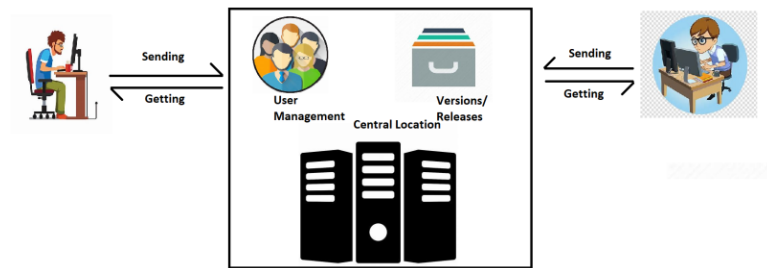MENU ☰

JANUARY 31, 2023

DevOps Classroomnotes 31/Jan/2023

Version Control Systems – Git an Introduction

- Version control system is a software system that allows
    - developers to store the code in common location
    - retains history of every change
    - allows to work for multiple versios/releases/customers

- Generally Version Control System will have two software components
    - Client Software (Used by Developers of code)
    - Server Software (Used to host code/Administration)

- Examples:

    - CVS
    - Visual Source Safe
    - Subversion
    - Perforce
    - Mercurial
    - IBM Clear Case
    - Team Foundation version control
    - Git

- Evolution of version control systems

    - Single System
    - Single System => Multi User
    - Client Server
   - Distributed

Git

- Git is a Distributed Version control system
- Terms:

    - Commit: This is a changeset
    - Local Repository: This is the local code
    - Remote Repository: This is the central code
    - Working Directory: This is the folder/directory where we make changes
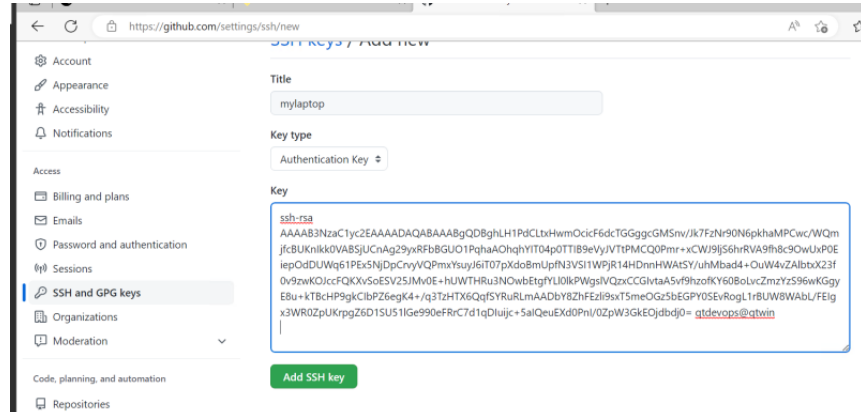
- Git Workflow for now

Four Areas of Git
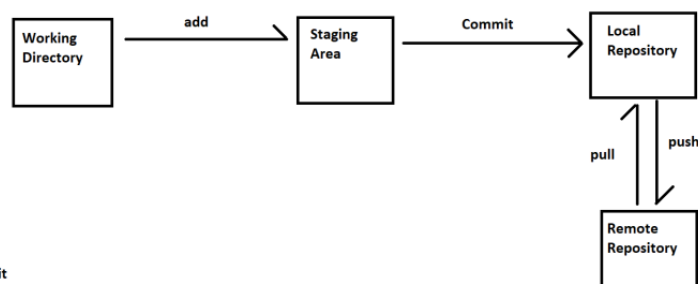
- Git Cheatsheet Refer Here
- Git can be connected with Server using two major ways
  - username and token/password
  - ssh key

