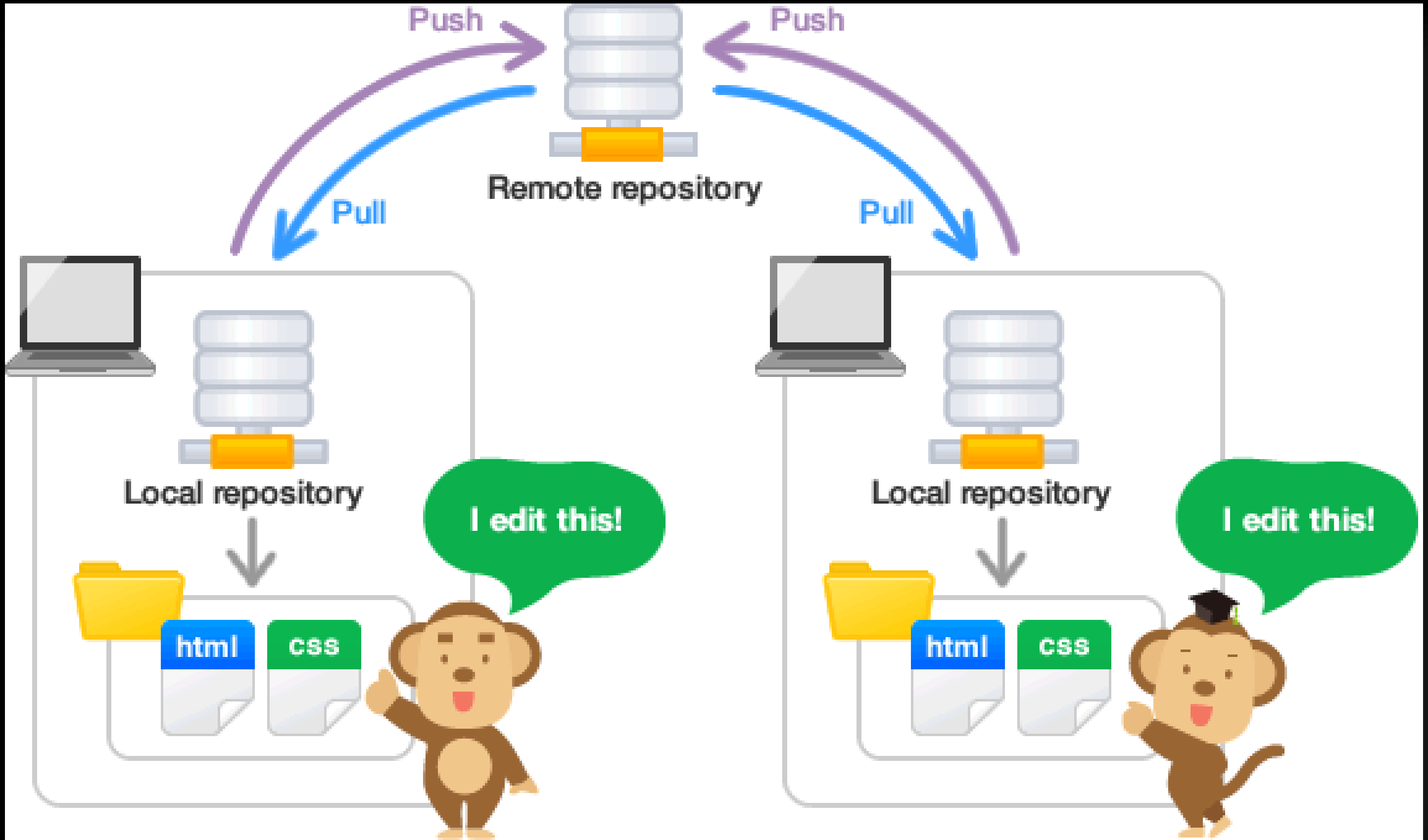
WHAT IS GIT?

Git is a **Distributed Version-control System (DVCS)** for tracking changes in source code during software development

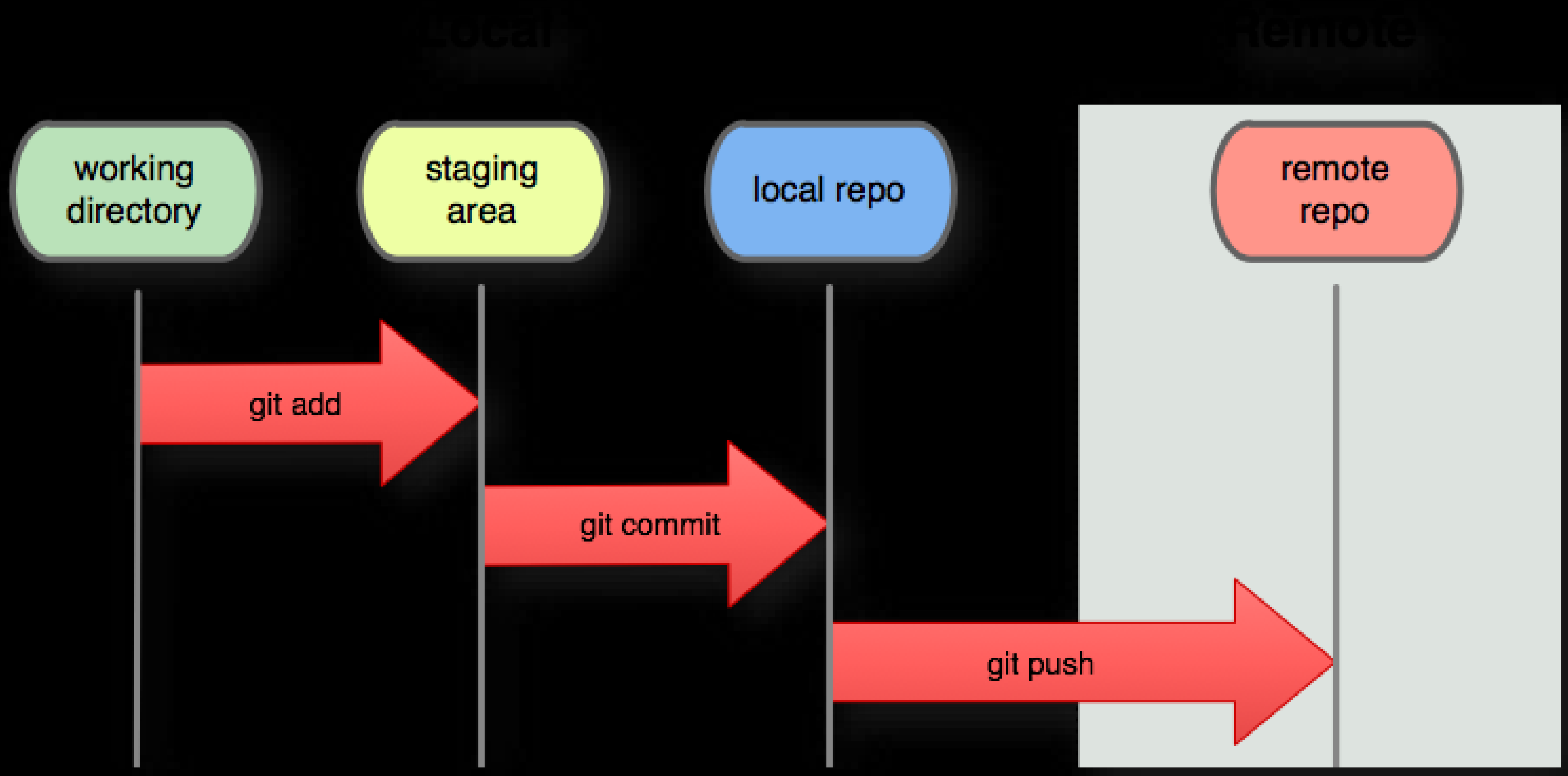
WHY GIT?

- As a Student
- As a Fresh Graduate
- As a Professional Developer
- As a Team Lead

HOW GIT WORKS?



HOW GIT WORKS?

