Subject Name: **Source Code Management**
Subject Code: **CS181**
Cluster: **BETA**
Department: **DCSE**

SUBMITTED BY:

Aahan Walia
2110990003
G1

SUBMITTED TO:

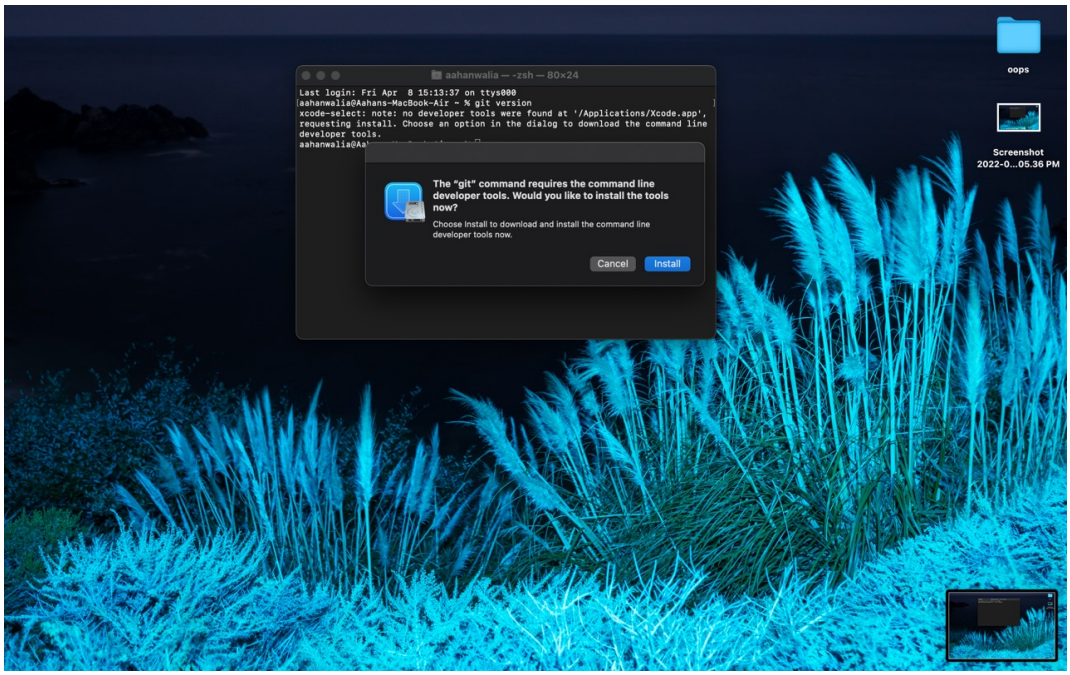
DR. MONIT KAPOOR