- Create a ssh-key
  - Launch Windows Powershell (Terminal)
  - Execute ssh-keygen

```
PS C:\Users\qtdevops> ssh-keygen
Generating public/private rsa key pair.
Enter file in which to save the key (C:\Users\qtdevops/.ssh/id_rsa):
Created directory 'C:\Users\qtdevops/.ssh'.
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in C:\Users\qtdevops/.ssh/id_rsa.
Your public key has been saved in C:\Users\qtdevops/.ssh/id_rsa.pub.
The key fingerprint is:
SHA256:MJDpHsLogZrrcmTrgtZbzid+uxRCVCr7UdPz5UxecFw qtdevops@qtwin
The key's randomart image is:
+---[RSA 3072]----+
|     .+..    ...E|
|     +.. .     o.|
|.o o ooo o   o . |
|o.o * .o. o * .  |
|o..+ + .S  . +   |
|o.o o o .        |
|.+.. o .         |
|=.o.+..o         |
|+=..o++oo        |
```
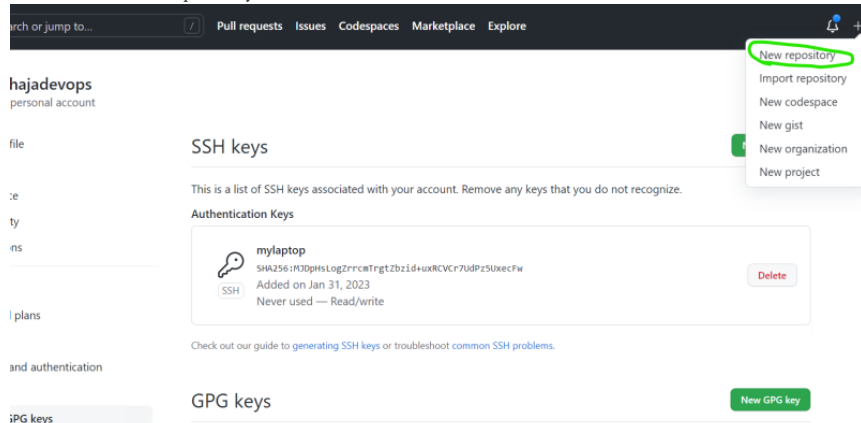
- Navigate to settings -> SSH and GPG keys and click on new ssh-key
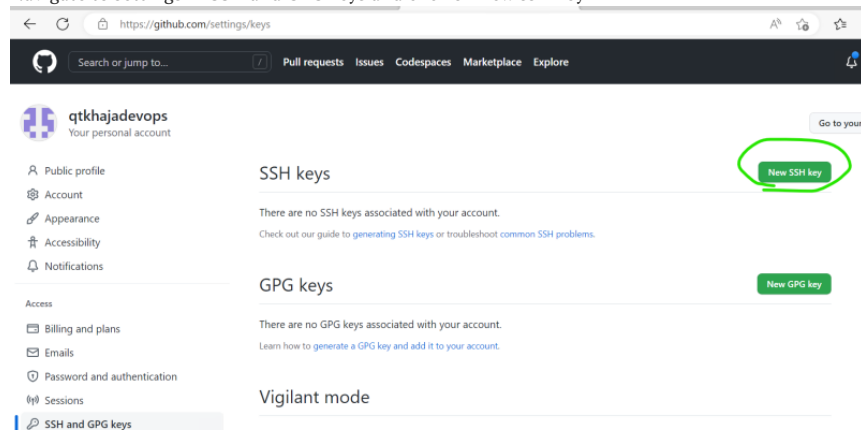
- Copy the contents of ~\.ssh\id_rsa.pub and paste as shown below and add ssh key

