



# Git and GitHub Fundamentals

Version 2.0

The world's largest global training provider

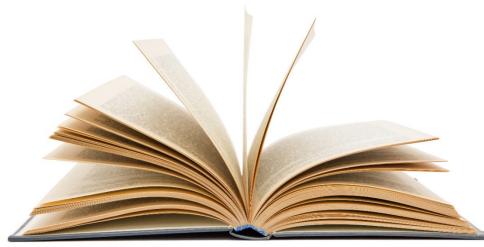
---

 | [info@theknowledgeacademy.com](mailto:info@theknowledgeacademy.com)

 | [theknowledgeacademy.com](http://theknowledgeacademy.com)

*the*knowledgeacademy

## Git and GitHub Fundamentals



© 2021 The Knowledge Academy Ltd

1

## About The Knowledge Academy

- World Class Training Solutions
- Subject Matter Experts
- Highest Quality Training Material
- Accelerated Learning Techniques
- Project, Programme, and Change Management, ITIL® Consultancy
- Bespoke Tailor Made Training Solutions
- PRINCE2®, MSP®, ITIL®, Soft Skills, and More



Project Management & Leadership

Business Architecture & Systems Design

Technical & Programming

Computer Software

*the*knowledgeacademy

Business & Corporate

Web, Marketing, PR & Social

Legal, Finance & Accounting

Workplace Requirements

© 2021 The Knowledge Academy Ltd

2

1

# Administration

- Trainer
- Fire Procedures
- Facilities
- Days/Times
- Breaks
- Additional Requirements
- Delegate ID check
- Phones and Mobile devices



*theknowledgeacademy*

# Outlines

- **Module 1:** Introducing Git and GitHub
- **Module 2:** Creating a Repository
- **Module 3:** Pull Requests
- **Module 4:** Social Networking



*theknowledgeacademy*

# Outlines

- **Module 5:** Forking a Repository
- **Module 6:** Git Branching Tools
- **Module 7:** Understanding Change Logs
- **Module 8:** Managing GitHub



*theknowledgeacademy*

# Outlines

- **Module 9:** Securing GitHub
- **Module 10:** Automating GitHub



*theknowledgeacademy*

# Module 1: Introducing Git and GitHub

*theknowledgeacademy*

## Introducing Git and GitHub

- Git is an open source code management system
- Git helps to keep track of different versions of text or code, so users can easily compare what has changed
- Linus Torvalds developed the Git that is utilised by numerous open source projects, most notably perhaps the Linux kernel. Git Lexicon:



*theknowledgeacademy*

# Introducing Git and GitHub

## GitHub

- It is a code hosting platform for collaboration and version control
- The user can edit files, view old versions of files, and access it from anywhere in the world
- It allows the user and other individuals to work together on projects from anyplace



*the*knowledgeacademy

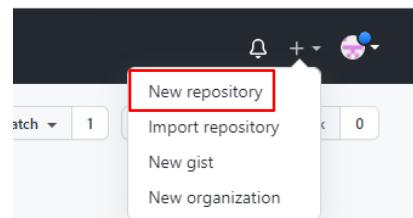
## Module 2: Creating a Repository

*the*knowledgeacademy

# Creating a Repository

- The following are the steps for creating a Repository

**Step 1:** Click on the drop-down menu in the upper-right corner of any page and choose **New repository**



the knowledge academy

© 2021 The Knowledge Academy Ltd

11

# Creating a Repository

**Step 2:** Enter a short name for your repository

## Create a new repository

A repository contains all project files, including the revision history. Already have a project repository elsewhere?  
[Import a repository.](#)

Owner \*

Repository name \*  

Great repository names are simple. Demo is available. Ie. Need inspiration? How about silver-barnacle?

the knowledge academy

© 2021 The Knowledge Academy Ltd

12

# Creating a Repository

## Step 3: Select repository visibility

### 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.

theknowledgeacademy

© 2021 The Knowledge Academy Ltd

13

# Creating a Repository

## Step 4: Choose Initialize this repository with Add a README file

### 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](#).

theknowledgeacademy

© 2021 The Knowledge Academy Ltd

14

# Creating a Repository

**Step 5:** Click on the **Create repository**

The screenshot shows the GitHub interface for creating a new repository. On the left, there is a message: "This will set `main` as the default branch. Change the default name in your [settings](#)." Below this is a green button labeled "Create repository" with a red border. A blue arrow points from this button to the right side of the screen. On the right, the repository page for "Delegateone / Demo" is shown. It has a header with navigation links: Code, Issues, Pull requests, Actions, Projects, Wiki, Security, and Insights. Below the header, it shows 1 main branch, 0 branches, 0 tags, and 1 commit. The commit is titled "Delegateone Initial commit" and was made by "Delegateone" at "10:57:13 in 2 minutes". The commit message is "Initial commit". The README file contains the text "Demo". At the bottom right of the slide, there is a watermark for "the knowledge academy".

