Unit - 6

https://github.com/

Objective

- Git and GitHub
- Difference between Git and GitHub
- Version Controlling
- Most popular Git commands
- Branching in Git
    - Connecting local with remote repo
- Hands-on GitHub

Git

- **Git** is software for tracking changes.
- Handle any set of files
- Used for work among programmers collaboratively
- Developing source code during software development.

GitHub

- Provider of Internet hosting for software development and version control.
- Offers the distributed version control
- Supports source code management (SCM)

Difference between git and GitHub

| Git	GitHub
Installed locally	Hosted in cloud
First released in 2005	Company launched in 2008
Maintained by The Linux Foundation	Purchase in 2018 by Microsoft
Focused on version control and code sharing	Focused on centralized source code hosting
Primarily a command-line tool	Administered through the web
No user management features	Built-in user management
Minimal external tool configuration features	Active marketplace for tool integration
Competes with Mercurial, Subversion, IBM, Rational Team Concert and ClearCase	Competes with Atlassian Bitbucket and GitLab
Open source licensed	Includes a free tier and pay-for-use tiers

Version Controlling

- **Version control** (also known as **revision control**, **source control**, or **source code management**)

- Responsible for managing changes to computer programs.

- Handle documents, large web sites, or other collections of information.

Distributed Version Control

Distributed version control system

Server
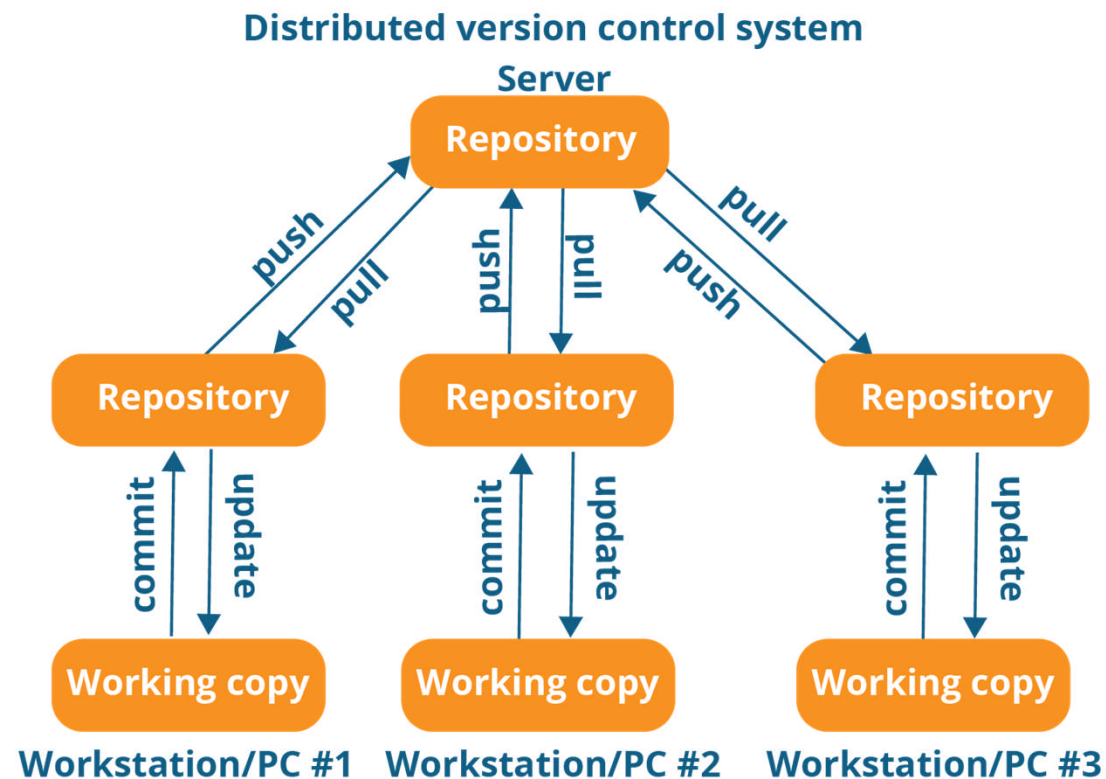
Distributed control

Centralized Version Control
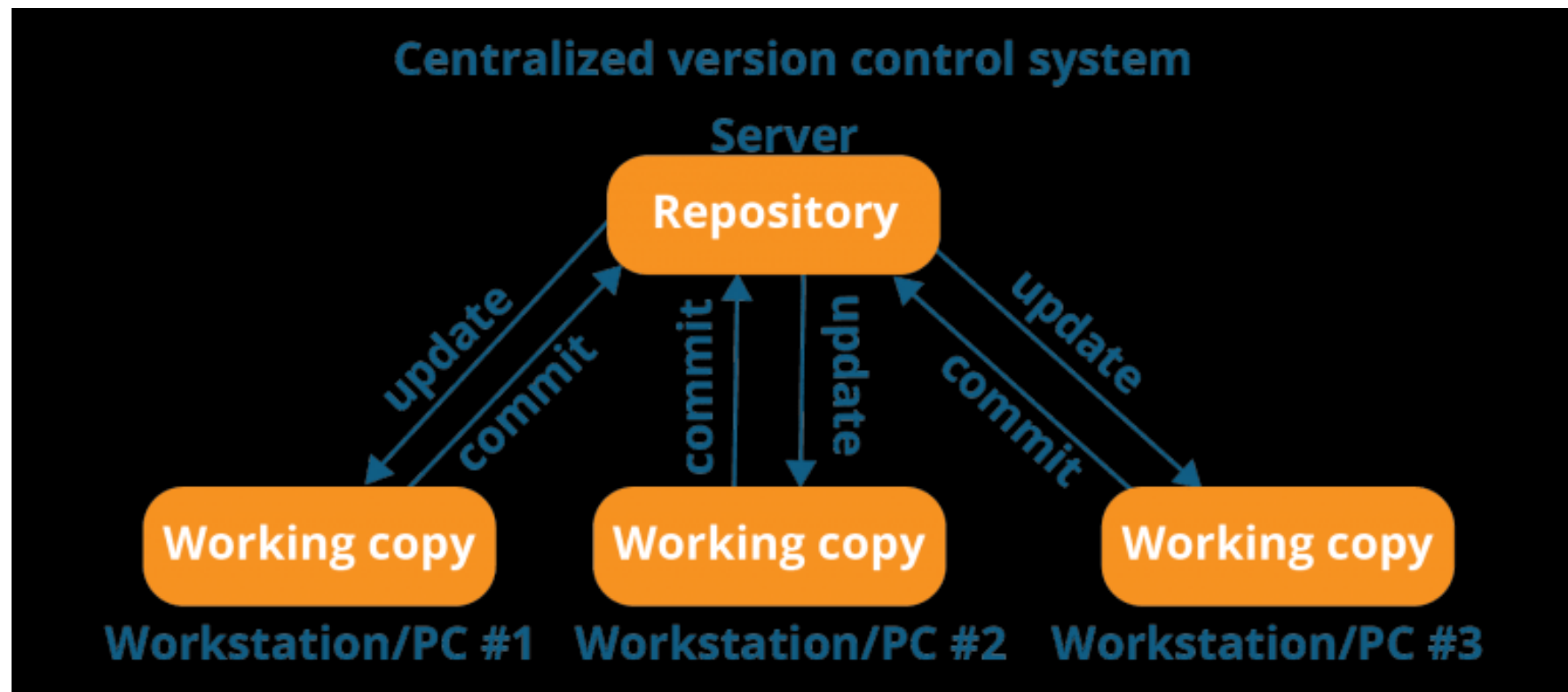
Centralised control

1. What is full form of VCS?

- Version Configuration System
- Version consolidated System
- Version configuration Solution
- Version Control System