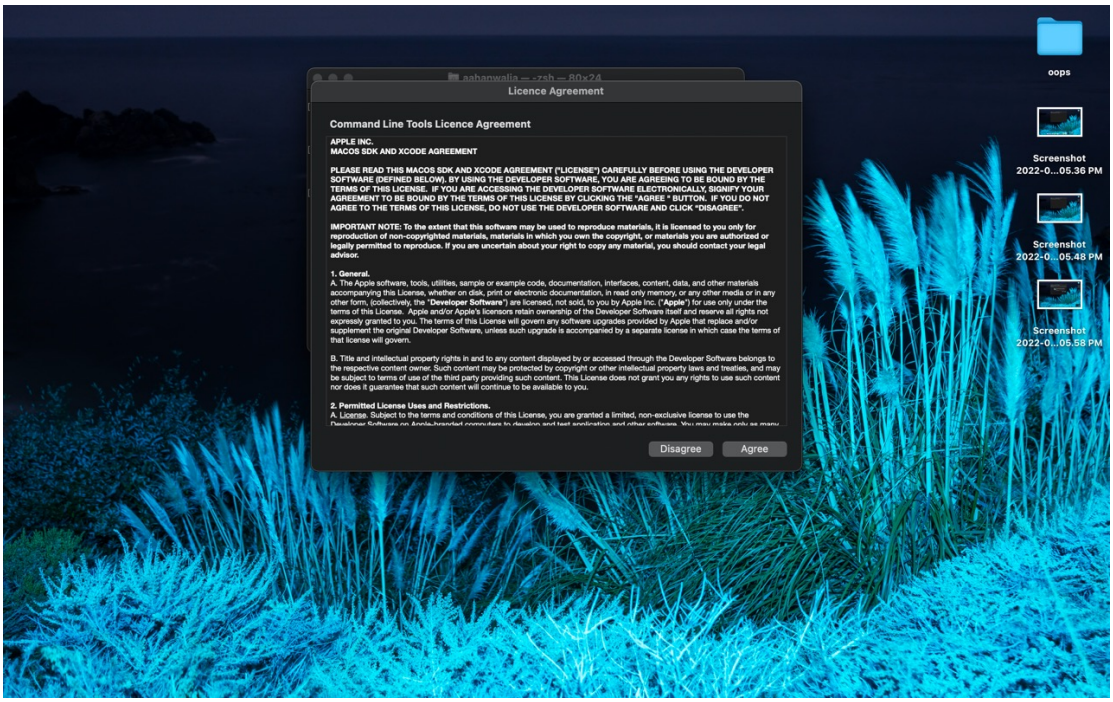
Aim: setting up Git Client

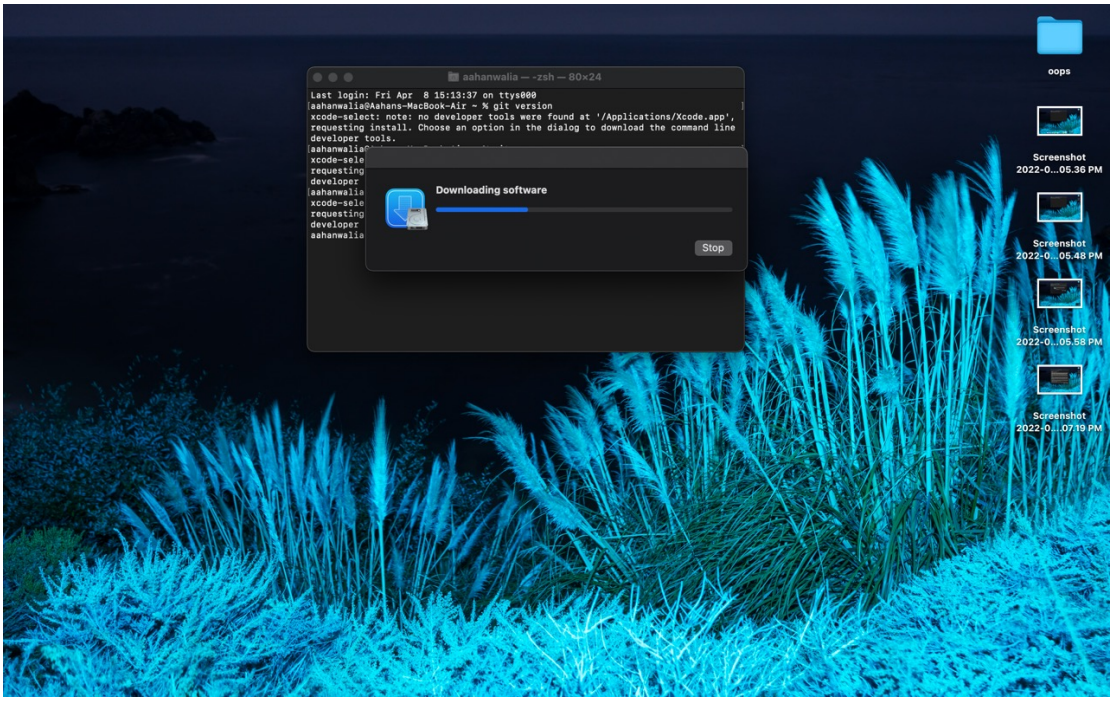
Steps to Install Git on mac

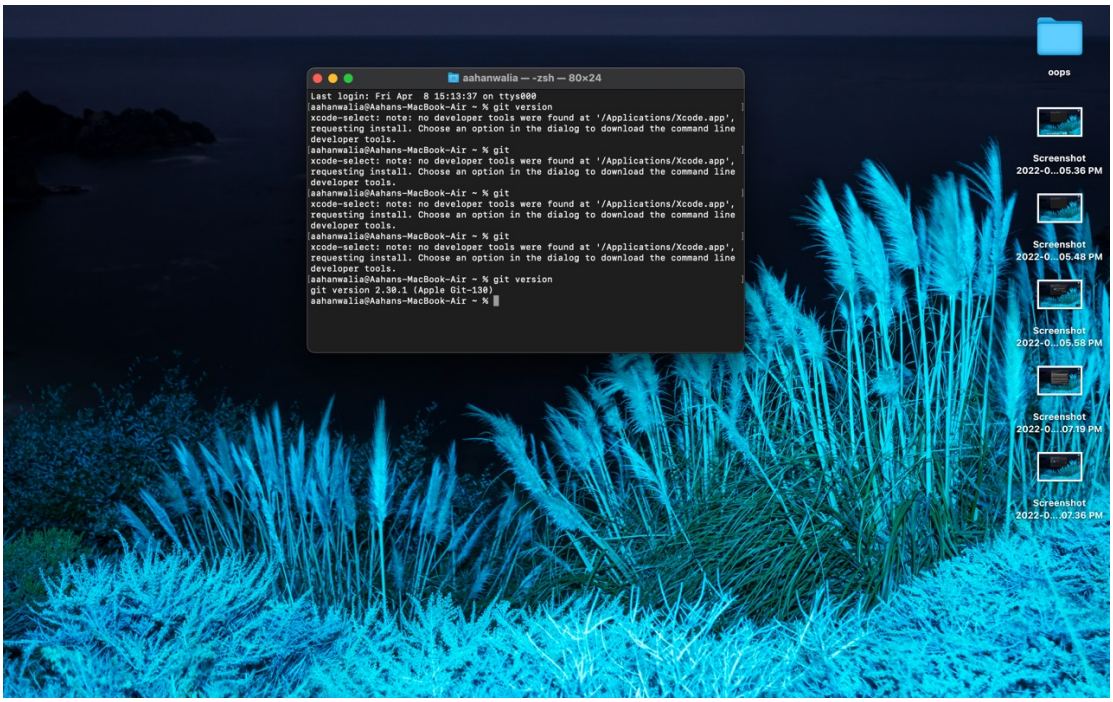
1. Open terminal in mac and type git.