© 2021 The Knowledge Academy Ltd

15

# Module 3: Pull Requests



© 2021 The Knowledge Academy Ltd

16

# About Pull Requests

- Pull requests helps the user to notify others about the modifications that are done by the user to a repository on GitHub
- After opening a pull request, a user can review and discuss the possible modifications with collaborators and add follow-up commits before merging the changes into the repository
- Once a pull request is initialised, the user can view a review page that displays an overview of the changes between the repository's base and the compare branch
- The summary of the recommended changes can be added by the user. The user can also review the modifications made by add labels, commits, assignees, milestones, and @mention individual contributors or teams

the knowledge academy

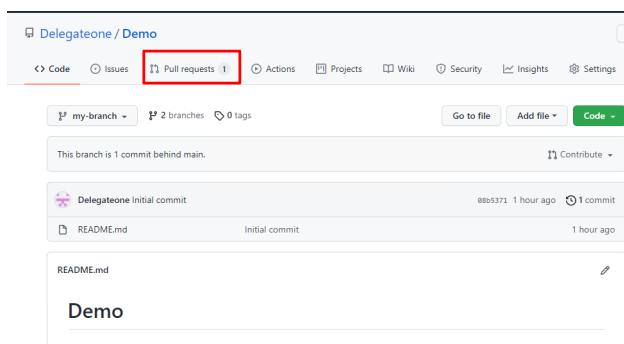
© 2021 The Knowledge Academy Ltd

17

# Creating Pull Requests

- The following are the steps for Creating Pull Request

## Step 1: Go to Pull request



The screenshot shows a GitHub repository interface for the 'Delegateone / Demo' repository. The top navigation bar has tabs for 'Code', 'Issues', 'Pull requests' (which is highlighted with a red box), 'Actions', 'Projects', 'Wiki', 'Security', 'Insights', and 'Settings'. Below the tabs, there are buttons for 'Go to file', 'Add file', and 'Code'. A status message says 'my-branch' is 1 commit behind 'main'. A commit card for 'Delegateone Initial commit' is shown, along with a diff viewer for 'README.md'. The commit was made 1 hour ago by user 'Delegateone' with 1 commit.

the knowledge academy

© 2021 The Knowledge Academy Ltd

18

# Creating Pull Requests

**Step 2:** Click on the **New pull request**

The screenshot shows a GitHub repository named 'Delegateone / Demo'. The 'Pull requests' tab is selected, indicated by a red underline. At the top right of the pull request list, there is a green button labeled 'New pull request' with a red border around it. The rest of the interface includes navigation tabs like 'Code', 'Issues', 'Actions', 'Projects', 'Wiki', 'Security', 'Insights', and 'Settings'. There are also filters for 'Labels', 'Milestones', and a search bar.

the knowledge academy

© 2021 The Knowledge Academy Ltd

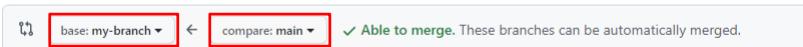
19

# Creating Pull Requests

**Step 3:** Use the **base** branch drop-down menu to choose the branch you want to merge your changes into, then use the **compare** branch drop-down menu to select the topic branch into which you committed your changes

## Comparing changes

Choose two branches to see what's changed or to start a new pull request. If you need to, you can also [compare across forks](#).



the knowledge academy

© 2021 The Knowledge Academy Ltd

20

10

# Creating Pull Requests

**Step 4:** Click on the **Create pull request**

The screenshot shows a GitHub repository page for 'Delegateone / Demo'. The user is comparing two branches: 'base: my-branch' and 'compare: main'. A message at the top says 'Able to merge. These branches can be automatically merged.' Below this, there's a text area for 'Discuss and review the changes in this comparison with others.' and a prominent green 'Create pull request' button. At the bottom, it shows '1 commit', '1 file changed', '0 comments', and '1 contributor'.

the knowledge academy

© 2021 The Knowledge Academy Ltd

21

# Creating Pull Requests

**Step 5:** Enter a description and title for your pull request

The screenshot shows the GitHub 'Create pull request' form. It has sections for 'Title' (with 'Add files via upload' and 'Leave a comment' fields), 'Description' (with 'Attach files by dragging & dropping, selecting or pasting them.'), and various settings like 'Reviewers', 'Assignees', 'Labels', 'Projects', 'Milestone', and 'Linked issues'. A note at the bottom says 'Remember, contributions to this repository should follow our GitHub Community Guidelines.'

