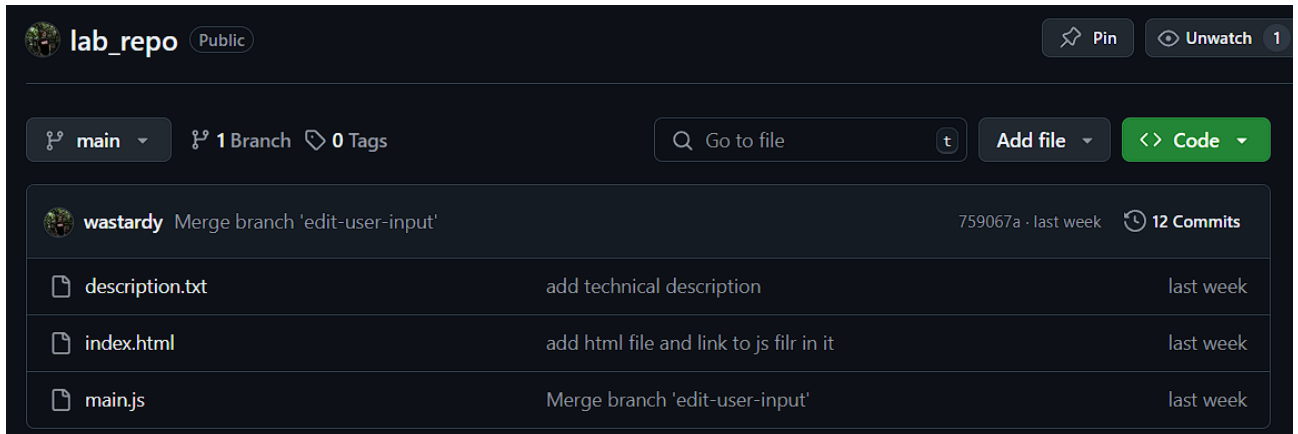
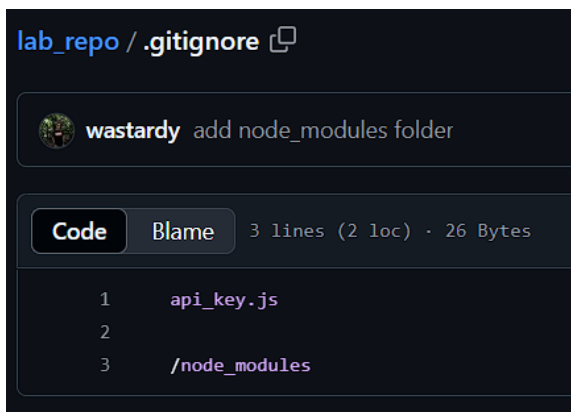


## Виконання лабораторної роботи № 5, робота з GitHub

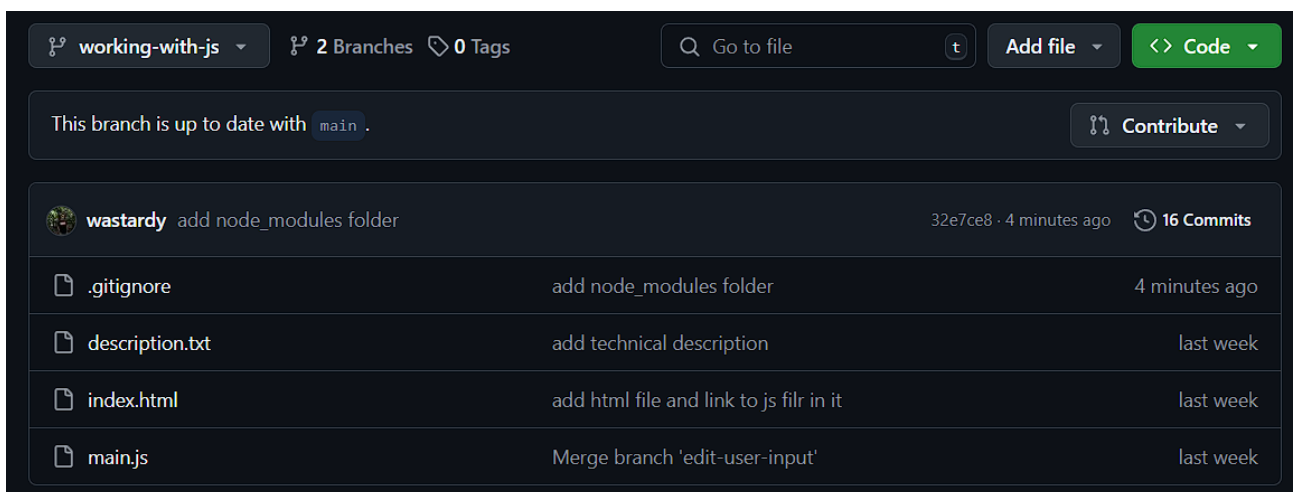
На даний момент в мене є репозиторій, який містить файли з попередньої лабораторної роботи. Тут я і буду вчитись працювати з GitHub