2. Add install the software.



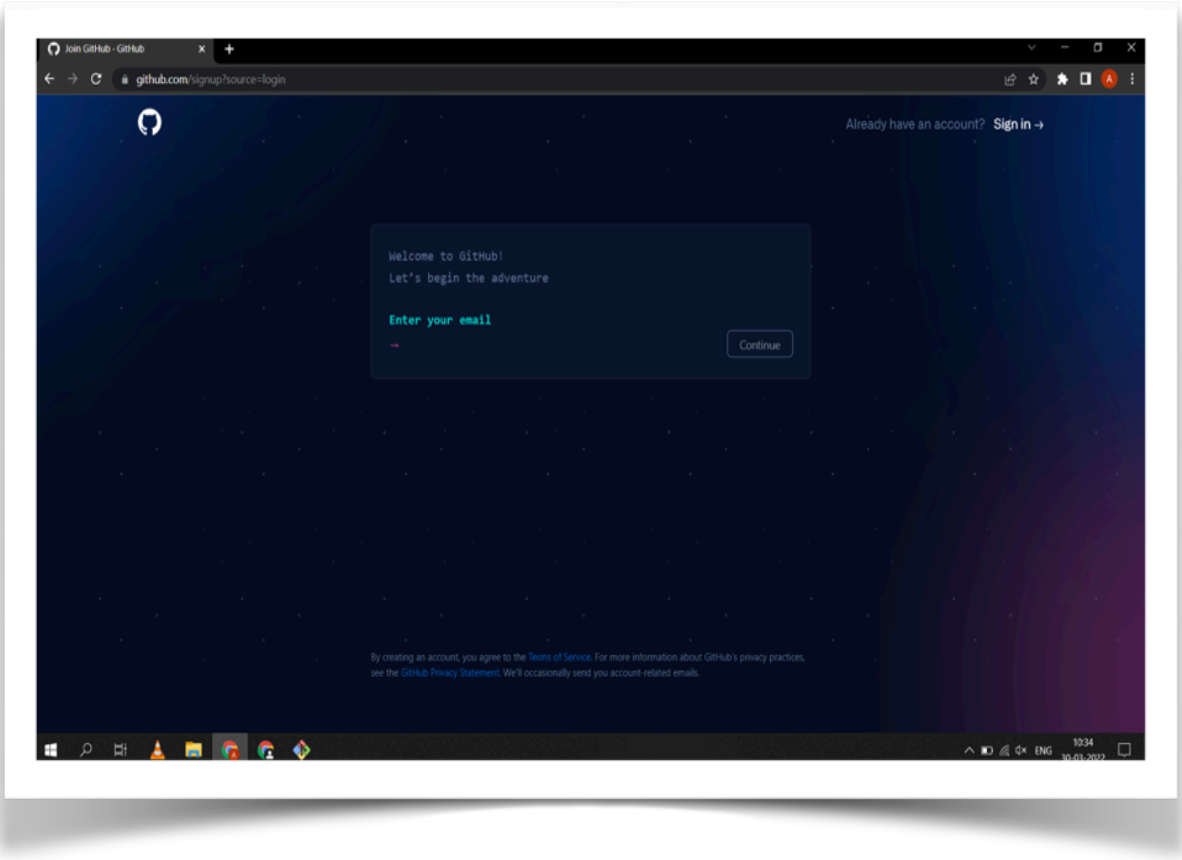

aahanwalia — -zsh — 80×24
Licence Agreement
Command Line Tools Licence Agreement
APPLE INC.
MACOS SDK AND XCODE AGREEMENT
PLEASE READ THIS MACOS SDK AND XCODE AGREEMENT ("LICENSE") CAREFULLY BEFORE USING THE DEVELOPER SOFTWARE (DEFINED BELOW). BY USING THE DEVELOPER SOFTWARE, YOU ARE AGREEING TO BE BOUND BY THE TERMS OF THIS LICENSE. IF YOU ARE ACCESSING THE DEVELOPER SOFTWARE ELECTRONICALLY, SIGNIFY YOUR AGREEMENT TO BE BOUND BY THE TERMS OF THIS LICENSE BY CLICKING THE "AGREE " BUTTON. IF YOU DO NOT AGREE TO THE TERMS OF THIS LICENSE, DO NOT USE THE DEVELOPER SOFTWARE AND CLICK "DISAGREE".
IMPORTANT NOTE: To the extent that this software may be used to reproduce materials, it is licensed to you only for reproduction of non-copyrighted materials, materials in which you own the copyright, or materials you are authorized or legally permitted to reproduce. If you are uncertain about your right to copy any material, you should contact your legal advisor.
1. General.
A. The Apple software, tools, utilities, sample or example code, documentation, interfaces, content, data, and other materials accompanying this License, whether on disk, print or electronic documentation, in read only memory, or any other media or in any other form, (collectively, the "Developer Software") are licensed, not sold, to you by Apple Inc. ("Apple") for use only under the terms of this License. Apple and/or Apple's licensors retain ownership of the Developer Software itself and reserve all rights not expressly granted to you. The terms of this License will govern any software upgrades provided by Apple that replace and/or supplement the original Developer Software, unless such upgrade is accompanied by a separate license in which case the terms of that license will govern.
B. Title and intellectual property rights in and to any content displayed by or accessed through the Developer Software belongs to the respective content owner. Such content may be protected by copyright or other intellectual property laws and treaties, and may be subject to terms of use of the third party providing such content. This License does not grant you any rights to use such content nor does it guarantee that such content will continue to be available to you.
2. Permitted License Uses and Restrictions.
A. License. Subject to the terms and conditions of this License, you are granted a limited, non-exclusive license to use the
Disagree
Agree
oops
Screenshot 2022-0...05.36 PM
Screenshot 2022-0...05.48 PM
Screenshot 2022-0...05.58 PM

