



HACK TOBER FEST



HACKTOBERFEST 2022

Organised by *BV Jagadeesh Science Centre* in
association with Dept. of Computer Science, National
College Jayanagar

Sigma Club members@workshop

Charan Yadav

Bhargav Ram

Shraddha

Spoorti K

Sindhuja

Vaishnavi

Abhishek YS

Sagar

Uday Kiran N

Rakshitha

Preetham

Venkatesh

Charan Kumar R

Contributions

- Chandan NL
- Shashank A
- Chandhan SS
- Navyashree



SESSION 1 - GETTING STARTED WITH GITHUB

FEW THINGS TO SAY

By the end of the workshop,

1. you would be collaborating with your friend on a project
2. workshop equips you to build and deploy
3. Learn how to build and deploy static site like *bvjsciencecentre.github.io*



WHAT IS GITHUB

GitHub is a code hosting platform for version control and collaboration. It lets you and others work together on projects from anywhere.

ARE THERE ALTERNATIVES TO GITHUB?

Yes, Gitlab, Gitea and many more



SIGNING UP FOR A NEW GITHUB ACCOUNT

INTRODUCTION

- Home page
- Explore, trending options
- Issues
- Notifications
- Repository
- Organisation
- Stars

DEMONSTRATION

How to search for a project and star it?

EXERCISE

Search for a project in your subject and star it.(some topics - open source university, awesome)

ACTIVITY 01

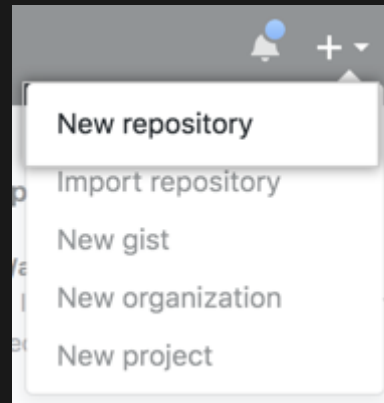
CREATING YOUR FIRST REPO AND EDITING IT

DEMONSTRATION - HOW TO CREATE A REPOSITORY?

ACTIVITY 01

1. CREATING A REPOSITORY

1. In the upper-right corner of any page, use the drop-down menu, and select New repository.





2. In the Repository name box, enter **Notes** as repo name
3. In the Description box, write a short description.(My Notes at Hacktoberfest)
4. Select Add a README file.
5. Select whether your repository will be Public or Private.
6. Select a license - MIT License



Click Create repository.

Owner *

Repository name *



octocat ▾



hello-world



Great repository names are short and memorable. Need inspiration? How about **ubiquitous-system**?

Description (optional)

My first repository



Public

Anyone on the internet can see this repository. You choose who can commit.



Private

You choose who can see and commit to this repository.

Initialize this repository with:

Skip this step if you're importing an existing repository.



Add a README file

This is where you can write a long description for your project. [Learn more.](#)



Add .gitignore

Choose which files not to track from a list of templates. [Learn more.](#)



Choose a license

A license tells others what they can and can't do with your code. [Learn more.](#)



This will set `main` as the default branch. Change the default name in your [settings](#).

Create repository

EXERCISE - CREATE YOUR FIRST REPOSITORY

Edit the README file with little introduction of file, comment and save changes, refer to markdown cheatsheet

