GIT AND GITHUB

DAY 1 OF 20

GitHub, Inc. [US] | github.com/facebook/react

Incognito

Search or jump to... /

Pull requests Issues Marketplace Explore

facebook / react

Used by ▾ 2,350,253 | Watch ▾ 6,643 | Unstar 134,638 | Fork 25,138

Code | Issues 603 | Pull requests 205 | Projects 0 | Wiki | Security | Insights

A declarative, efficient, and flexible JavaScript library for building user interfaces. https://reactjs.org

javascript react frontend declarative ui library

11,236 commits | 31 branches | 305 contributors | MIT

Branch: master ▾ | New pull request | pload files | Find File | Clone or download ▾

bvaughn Added DevTools 4.0.4 CHANGELOG entry | Latest commit e89c19d yesterday

.circleci	[CI] Disable	6 days ago
.github	Reword issue	2 years ago
fixtures	[Partial Hydrati ...	6 days ago
packages	Added DevTools 4.	yesterday
scripts	Added better error repo ... (#16394)	5 days ago
.editorconfig	Add insert_final_newline to editorconfig	4 years ago
.eslintignore	Use Yarn Workspaces (#11252)	2 years ago
.eslintrc.js	Revert "[Scheduler] Profiling features (#16145)" (#16392)	5 days ago
.gitattributes	.gitattributes to ensure LF line endings when we should	6 years ago
.gitignore	Parallelizes the build script across multiple processes (#15716)	3 months ago

PREPARED BY DSEA

CONTENT

- Version Control
- Version Control Tools
- Git and Github
- Git Features
- Git Operations &Commands

VERSION CONTROL

VERSION CONTROL SYS

WHAT IS VERSION CONTROL SYSTEM

- A Management system that manages the changes that you make to the project till the end
- Every time you make a change snapshots are taken
- A Snapshot is the entire state of your project

#WHY VERSION CONTROL

- Collaboration among developers

  Different developers can work on a single projec
  irrespective of their location.
  Changes made on a single project are visualized t
  everyone working on the project.

- Storage of versions

  Snapshots of all versions are properly
  documented and stored

- Backup of data

fetching from the central server backup is always available in your local server

ASSIGNMENT

- Research more on version control
- Document your work

VERSION CONTROL TOOLS

- Git
- SVN(Apache SubVersion)
- CVS(Concurrent Versions System)
- ,Mercurial l

GIT AND GITHUB

GIT

Distribute vc tool that supports non-linear workflows by providing data assurance