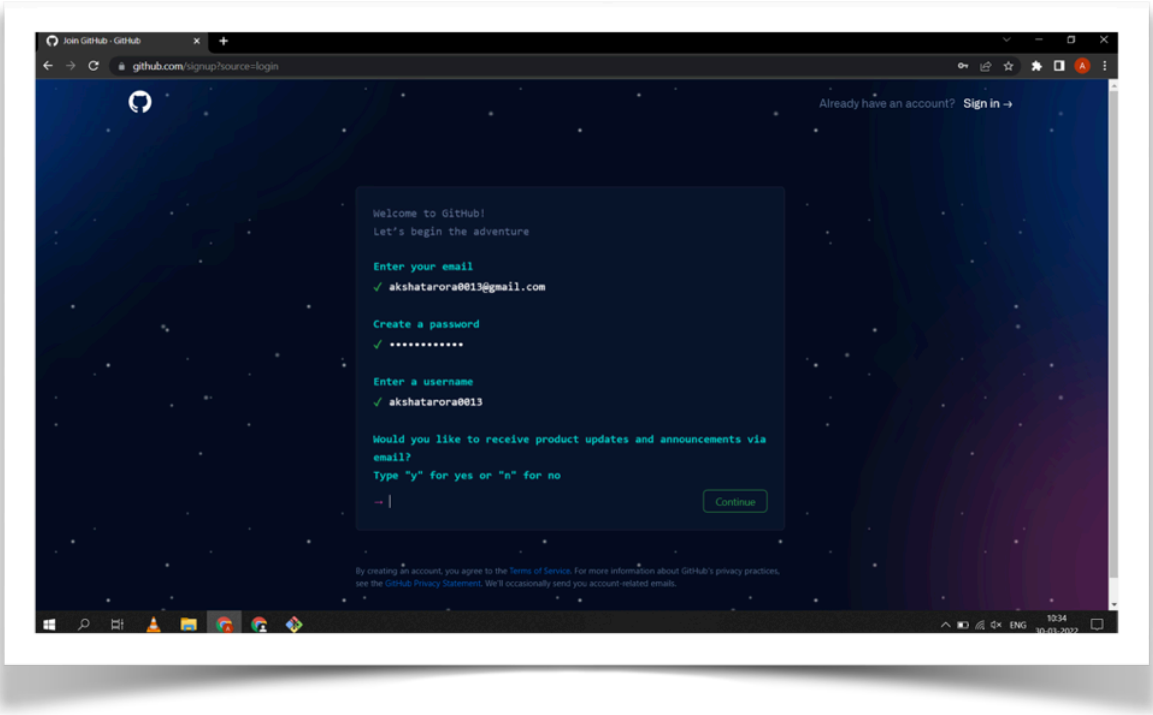

aahanwalia — -zsh — 80×24
Last login: Fri Apr 8 15:13:37 on ttys000
aahanwalia@Aahans-MacBook-Air ~ % git version
xcode-select: note: no developer tools were found at '/Applications/Xcode.app', requesting install. Choose an option in the dialog to download the command line developer tools.
Downloading software
Stop
oops
Screenshot 2022-0...05.36 PM
Screenshot 2022-0...05.48 PM
Screenshot 2022-0...05.58 PM
Screenshot 2022-0...07.19 PM


aahanwalia — -zsh — 80×24
Last login: Fri Apr 8 15:13:37 on ttys000
aahanwalia@Aahans-MacBook-Air ~ % git version
xcode-select: note: no developer tools were found at '/Applications/Xcode.app', requesting install. Choose an option in the dialog to download the command line developer tools.
aahanwalia@Aahans-MacBook-Air ~ % git
xcode-select: note: no developer tools were found at '/Applications/Xcode.app', requesting install. Choose an option in the dialog to download the command line developer tools.
aahanwalia@Aahans-MacBook-Air ~ % git
xcode-select: note: no developer tools were found at '/Applications/Xcode.app', requesting install. Choose an option in the dialog to download the command line developer tools.
aahanwalia@Aahans-MacBook-Air ~ % git
xcode-select: note: no developer tools were found at '/Applications/Xcode.app', requesting install. Choose an option in the dialog to download the command line developer tools.
aahanwalia@Aahans-MacBook-Air ~ % git version
git version 2.30.1 (Apple Git-130)
aahanwalia@Aahans-MacBook-Air ~ %
oops
Screenshot 2022-0...05.36 PM
Screenshot 2022-0...05.48 PM
Screenshot 2022-0...05.58 PM
Screenshot 2022-0...07.19 PM
Screenshot 2022-0...07.36 PM

Aim: Setting up Git account

1. Go to github.com and click on signup option at the top right corner of the screen
2. Follow up the instructions popping up on screen to create your GitHub Account





How to Set Up Default Credentials for Git Config?