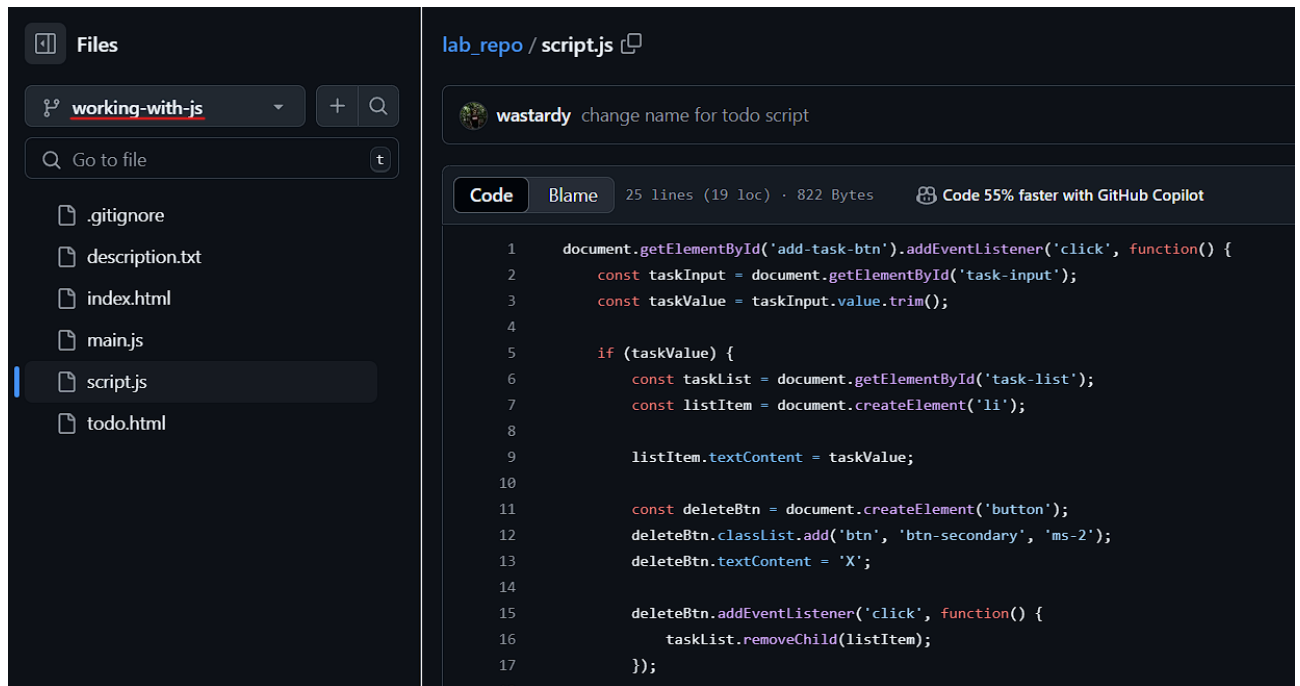
Для того, щоб уникнути завантаження конфіденційних, непотрібних, тимчасових файлів, або модулів, які займають багато місця і є такими, які можна в будь-який момент встановити заново, їх потрібно записувати в гітїгнор. Таким чином ці файли не завантажаться в репозиторій і відповідно їх ніхто не буде бачити в репозиторії після виконання push команди і також не зможе скачати з репозиторію