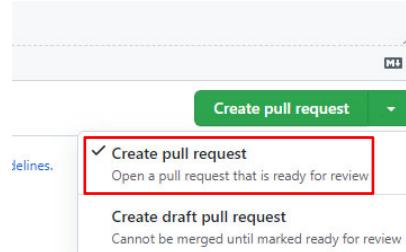
the knowledge academy

© 2021 The Knowledge Academy Ltd

22

# Creating Pull Requests

**Step 6:** To create a pull request that is ready for review, click on the **Create pull request**. To create a draft pull request, use the drop-down and choose to **Create draft pull request**, then click on the Draft Pull Request



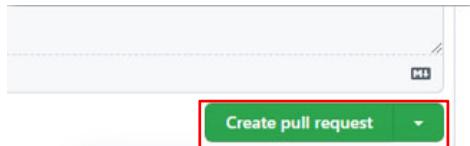
theknowledgeacademy

© 2021 The Knowledge Academy Ltd

23

# Creating Pull Requests

**Step 7:** Click on the **Create pull request**



theknowledgeacademy

© 2021 The Knowledge Academy Ltd

24

# Pull Requests and Forks

- The following are the steps for Creating a pull request from a fork:

## Step 1: Go to Pull request

A screenshot of a GitHub repository page for 'Delegateone / Demo'. The 'Pull requests' tab is selected, highlighted with a red box. The page shows one pull request: 'Delegateone Initial commit' by Delegateone, which has been merged. The commit message is 'Initial commit'. The pull request was created 1 hour ago and merged 1 hour ago. The repository has 2 branches and 0 tags.

the knowledge academy

© 2021 The Knowledge Academy Ltd

25

# Pull Requests and Forks

## Step 2: Click on the New pull request

A screenshot of a GitHub repository page for 'Delegateone / Demo'. The 'Pull requests' tab is selected. A modal window titled 'Label issues and pull requests for new contributors' is open, prompting the user to help potential first-time contributors discover issues labeled with 'good first issue'. At the bottom right of the modal, a red box highlights the 'New pull request' button.

the knowledge academy

© 2021 The Knowledge Academy Ltd

26

# Pull Requests and Forks

**Step 3:** Click on the **compare across forks**

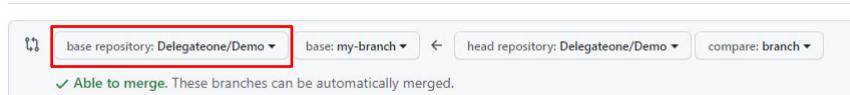
## Compare changes

Compare changes across branches, commits, tags, and more below. If you need to, you can also [compare across forks.](#)

*the*knowledgeacademy

# Pull Requests and Forks

**Step 4:** In the **base branch** drop-down menu, choose the branch of the upstream repository you want to merge changes into

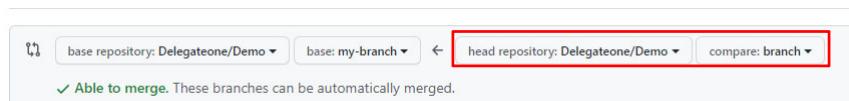


The screenshot shows a user interface for a pull request. At the top, there are four dropdown menus: 'base repository' set to 'Delegateone/Demo', 'base' set to 'my-branch', 'head repository' set to 'Delegateone/Demo', and 'compare' set to 'branch'. Below these menus, a green checkmark icon and the text 'Able to merge. These branches can be automatically merged.' are displayed.

*the*knowledgeacademy

# Pull Requests and Forks

**Step 5:** Choose your fork in the **head fork** drop-down menu, then use the **compare branch** drop-down menu to choose the branch you made your changes in



the knowledge academy

© 2021 The Knowledge Academy Ltd

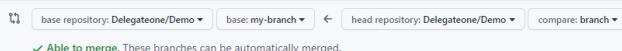
29

# Pull Requests and Forks

**Step 6:** Click on the **Create pull request**

## Comparing changes

Choose two branches to see what's changed or to start a new pull request. If you need to, you can also compare across forks.



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

**Create pull request**

the knowledge academy

© 2021 The Knowledge Academy Ltd

30

# Pull Requests and Forks

**Step 7:** Enter a title and description for your pull request

Open a pull request

Create a new pull request by comparing changes across two branches. If you need to, you can also compare across forks.

base: my-branch ▾ ← compare: branch ▾ ✓ Able to merge. These branches can be automatically merged.

