

[github.com](https://github.com)

## Description

GitHub, Inc. is an Internet hosting service for software development and version control using Git. It provides distributed version control access, bug tracking, software feature requests, task management, and continuous integration for every project.

[Wikipedia](#)

Parent organization: [Microsoft Corporation](#)

CEO: [Thomas Dohmke](#) (15 Nov 2021–)

Founders: [Tom Preston-Werner](#), [Chris Wanstrath](#), [P. J. Hyett](#), [Scott Chacon](#)

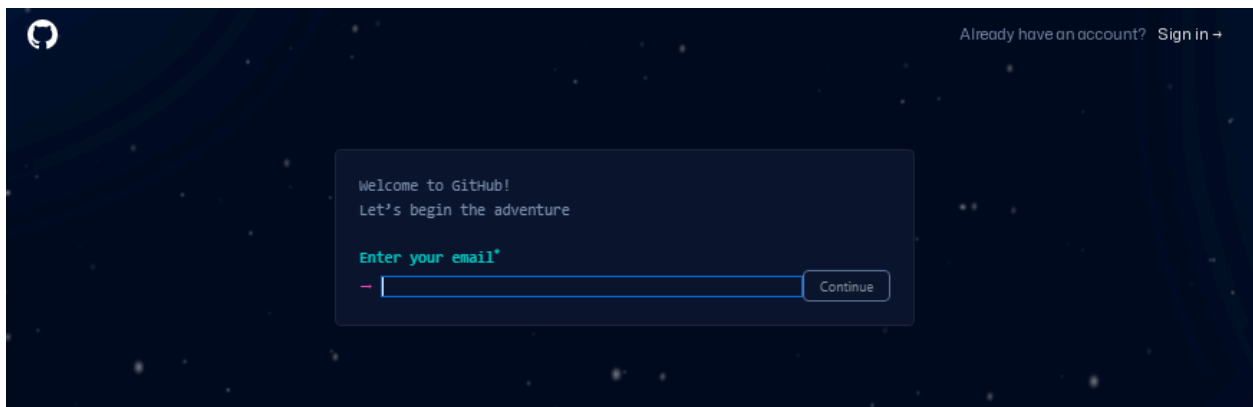
Founded: 2008

Headquarters: [San Francisco, California, United States](#)

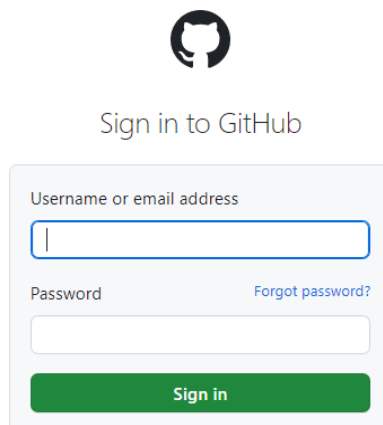
# What is a Version Control System?

The Git version control system, as the name suggests, is a system that records all the modifications made to a file or set of data so that a specific version may be called up later if needed. The system makes sure that all the team members are working on the file's latest version, and everyone can work simultaneously on the same project.

Sign up -

A screenshot of the GitHub sign-up page. The background is dark blue with a starry space pattern. In the top left corner is the GitHub logo. In the top right corner, there is a link that says "Already have an account? Sign in →". In the center, there is a dark blue rectangular box containing the text "Welcome to GitHub!" and "Let's begin the adventure". Below this text, there is a label "Enter your email\*" in green. Underneath the label is a text input field with a small red dash on the left. To the right of the input field is a button labeled "Continue".