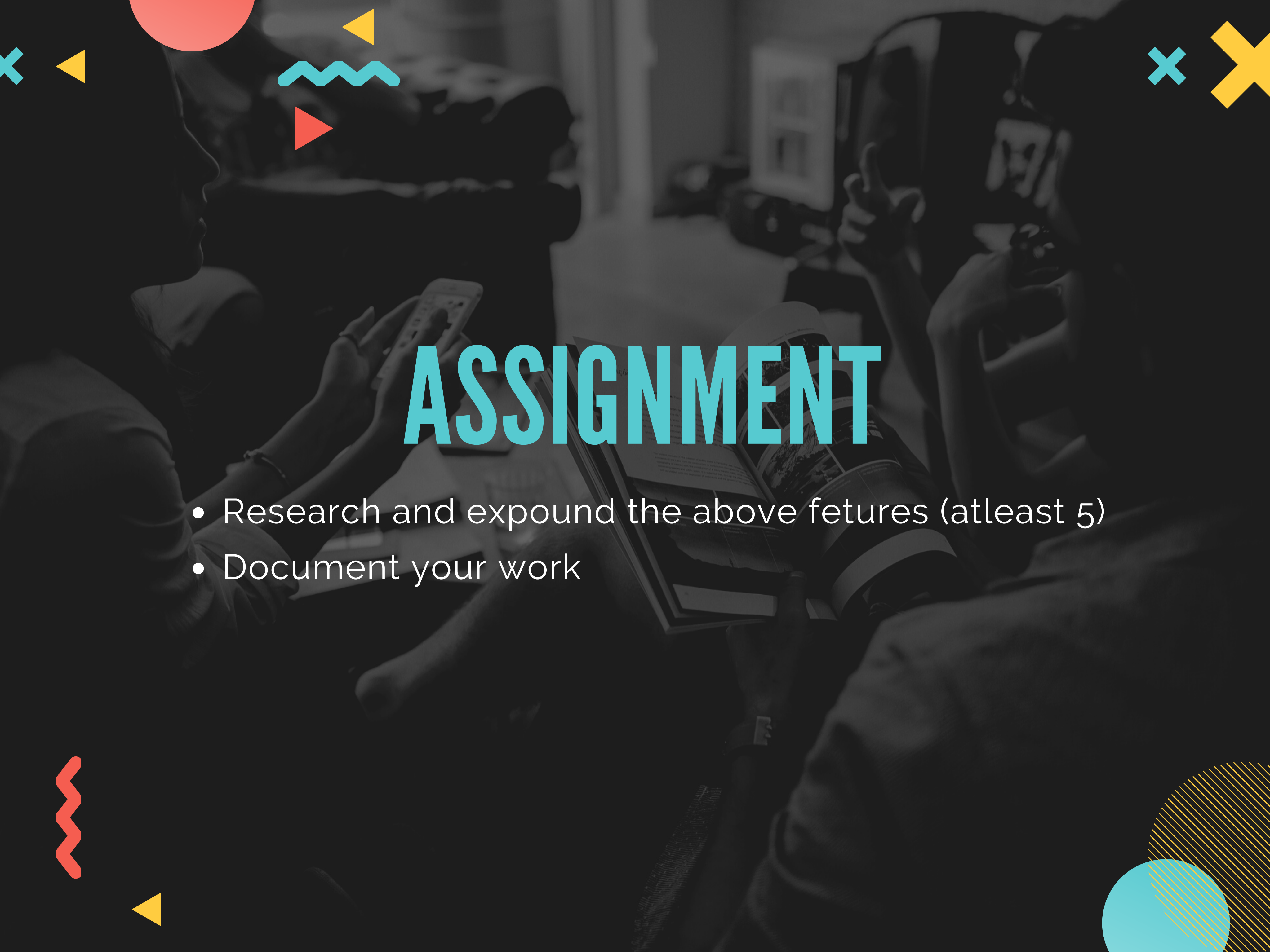
GITHUB

Cloud hosting platform for your files

A service that allows you to upload your code using git and to manage your code with a nice web interface

GIT FEATURES

- It is compatible with other version control system
- It is non- linear
- It is secure
- Very economical
- lightweight
- Branching
- Speed