New pull request

Write Preview H B I  $\neq$   $\leftrightarrow$   $\oplus$   $\exists$   $\exists$   $\otimes$   $\otimes$   $\otimes$   $\otimes$   $\otimes$

Testing

Attach files by dragging & dropping, selecting or pasting them.

Create pull request

Remember, contributions to this repository should follow our GitHub Community Guidelines.



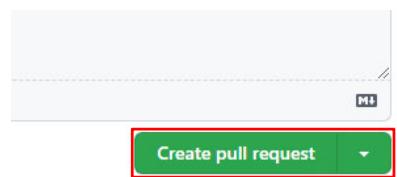
the knowledge academy

© 2021 The Knowledge Academy Ltd

31

# Pull Requests and Forks

**Step 8:** Click on the **Create pull request** button



the knowledge academy

© 2021 The Knowledge Academy Ltd

32

## Module 4: Social Networking

*theknowledgeacademy*

### Using GitHub as a social network

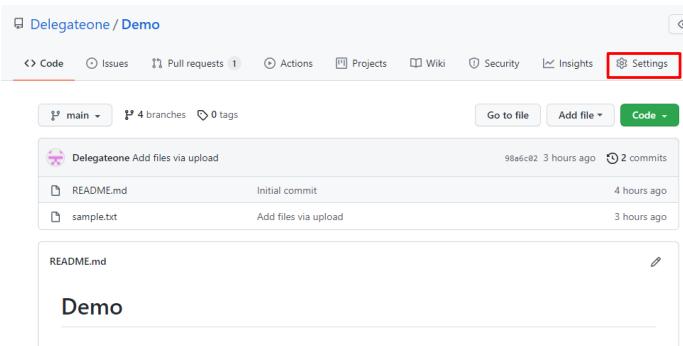
- GitHub is a social network built. Programmers can collaborate around repositories of code by using the GitHub which enables them to fork, download, and commit back code in any format
- GitHub repositories can include text files, HTML, and various other formats that can be edited by anybody, which allows the power of version control and collaboration to occur between anyone, in any business sector
- GitHub holds a phenomenal amount of untapped potential. For effective and meaningful social interactions, GitHub opens up unlimited opportunities
- With social interactions anyone can take benefit from the successful phases of open source software development

*theknowledgeacademy*

# Collaboration

- The following are the steps for Inviting collaborators to a personal repository:

## Step 1: Click on the Settings



The screenshot shows a GitHub repository named 'Delegateone / Demo'. The 'Code' tab is selected. At the top right, there is a red box around the 'Settings' tab. Below the tabs, there is a summary bar with 'main' branch, '4 branches', '0 tags', and file counts for 'README.md' and 'sample.txt'. A commit history is shown with three commits: 'Add files via upload' by 'Delegateone' (3 hours ago), 'Initial commit' by 'Delegateone' (4 hours ago), and another 'Add files via upload' (3 hours ago). The repository name 'Demo' is displayed at the bottom.

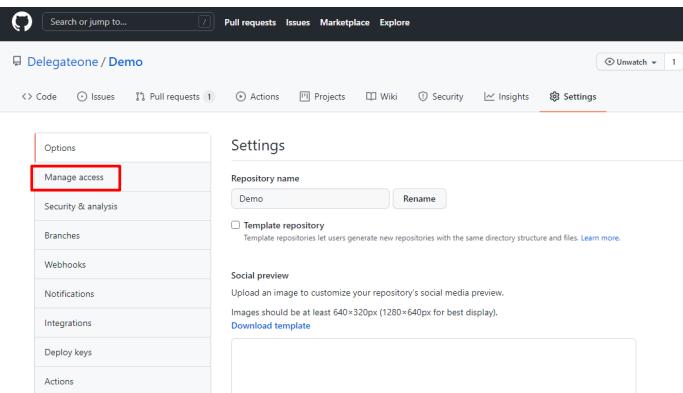
the knowledge academy

© 2021 The Knowledge Academy Ltd

35

# Collaboration

## Step 2: Click on the **Manage access**



The screenshot shows the 'Settings' page for the same GitHub repository. On the left, there is a sidebar with options: 'Options' (selected), 'Manage access' (highlighted with a red box), 'Security & analysis', 'Branches', 'Webhooks', 'Notifications', 'Integrations', 'Deploy keys', and 'Actions'. The main area is titled 'Settings' and contains fields for 'Repository name' (set to 'Demo') and 'Rename'. There is also a checkbox for 'Template repository' and a section for 'Social preview' with a placeholder for a social media preview image.