Download Markdown All in one extension in VSCode

ACTIVITY 01

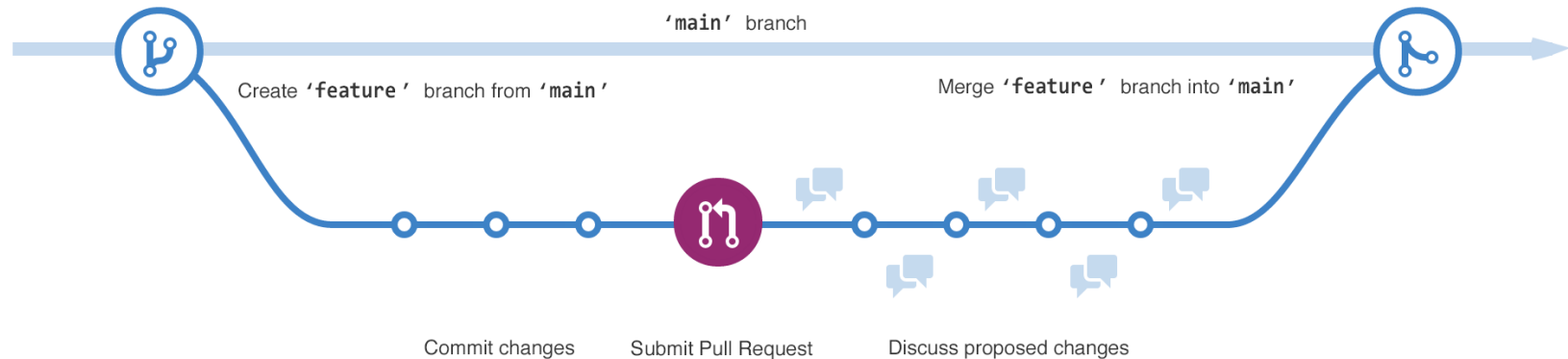
2. CREATING A BRANCH

DEMONSTRATION - CREATING A BRANCH IN THE FIRST REPO.

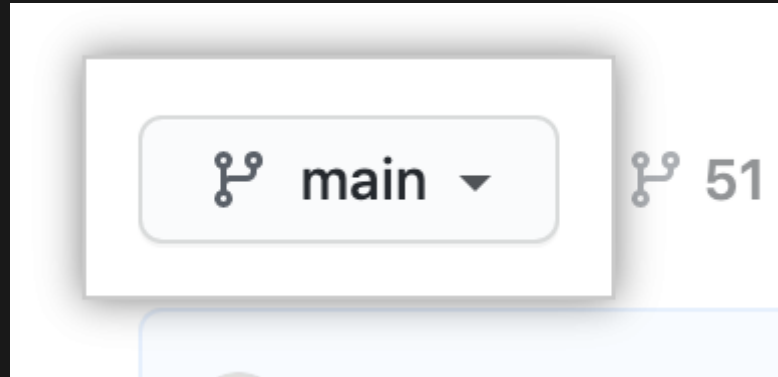
Have you ever saved different versions of a file?
Something like:

- design1.html
- design-with-navbar.html
- final-design.html

WHAT IS A GOOD SOLUTION FOR THIS?



1. Click the Code tab of your **Notes** repository.
2. Click the drop down at the top of the file list that says main.





5. Type a branch name - **activity1-notes**, into the text box.
4. Click Create branch: activity1-notes from main.

The image shows a GitHub interface for creating a new branch. At the top, there is a dropdown menu for the current branch, which is set to 'main'. To the right of this are links for 'Branches' and 'Tags'. Below this is a modal window titled 'Switch branches/tags' with a close button (X) in the top right corner. Inside the modal, there is a text input field containing the text 'readme-edits'. Below the input field are two tabs: 'Branches' (which is selected) and 'Tags'. Under the 'Branches' tab, there is a button that says 'Create branch: readme-edits from 'main''. At the bottom of the modal, there is a link that says 'View all branches'.



Now you have two branches, main and activity1-notes. Right now, they look exactly the same. Next you'll add changes to the new branch.