- Open source
- Reliable
- Secure -SHA1 Key to rename



ASSIGNMENT

- Research and expound the above fetures (atleast 5)
- Document your work

INSTALLING GIT

LINUX USERS

- **Update the system**

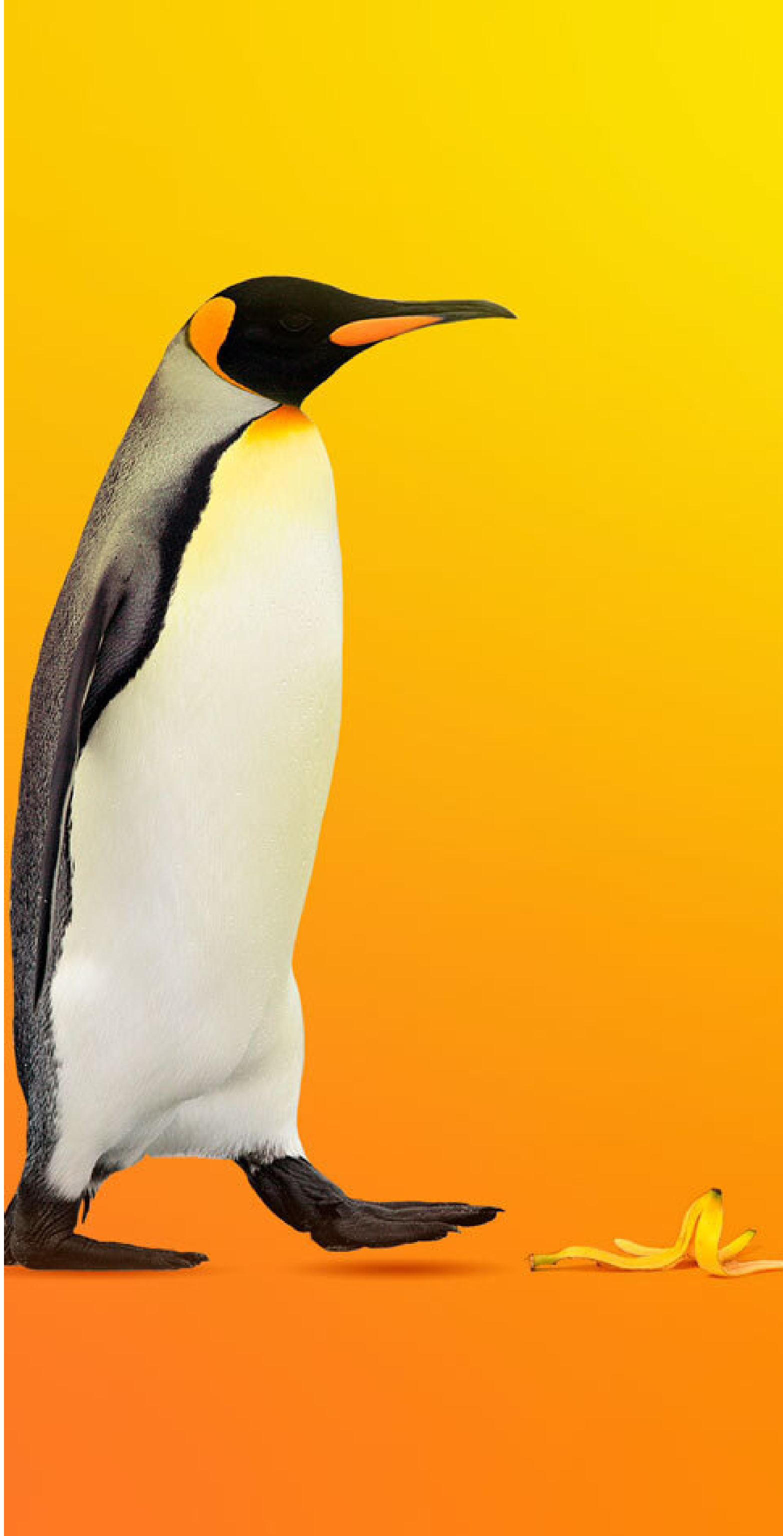
Run these commands in the terminal to update the Linux system:

```
sudo apt update
sudo  apt upgrade
```