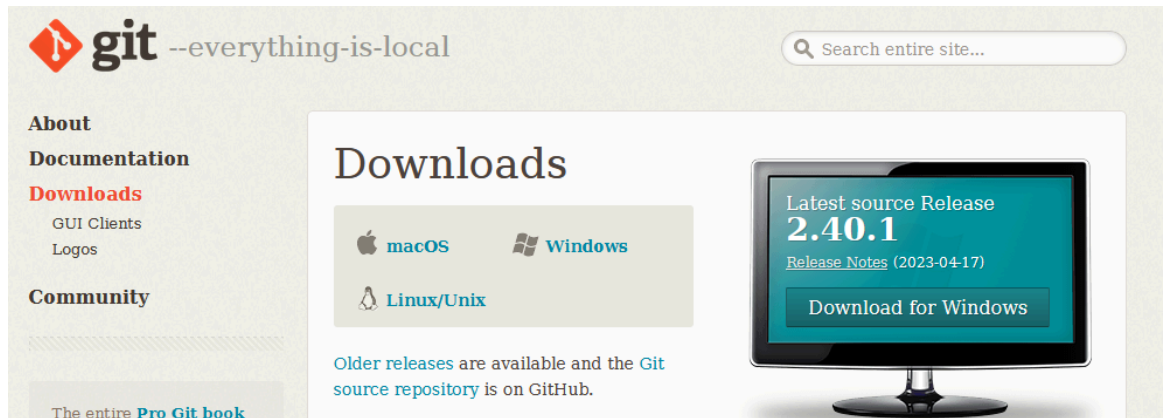
After sign up - login to github

A screenshot of the GitHub login page. At the top center is the GitHub logo. Below the logo, the text "Sign in to GitHub" is displayed. The login form is a light gray box containing two input fields. The first field is labeled "Username or email address" and has a cursor inside. The second field is labeled "Password" and has a link "Forgot password?" to its right. At the bottom of the form is a green button labeled "Sign in".

## Git Bash -

To run git commands, we first need to install git bash.

<https://git-scm.com/downloads>



## Git commands -

> create your git account from github.com

> First install git bash.

> go to your code root folder , where you have your package.json file.

> **git init**

> create a .gitignore file and add node\_modules like below.

/node\_modules

> **git add .** (adding all new or modified files also known as staging )

> **git commit -m "commit message here"** (commit your files to master branch)

### Now create a new repo in your github.

> **git remote add origin your\_repo\_url**

> **git push -u origin master**

### ===== Take Latest =====

> **git pull origin** (take latest before start work)

> **git status** (check modified files)

> **git diff** (show diff)

### ===== Stashing your changes =====

> **git stash**

```
> git stash apply
```

```
=====
```

```
-----
```

```
BUDHA@BUDHA-PC MINGW32 /d/Eduriefy/todoApp/todo-app (master)
```

```
$ git remote -v
```

```
BUDHA@BUDHA-PC MINGW32 /d/Eduriefy/todoApp/todo-app (master)
```

```
$ git remote add origin
```

```
https://github.com/sandeepkr Gupta/ToDoApp.git
```

```
BUDHA@BUDHA-PC MINGW32 /d/Eduriefy/todoApp/todo-app (master)
```

```
$ git remote -v
```

```
origin https://github.com/sandeepkr Gupta/ToDoApp.git (fetch)
```

```
origin https://github.com/sandeepkr Gupta/ToDoApp.git (push)
```

```
BUDHA@BUDHA-PC MINGW32 /d/Eduriefy/todoApp/todo-app (master)
```

```
$ git add .
```

```
warning: LF will be replaced by CRLF in package-lock.json.
```

The file will have its original line endings in your working directory

warning: LF will be replaced by CRLF in package.json.

The file will have its original line endings in your working directory

warning: LF will be replaced by CRLF in src/App.js.

The file will have its original line endings in your working directory

warning: LF will be replaced by CRLF in src/index.js.