**EXERCISE - CREATE A NEW HELLO-WORLD REPO AND
CREATE A NEW BRANCH CALLED 1-EDIT-README**

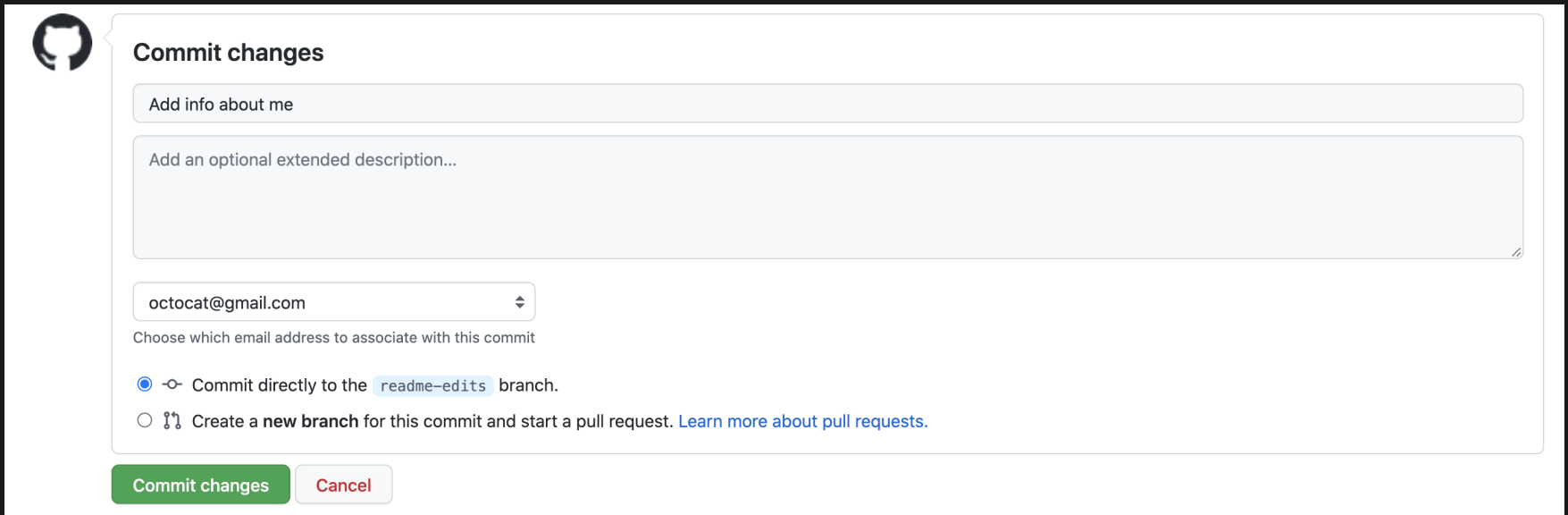
ACTIVITY 01

3. EDITING AND COMMITTING CHANGES

DEMONSTRATION - MAKING CHANGES AND COMMITTING IT IN THE FIRST REPO.

1. Under the activity1-notes branch you created, click the [README.md](#) file.
2. Click at pencil symbol to edit the file.
3. In the editor, write a bit about yourself. Try using different Markdown elements.
4. In the Commit changes box, write a commit message that describes your changes.

5. Click Commit changes.



The image shows the GitHub 'Commit changes' dialog box. It features a GitHub logo in the top left corner. The main title is 'Commit changes'. Below the title, there are two text input fields: 'Add info about me' and 'Add an optional extended description...'. Below these fields is a dropdown menu showing 'octocat@gmail.com' with a small arrow icon. Under the dropdown, it says 'Choose which email address to associate with this commit'. There are two radio button options: the first is selected and says '-o- Commit directly to the `readme-edits` branch.', and the second is unselected and says '🔑 Create a new **branch** for this commit and start a pull request. [Learn more about pull requests.](#)'. At the bottom, there are two buttons: a green 'Commit changes' button and a grey 'Cancel' button.

Commit changes

Add info about me

Add an optional extended description...

octocat@gmail.com

Choose which email address to associate with this commit

☒ -o- Commit directly to the `readme-edits` branch.

☐ 🔑 Create a new **branch** for this commit and start a pull request. [Learn more about pull requests.](#)

Commit changes Cancel



EXERCISE - CREATE A BRANCH IN YOUR FIRST REPO

ACTIVITY 01

4. OPENING A PULL REQUEST


DEMONSTRATION - CREATING A PR IN MY REPO, AND MENTIONING MY FRIEND'S NAME



1. Click the Pull requests tab of your activity1-notes repository.
2. Click New pull request
3. In the Example Comparisons box, select the branch you made, activity1-notes, to compare with main (the original).

4. Look over your changes in the diffs on the Compare page, make sure they're what you want to submit.

Showing 1 changed file with 3 additions and 3 deletions. Split Unified

6 ■■■■■ README.md 

...	...	@@ -1,3 +1,3 @@
1	-	# test-area-2
2	-	edit1
3	-	edit2
	1	+ # About me
	2	+
	3	+ My name is Mona Lisa.

- Pull requests are the heart of collaboration on GitHub. When you open a pull request, you're proposing your changes and requesting that someone review and pull in your contribution and merge them into their branch. Pull requests show diffs, or differences, of the content from both branches. The changes, additions, and subtractions are shown in different colors.

- As soon as you make a commit, you can open a pull request and start a discussion, even before the code is finished.
- By using GitHub's @mention feature in your pull request message, you can ask for feedback from specific people or teams, whether they're down the hall or 10 time zones away.

- You can even open pull requests in your own repository and merge them yourself. It's a great way to learn the GitHub flow before working on larger projects.