1. What is full form of VCS?

- Version Configuration System
- Version consolidated System
- Version configuration Solution
- Version Control System

2. Git is _____ Version Control system

A. Distributed
B. Centralized

2. Git is _____ Version Control system
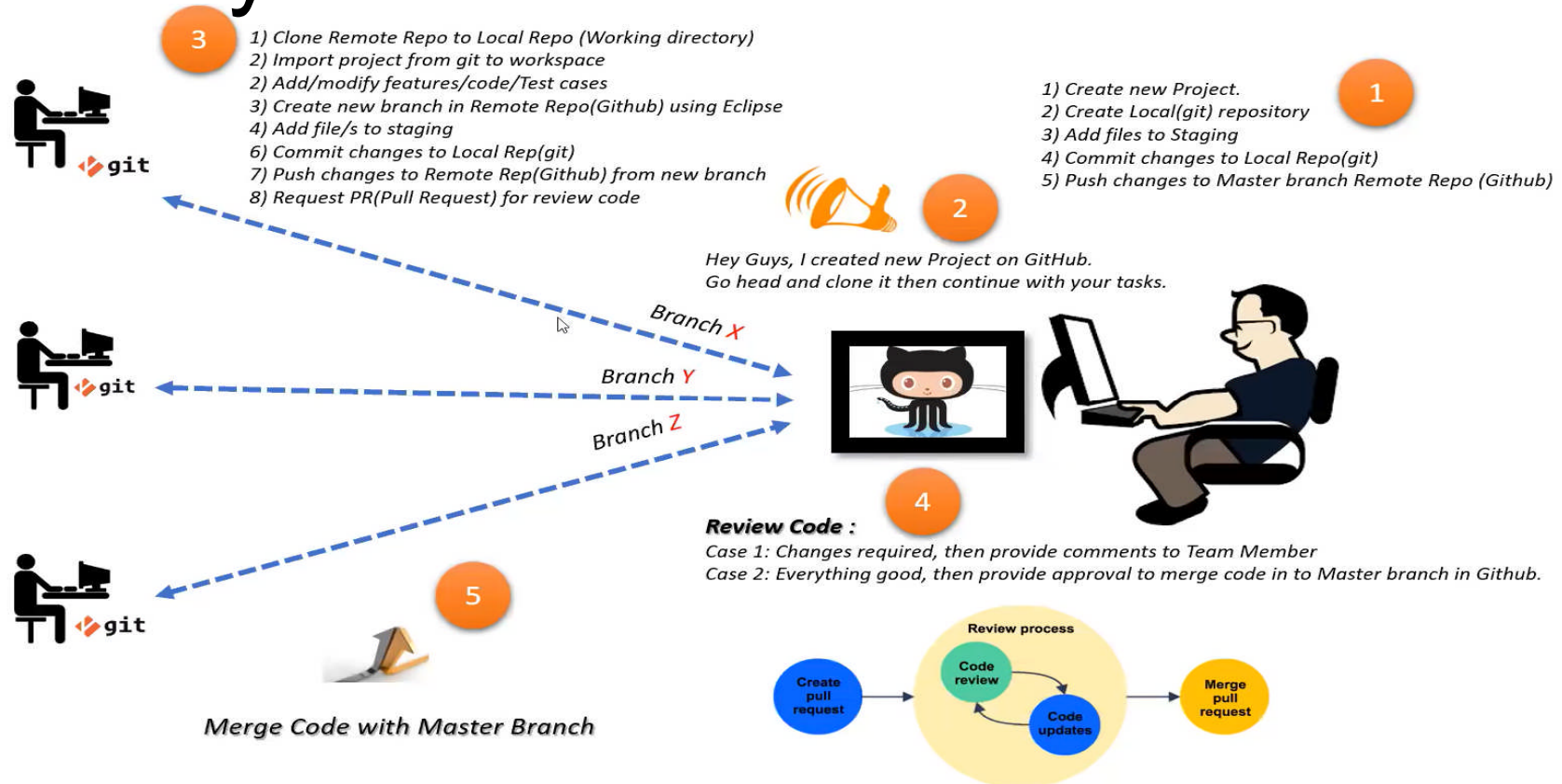
A. Distributed

B. Centralized

3. Match the following:

A. Git   Cloud Service

B. Github   Software to track versions

3. Match the following:

A.   Git   Cloud Service

B.   Github   Software to track versions

Code-Cycle

3
1) Clone Remote Repo to Local Repo (Working directory)
2) Import project from git to workspace
2) Add/modify features/code/Test cases
3) Create new branch in Remote Repo(Github) using Eclipse
4) Add file/s to staging
6) Commit changes to Local Rep(git)
7) Push changes to Remote Rep(Github) from new branch
8) Request PR(Pull Request) for review code

1
1) Create new Project.
2) Create Local(git) repository
3) Add files to Staging
4) Commit changes to Local Repo(git)
5) Push changes to Master branch Remote Repo (Github)

2

Hey Guys, I created new Project on GitHub.
Go head and clone it then continue with your tasks.

Branch X

Branch Y

Branch Z

4

Review Code :
Case 1: Changes required, then provide comments to Team Member
Case 2: Everything good, then provide approval to merge code in to Master branch in Github.