- **Install git**

Most likely you have git installed already,but to make sure that you have the most up to date version of git, run the following commands:

```
sudo add-apt-repository ppa:git-core/ppa
sudo apt update
sudo
apt install git
```



LINUX USERS

- **Verify the version**

Make sure your git version is at least 2.28 by running this command:

```
git--version
```

- **If the version number is less than 2.28, follow the instructions again.**

- **Configure git and github**

For Git to work properly,you need to let it know who we are so that it can link a local Git user (you) to GitHub.
When working on a team, this allows people to see what you have
committed and who committed each line of code.

LINUX USERS

The commands below will configure Git. Be sure to enter your own information inside the quotes **(but include the quotation marks)!**

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

GitHub recently changed the default branch on new repositories from master to main,**change the default branch** for Git using this command:

```
git config --global init.defaultBranch main
```

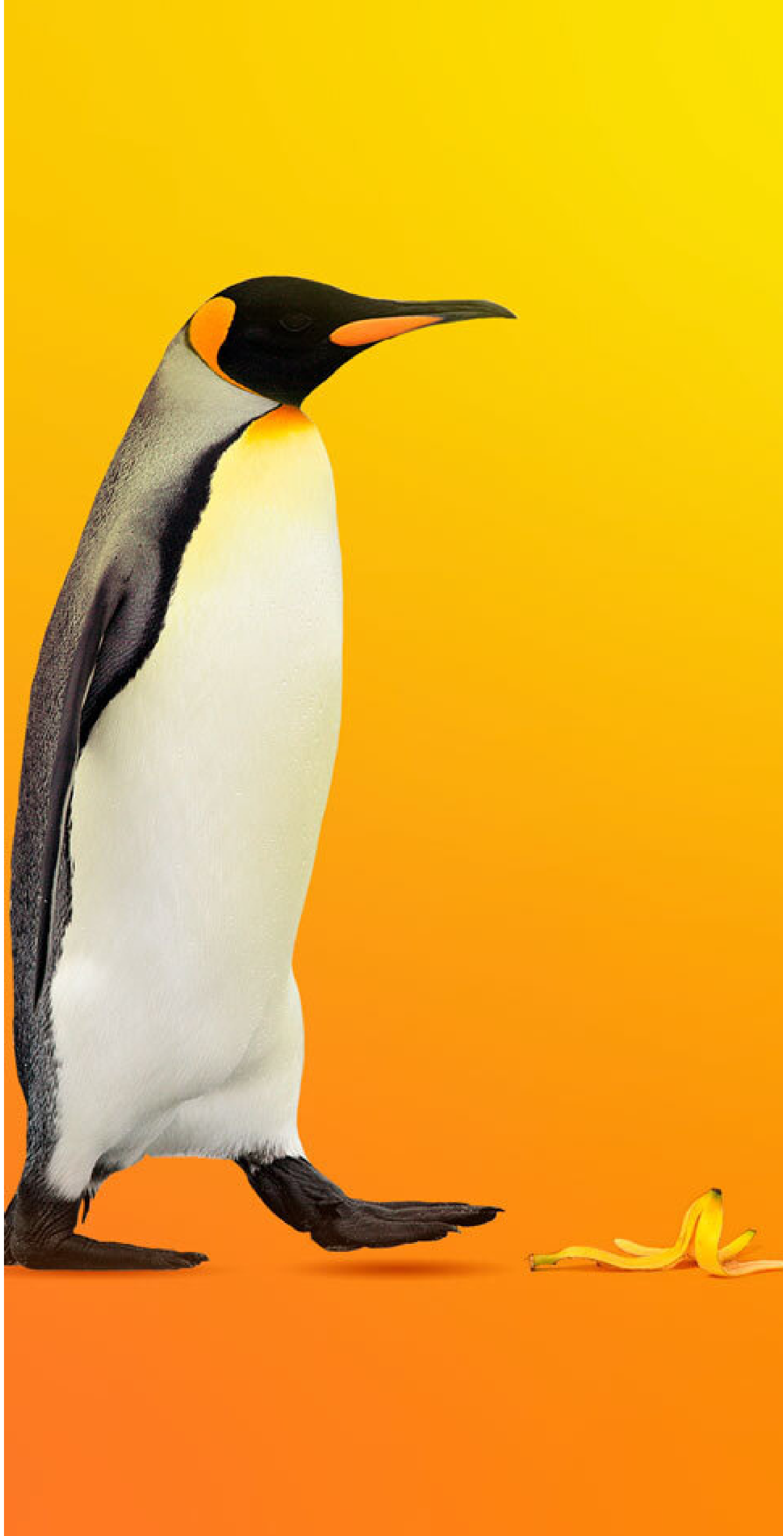
LINUX USERS

To enable colorful output with git, type

```
git config --global color.ui auto
```

To verify things are working properly, enter these commands and verify that the output **matches your name and email address.**

```
git config --get user.name
git config --get user.email
```



LINUX USERS

- **Create a GitHub Account or Sign In**

  Go to github and create an account!
  If you already have an account, sign in.
  You do not need to use the same email address you used before, but it might be a good idea to use the same one to keep things simple.

- **Create an SSH Key**

An SSH key is a cryptographically secure identifier. It's like a really long password used to identify your machine. GitHub uses SSH keys to allow you to upload to your repository without having to type in

LINUX USERS

your username and password every time.