**EXERCISE - CREATING A PR IN HELLO-WORLD FROM
BRANCH(1-EDIT-README) AND MENTION YOUR
FRIEND'S NAME THAT YOU HAVE FINISHED DOING IT**

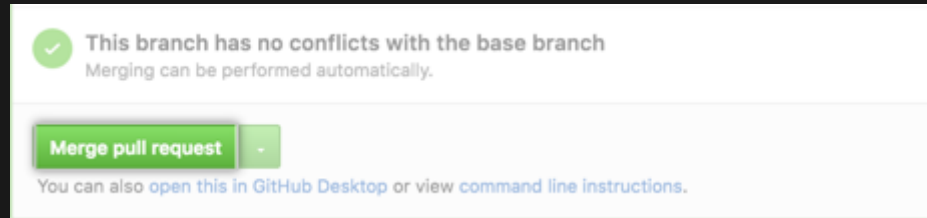
ACTIVITY 01

5. MERGING YOUR PULL REQUEST



**DEMONSTRATION - MERGE YOUR ACTIVITY1-NOTES
BRANCH INTO THE MAIN BRANCH.**

1. Click Merge pull request to merge the changes into main.



2. Click **Confirm merge**. You will receive a message that the request was successfully merged and the request was closed.

3. Click **Delete branch.**

Now that your pull request is merged and your changes are on main, you can safely delete the 1-edit-readme branch. If you want to make more changes to your project, you can always create a new branch and repeat this process.

EXERCISE - MERGE YOUR 1-EDIT-README BRANCH INTO THE MAIN BRANCH IN HELLO-WORLD REPO.

After you merge your pull request, the changes on your 1-edit-readme branch will be incorporated into main.

ACTIVITY 02

CREATE YOUR PROFILE USING README

You can create your custom profile that would be displayed whenever some user visits your profile.

EXERCISE - CREATE A REPO AND NAME IT YOUR USERNAME AND INCLUDE A README FILE WHILE CREATING THE REPO.

- ☒ Use the markdown cheatsheet
- ☐ For comprehensive understanding, refer to this article from Github

Markdown consists of a set of symbols inserted in a document ending with .md suffix to control its structure, formatting, or the relationship between its parts.

ACTIVITY 03

FORK SPOON-KNIFE REPOSITORY

WHY FORK A REPO?

- Propose changes to someone else's project
- Use someone else's project as a starting point for your own idea

AM I VIOLATING ETHICS OR DOING SOMETHING ILLEGAL?

No, Open source software is based on the idea that by sharing code, we can make better, more reliable software.

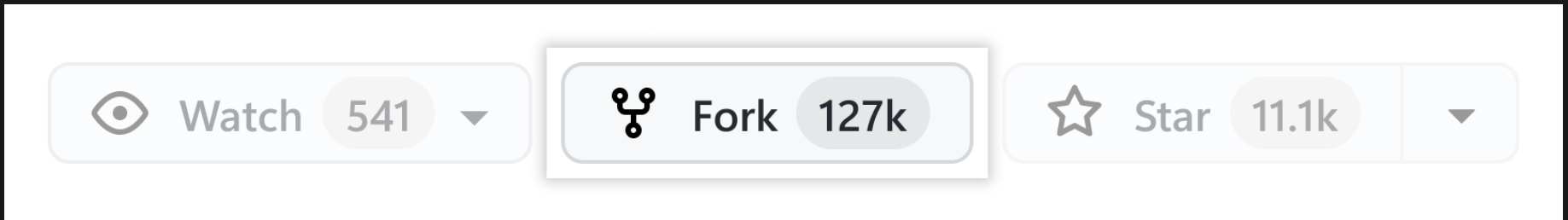
WHO/WHAT DETERMINES WHO CAN USE THE CODE OF A PROJECT?

Repo owner can choose various licenses that determine how you want your project to be shared with others.

ACTIVITY 03

1. FORKING A REPOSITORY

1. On [GitHub.com](https://github.com/octocat/Spoon-Knife), navigate to the [octocat/Spoon-Knife](https://github.com/octocat/Spoon-Knife) repository.
2. In the top-right corner of the page, click Fork.



3. Select an owner for the forked repository.

Create a new fork

A *fork* is a copy of a repository. Forking a repository allows you to freely experiment with changes without affecting the original project. [View existing forks.](#)

Owner *



Repository name *

Spoon-Knife



By default, forks are named the same as their parent repository. You can customize the name to distinguish it further.

4. By default, forks are named the same as their parent repositories. You can change the name of the fork to distinguish it further.