Тепер створюю нову вітку, для подальшої роботи з нею



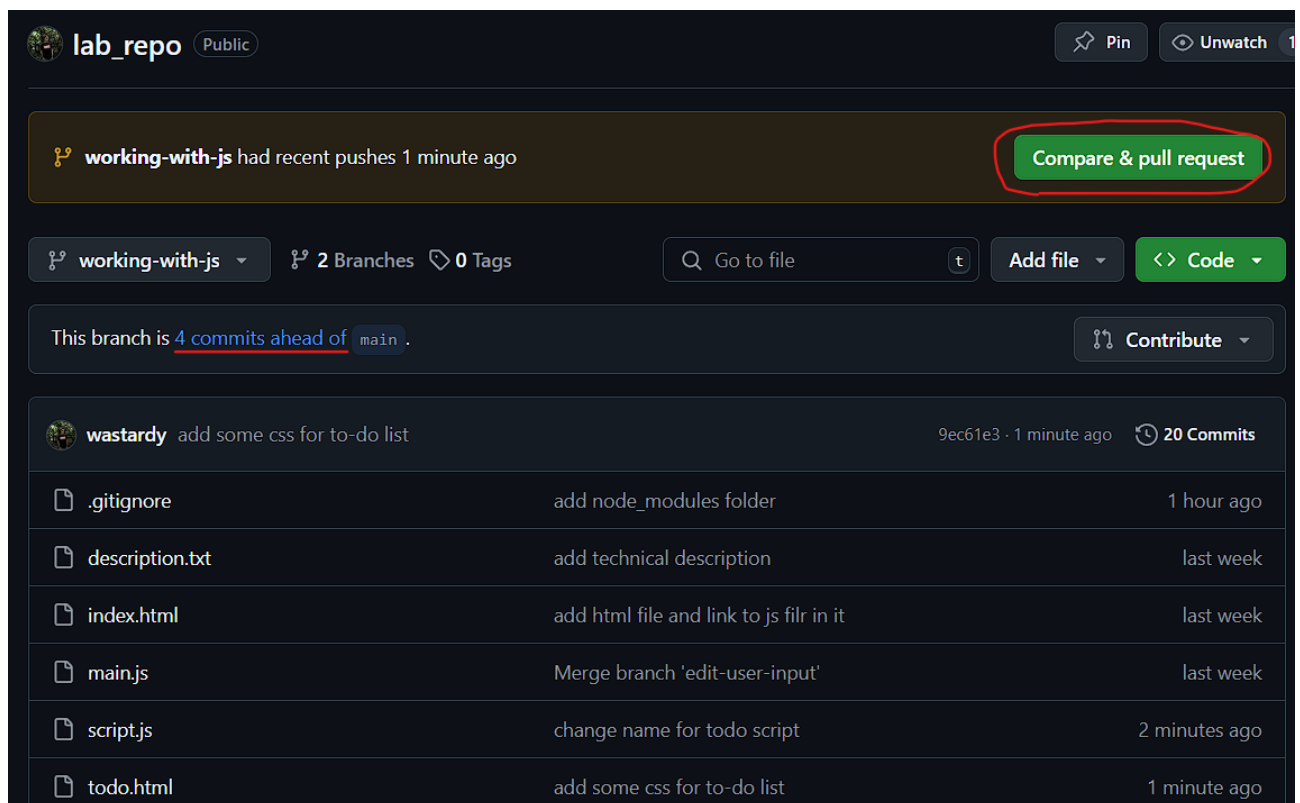
На вітці я хочу розробити програму to-do list на вебсторінці



The screenshot shows the GitHub interface for a repository named 'lab\_repo'. On the left, a file explorer shows the directory structure: .gitignore, description.txt, index.html, main.js, script.js (selected), and todo.html. The main area displays the content of 'script.js' with the following code:

```
1 document.getElementById('add-task-btn').addEventListener('click', function() {
2   const taskInput = document.getElementById('task-input');
3   const taskValue = taskInput.value.trim();
4
5   if (taskValue) {
6     const taskList = document.getElementById('task-list');
7     const listItem = document.createElement('li');
8
9     listItem.textContent = taskValue;
10
11    const deleteBtn = document.createElement('button');
12    deleteBtn.classList.add('btn', 'btn-secondary', 'ms-2');
13    deleteBtn.textContent = 'X';
14
15    deleteBtn.addEventListener('click', function() {
16      taskList.removeChild(listItem);
17    });
18  }
```


GitHub повідомляє, що на вітці working-with-js я зробив 4 комміти від основної гілки і також пропонує виконати pull request