- Now create a new repository

- If you dont have a copy of remote repo i.e. local repo we perform an operation which is called as clone.
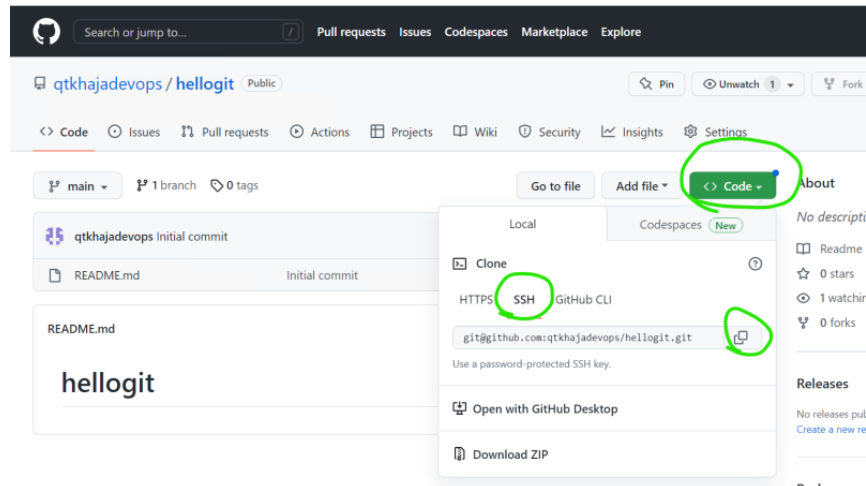
- Copy the url for clone
- Navigate to any folder in your system and execute `git clone <url>`

```
PS C:\Users\qtdevops> cd D:
PS D:\> git clone git@github.com:qtkhajadevops/hellogit.git
Cloning into 'hellogit'...
The authenticity of host 'github.com (140.82.113.3)' can't be established.
ED25519 key fingerprint is SHA256:+DiY3wvvV6TuJJhbpZisF/zLDA0zPMSvHdkr4UvCOq
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added 'github.com' (ED25519) to the list of known hosts
remote: Enumerating objects: 3, done.
remote: Counting objects: 100% (3/3), done.
remote: Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
Receiving objects: 100% (3/3), done.
PS D:\>
```
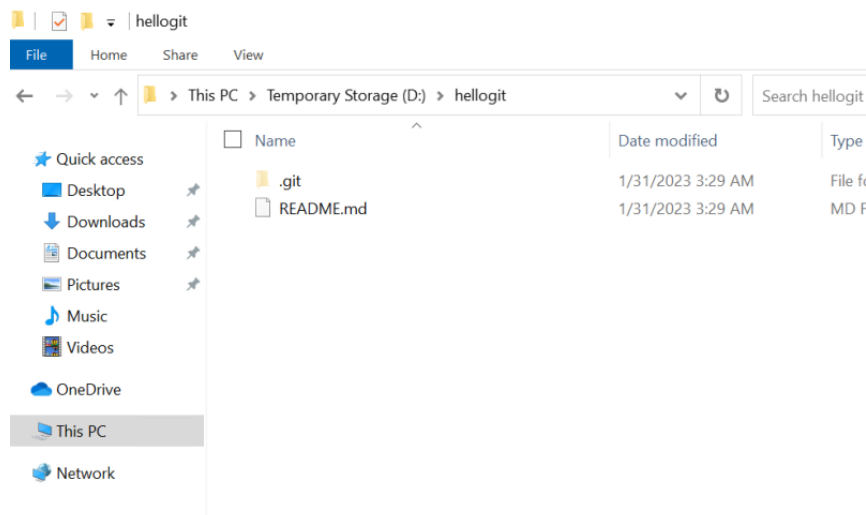
- Now lets make our first change and try executing the workflow

```
PS D:\hellogit> git add '.\New Text Document.txt'
PS D:\hellogit> git commit -m "this is my first commit"
Author identity unknown

*** Please tell me who you are.

Run

  git config --global user.email "you@example.com"
  git config --global user.name "Your Name"

to set your account's default identity.
Omit --global to set the identity only in this repository.

fatal: unable to auto-detect email address (got 'qtdevops@qtwin.(none)')
PS D:\hellogit>  git config --global user.email "qtkhajadevops@gmail.com"
PS D:\hellogit> git config --global user.name "qtkhaja"
PS D:\hellogit> git commit -m "this is my first commit"
[main e394a0d] this is my first commit
 1 file changed, 0 insertions(+), 0 deletions(-)
 create mode 100644 New Text Document.txt
```
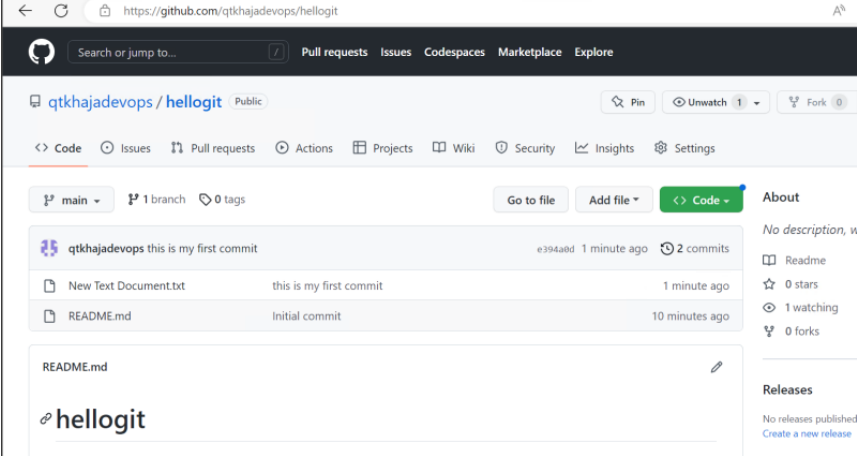
```
PS D:\hellogit> git push
Enumerating objects: 4, done.
Counting objects: 100% (4/4), done.
Delta compression using up to 2 threads
Compressing objects: 100% (2/2), done.
Writing objects: 100% (3/3), 294 bytes | 294.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
To github.com:qtkhajadevops/hellogit.git
   0a6d062..e394a0d  main -> main
PS D:\hellogit>
```


-

Leave a Reply

Enter your comment here...

This site uses Akismet to reduce spam. Learn how your comment data is processed.

About continuous learner

devops & cloud enthusiastic learner

VIEW ALL POSTS

◄ PREVIOUS POST

Azure Classroomnotes 31/Jan/2023

NEXT POST

AWS Classroomnotes 31/Jan/2023