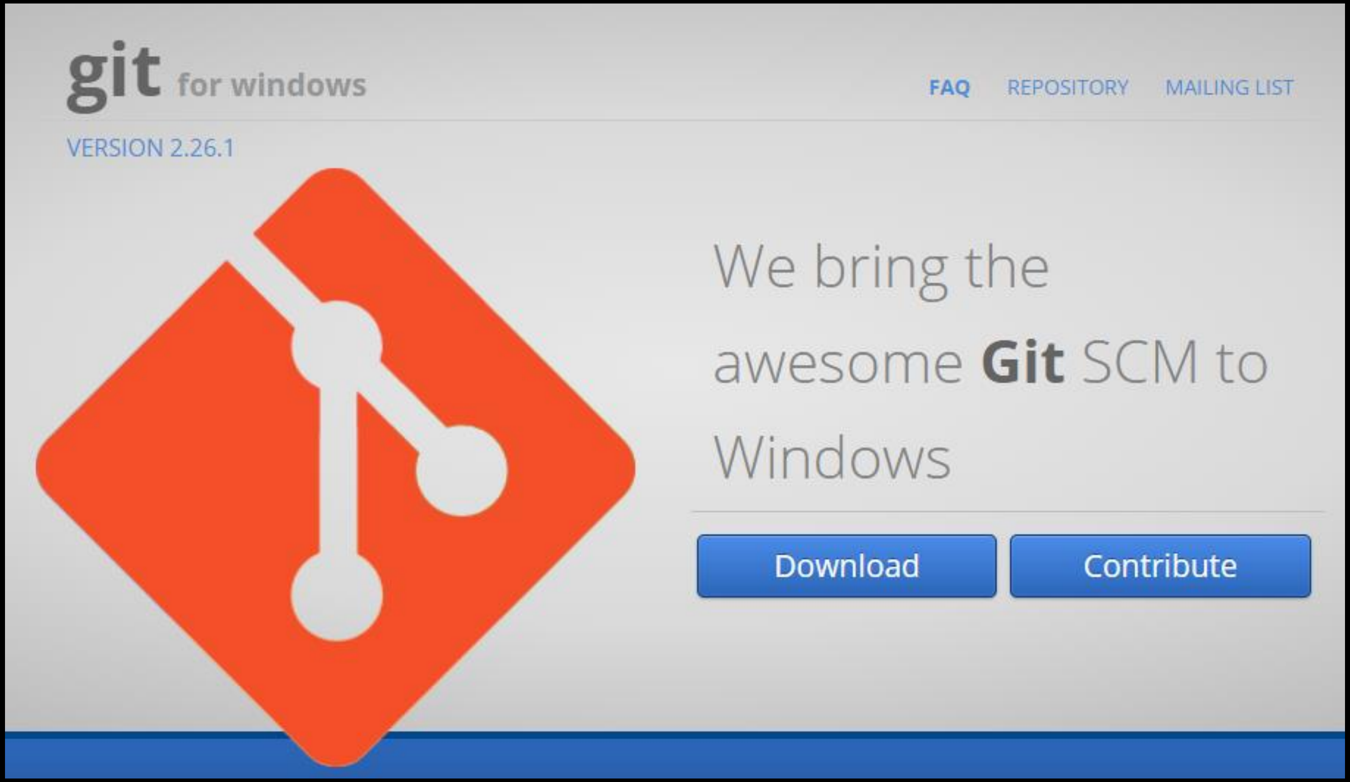


GIT COMMANDS

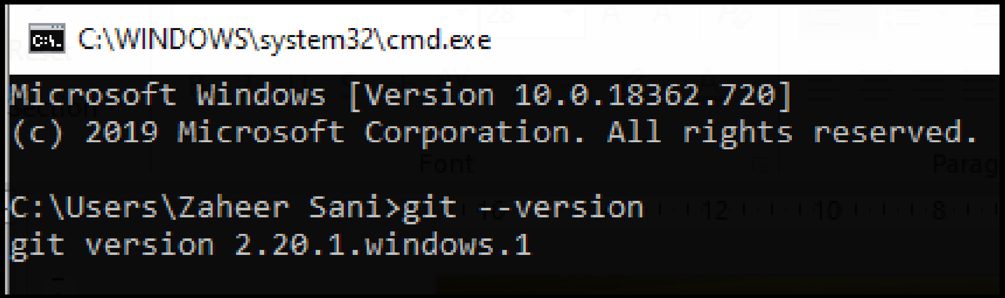
• `git init`	// Initialize Local Git Repository or Repo
• `git add`	// Add File(s)
• `git status`	// Check Status of Working Tree
• `git commit`	// Commit Changes
• `git push`	// Push To Remote Repository

GIT INSTALLATION

- Visit https://gitforwindows.org/

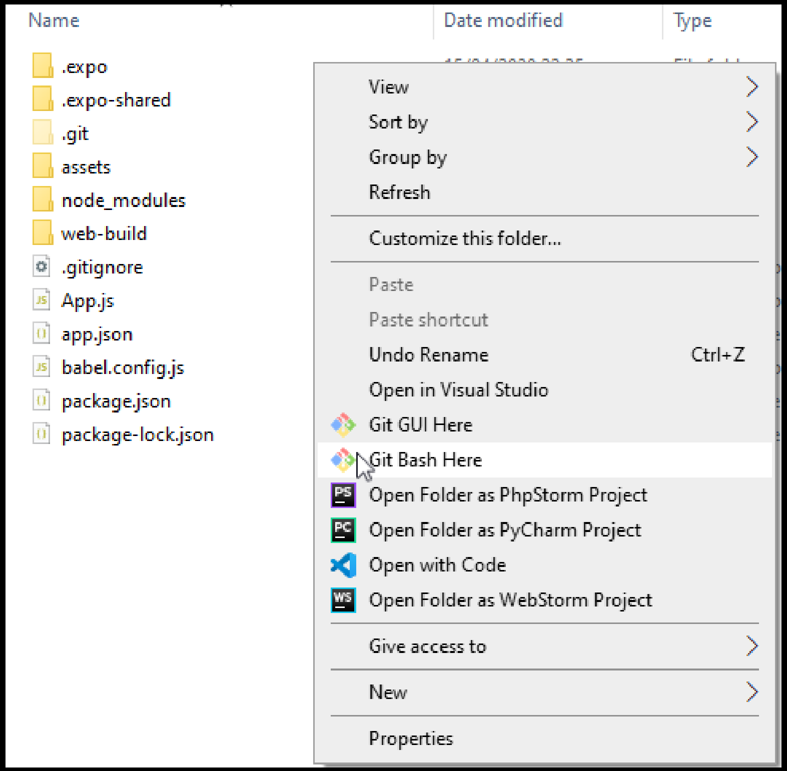


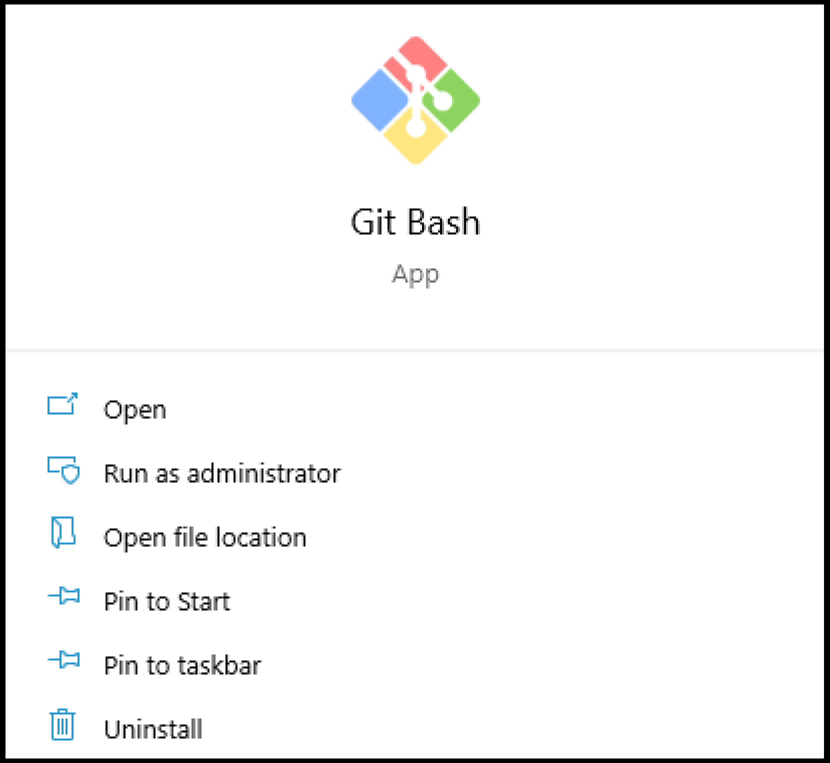
We bring the awesome **Git** SCM to Windows

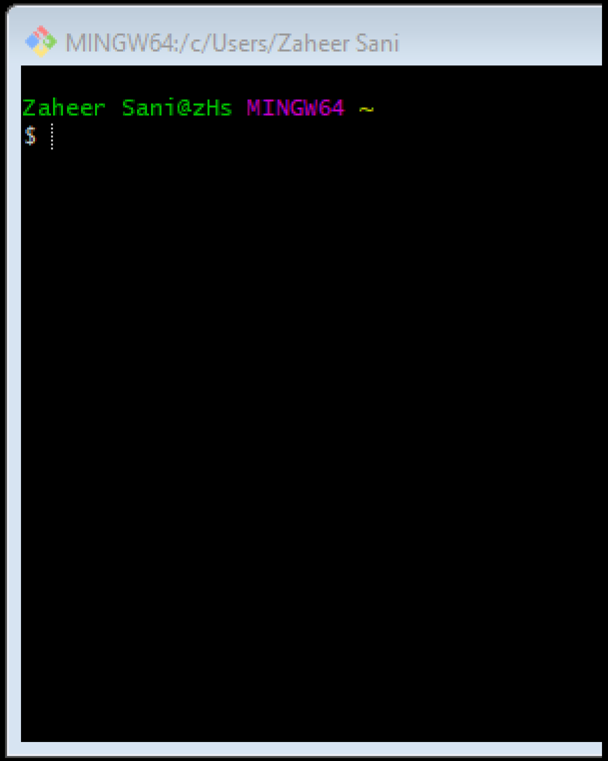


```
Microsoft Windows [Version 10.0.18362.720]
(c) 2019 Microsoft Corporation. All rights reserved.

C:\Users\Zaheer Sani>git --version
git version 2.20.1.windows.1
```