the knowledge academy

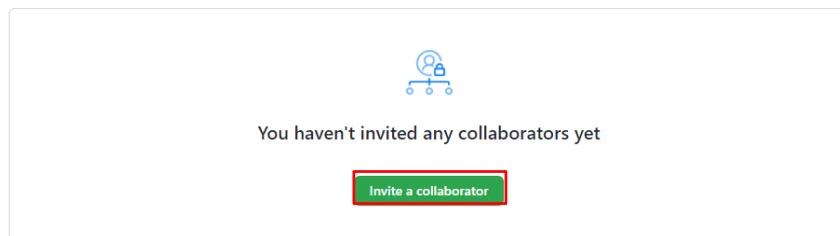
© 2021 The Knowledge Academy Ltd

36

# Collaboration

**Step 3:** Click on the **Invite a collaborator**

Manage access



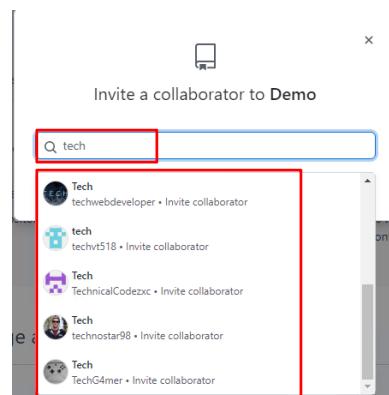
*the*knowledgeacademy

© 2021 The Knowledge Academy Ltd

37

# Collaboration

**Step 4:** Enter the name of the person you need to invite, then click on a name in the list of matches



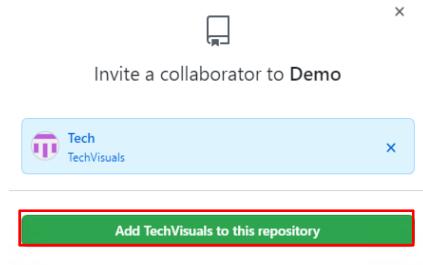
*the*knowledgeacademy

© 2021 The Knowledge Academy Ltd

38

# Collaboration

**Step 5:** Click on the Add NAME to REPOSITORY



**Step 6:** The user will receive an email inviting them to the repository. You will have collaborative access to your repository once they accept your invitation

*the*knowledgeacademy

© 2021 The Knowledge Academy Ltd

39

# Module 5: Forking a Repository

*the*knowledgeacademy

© 2021 The Knowledge Academy Ltd

40

# How to Fork

- A fork can be described as a copy of a repository
- Forking a repository enables the user to experiment with changes without influencing the original project
- Forks are utilised to either use someone else's project as a beginning point for your own idea or to propose changes to someone else's project
- The following are the steps for Forking a repository:

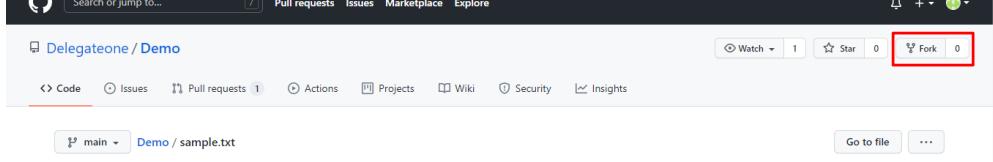
**Step 1:** To share the fork repository, copy the URL of the fork repository and paste in the another browser where the second user is logged in



The screenshot shows a web browser window with the URL bar containing <https://github.com/Delegateone/Demo/blob/main/sample.txt>. The URL is highlighted with a red box. The browser interface includes back, forward, and search buttons. The page title is "Delegateone / Demo". On the right side of the browser, there is a watermark for "the knowledge academy". At the bottom left, it says "© 2021 The Knowledge Academy Ltd".

# How to Fork

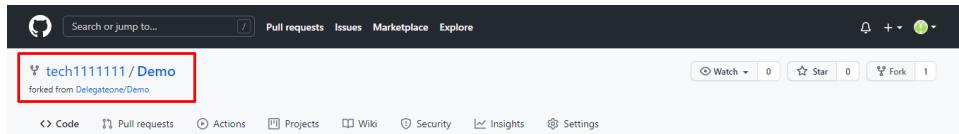
**Step 2:** Click on the **Fork** button



The screenshot shows a GitHub repository page for "Delegateone / Demo". The "Code" tab is selected. At the top right, there are buttons for "Watch" (with 1 follower), "Star" (with 0 stars), and "Fork" (with 0 forks). The "Fork" button is highlighted with a red box. Below the header, there are links for "Issues", "Pull requests", "Actions", "Projects", "Wiki", and "Insights". At the bottom, there are buttons for "Go to file" and "...". The page title is "Delegateone / Demo". On the right side, there is a watermark for "the knowledge academy". At the bottom left, it says "© 2021 The Knowledge Academy Ltd".