Review process

Create pull request → Code review ⟷ Code updates → Merge pull request

5

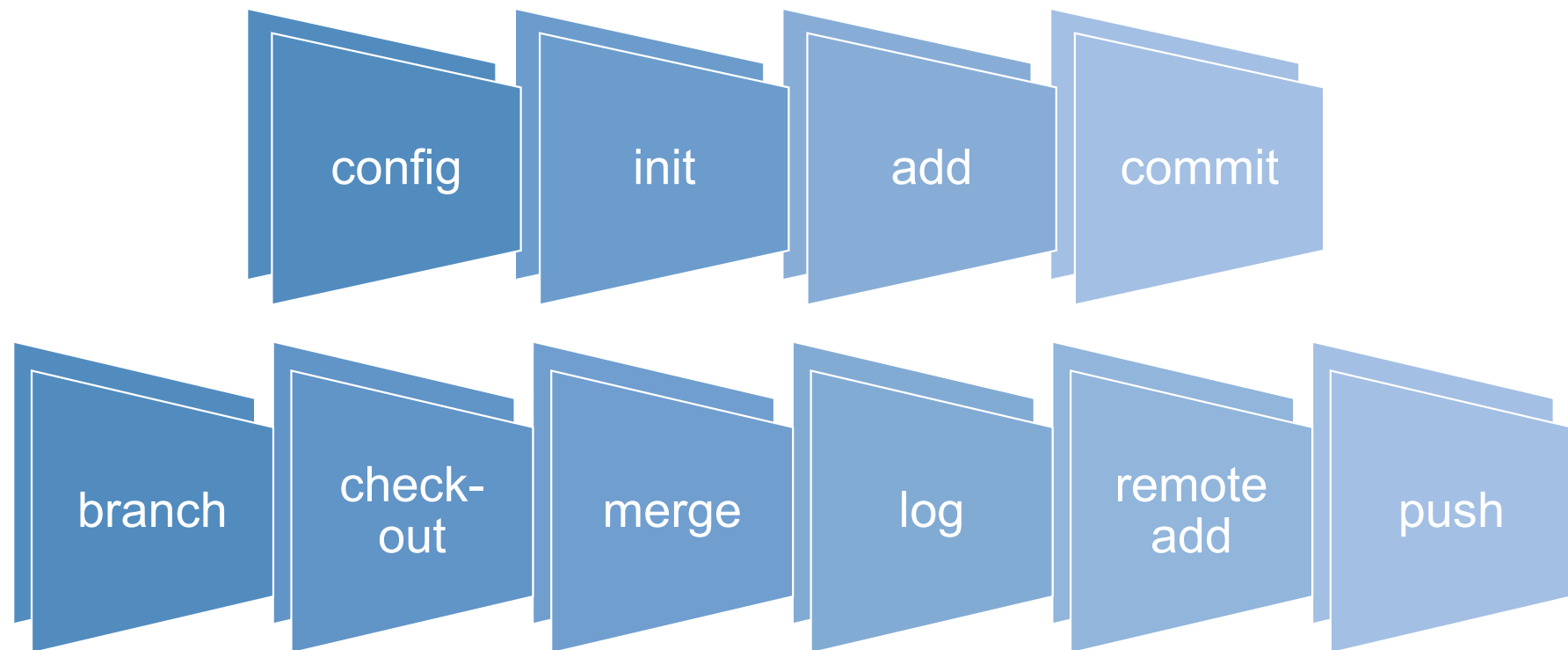**Merge Code with Master Branch**

Flowgraph

Git CMD Vs Git bash

•Git CMD just like Windows CMD.
• Can call all Git features

•Git Bash emulates bash environment
• Also support Unix commands

Git-bash

Git commands

config

init

add

commit