LAUNCH GIT BASH



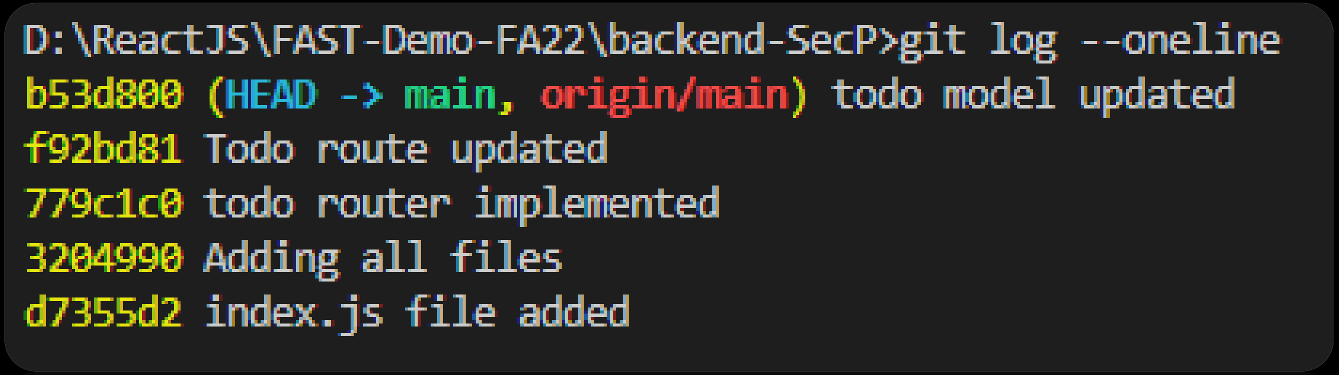




```
Zaheer Sani@zHs MINGW64 ~
$
```

git log

- You can check your commit history by running:
  - `git log`
- The output shows a log of all the commits you have made, who made the commit, the date, and the commit notes.
- Adding the `--oneline` flag shows the commit history condensed in one line. Omitting the flag shows a detailed commit history.
  - `git log --oneline`



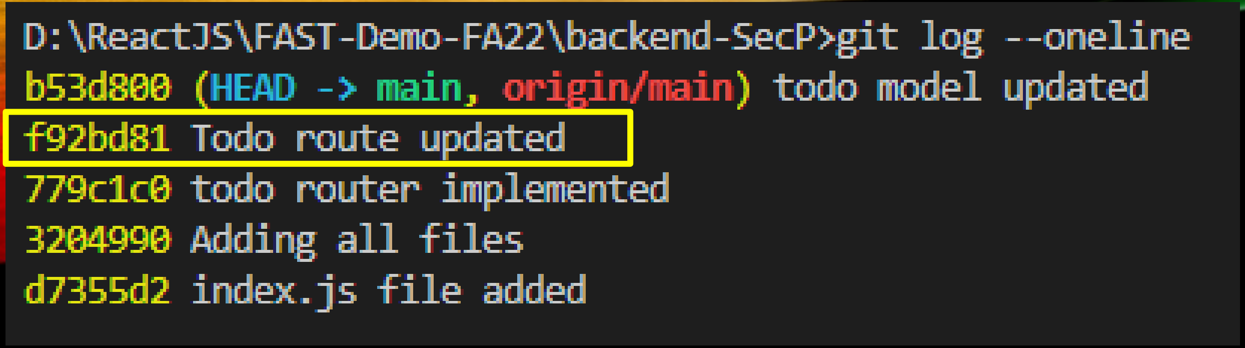
```
D:\ReactJS\FAST-Demo-FA22\backend-SecP>git log --oneline
b53d800 (HEAD -> main, origin/main) todo model updated
f92bd81 Todo route updated
779c1c0 todo router implemented
3204990 Adding all files
d7355d2 index.js file added
```

git revert

- If you have made mistakes during your project development or want to revert a commit for any reason, **`git revert`** allows you to do so.
- The git revert command reverts a particular commit, i.e., undoes the commit you made to remove the changes from the master branch.
  - `git revert [commit_ID]`
- Find commit IDs by running `git log`. The 7-character code is the commit ID.

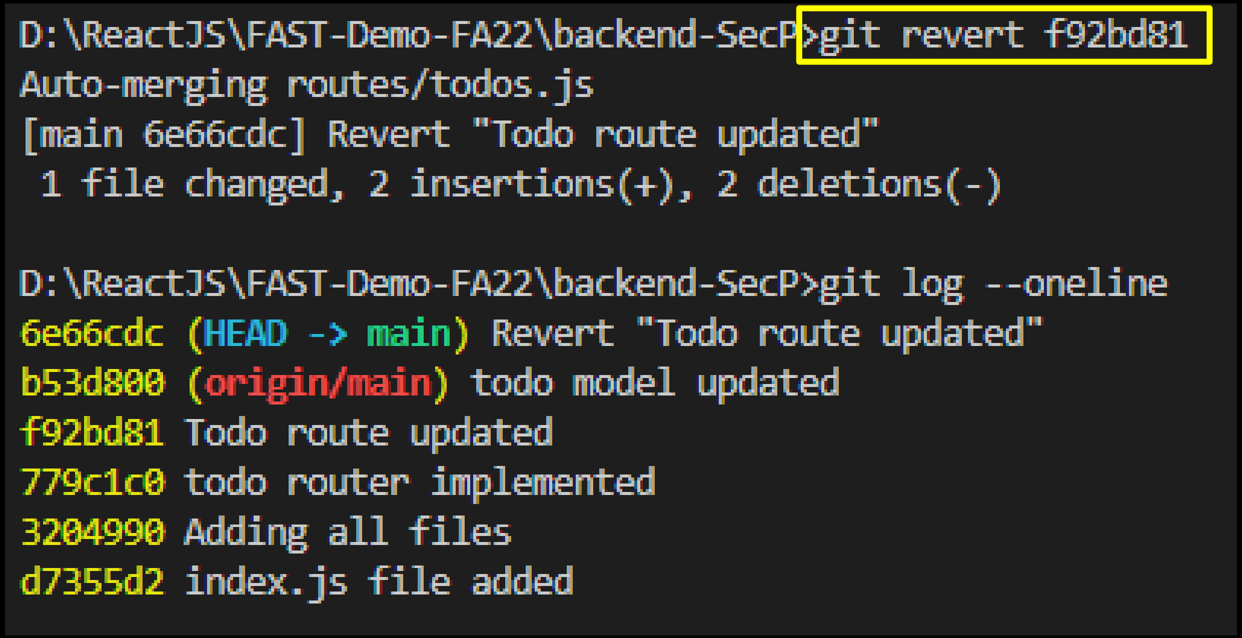
git revert

1

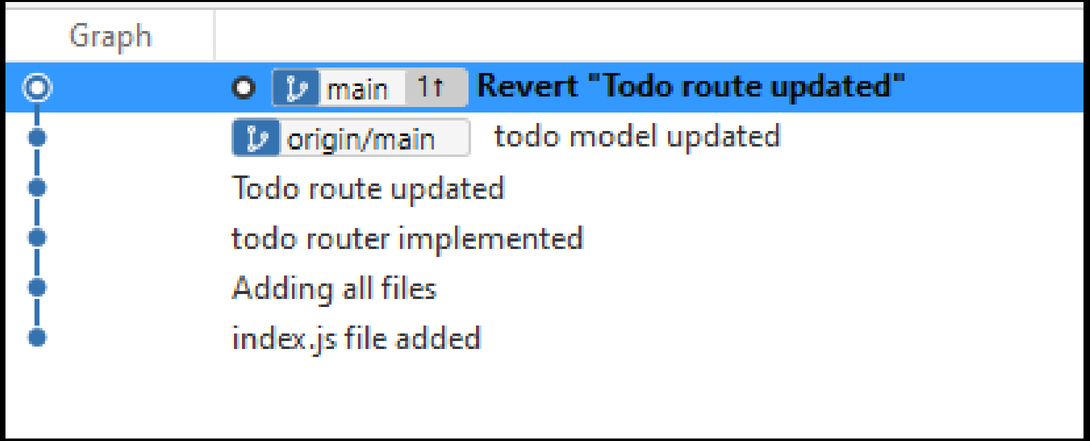


```
D:\ReactJS\FAST-Demo-FA22\backend-SecP>git log --oneline
b53d800 (HEAD -> main, origin/main) todo model updated
f92bd81 Todo route updated
779c1c0 todo router implemented
3204990 Adding all files
d7355d2 index.js file added
```