# How to Fork

**Step 3:** Your repository has been forked



the knowledge academy

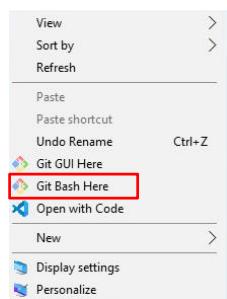
© 2021 The Knowledge Academy Ltd

43

# Keeping a Fork Synced

- The following are the steps for Syncing a fork from the command line

**Step 1:** Open Git Bash



the knowledge academy

© 2021 The Knowledge Academy Ltd

44

# Keeping a Fork Synced

**Step 2:** Change the current working directory to the local project

```
Administrator@EC2AMAZ-J34LJMS MINGW64 ~/Desktop
$ cd Demo/Demo/
Administrator@EC2AMAZ-J34LJMS MINGW64 ~/Desktop/Demo/Demo (main)
$ |
```

*the*knowledgeacademy

© 2021 The Knowledge Academy Ltd

45

# Keeping a Fork Synced

**Step 3:** From the upstream repository, Fetch the branches and their respective commits.  
Commits to BRANCHNAME will be saved in the local branch upstream/BRANCHNAME

```
Administrator@EC2AMAZ-J34LJMS MINGW64 ~/Desktop/Demo/Demo (main)
$ git fetch upstream
From https://github.com/Delegateone/Demo
 * [new branch]      My-branch-2 -> upstream/My-branch-2
 * [new branch]      branch      -> upstream/branch
 * [new branch]      main        -> upstream/main
 * [new branch]      my-branch   -> upstream/my-branch
```

*the*knowledgeacademy

© 2021 The Knowledge Academy Ltd

46

# Keeping a Fork Synced

**Step 4:** Merge the changes from the upstream default branch

```
Administrator@EC2AMAZ-334LJMS MINGW64 ~/Desktop/Demo/Demo (main)
$ git merge upstream/main
Already up to date.
```

- In this case it is already up to date

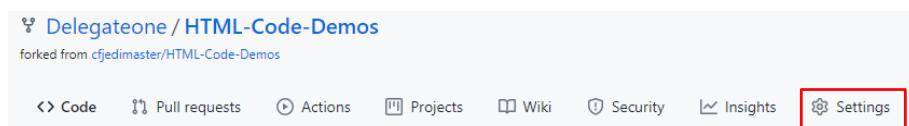
# Module 6: Git Branching Tools

# Archiving repositories

- The following are the steps for archiving repositories:

**Step 1:** Navigate to the main page of the repository

**Step 2:** click on the Settings, under your repository name



the knowledge academy

© 2021 The Knowledge Academy Ltd

49

# Archiving repositories

**Step 3:** Click on **Archive this repository**, Under **Danger Zone**

## Danger Zone

### Change repository visibility

For security reasons, you cannot change the visibility of a fork.

[Change visibility](#)

### Transfer ownership

Transfer this repository to another user or to an organization where you have the ability to create repositories.

[Transfer](#)

### Archive this repository

Mark this repository as archived and read-only.

[Archive this repository](#)

### Delete this repository

Once you delete a repository, there is no going back. Please be certain.

[Delete this repository](#)

the knowledge academy

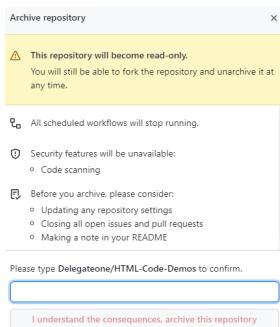
© 2021 The Knowledge Academy Ltd

50

# Archiving repositories

**Step 4:** Read the warnings and type the name of the repository you need to archive

**Step 5:** Click on the I understand the consequences, archive this repository



the knowledge academy

© 2021 The Knowledge Academy Ltd

51

# Showing Recent Branches

- Working on several branches mean that you did not remember which branch you worked on last week

```
Administrator@EC2AMAZ-J34LJMS MINGW64 ~/Desktop/Demo/Demo (main)
$ git branch -a
* main
  remotes/origin/HEAD -> origin/main
  remotes/origin/My-branch-2
  remotes/origin/branch
  remotes/origin/main
  remotes/origin/my-branch
  remotes/upstream/My-branch-2
  remotes/upstream/branch
  remotes/upstream/main
  remotes/upstream/my-branch
```

- Lists the several branches checked out over the history, including the date and last commit date on that branch

the knowledge academy

© 2021 The Knowledge Academy Ltd

52

## Better Branch Descriptions

- If there is upstream, then it will also display what is already merged into upstream by patch

```
git branch-description [branchname] [upstream]
```

- And install the git-post-checkout-nagging-hook as your .git/hooks/post-checkout to make sure you get reminded to set the branch description