The screenshot shows the GitHub repository page for 'lab\_repo' (Public). A notification bar at the top states: 'working-with-js had recent pushes 1 minute ago'. A green button labeled 'Compare & pull request' is highlighted with a red circle. Below the notification, the repository details show '2 Branches' and '0 Tags'. A status bar indicates 'This branch is 4 commits ahead of main'. The commit history table is as follows:





Commit Message	Commit Hash	Time Ago
add some css for to-do list	9ec61e3	1 minute ago
add node_modules folder		1 hour ago
add technical description		last week
add html file and link to js filr in it		last week
Merge branch 'edit-user-input'		last week
change name for todo script		2 minutes ago
add some css for to-do list		1 minute ago

Вітка main не містить моїх нових файлів, які створив на іншій вітці

 **working-with-js** had recent pushes 5 minutes ago [Compare & pull request](#)




 **main**  **2 Branches**  **0 Tags**   **Add file**  **Code**

 **wastardy** add node\_modules folder 32e7ce8 · 1 hour ago 16 Commits


 .gitignore	add node_modules folder	1 hour ago
 description.txt	add technical description	last week
 index.html	add html file and link to js filr in it	last week
 main.js	Merge branch 'edit-user-input'	last week

Отже щоб з'єднати вітку working-with-js з main, потрібно виконати merge. Для цього я спершу створюю pull request

Discuss and review the changes in this comparison with others. [Learn about pull requests](#) [Create pull request](#)


 **4 commits**  **2 files changed**  **1 contributor**

Commits on Nov 15, 2024

 **wastardy** authored 11 minutes ago


add html file for to-do list

Verified eb5fd63 <>

 **wastardy** authored 10 minutes ago


add script file for to-do list

Verified ca0114b <>

 **wastardy** authored 9 minutes ago

change name for todo script


Verified 0faf593 <>


 **wastardy** authored 7 minutes ago


add some css for to-do list


Verified 9ec61e3 <>


GitHub перевіряє, чи виникнуть merge-конфлікти і повідомляє про це



 **wastardy** added 4 commits 13 minutes ago


 add html file for to-do list Verified eb5fd63

 add script file for to-do list Verified ca0114b

 change name for todo script Verified 0faf593

 add some css for to-do list Verified 9ec61e3

  **Require approval from specific reviewers before merging** [Rulesets](#) ensure specific people approve pull requests before they're merged. [Add rule](#) ×

 **This branch has no conflicts with the base branch**  
Merging can be performed automatically.

[Merge pull request](#) You can also [open this in GitHub Desktop](#) or view [command line instructions](#).

Виконую merge і тепер вітка main містить мій код до To-Do List

The screenshot shows the GitHub interface for a repository named 'lab\_repo'. At the top, there's a header with the repository name, a 'Public' badge, and buttons for 'Pin' and 'Unwatch'. Below this, a navigation bar shows the current branch 'main', '2 Branches', and '0 Tags'. A search bar 'Go to file' and buttons 'Add file' and 'Code' are also present. The main content area displays a merge pull request #1 from 'wastardy/working-with-js'. Below the pull request, a list of files is shown with their commit messages and timestamps:

File	Commit Message	Time
.gitignore	add node_modules folder	1 hour ago
description.txt	add technical description	last week
index.html	add html file and link to js filr in it	last week
main.js	Merge branch 'edit-user-input'	last week
script.js	change name for todo script	12 minutes ago
todo.html	add some css for to-do list	11 minutes ago

Створюю нову вітку, щоб навчитись розв'язувати merge-конфлікт. Змінюю вигляд файлу

The screenshot shows the GitHub web editor interface. On the left, a sidebar lists files: '.gitignore', 'description.txt', and 'index.html'. The main area shows the 'Edit' view of 'description.txt' in the 'conflict-branch'. The content of the file is:

```
1 here is some technical description for my project
2 - i wanna create simple calculator using website
3 - calculator abilities: add, subtract, multiple, divide
4 - user input from web prompts
5 - add To-Do List Program
6 - will be a plus to create Readme file
```

Тепер в main вітці теж зміню цей файл

The screenshot shows the GitHub web editor interface for the 'main' branch. The sidebar lists files: '.gitignore', 'description.txt', 'index.html', 'main.js', and 'script.js'. The main area shows the 'Edit' view of 'description.txt'. The content of the file is:

```
1 here is some technical description for my project
2
3 - i wanna create simple calculator using website
4
5 - calculator abilities: add, subtract, multiple, divide
6
7 - user input from web prompts
8
9 - will be a plus to create Readme file with photos
```

При спробі виконати merge отримую повідомлення про конфлікт, а також кнопку для переходу до вирішення конфлікту

The screenshot shows a GitHub message box with a warning icon and the text: 'This branch has conflicts that must be resolved'. Below this, it says 'Use the [web editor](#) or the [command line](#) to resolve conflicts.' and 'Conflicting files' followed by 'description.txt'. At the bottom, there is a 'Merge pull request' button and a link to 'open this in GitHub Desktop' or 'view [command line instructions](#)'.