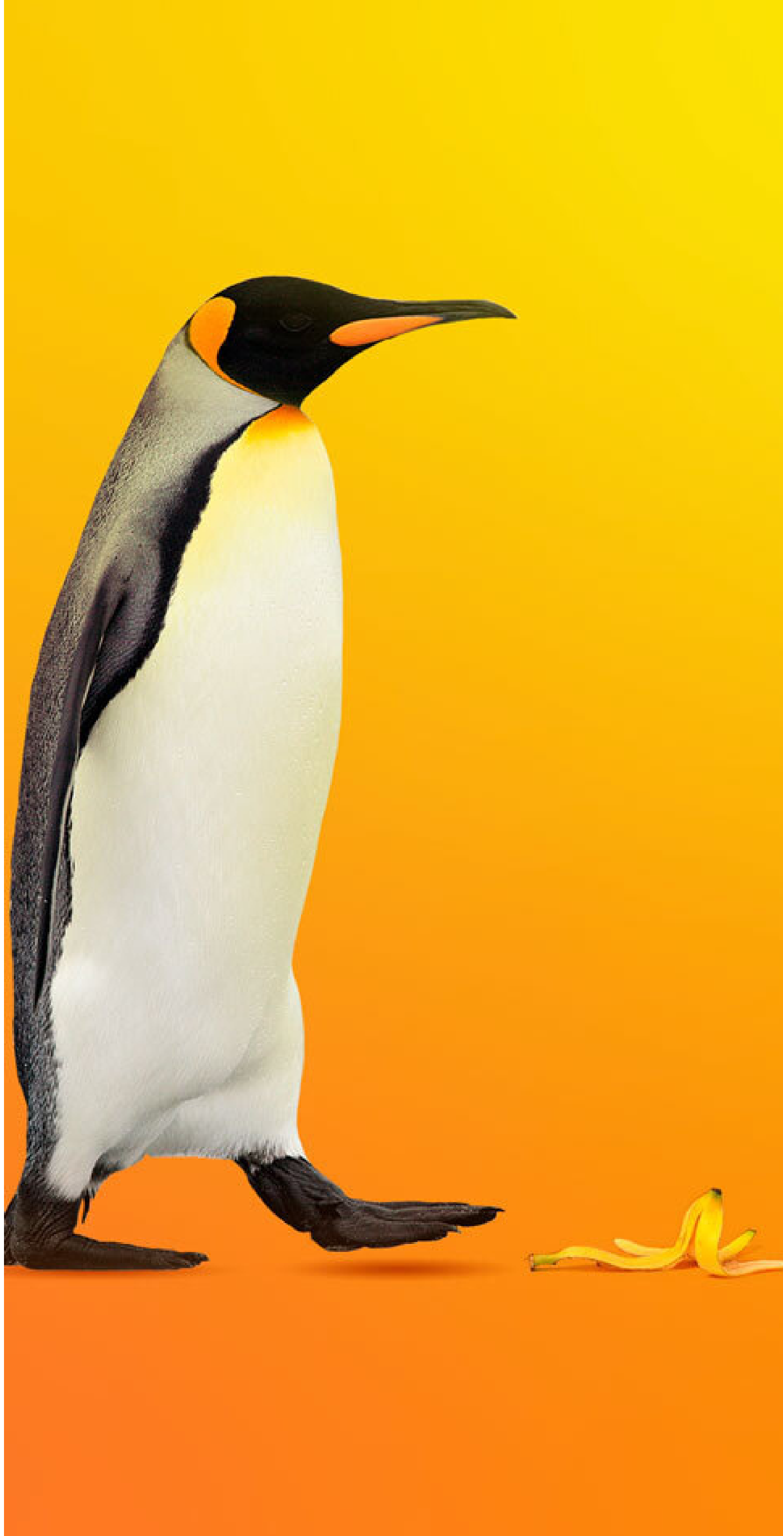
First, we need to see if you have an SSH key already installed. Type this into the terminal:

```
ls~/.ssh/id_rsa.pub
```

If a message appears in the console containing the text **"No such file or directory"**, then you **do not yet have an SSH key**, and you will need to create one.

If **no message** has appeared in the console output, you already have a key and can proceed to linking your key

- **To create a new SSH key, run the following command inside your terminal.**
- **The -C flag followed by your email address ensures that GitHub knows who you are.**