## Creating Patch Sets

- creating a patch for this single file that was edited. git diff > gfg-intro.patch will be used where gfg-intro is the patch name

```
Administrator@EC2AMAZ-J34LJMS MINGW64 ~/Desktop/Demo/Demo (main)
$ git diff > gfg-intro.patch

Administrator@EC2AMAZ-J34LJMS MINGW64 ~/Desktop/Demo/Demo (main)
$ ls
Demo.txt  README.md  gfg-intro.patch  sample.txt  some-changes.patch
```

# Splitting Commits

- During rebase when the user realises that the commit needs to be divided into two

```
$> git rebase -i  
# mark commit to split as "edit"  
$> git split-commit
```

- The tool calls git add -p, that allows the user to add the required hunks
- Finish up; it will re-commit with the identical message and drop the user back to the shell to finish up and proceed

# Squashing Commits

- Git config's rebase.autosquash allows automatic squashing and re-ordering of commits beginning with "fixup!" or "squash!", followed by a subject line that matches a previous commit
- Git-squash is a wrapper script around that so that it can be remembered easily

```
$> git add myfile yourfile  
$> git squash  
[1] this is the most recent commit  
[2] previous commit  
[3] some other commit  
[4] a commit message  
Number to squash in: 3
```

## Commit Message Tags

- Some projects request Signed-off-by, Bugzilla references, and Reviewed-by to be added to a commit message
- The script to add these to commit messages git-add-tag is used – see below

```
$> git add-tag HEAD~6..
```

- An editor window will be opened
- Write the line which is required to be added to the commit range and then exit. The line will be added to all the commits in the range

## Testing Commits in Sequence

- Once a set of patches are merged, assure that every patch fulfils certain conditions
- \$> git for-each origin/master.. make check
- From origin/master to HEAD, runs make check on every commit and prints a log

# Module 7: Understanding Change Logs

*the*knowledgeacademy

## Understanding Change Logs

- A changelog is a file that shares a chronologically ordered list of your changes to your project
- It is often organised by the version with the date followed by a list of added, removed, and improved features
- There are two methods to write a changelog:
  - the usual way: create a text file and start to enumerate all your changes with a definite date
  - the developer choice (alias the lazy option): auto-generate your changelog from your commit messages

*the*knowledgeacademy

# Understanding Change Logs

(Continued)

- A changelog is a summary of all your changes. It should be simple to understand by both the users using your project as well as the developers working on it
- You can find a CHANGELOG.md file in the GitHub repository if you're working on an open-source project
- This file is intended to provide contributors with the most recent project updates

*the*knowledgeacademy

# Understanding Change Logs

## Command for Generating a changelog

```
administrator@EC2AMAZ-134LMS MINGW64 ~/Desktop/Demo/Demo (main)
$ git log
commit 5fbed2fda209916978b11db7bcfc6f32c143b93 (HEAD -> main)
Author: Delegateone <studentone448@gmail.com>
Date:   Sat Sep 4 08:53:34 2021 +0000

    Initial commit

commit 98a6c02197dd2cabecc73522a9faa4e73912ed75 (upstream/main, upstream/branch, upstream/My-branch-2, origin/main,
Author: Delegateone <89916523+Delegateone@users.noreply.github.com>
Date:   Thu Sep 2 06:45:59 2021 +0100

    Add files via upload

commit 08b5371214f870d4c3150c614cc1495515df90a0
Author: Delegateone <89916523+Delegateone@users.noreply.github.com>
Date:   Thu Sep 2 05:38:31 2021 +0100

    Initial commit
```

*the*knowledgeacademy

# Module 8: Managing GitHub

theknowledgeacademy

## Adding Files

- The following are the steps for Adding a file to a repository:

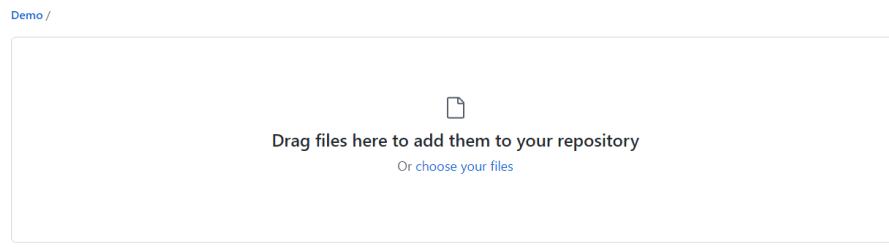
**Step 1:** Click on the **Add file** drop-down, then choose **Upload files**