Отже вибираю зміни які мені підходять і підтверджую зміни

```
description.txt 1 conflict

1  here is some technical description for my project
2  - i wanna create simple calculator using website
3  - calculator abilities: add, subtract, multiple, divide
4  <<<<<< conflict-branch
5  - user input from web prompts
6  - add To-Do List Program
7  - will be a plus to create Readme file
8  =====
9
10 - user input from web prompts
11
12 - will be a plus to create Readme file with photos
13 >>>>>> main
14
```


```
description.txt 1 conflict Prev ^ Next v ⚙️ Mark as resolved

1  here is some technical description for my project
2  - simple calculator using website
3  - calculator abilities: add, subtract, multiple, divide
4  - user input from web prompts
5
6  - add To-Do List Program
7  - will be a plus to create Readme file with photos
```

Тепер є повідомлення, що конфліктів немає, тому виконую merge

Update description.txt Verified f2c386b

Merge branch 'main' into conflict-branch Verified 10803fc

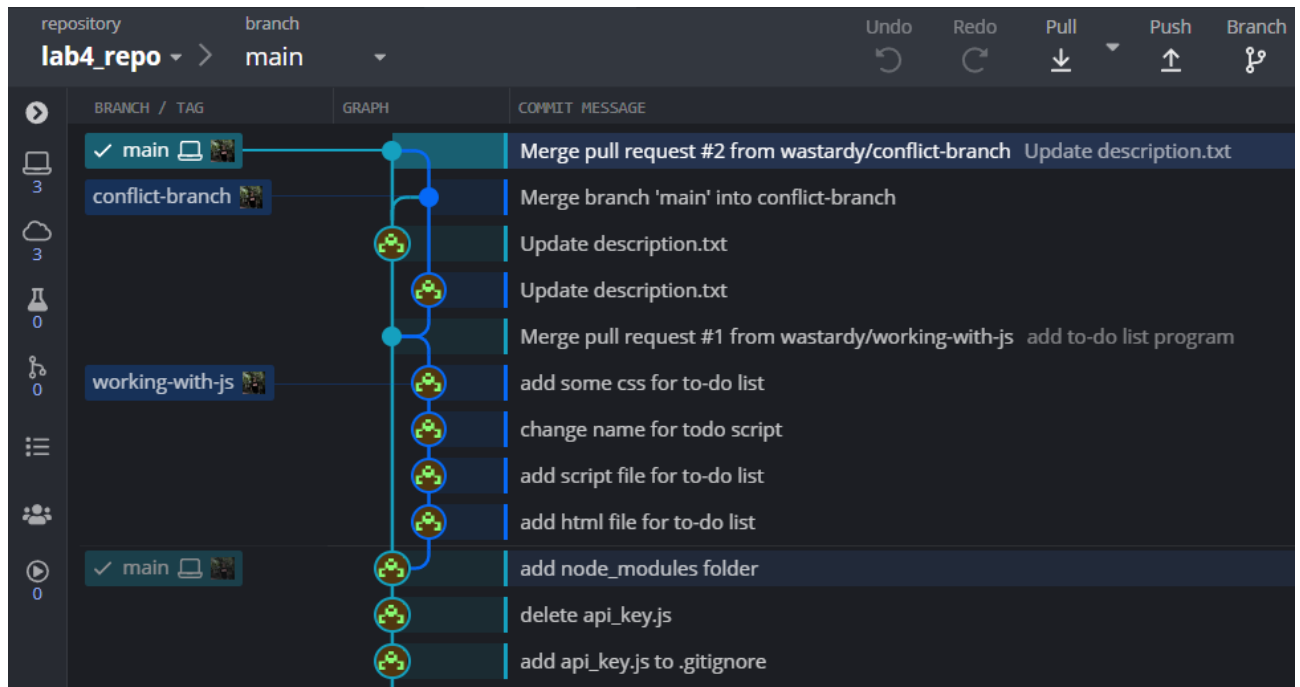
 Require approval from specific reviewers before merging  
[Rulesets](#) ensure specific people approve pull requests before they're merged. Add rule ×

✓

 This branch has no conflicts with the base branch  
Merging can be performed automatically.