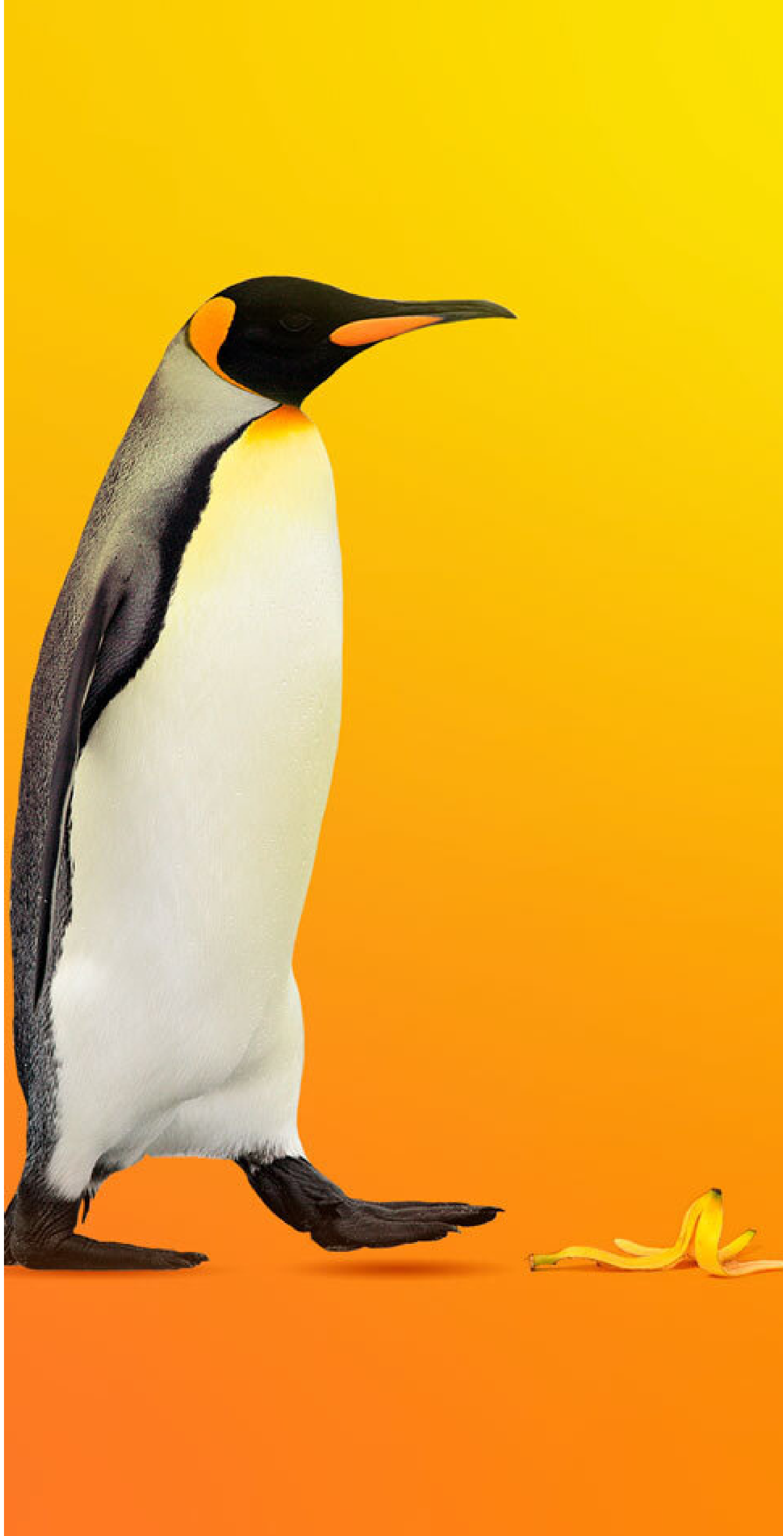


LINUX USERS

IF YOU DON'T HAVE AN SSH KEY

- **To create a new SSH key, run the following command inside your terminal.**
- **The -C flag followed by your email address ensures that GitHub knows who you are.**

Note: The angle brackets (< >) in the code snippet below indicate that you should replace that part of the command with the appropriate information.
Do not include the brackets themselves in your command.
For example, if your email address is
odin@valhalla.com, then you would type ssh-keygen -C odin@valhalla.com. You
will see this convention of using angle brackets to indicate placeholder text



LINUX USERS

. **For example**

if your email address is odin@valhalla.com,

then you would type

```
ssh-keygen -C odin@valhalla.com.
```

You will see this convention of using angle brackets to **indicate placeholder text** used throughout The Odin Project's curriculum and other coding websites,

So it's good to be familiar with what it means.

```
ssh-keygen -C <youremail>
```

When it prompts you for a location to save the generated key, just push Enter.

Next, it will ask you for a password; enter one if you wish, but it's not required.

LINUX USERS

- **Link Your SSH Key with GitHub**

Now, you need to tell GitHub what your **SSH key** is so that you can push your code without typing in a password every time.

First, you'll navigate to where GitHub receives our SSH key.
Log into GitHub and **click on your profile picture** in the top right corner.
Then, click on Settings in the drop-down menu.