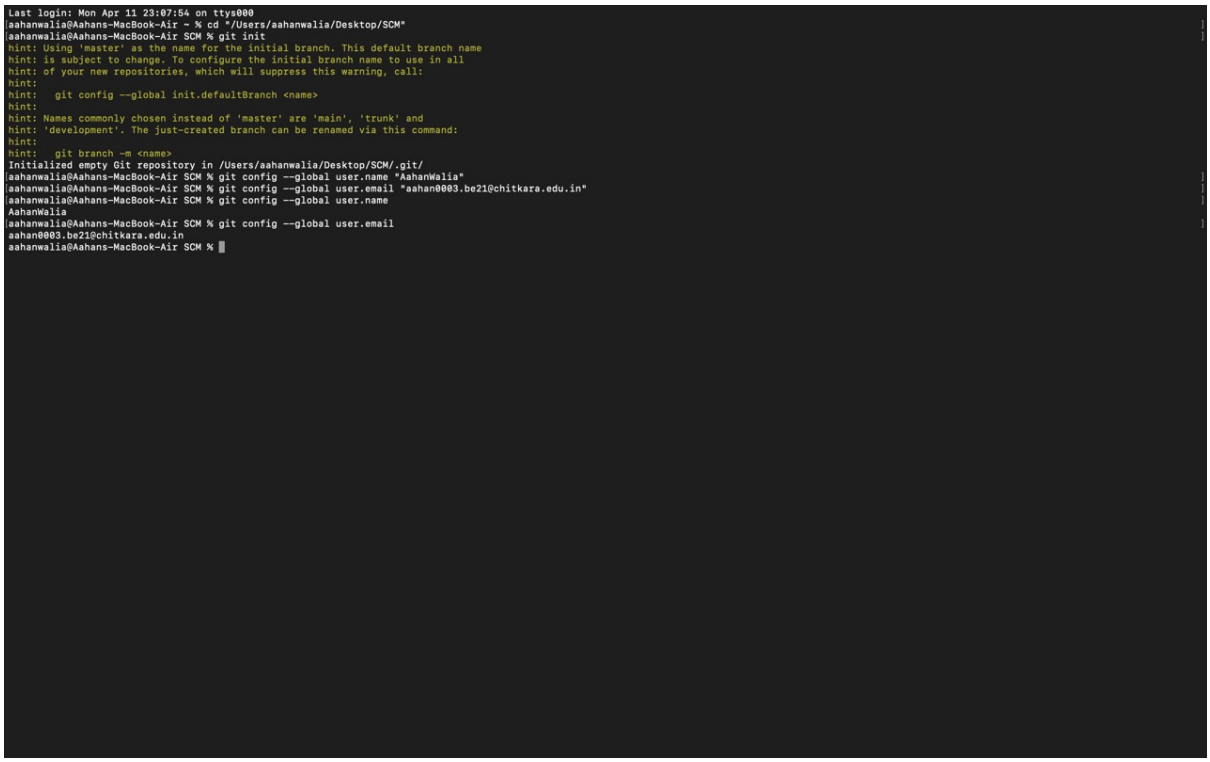
Set user's UserName in Git Config

The first change that we will be making inside our config file will be changing our username in Git. To change our username, follow these steps.

Open Git Bash in your system. Type the following command with your username: git config --global user.name "Your User Name"

Set user's Email in Git Config

After executing the above command successful, now we will change our email. By Type the following command.



git config --global user.email "Your EmailID”

git config --global user.name “Your Name ”

CHITKARA UNIVERSITY

AIM:Program to generate logs

git log command is used to generate logs

SCM — -zsh — 80×24
Last login: Mon Apr 11 23:25:34 on ttys000
aahanwalia@Aahans-MacBook-Air ~ % cd "/Users/aahanwalia/Desktop/SCM"
aahanwalia@Aahans-MacBook-Air SCM % git log
commit 39763e00ec5982e85bc18e4817aeaa37c2ab73da (HEAD -> master, activity1)
Author: AahanWalia <aahan0003.be21@chitkara.edu.in>
Date: Mon Apr 11 23:16:16 2022 +0530
created files in activity1
commit f16b440baf026fa6c51ccc7014bd3ea8562a8a19
Author: AahanWalia <aahan0003.be21@chitkara.edu.in>
Date: Mon Apr 11 23:14:25 2022 +0530
Created files
aahanwalia@Aahans-MacBook-Air SCM %
SCM
oops
Screenshot 2022-0...05.36 PM
Screenshot 2022-0...13.53 PM
Screenshot 2022-0...05.48 PM
Screenshot 2022-0...16.47 PM
Screenshot 2022-0...05.58 PM
Screenshot 2022-0...16.49 PM
Screenshot 2022-0...07.19 PM
Screenshot 2022-0...07.36 PM
Screenshot 2022-0...49.49 PM
Terminal

AIM : Create and visualize branches

HOW TO CHANGE BRANCH IN GIT

The git branch command lets you create, list, rename, and delete branches. It
doesn’t let you switch between branches or put a forked history back together
again. For this reason, git branch is tightly integrated with the git checkout and git merge commands.

Why do we need a Branch in Git and Why Branches Are Important?

Git branches come to the rescue at many different places during the development of a project. As mentioned above, branches create another line of development that is entirely different or isolated from the main stable master branch. There are many advantages to doing so.

Consider that you are developing a project with your team, and you finish a feature. You contact the client to request them to see the feature, but they are too busy, so you send them the link to have a look at the project. Okay, it's lengthy to explain in words. Let's see the same project development in different phases through images.

How to view a Local Branch in Git?