Merge pull request ▾ You can also [open this in GitHub Desktop](#) or view [command line instructions](#).

Якщо глянути на дерево коммітів, бачу, що коли виконується merge, то мої зміни з вітки переносяться на головну вітку



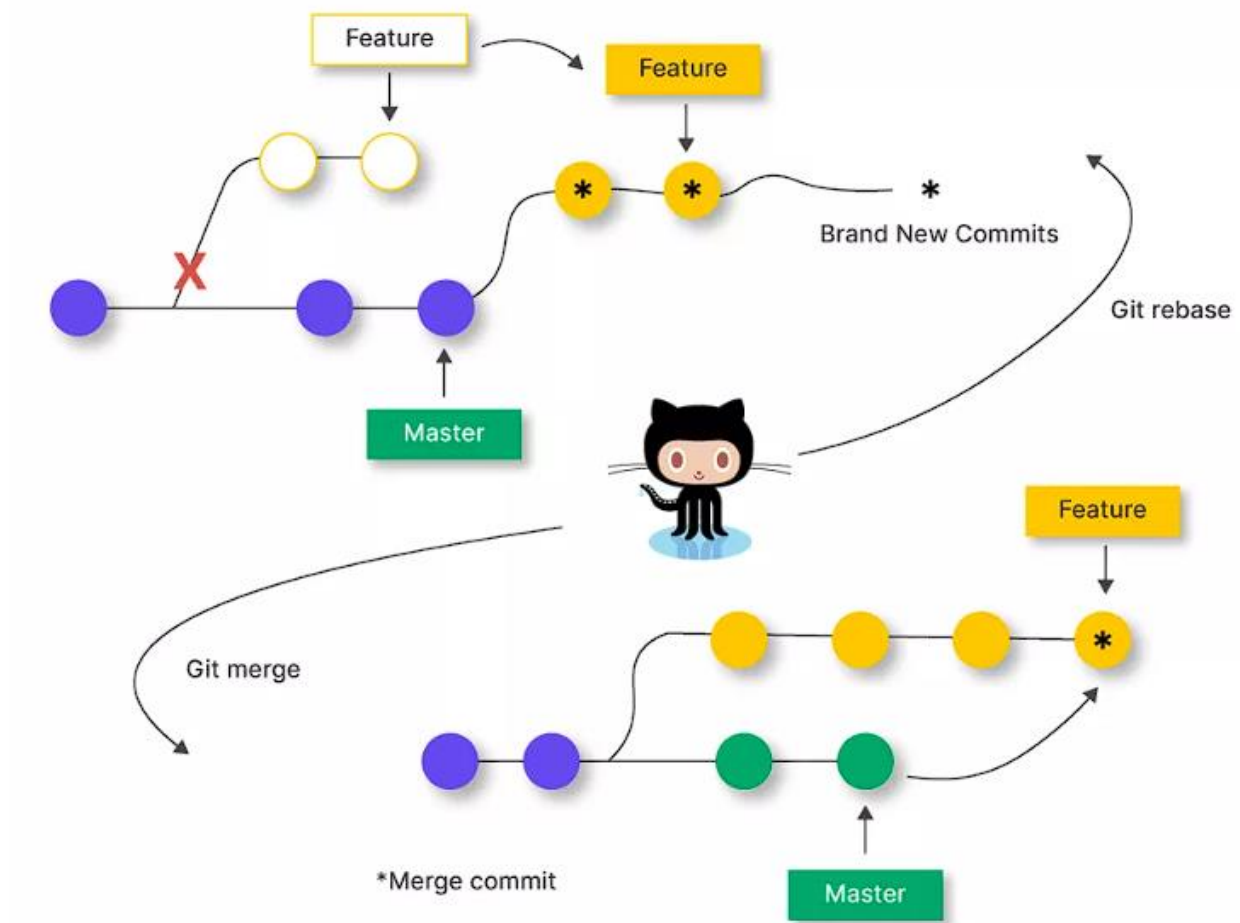
Також merge зберігає історію коммітів

Commits			
main	All users	All time	
Commits on Nov 15, 2024			
Merge pull request #2 from wastardy/conflict-branch	Verified	542ec37	<>
Merge branch 'main' into conflict-branch	Verified	18803fc	<>
Update description.txt	Verified	36faa8b	<>
Update description.txt	Verified	f2c386b	<>
Merge pull request #1 from wastardy/working-with-js	Verified	fe50bb8	<>