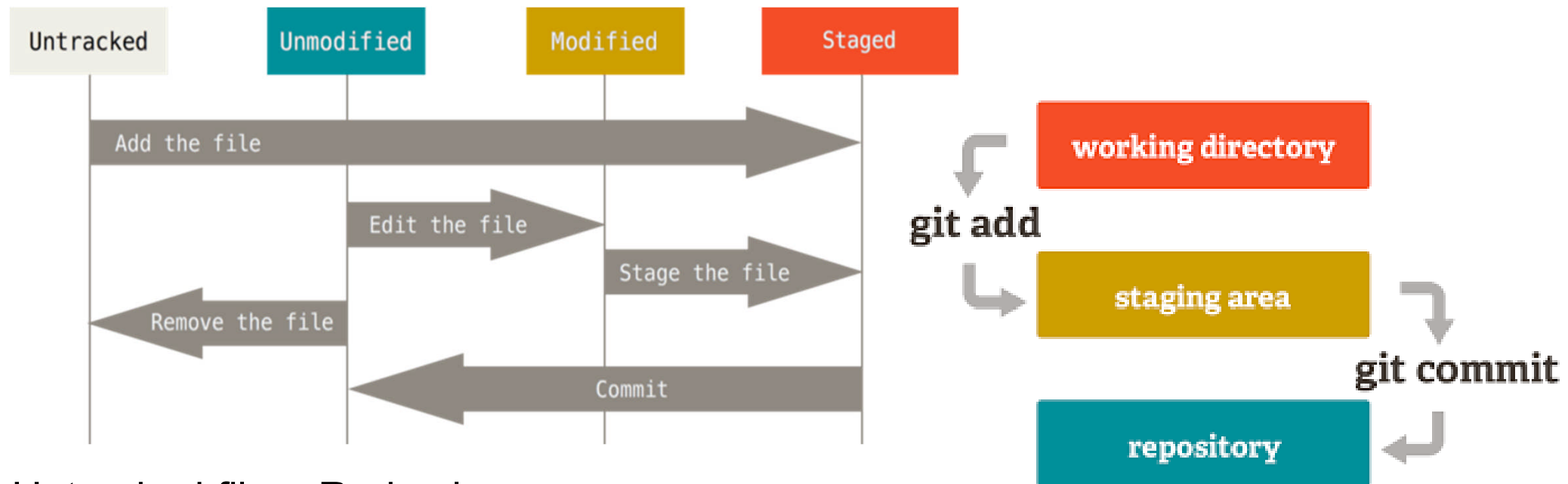
branch

check-out

merge

log

remote add

push

Processing Stages

- **Untracked :** Files which are newly created in working directory and git does not aware of these files.

- **Staged:** Files which are added to staging area. These files are ready to commit.

- **Committed:** Files which is committed and placed in local repository/ Committed State.

