Git is source code management

There are two SCM system

1 CVCS centralized

2. DVCS distributed version control system

Github is a service,

Git is a software which is use on local

**Git Process**: There are three logical area are created in GIT directory. Working space, staging area, local repository first code will be sent(commit) to the Staging area then code will be finalized and snapshot will be created in local repository git uses HSA1 function for security purpose

**Stages**:

1. Workspace
2. Staging
3. Local

**Data Flow:**

Workspace Staging Local Remote Location

**Commands**:

**Git** **init**: to initialize git

**Git add –a**: to add changes into staging area ( --a indicates to all data we can also use dot(.)instead of --a

**Commit**: send data from working space to local repo. Commit id is unique 40 alphanumeric long

**Git Push origin master**: send data from local repo to remote location

Git checkout –f: use to restore all files till last commit

**Git Pull origin {branch name}**: get data from remote location to local repo

**Git remote add origin {Remote Path}**: to add remote location Ex: Github, Gitlab

**Authentication Note**: to push data from local to remote Repository we have to make an authentication for that we have to generate an authentication(SSH key) by gitbash and save that key into our remote account(Github, Gitlab)

**Git restore {path}:** to discard changes all changes will be discard on specific path

**Git diff**: compare staging area with working directory

**Git diff –staged**: compare last commit with staging area

**Git checkout –b {branch name}:** create new branch

**Git checkout {branch name}**: switch branch

**Git push origin {branch name}:** to push branch to server

**Git merge {branch name}:** to merge branch

**Git branch –merged**: to check which branch has been merged

**Git branch –no—merged**: which branch has not been merged

**Git branch –d {branch name}:** delete branch

**Git log**: to check log only

**Git log –p**: to check log with changes

**Git log –stat**: to check count changes

**Add**: When we send data (commit) from working space to Staging stage this process called add

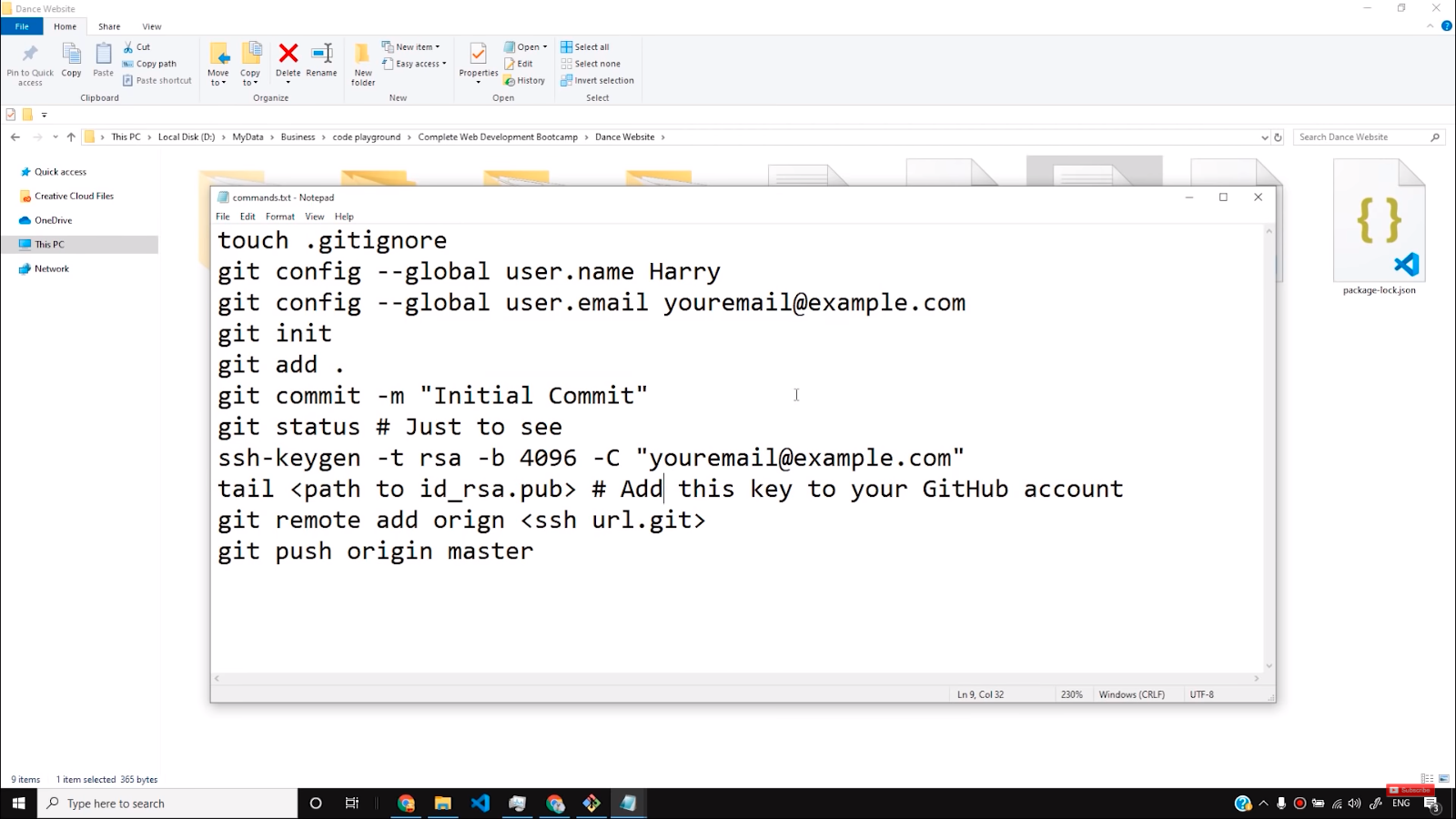
**Staging:** finalize data orfinally which data has to commit is decide on staging process.

**Branch**: create a separate copy of existing branch

**Merge**: combine two branches data into one

**Git Revert**: this works on publicly (Github)

**Git Reset**: works on private files (local Repo)



* *GroupId* – a unique base name of the company or group that created the project
* *ArtifactId* – a unique name of the project
* *Version* – a version of the project
* *Packaging* – a packaging method (e.g. *WAR*/*JAR*/*ZIP*)