The file will have its original line endings in your working directory

BUDHA@BUDHA-PC MINGW32 /d/Eduriefy/todoApp/todo-app (master)

**\$ git status**

On branch master

Changes to be committed:

(use "git restore --staged <file>..." to unstage)

```
modified:   package-lock.json

modified:   package.json

modified:   src/App.js

new file:   src/component/ToDoApp.js

new file:   src/component/todo.css

modified:   src/index.js

new file:   src/redux/actions.js

new file:   src/redux/reducer.js

new file:   src/reduxStore.js
```

```
BUDHA@BUDHA-PC MINGW32 /d/Eduriefy/todoApp/todo-app (master)
```

```
$ git commit -m "committing All TOdo App Files"
```

```
[master a77b182] committing All TOdo App Files
```

```
9 files changed, 354 insertions(+), 43 deletions(-)
```

```
BUDHA@BUDHA-PC MINGW32 /d/Eduriefy/todoApp/todo-app (master)
```

```
$ git push -u origin master
```

Enumerating objects: 36, done.

Counting objects: 100% (36/36), done.

Delta compression using up to 4 threads

Compressing objects: 100% (36/36), done.

Writing objects: 100% (36/36), 315.37 KiB | 2.63 MiB/s, done.

Total 36 (delta 5), reused 0 (delta 0), pack-reused 0

BUDHA@BUDHA-PC MINGW32 /d/Eduriefy/todoApp/todo-app (master)

**\$ git status**

On branch master

Your branch is up to date with 'origin/master'.

nothing to commit, working tree clean

BUDHA@BUDHA-PC MINGW32 /d/Eduriefy/todoApp/todo-app (master)

**\$ git push origin master**

Everything up-to-date

BUDHA@BUDHA-PC MINGW32 /d/Eduriefy/todoApp/todo-app (master)



## **RSA Token**

```
ssh-keygen -t rsa -b 4096 -C "your_email@example.com"
```

```
$ git pull
```

```
remote: Enumerating objects: 3, done.
```

```
remote: Counting objects: 100% (3/3), done.
```

```
remote: Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
```

```
Unpacking objects: 100% (3/3), 625 bytes | 4.00 KiB/s, done.
```

```
From https://github.com/sandeepkrgupt/ToDoApp
```

```
* [new branch]      main      -> origin/main
```

```
Already up to date.
```

```
BUDHA@BUDHA-PC MINGW32 /d/Eduriefy/todoApp/todo-app (master)
```

```
$ git branch
```

```
* master
```

```
BUDHA@BUDHA-PC MINGW32 /d/Eduriefy/todoApp/todo-app (master)
```

```
$ ssh-keygen
```

Generating public/private rsa key pair.

Enter file in which to save the key

(/c/Users/BUDHA/.ssh/id\_rsa):

Created directory '/c/Users/BUDHA/.ssh'.

Enter passphrase (empty for no passphrase):

Enter same passphrase again:

Your identification has been saved in /c/Users/BUDHA/.ssh/id\_rsa

Your public key has been saved in /c/Users/BUDHA/.ssh/id\_rsa.pub

The key fingerprint is:

SHA256:hxTcgeqPV2YqO+kOR8KMC0SOKv7WUf0Vykp1l4PKwv0

BUDHA@BUDHA-PC

## **Create New Branch From Master and push files -**

BUDHA@BUDHA-PC MINGW32 /d/FoodBlog/food-blog (master)

**\$ git checkout -b sandeep\_figma\_proj**

BUDHA@BUDHA-PC MINGW32 /d/FoodBlog/food-blog (master)

```
$ git branch
```

```
* master
```

```
sandeep_figma_assignment
```

```
sandeep_figma_proj
```

```
BUDHA@BUDHA-PC MINGW32 /d/FoodBlog/food-blog (master)
```

```
$ git status
```

```
    modified:   src/App.js
```

```
BUDHA@BUDHA-PC MINGW32 /d/FoodBlog/food-blog  
(sandeep_figma_proj)
```

```
$ git add .
```

```
BUDHA@BUDHA-PC MINGW32 /d/FoodBlog/food-blog  
(sandeep_figma_proj)
```

```
$ git commit -m "app js file"
```

```
BUDHA@BUDHA-PC MINGW32 /d/FoodBlog/food-blog  
(sandeep_figma_proj)
```

```
$ git push origin sandeep_figma_proj
```