add some css for to-do list	Verified	9ec61e3	<>
change name for todo script	Verified	0faF593	<>
add script file for to-do list	Verified	ca0114b	<>

Але при великому масиві віток і розробників, після кожного мурджу робиться комміт, таких merge-коммітів може бути дуже багато й історія коммітів стане великою і не читабельною

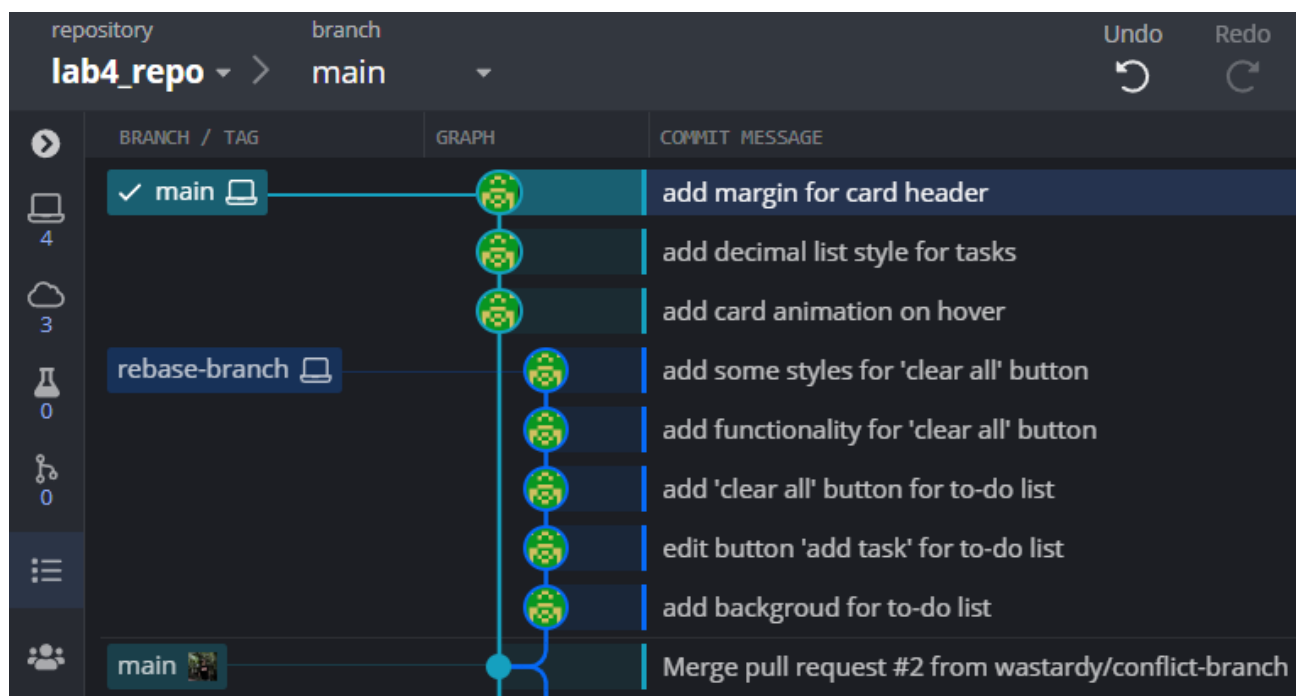
Тому для оновлення моєї вітки від main краще використовувати rebase команду. Вона замість того, щоб робити merge-комміт, змінить базову точку для моєї вітки і вона буде виглядати так, ніби всі її комміти були створені поверх main. Так історія розробки буде лінійною і зрозумілою для читання.