2



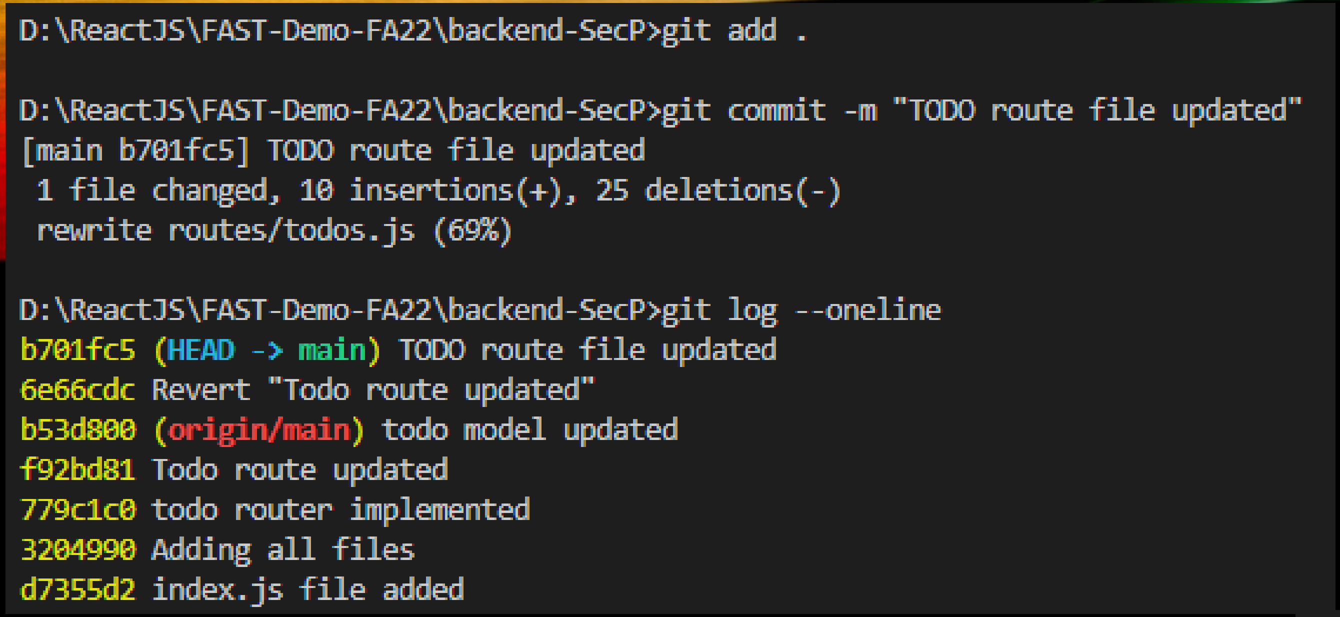
```
D:\ReactJS\FAST-Demo-FA22\backend-SecP>git revert f92bd81
Auto-merging routes/todos.js
[main 6e66cdc] Revert "Todo route updated"
 1 file changed, 2 insertions(+), 2 deletions(-)

D:\ReactJS\FAST-Demo-FA22\backend-SecP>git log --oneline
6e66cdc (HEAD -> main) Revert "Todo route updated"
b53d800 (origin/main) todo model updated
f92bd81 Todo route updated
779c1c0 todo router implemented
3204990 Adding all files
d7355d2 index.js file added
```



```
main 1↑ Revert "Todo route updated"
origin/main todo model updated
Todo route updated
todo router implemented
Adding all files
index.js file added
```

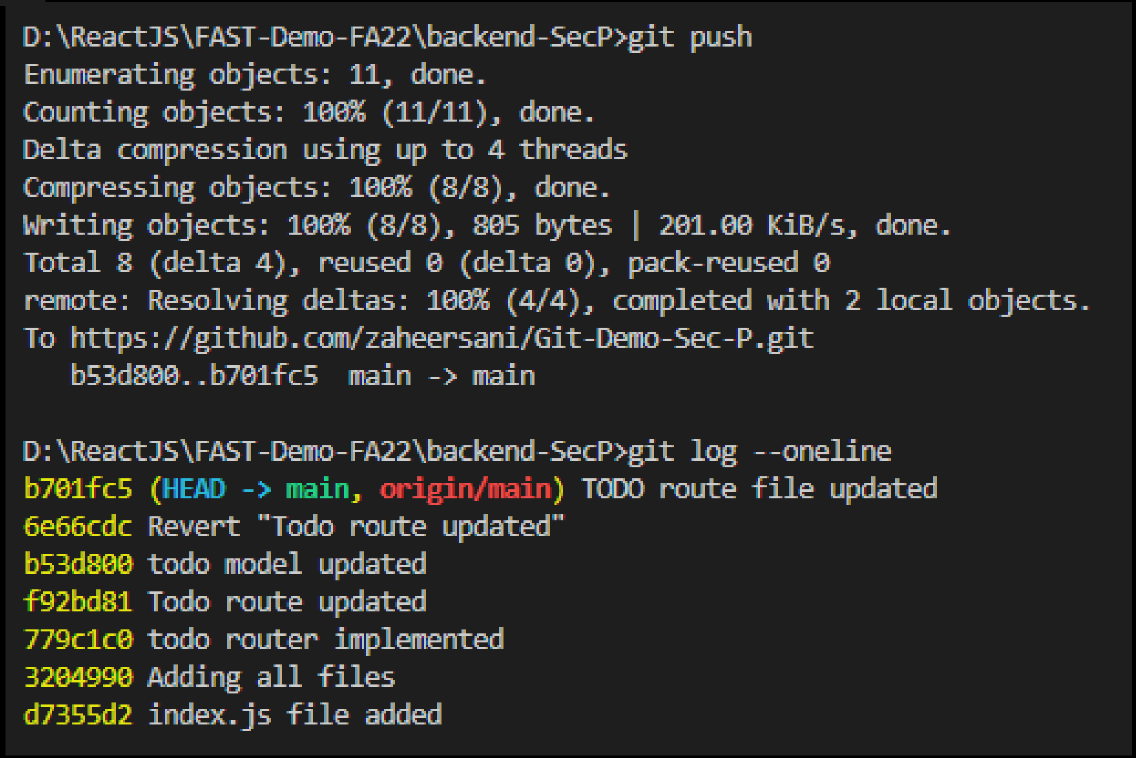
git revert

3



```
D:\ReactJS\FAST-Demo-FA22\backend-SecP>git add .

D:\ReactJS\FAST-Demo-FA22\backend-SecP>git commit -m "TODO route file updated"
[main b701fc5] TODO route file updated
 1 file changed, 10 insertions(+), 25 deletions(-)
 rewrite routes/todos.js (69%)

D:\ReactJS\FAST-Demo-FA22\backend-SecP>git log --oneline
b701fc5 (HEAD -> main) TODO route file updated
6e66cdc Revert "Todo route updated"
b53d800 (origin/main) todo model updated
f92bd81 Todo route updated
779c1c0 todo router implemented
3204990 Adding all files
d7355d2 index.js file added
```

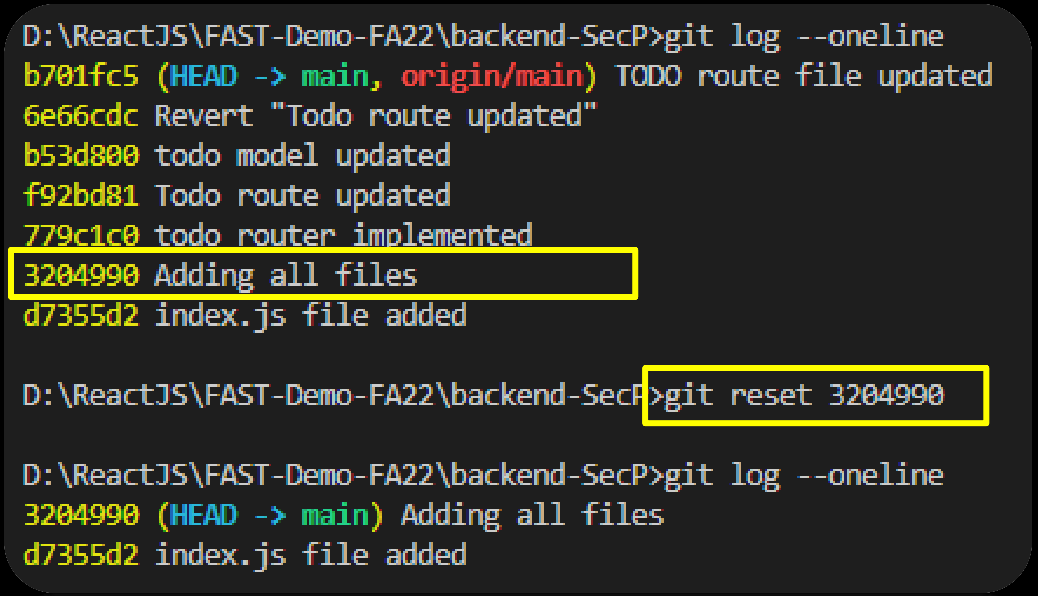
4



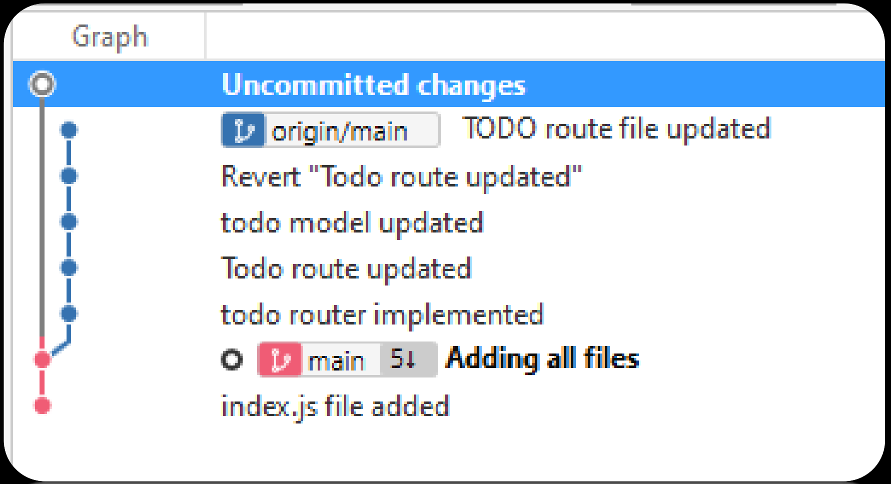
```
D:\ReactJS\FAST-Demo-FA22\backend-SecP>git push
Enumerating objects: 11, done.
Counting objects: 100% (11/11), done.
Delta compression using up to 4 threads
Compressing objects: 100% (8/8), done.
Writing objects: 100% (8/8), 805 bytes | 201.00 KiB/s, done.
Total 8 (delta 4), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (4/4), completed with 2 local objects.
To https://github.com/zaheersani/Git-Demo-Sec-P.git
   b53d800..b701fc5  main -> main

D:\ReactJS\FAST-Demo-FA22\backend-SecP>git log --oneline
b701fc5 (HEAD -> main, origin/main) TODO route file updated
6e66cdc Revert "Todo route updated"
b53d800 todo model updated
f92bd81 Todo route updated
779c1c0 todo router implemented
3204990 Adding all files
d7355d2 index.js file added
```

git reset

- The `git reset` command permanently takes you back to a certain point in development.
- All the files and changes added after that point in time are unstaged if you want to re-add them.



```
D:\ReactJS\FAST-Demo-FA22\backend-SecP>git log --oneline
b701fc5 (HEAD -> main, origin/main) TODO route file updated
6e66cdc Revert "Todo route updated"
b53d800 todo model updated
f92bd81 Todo route updated
779c1c0 todo router implemented
3204990 Adding all files
d7355d2 index.js file added

D:\ReactJS\FAST-Demo-FA22\backend-SecP>git reset 3204990

D:\ReactJS\FAST-Demo-FA22\backend-SecP>git log --oneline
3204990 (HEAD -> main) Adding all files
d7355d2 index.js file added
```