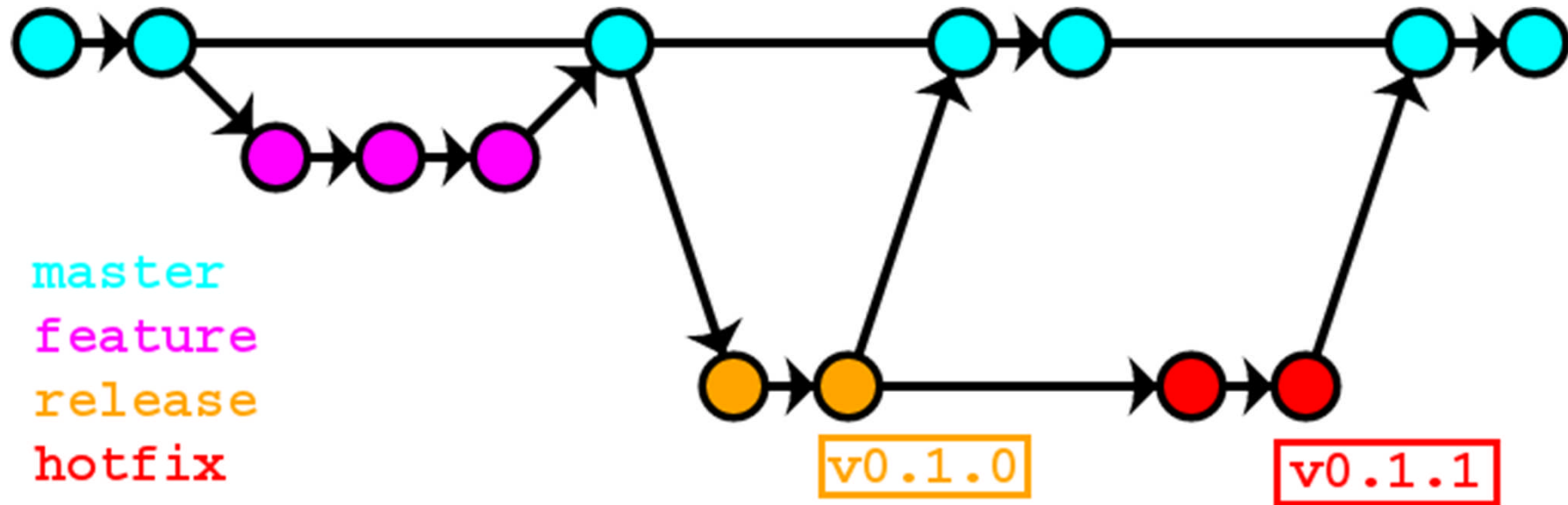- **Modified:** File which is already tracked by git. But is modified in working directory .

Git-flow

Untracked file – Red color
Staging Area – Green Color

Staging Tree

Branching

master
feature
release
hotfix

v0.1.0   v0.1.1

Master and branch

Create account on GitHub

https://github.com/join?ref_cta=Sign+up&ref_loc=header+logged+out&ref_page=%2F&source=header-home

References

1. https://en.wikipedia.org/wiki/Git
2. Jason Long, CC BY 3.0 <https://creativecommons.org/licenses/by/3.0>, via Wikimedia Commons
3. https://en.wikipedia.org/wiki/GitHub
4. GitHub, Public domain, via Wikimedia Commons
5. https://www.geeksforgeeks.org/difference-between-git-and-github/
6. https://en.wikipedia.org/wiki/Version_control
7. Qeef, CC BY-SA 4.0 <https://creativecommons.org/licenses/by-sa/4.0>, via Wikimedia Commons
8. Scott Chacon and Ben Straub , *Pro Git , Second edition , Apress , 2014.*
9. Git and GitHub for beginners - Crash course by freeCodeCamp.org - https://www.youtube.com/watch?v=RGOj5yH7evk&t=408s