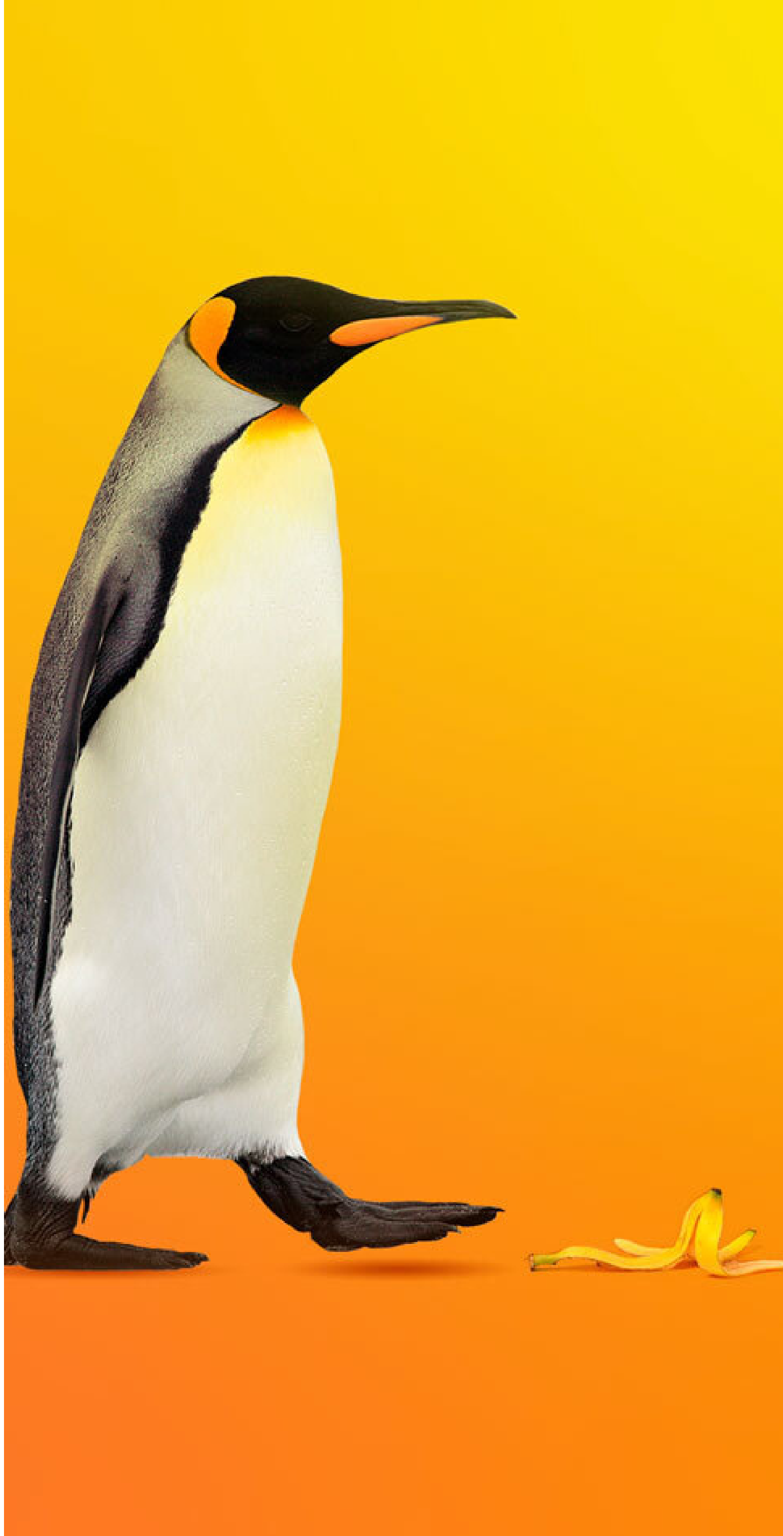
LINUX USERS

Next, on the left-hand side,**click SSH and GPG keys**.Then, click the green button in the top right corner that says **New SSH Key.** Name your key something that is **descriptive enough** for you to remember where it came from. Leave this window open while you do the next steps.

Now you need to copy your public SSH key.

To do this, we're going to use a command called **cat** to read the file to the console. (**Note that the .pub file extension is important in this case.)**

```
cat ~/.ssh/id_rsa.pub
```



LINUX USERS

- **Testing your key**

Follow the directions in https://docs.github.com/en/free-pro-team@latest/github/authenticating-to-github/testing-your-ssh-connection to verify your SSH connection.

WINDOW USERS

- **Install Git on Windows**

  Download the latest **Git for Windows installer**

- **Step 2**

  When you've successfully started the installer, you should see the Git Setup wizard screen.

  Follow the Next and Finish prompts to complete the installation. The default options are pretty sensible for most users.

WINDOW USERS

- **Install Git on Windows**

  Download the latest **Git for Windows installer**

- **Step 3**
  Open a command Prompt (or Git Bash if during installation you elected not to use Git from the Windows Command Prompt).

- **Step 4**

Run the following commands to configure your Git username and email using the following commands, **replacing** Emma's name with your own.