Uncommitted changes

origin/main TODO route file updated

Revert "Todo route updated"

todo model updated

Todo route updated

todo router implemented

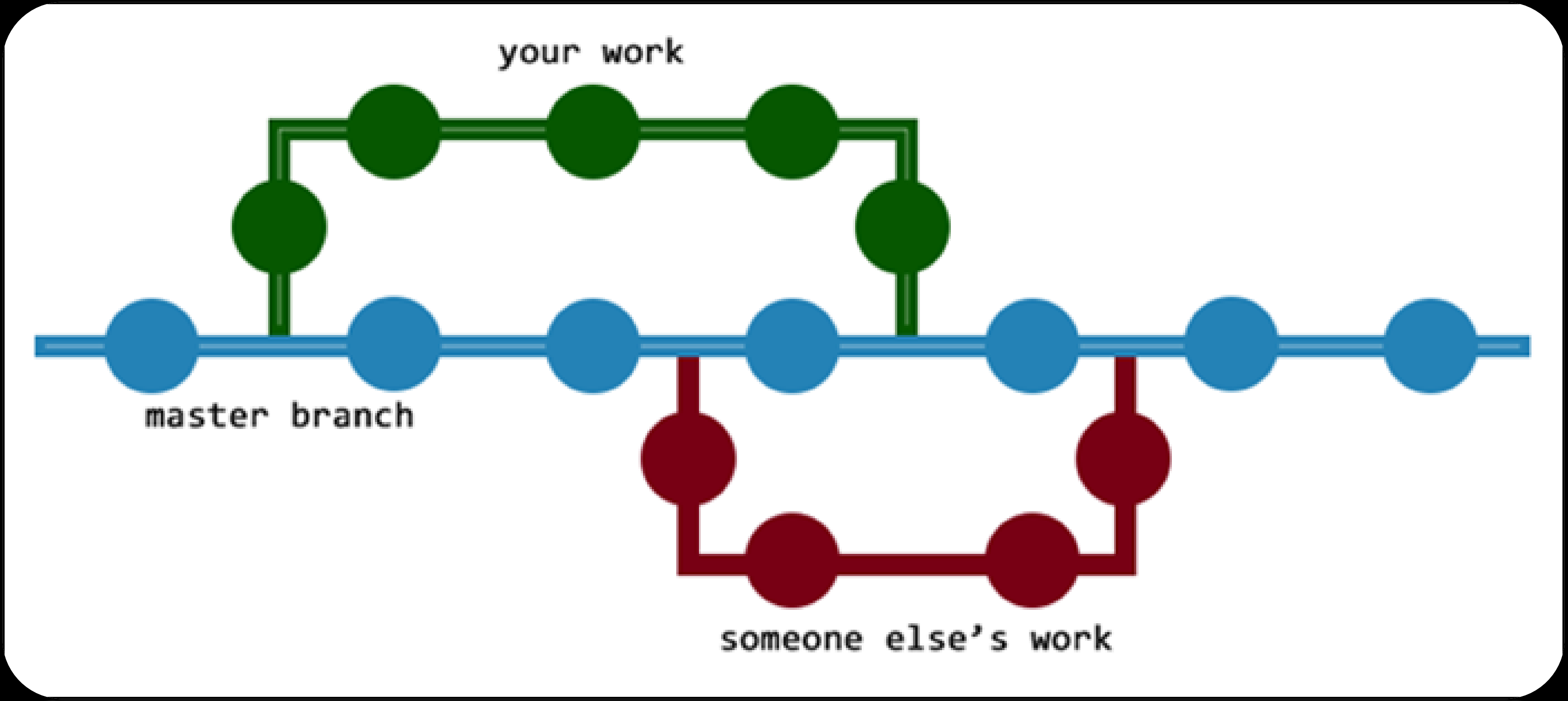
main 5↓ Adding all files

index.js file added

BRANCHING

- Branching is a feature in Git that allows developers to work on a copy of the original code to fix bugs or develop new features.
- By working on a branch, developers don't affect the master branch until they want to implement the changes.
- The master branch generally represents the stable version of your code, which is released or published.
- That's why you should avoid adding new features and new code to the master branch if they are unstable.
- Branching creates an isolated environment to try out the new features, and if you like them, you can merge them into the master branch.
- If something goes wrong, you can delete the branch, and the master branch remains untouched.

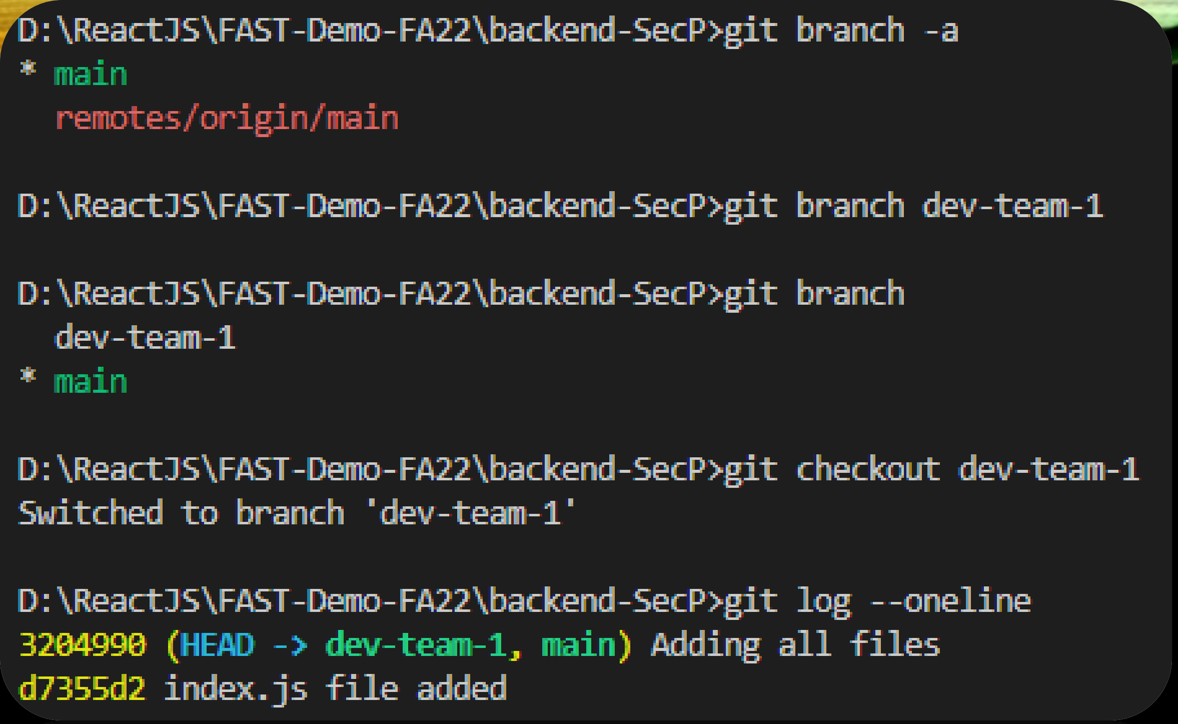
BRANCHING



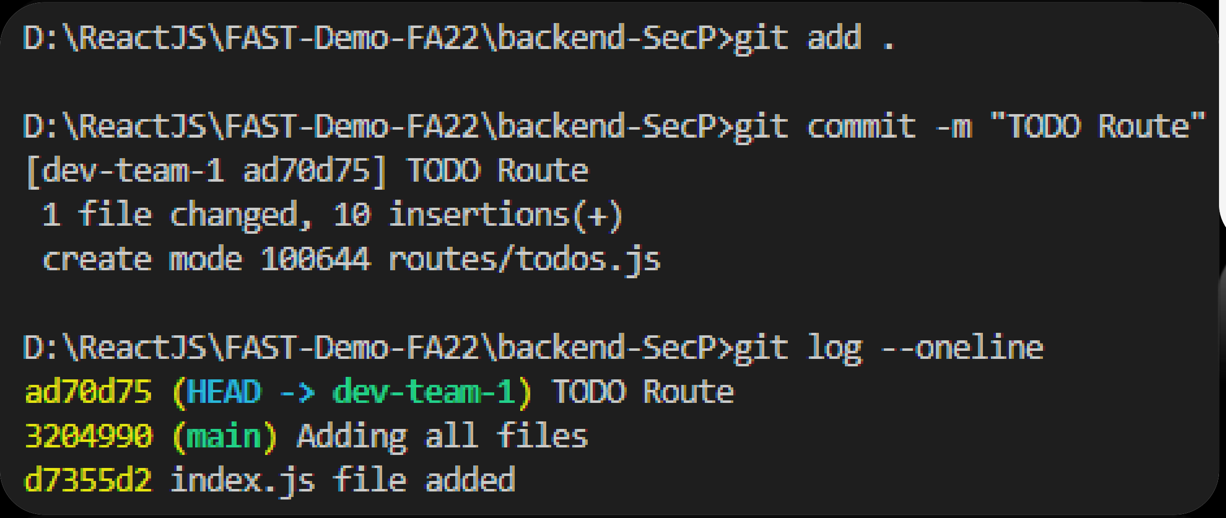
BRANCHING

- Check the branches
  - `git branch -a`
- Create new branch
  - `git branch [branch-name]`