Візуальне порівняння:



\* я не знайшов, як це робити на GitHub, тому просто виконаю rebase в терміналі і покажу візуальні зміни

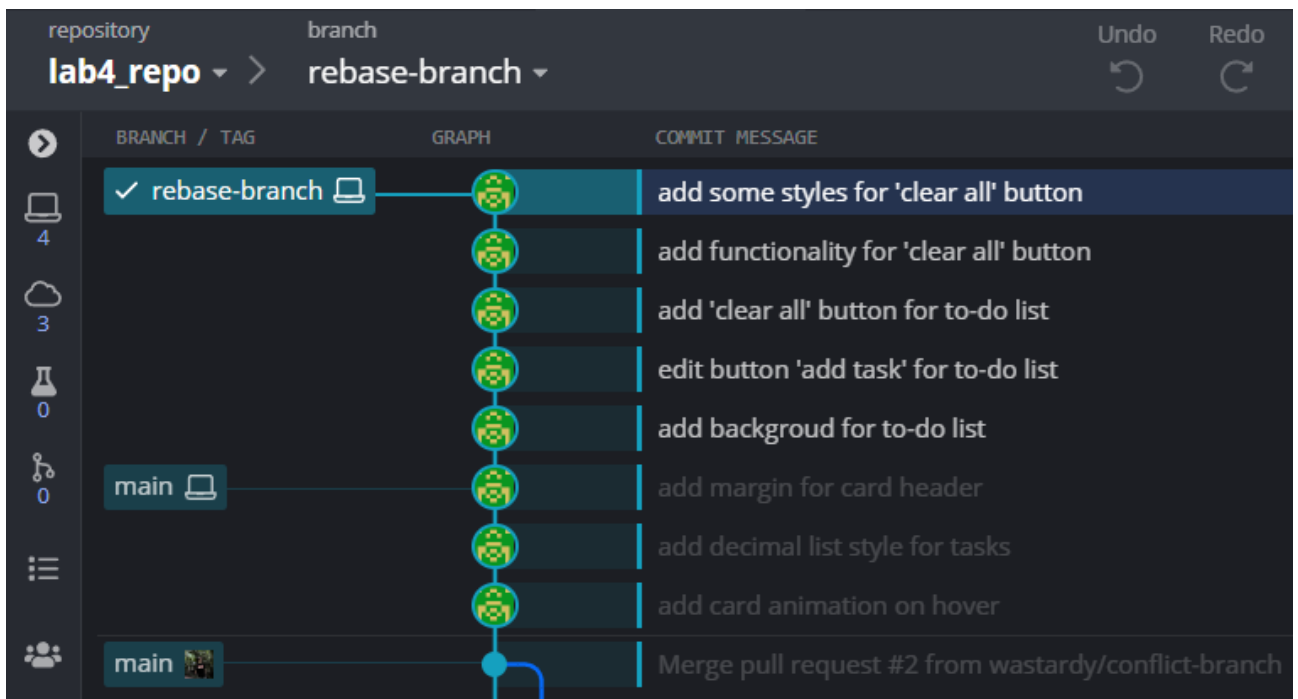
Отже на момент ДО rebase дерево коммітів виглядає так:



Я можу зробити мердж комміт собі в rebase-branch, але тоді в мене буде непотрібний комміт в історії, якого я хочу уникнути. Тому я виконаю rebase

```
PS C:\Users\Користувач\OneDrive\Робочий стіл\git\lab4_repo> git branch
develop-subtract-func
edit-user-input
* main
rebase-branch
PS C:\Users\Користувач\OneDrive\Робочий стіл\git\lab4_repo> git rebase main rebase-branch
Successfully rebased and updated refs/heads/rebase-branch.
PS C:\Users\Користувач\OneDrive\Робочий стіл\git\lab4_repo>
```

Тепер моя вітка rebase-branch починається з останнього комміту вітки main, + на вітці rebase-branch немає merge-комміту, тому історія буде виглядати набагато краще)



Якщо є вітка, яка вже не потрібна, її можна видалити у вкладці branches. Якщо не виконати merge, вся робота з вітки зникне разом з нею

Branches						New branch
Overview Yours Active Stale All						
Search branches...						
Default						
Branch	Updated	Check status	Behind	Ahead	Pull request	
main	3 minutes ago				Default	🗑️ ⋮
Your branches						
Branch	Updated	Check status	Behind	Ahead	Pull request	
tem-branch	now		2	1		🗑️ ⋮
conflict-branch	1 hour ago		12	0	#2	🗑️ ⋮
working-with-js	2 hours ago		16	0	#1	🗑️ ⋮