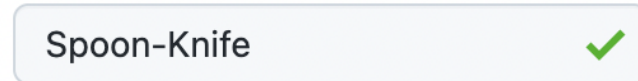
Create a new fork

A *fork* is a copy of a repository. Forking a repository allows you to freely experiment with changes without affecting the original project. [View existing forks.](#)

Owner *



Repository name *



By default, forks are named the same as their parent repository. You can customize the name to distinguish it further.


5. Optionally, add a description of your fork.

Create a new fork

A *fork* is a copy of a repository. Forking a repository allows you to freely experiment with changes without affecting the original project. [View existing forks.](#)

Owner *

Repository name *

 octocat ▾ / Spoon-Knife ✓

By default, forks are named the same as their parent repository. You can customize the name to distinguish it further.

Description (optional)

This repo is for demonstration purposes only.



Choose whether to copy only the default branch or all branches to the new fork. For many forking scenarios, such as contributing to open-source projects, you only need to copy the default branch. By default, only the default branch is copied.

Owner *

Repository name *

Select an owner ▼

/

Spoon-Knife

By default, forks are named the same as their parent repository. You can customize the name to distinguish it further.

Description (optional)

This repo is for demonstration purposes only.

☒ **Copy the `main` branch only**

Contribute back to octocat/Spoon-Knife by adding your own branch. [Learn more.](#)

7. Click Create fork.

☒ Copy the **main** branch only

Contribute back to octocat/Spoon-Knife by adding your own branch. [Learn more.](#)

 You are creating a fork in your personal account.

Create fork

EXERCISE - FORK YOUR FRIEND'S REPO (HELLO-WORLD / NOTES)

- create a separate branch(ex: update-code) from main branch
- **(Up for a challenge?)** open a Pull request from it.

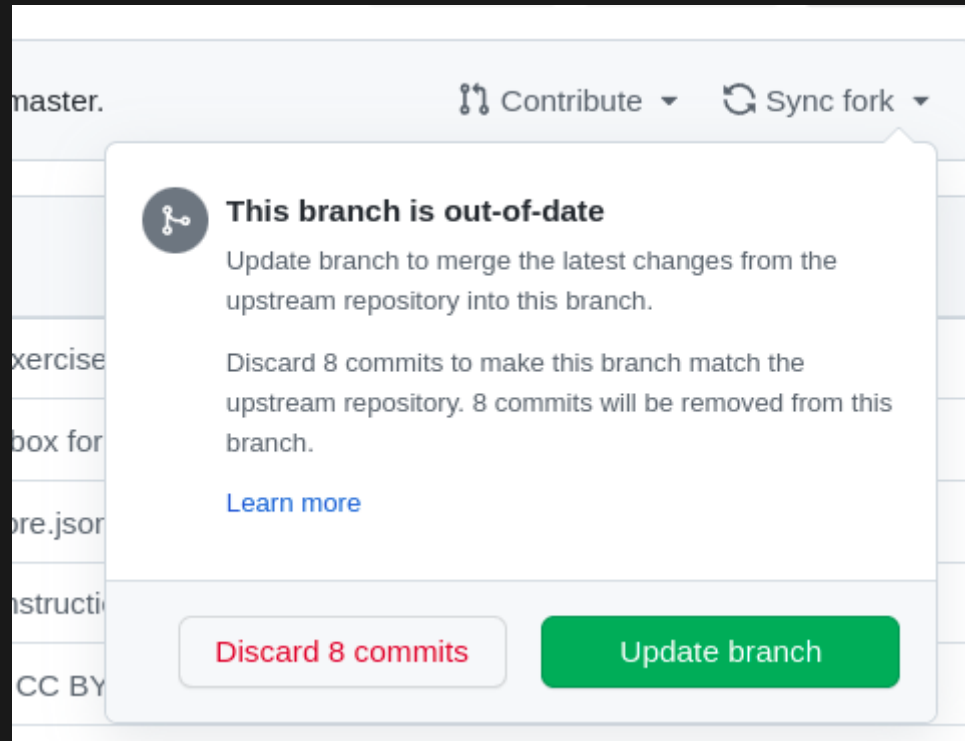


HOW TO UPDATE YOUR FORKED REPO, IF YOUR FRIEND ACCEPTS AND MERGES YOUR PULL REQUEST



After your friend merges your pull request, Go to forked repo.

- You will find **Sync fork** option
- click on **update branch**



COMPLICATED RIGHT?

Please repeat this exercise a number of times to get an idea of whole process

SESSION 1 - Getting Started with Github

THE END