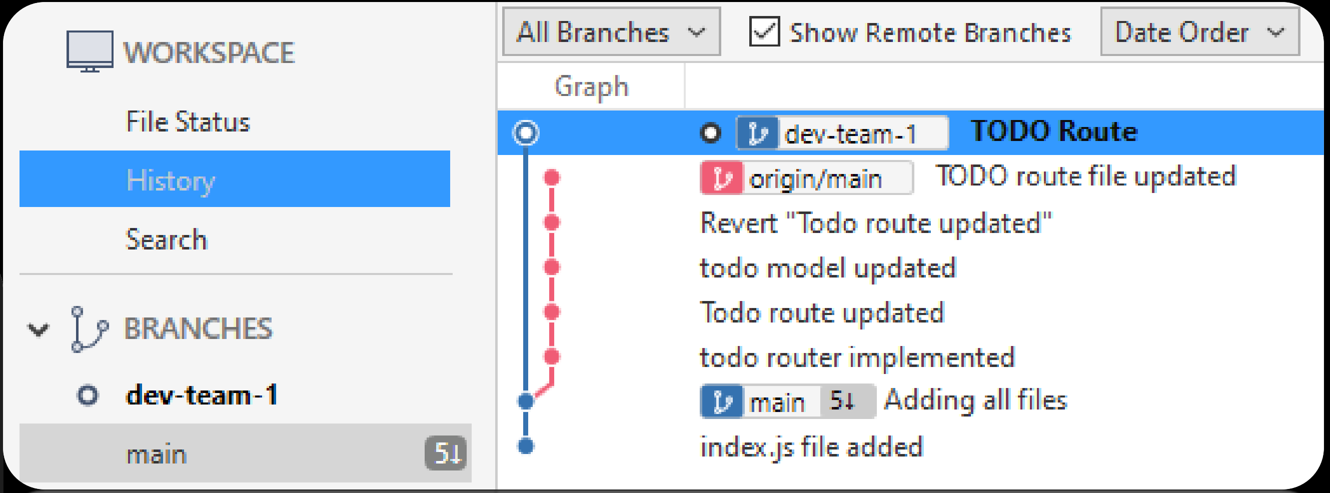
BRANCHING



```
D:\ReactJS\FAST-Demo-FA22\backend-SecP>git branch -a
* main
  remotes/origin/main

D:\ReactJS\FAST-Demo-FA22\backend-SecP>git branch dev-team-1

D:\ReactJS\FAST-Demo-FA22\backend-SecP>git branch
  dev-team-1
* main

D:\ReactJS\FAST-Demo-FA22\backend-SecP>git checkout dev-team-1
Switched to branch 'dev-team-1'

D:\ReactJS\FAST-Demo-FA22\backend-SecP>git log --oneline
3204990 (HEAD -> dev-team-1, main) Adding all files
d7355d2 index.js file added
```



```
D:\ReactJS\FAST-Demo-FA22\backend-SecP>git add .

D:\ReactJS\FAST-Demo-FA22\backend-SecP>git commit -m "TODO Route"
[dev-team-1 ad70d75] TODO Route
 1 file changed, 10 insertions(+)
 create mode 100644 routes/todos.js

D:\ReactJS\FAST-Demo-FA22\backend-SecP>git log --oneline
ad70d75 (HEAD -> dev-team-1) TODO Route
3204990 (main) Adding all files
d7355d2 index.js file added
```

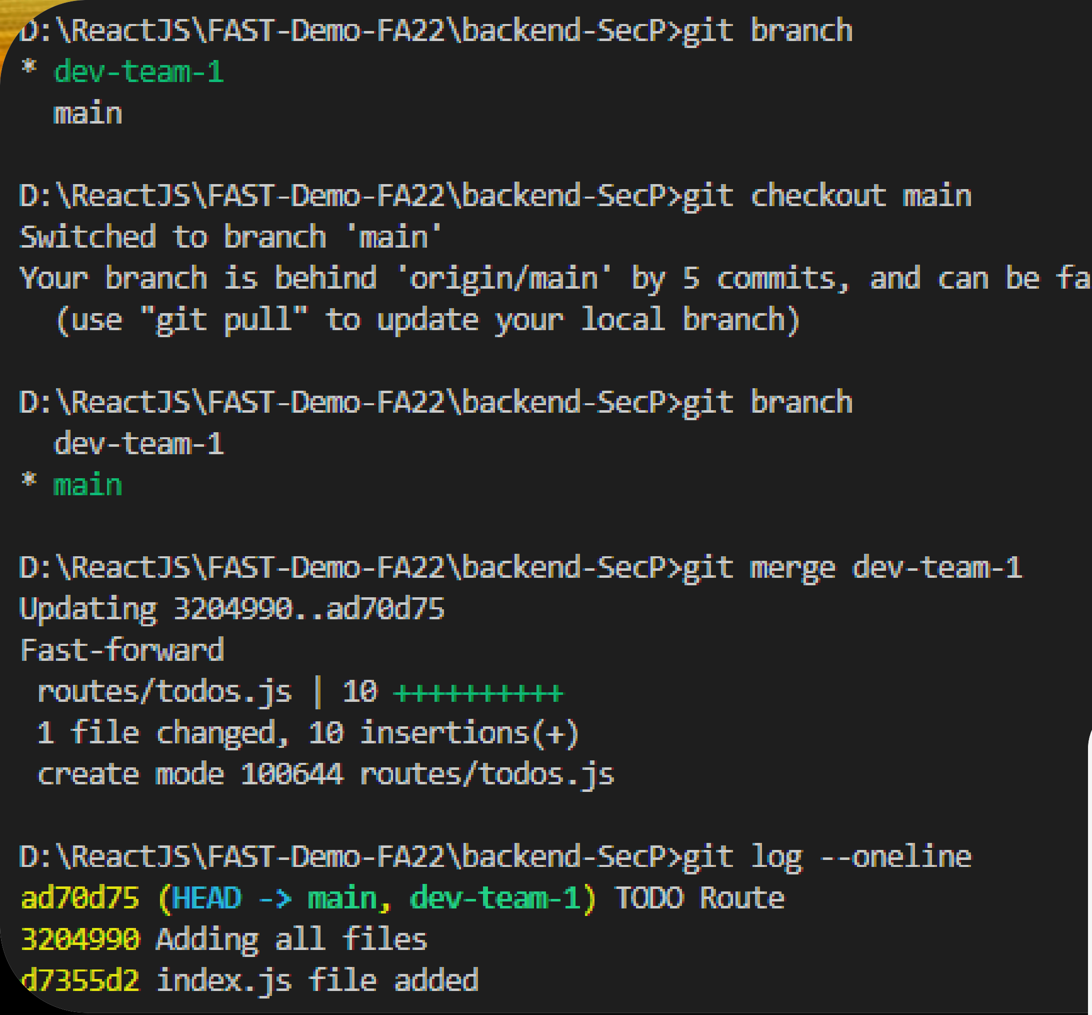


```
dev-team-1      TODO Route
origin/main     TODO route file updated
Revert "Todo route updated"
todo model updated
Todo route updated
todo router implemented
main 5↓         Adding all files
index.js file added
```

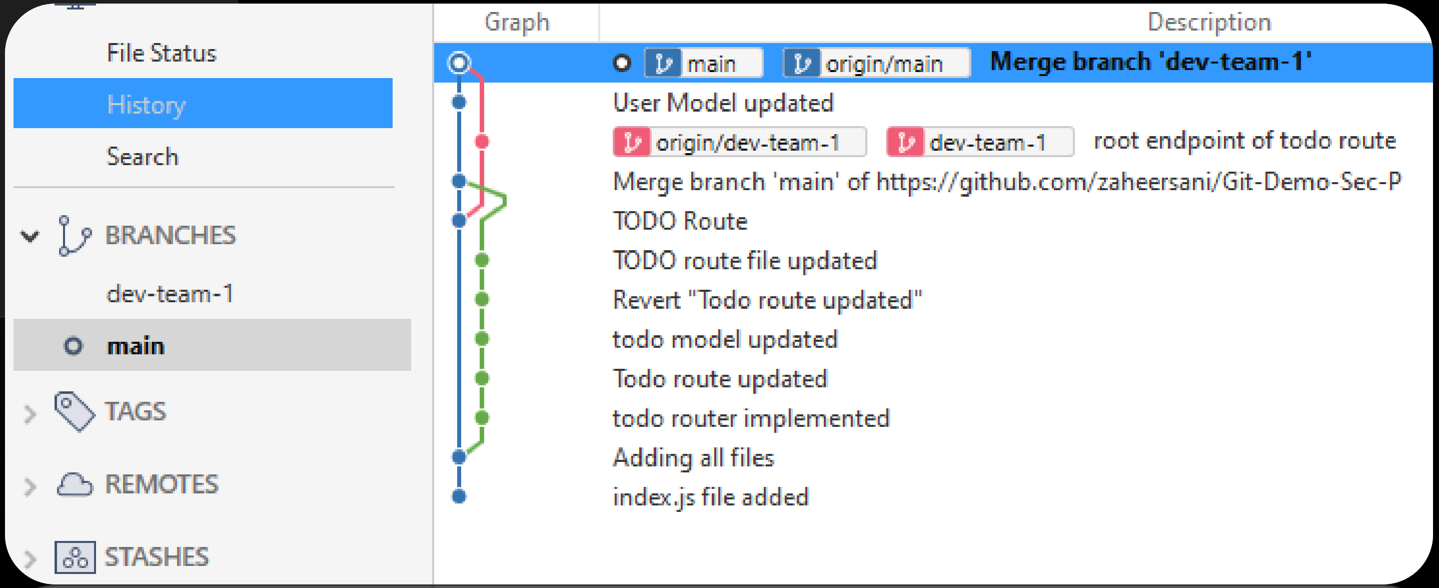
BRANCHING

- Switch to the master branch.
  - The git merge command requires you to be on the merge-receiving branch.
  - `git checkout [master-branch-name]`
- After switching to the master branch, use the following syntax to merge your changes:
  - `git merge [branch-name]`