Open Git Bash and navigate to the local working repository. Type the following command to view all of your branches.

```
Last login: Mon Apr 11 23:07:54 on ttys000
aahanwalia@Aahans-MacBook-Air ~ % cd "/Users/aahanwalia/Desktop/SCM"
aahanwalia@Aahans-MacBook-Air SCM % git init
hint: Using 'master' as the name for the initial branch. This default branch name
hint: is subject to change. To configure the initial branch name to use in all
hint: of your new repositories, which will suppress this warning, call:
hint:
hint:   git config --global init.defaultBranch <name>
hint:
hint: Names commonly chosen instead of 'master' are 'main', 'trunk' and
hint: 'development'. The just-created branch can be renamed via this command:
hint:
hint:   git branch -m <name>
Initialized empty Git repository in /Users/aahanwalia/Desktop/SCM/.git/
aahanwalia@Aahans-MacBook-Air SCM % git config --global user.name "AahanWalia"
aahanwalia@Aahans-MacBook-Air SCM % git config --global user.email "aahan0003.be21@chitkara.edu.in"
aahanwalia@Aahans-MacBook-Air SCM % git config --global user.name
AahanWalia
aahanwalia@Aahans-MacBook-Air SCM % git config --global user.email
aahan0003.be21@chitkara.edu.in
aahanwalia@Aahans-MacBook-Air SCM % git status
On branch master

No commits yet

Untracked files:
  (use "git add <file>..." to include in what will be committed)
        1.rtf
        2.rtf

nothing added to commit but untracked files present (use "git add" to track)
aahanwalia@Aahans-MacBook-Air SCM % git add --a
aahanwalia@Aahans-MacBook-Air SCM % git commit -m "Created files"
[master (root-commit) f16b440] Created files
 2 files changed, 12 insertions(+)
 create mode 100644 1.rtf
 create mode 100644 2.rtf
aahanwalia@Aahans-MacBook-Air SCM % git branch
* master
aahanwalia@Aahans-MacBook-Air SCM % git branch activity1
aahanwalia@Aahans-MacBook-Air SCM % git branch
  activity1
* master
aahanwalia@Aahans-MacBook-Air SCM % git checkout activity1
Switched to branch 'activity1'
aahanwalia@Aahans-MacBook-Air SCM % git branch
* activity1
  master
aahanwalia@Aahans-MacBook-Air SCM % git status
On branch activity1
Untracked files:
  (use "git add <file>..." to include in what will be committed)
        3.rtf
        4.rtf

nothing added to commit but untracked files present (use "git add" to track)
aahanwalia@Aahans-MacBook-Air SCM % git add --a
aahanwalia@Aahans-MacBook-Air SCM % git commit -m "created files in activity1"
[activity1 39763e0] created files in activity1
 2 files changed, 12 insertions(+)
 create mode 100644 3.rtf
 create mode 100644 4.rtf
aahanwalia@Aahans-MacBook-Air SCM % git checkout master
```

How to Create a Local Branch in Git?

Let's create a new branch now in our local working repository. Type the following command to create a new branch named "activity1"

git branch <branch_name>


```
Last login: Mon Apr 11 23:07:54 on ttys000
aahanwalia@Aahans-MacBook-Air ~ % cd "/Users/aahanwalia/Desktop/SCM"
aahanwalia@Aahans-MacBook-Air SCM % git init
hint: Using 'master' as the name for the initial branch. This default branch name
hint: is subject to change. To configure the initial branch name to use in all
hint: of your new repositories, which will suppress this warning, call:
hint:
hint:   git config --global init.defaultBranch <name>
hint:
hint: Names commonly chosen instead of 'master' are 'main', 'trunk' and
hint: 'development'. The just-created branch can be renamed via this command:
hint:
hint:   git branch -m <name>
Initialized empty Git repository in /Users/aahanwalia/Desktop/SCM/.git/
aahanwalia@Aahans-MacBook-Air SCM % git config --global user.name "AahanWalia"
aahanwalia@Aahans-MacBook-Air SCM % git config --global user.email "aahan0003.be21@chitkara.edu.in"
aahanwalia@Aahans-MacBook-Air SCM % git config --global user.name
AahanWalia
aahanwalia@Aahans-MacBook-Air SCM % git config --global user.email
aahan0003.be21@chitkara.edu.in
aahanwalia@Aahans-MacBook-Air SCM % git status
On branch master

No commits yet

Untracked files:
  (use "git add <file>..." to include in what will be committed)
        1.rtf
        2.rtf

nothing added to commit but untracked files present (use "git add" to track)
aahanwalia@Aahans-MacBook-Air SCM % git add --a
aahanwalia@Aahans-MacBook-Air SCM % git commit -m "Created files"
[master (root-commit) f16b440] Created files
 2 files changed, 12 insertions(+)
 create mode 100644 1.rtf
 create mode 100644 2.rtf
aahanwalia@Aahans-MacBook-Air SCM % git branch
* master
aahanwalia@Aahans-MacBook-Air SCM % git branch activity1
aahanwalia@Aahans-MacBook-Air SCM % git branch
  activity1
* master
aahanwalia@Aahans-MacBook-Air SCM % git checkout activity1
Switched to branch 'activity1'
aahanwalia@Aahans-MacBook-Air SCM % git branch
* activity1
  master
aahanwalia@Aahans-MacBook-Air SCM % git status
On branch activity1
Untracked files:
  (use "git add <file>..." to include in what will be committed)
        3.rtf
        4.rtf

nothing added to commit but untracked files present (use "git add" to track)
aahanwalia@Aahans-MacBook-Air SCM % git add --a
aahanwalia@Aahans-MacBook-Air SCM % git commit -m "created files in activity1"
[activity1 39763e0] created files in activity1
 2 files changed, 12 insertions(+)
 create mode 100644 3.rtf
 create mode 100644 4.rtf
aahanwalia@Aahans-MacBook-Air SCM % git checkout master
```