WINDOW USERS

These details will be associated with any commits that you create:

```
git config --global user.name "Emma Paris"
$ git config --global user.email
"eparis@atlassian.com"
```

- **Step 5**

  Optional:
  Install the Git credential helper on Windows
  Bitbucket supports pushing and pulling over HTTP t
  your remote Git repositories on Bitbucket.
  Every time you interact with the remote repository, you must supply a username/password combination.

WINDOW USERS

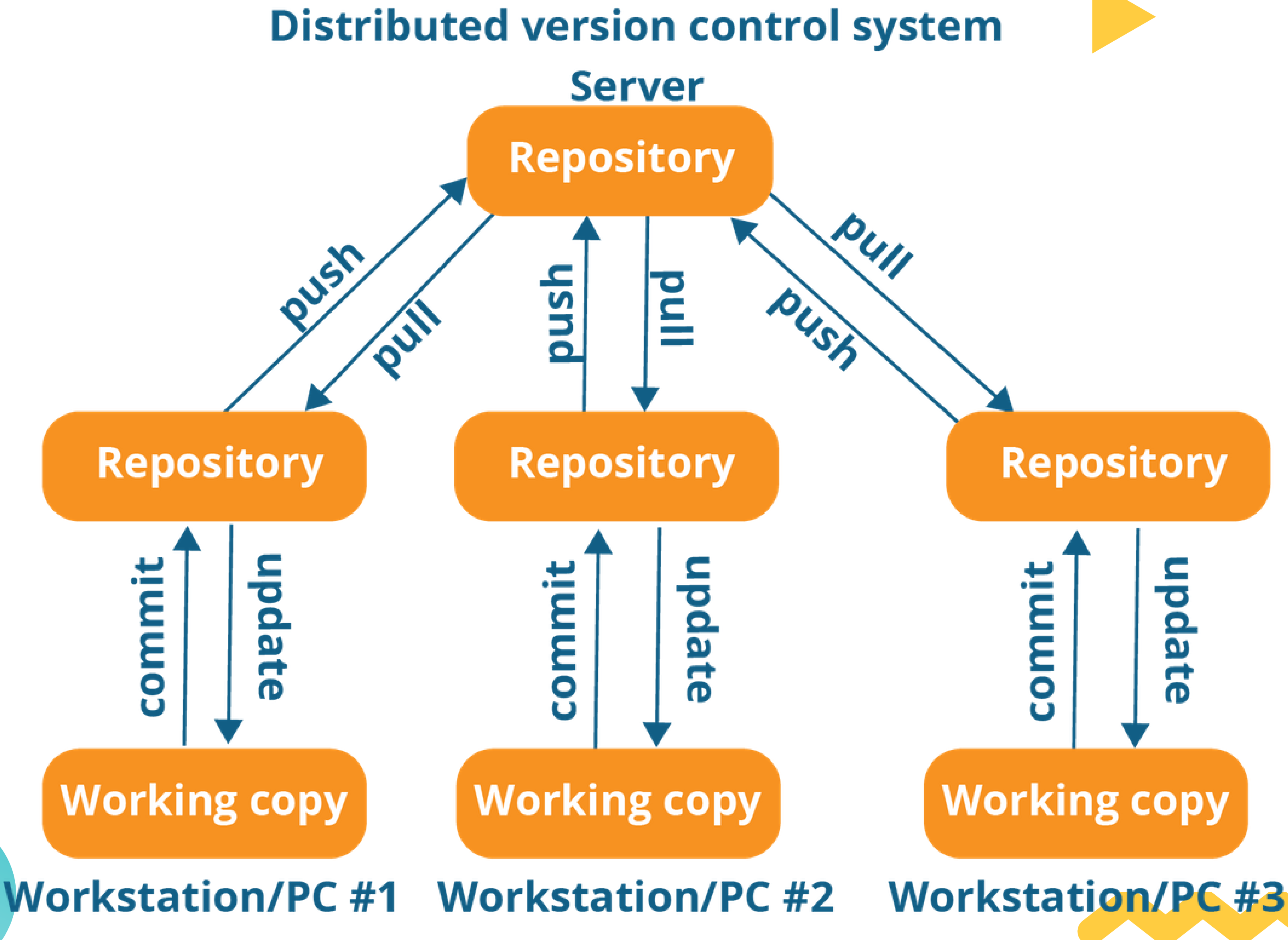
Optional:

You can store these credentials, instead of supplying the combination every time, with the Git Credential Manager for Windows.

GIT COMMANDS

Read about Git commands and create your own cheat sheet

SUMMARY OF GIT COMMANDS



WHERE IS YOUR CHEAT SHEET ?