BRANCHING



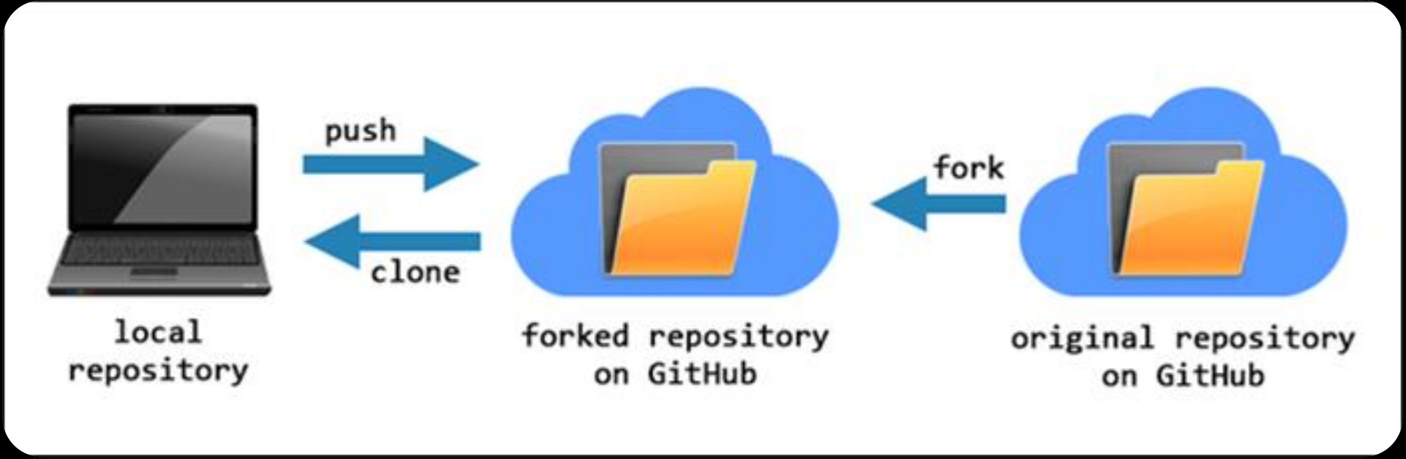
```
D:\ReactJS\FAST-Demo-FA22\backend-SecP>git branch
* dev-team-1
  main

D:\ReactJS\FAST-Demo-FA22\backend-SecP>git checkout main
Switched to branch 'main'
Your branch is behind 'origin/main' by 5 commits, and can be fast-forwarded.
  (use "git pull" to update your local branch)

D:\ReactJS\FAST-Demo-FA22\backend-SecP>git branch
  dev-team-1
* main

D:\ReactJS\FAST-Demo-FA22\backend-SecP>git merge dev-team-1
Updating 3204990..ad70d75
Fast-forward
 routes/todos.js | 10 ++++++++++
 1 file changed, 10 insertions(+)
 create mode 100644 routes/todos.js

D:\ReactJS\FAST-Demo-FA22\backend-SecP>git log --oneline
ad70d75 (HEAD -> main, dev-team-1) TODO Route
3204990 Adding all files
d7355d2 index.js file added
```



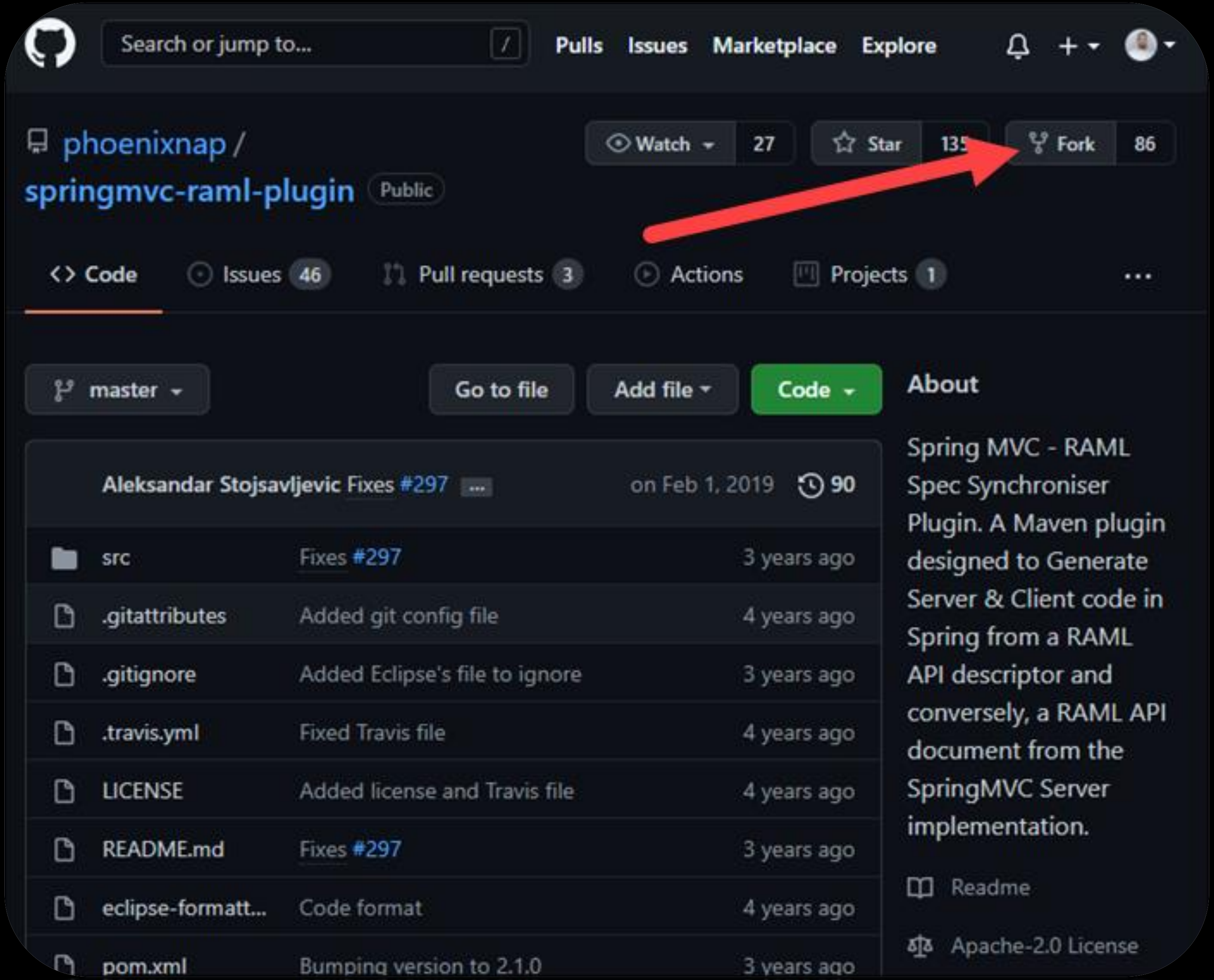
```
main  origin/main  Merge branch 'dev-team-1'
User Model updated
origin/dev-team-1  dev-team-1  root endpoint of todo route
Merge branch 'main' of https://github.com/zaheersani/Git-Demo-Sec-P
TODO Route
TODO route file updated
Revert "Todo route updated"
todo model updated
Todo route updated
todo router implemented
Adding all files
index.js file added
```

FORKING

- A fork is a complete copy of an existing repository that allows you to make changes and experiment without affecting the original project.
- Forking is a way for someone to propose changes to an existing project, or it can be a starting point for a project of your own if the code is open source.
- If you want to propose a change or a bug fix for a project, you can fork a repository, make the fix, and make a pull request to the project owner.



Visit the repository page on github and click the fork option.