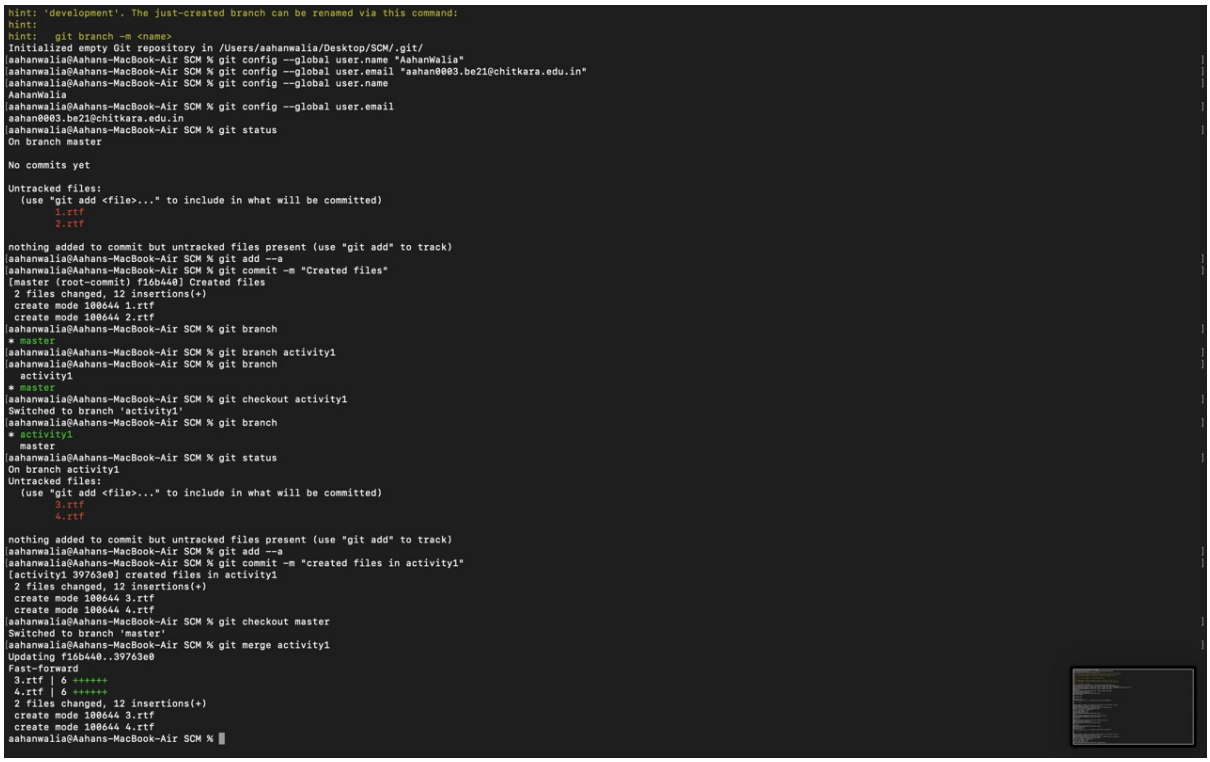

How to Switch Branch in Git?

Check the branch you are currently on, which is visible alongside the directory name.
Switch to the "activity1" by executing the command:
git checkout activity1