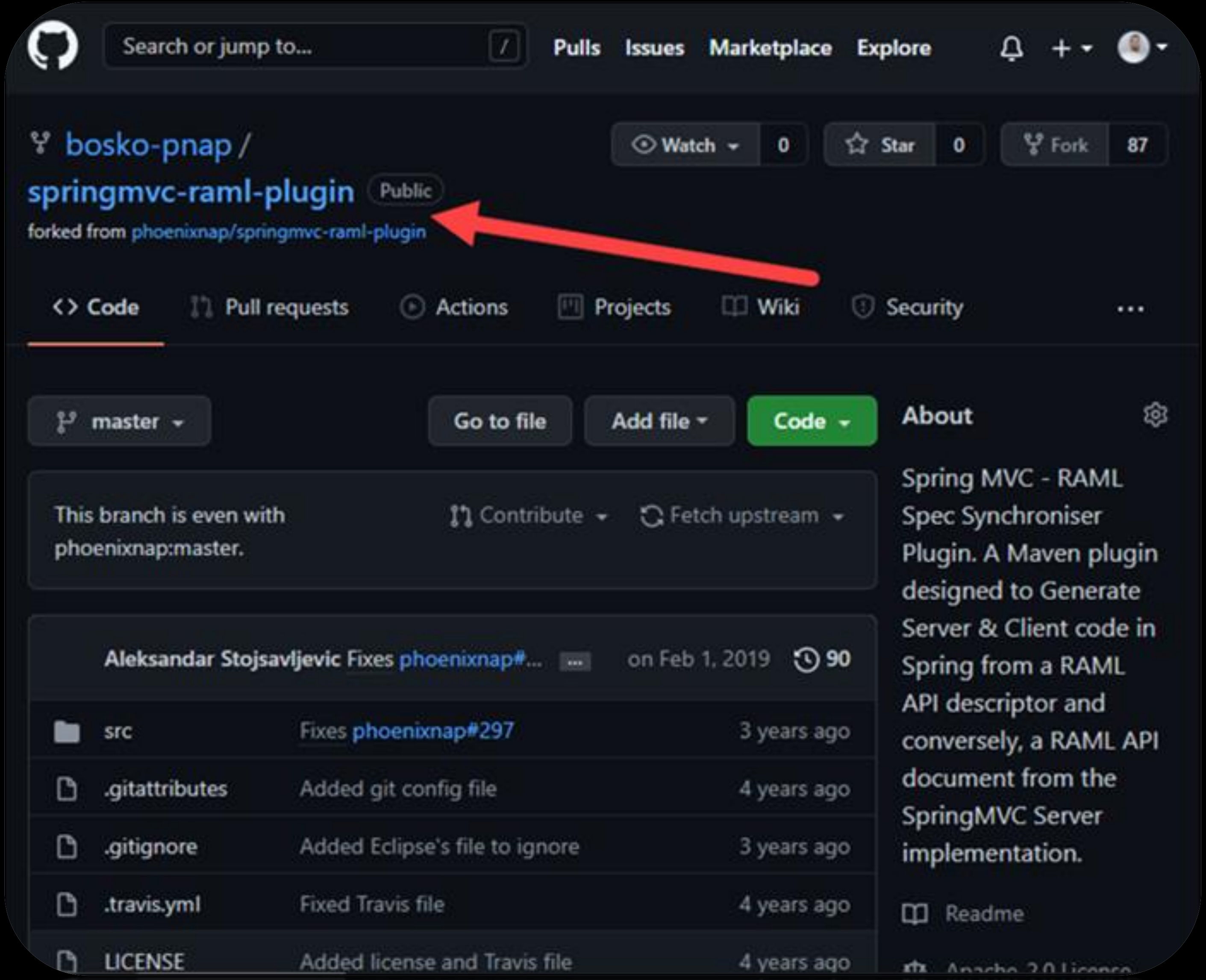


Aleksandar Stojsavljevic Fixes #297 ... on Feb 1, 2019 90

src	Fixes #297	3 years ago
.gitattributes	Added git config file	4 years ago
.gitignore	Added Eclipse's file to ignore	3 years ago
.travis.yml	Fixed Travis file	4 years ago
LICENSE	Added license and Travis file	4 years ago
README.md	Fixes #297	3 years ago
eclipse-formatt...	Code format	4 years ago
pom.xml	Bumping version to 2.1.0	3 years ago

Spring MVC - RAML Spec Synchroniser Plugin. A Maven plugin designed to Generate Server & Client code in Spring from a RAML API descriptor and conversely, a RAML API document from the SpringMVC Server implementation.

Wait for the forking process to complete. When it finishes, you will have a copy of the repository on your GitHub account.



This branch is even with phoenixnap:master.

Contribute

Fetch upstream

Aleksandar Stojsavljevic Fixes phoenixnap#... on Feb 1, 2019 90

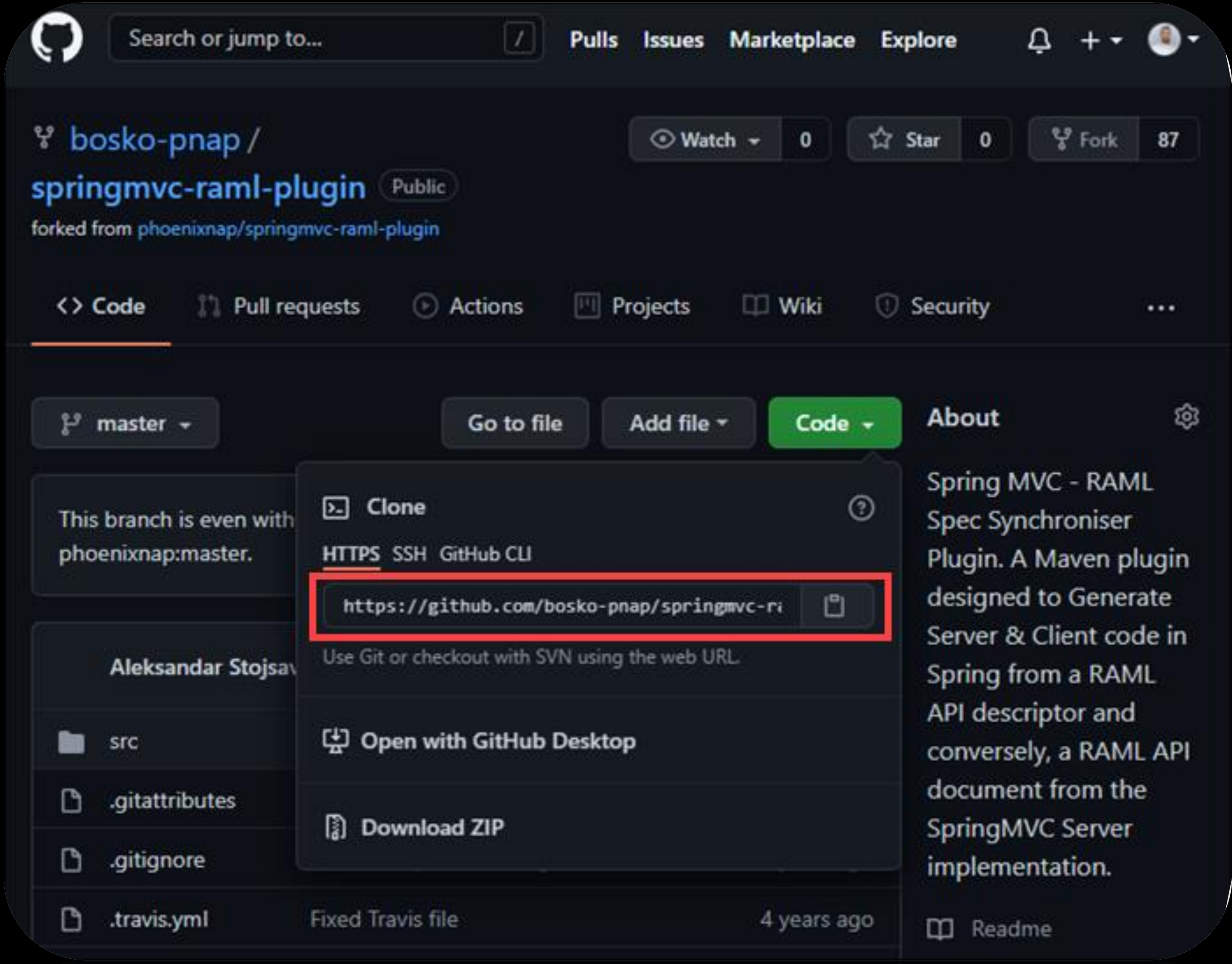
src	Fixes phoenixnap#297	3 years ago
.gitattributes	Added git config file	4 years ago
.gitignore	Added Eclipse's file to ignore	3 years ago
.travis.yml	Fixed Travis file	4 years ago
LICENSE	Added license and Travis file	4 years ago

Spring MVC - RAML Spec Synchroniser Plugin. A Maven plugin designed to Generate Server & Client code in Spring from a RAML API descriptor and conversely, a RAML API document from the SpringMVC Server implementation.

Readme

Apache 2.0 License

- The next step is to take the repository URL from the Code section and clone the repository to your local machine.
- Clone the repository using the following syntax:
  - `git clone [repository URL]`



bosko-pnap / springmvc-raml-plugin Public

forked from phoenixnap/springmvc-raml-plugin

This branch is even with phoenixnap:master.

https://github.com/bosko-pnap/springmvc-ra

Use Git or checkout with SVN using the web URL.

About

Spring MVC - RAML Spec Synchroniser Plugin. A Maven plugin designed to Generate Server & Client code in Spring from a RAML API descriptor and conversely, a RAML API document from the SpringMVC Server implementation.

.travis.yml	Fixed Travis file	4 years ago

REFERENCES

- Git Download Link
    - https://gitforwindows.org/
- Video Tutorial (Git + React Native by Code Synergy)
    - https://www.youtube.com/watch?v=R79NZHD1A4I
- Git Docs
    - https://git-scm.com/docs/git
- Git Official Videos
    - https://git-scm.com/videos
- Advance Git Commands [Forking, Branching, Merging, Rebasing etc]
    - https://phoenixnap.com/kb/how-git-works
- 20 Git Commands with Examples
    - https://dzone.com/articles/top-20-git-commands-with-examples