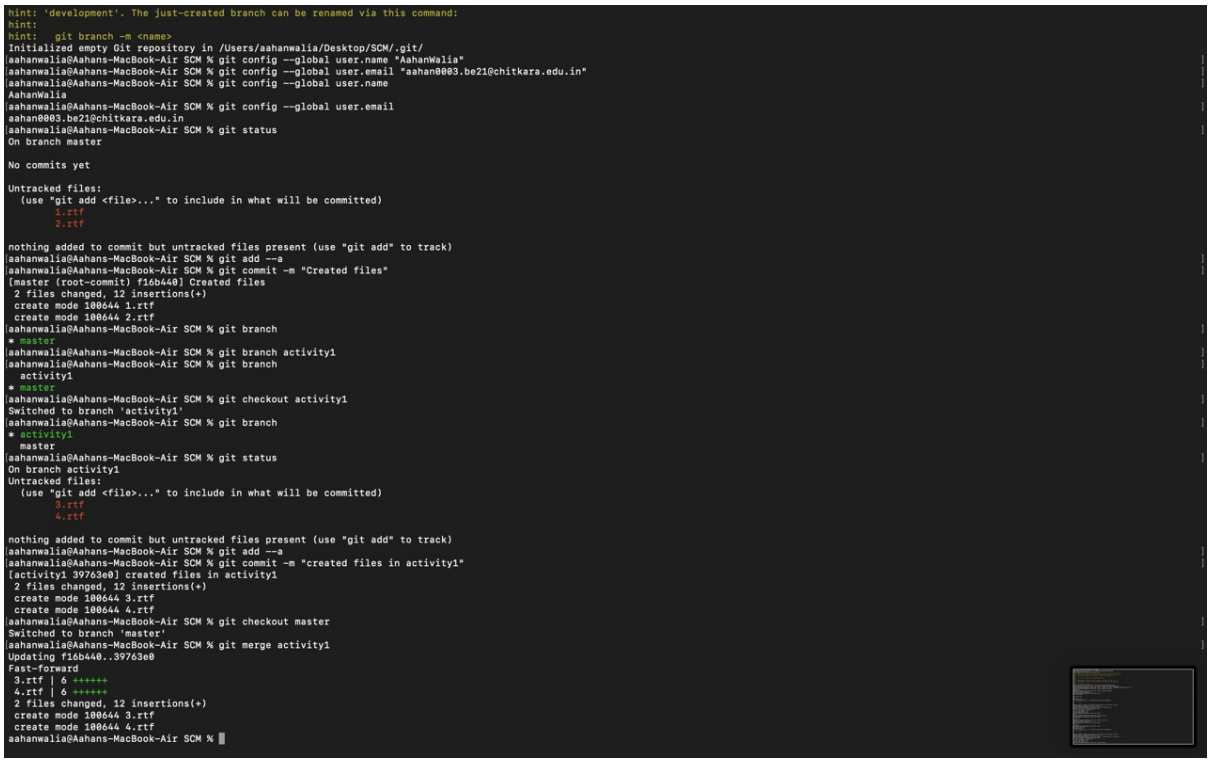
```
Last login: Mon Apr 11 23:07:54 on ttys000
aahanwalia@Aahans-MacBook-Air ~ % cd "/Users/aahanwalia/Desktop/SCM"
aahanwalia@Aahans-MacBook-Air SCM % git init
hint: Using 'master' as the name for the initial branch. This default branch name
hint: is subject to change. To configure the initial branch name to use in all
hint: of your new repositories, which will suppress this warning, call:
hint:
hint:   git config --global init.defaultBranch <name>
hint:
hint: Names commonly chosen instead of 'master' are 'main', 'trunk' and
hint: 'development'. The just-created branch can be renamed via this command:
hint:
hint:   git branch -m <name>
Initialized empty Git repository in /Users/aahanwalia/Desktop/SCM/.git/
aahanwalia@Aahans-MacBook-Air SCM % git config --global user.name "AahanWalia"
aahanwalia@Aahans-MacBook-Air SCM % git config --global user.email "aahan0003.be21@chitkara.edu.in"
aahanwalia@Aahans-MacBook-Air SCM % git config --global user.name
AahanWalia
aahanwalia@Aahans-MacBook-Air SCM % git config --global user.email
aahan0003.be21@chitkara.edu.in
aahanwalia@Aahans-MacBook-Air SCM % git status
On branch master

No commits yet

Untracked files:
  (use "git add <file>..." to include in what will be committed)
        1.rtf
        2.rtf

nothing added to commit but untracked files present (use "git add" to track)
aahanwalia@Aahans-MacBook-Air SCM % git add --a
aahanwalia@Aahans-MacBook-Air SCM % git commit -m "Created files"
[master (root-commit) f16b440] Created files
 2 files changed, 12 insertions(+)
 create mode 100644 1.rtf
 create mode 100644 2.rtf
aahanwalia@Aahans-MacBook-Air SCM % git branch
* master
aahanwalia@Aahans-MacBook-Air SCM % git branch activity1
aahanwalia@Aahans-MacBook-Air SCM % git branch
  activity1
* master
aahanwalia@Aahans-MacBook-Air SCM % git checkout activity1
Switched to branch 'activity1'
aahanwalia@Aahans-MacBook-Air SCM % git branch
* activity1
  master
aahanwalia@Aahans-MacBook-Air SCM % git status
On branch activity1
Untracked files:
  (use "git add <file>..." to include in what will be committed)
        3.rtf
        4.rtf

nothing added to commit but untracked files present (use "git add" to track)
aahanwalia@Aahans-MacBook-Air SCM % git add --a
aahanwalia@Aahans-MacBook-Air SCM % git commit -m "created files in activity1"
[activity1 39763e0] created files in activity1
 2 files changed, 12 insertions(+)
 create mode 100644 3.rtf
 create mode 100644 4.rtf
aahanwalia@Aahans-MacBook-Air SCM % git checkout master
```

How to merge branch in Git to another branch?

1 -First switch to new branch, and make a new file by using following command.



2 - Now after making changes in activity1 use the following command to merge the two branches.



git merge <branch name>

AIM : Git life cycle description

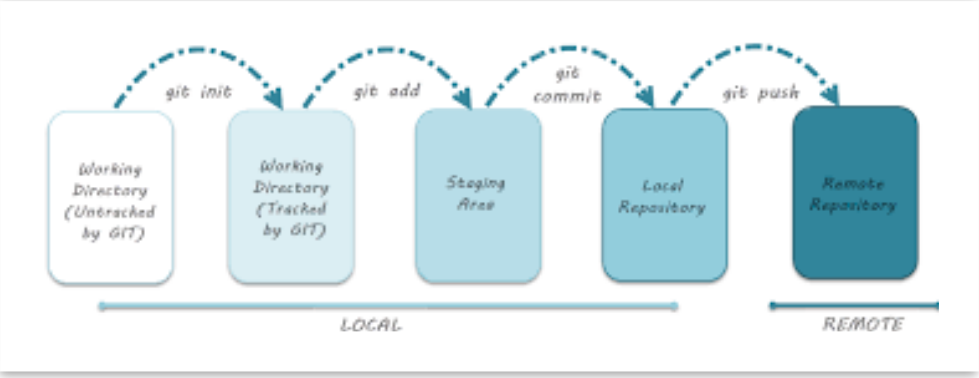
General workflow is as follows −

• You clone the Git repository as a working copy.

• You modify the working copy by adding/editing files.

• If necessary, you also update the working copy by taking other developer's changes.

• You review the changes before commit.

• You commit changes. If everything is fine, then you push the changes to the repository.

• After committing, if you realize something is wrong, then you correct the last commit and push the changes to the repository.