A screenshot of a GitHub repository interface. At the top, there are buttons for 'main', '4 branches', '0 tags', 'Go to file', 'Add file', and 'Code'. A dropdown menu is open over the 'Add file' button, showing options 'Create new file' and 'Upload files'. Below this, a list of files is shown: 'README.md' (Initial commit, yesterday) and 'sample.txt' (Add files via upload, yesterday). The 'Upload files' option is highlighted with a red box.

theknowledgeacademy

# Adding Files

**Step 2:** Drag and drop the folder or file you want to upload to your repository onto the file tree



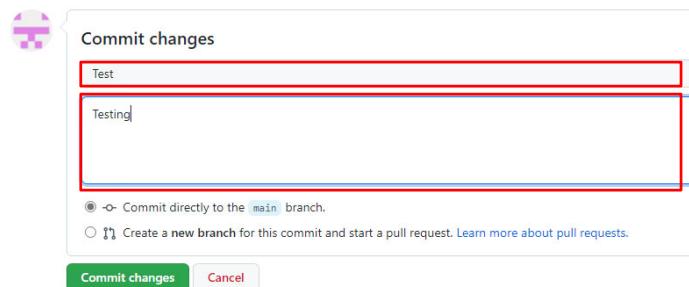
the knowledge academy

© 2021 The Knowledge Academy Ltd

65

# Adding Files

**Step 3:** At the bottom of the page, Enter a short, commit message that explains the change you have made to the file. You can attribute a commit to one or more authors in a commit message



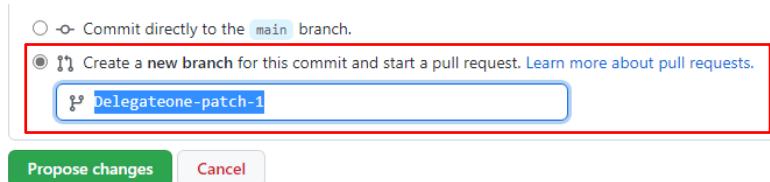
the knowledge academy

© 2021 The Knowledge Academy Ltd

66

# Adding Files

**Step 4:** Choose to add a commitment to the current branch or to a new branch below the commit message fields. If the current branch is the default one, you should choose to create a new branch and create a pull request for your commit



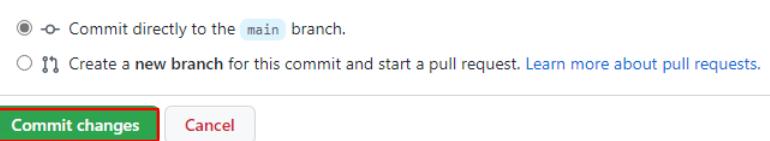
the knowledge academy

© 2021 The Knowledge Academy Ltd

67

# Adding Files

**Step 5:** Click on the **Commit changes**



the knowledge academy

© 2021 The Knowledge Academy Ltd

68

# Editing Files from Other Users

- The following are the steps for Editing files in another user's repository:

**Step 1:** In another user's repository, browse to the folder that includes the file you need to edit. Click on the name of the file you need to edit

**Step 2:** Click on point above the file, GitHub forks the repository for you

A screenshot of a GitHub file editor. The URL is 'Demo / sample.txt'. The file content is a simple HTML page with a title, a link to a stylesheet, and an octocat image. A red box highlights the 'Edit' button at the top right of the code area. The GitHub interface shows 23 lines (18 code) and 411 Bytes. It also shows 1 commit by 'tech11111' and 2 contributors.

the knowledge academy

© 2021 The Knowledge Academy Ltd

69

# Editing Files from Other Users

**Step 3:** Make any changes you require to the file

A screenshot of a GitHub file editor showing changes made to the 'sample.txt' file. The changes include adding a new line 14 with the text '<!-- Feel free to change this is text here --&gt;' and line 15 with the text 'Fork me! Fork you, @octocat!'. The rest of the file remains the same as in the previous screenshot.</p>

the knowledge academy

© 2021 The Knowledge Academy Ltd

70

# Editing Files from Other Users

**Step 4:** Click on the **Preview changes**

Demo / sample.txt in main

Preview changes

```
@@ -14,6 +14,7 @@
14 14 <!-- Feel free to change this is text here -->
15 15 <p>
16 16 Fork me? Fork you, @octocat!
17 + Edit
18 </p>
19 <p>
20 Sean made a change
...
```

the knowledge academy

© 2021 The Knowledge Academy Ltd

71

# Editing Files from Other Users

**Step 5:** Enter a short, commit message that explains the change you made to the file. You can attribute the commit to one or more authors in the commit message

Commit changes

Update sample.txt

Add an optional extended description...

→ Commit directly to the `main` branch.

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

Commit changes Cancel

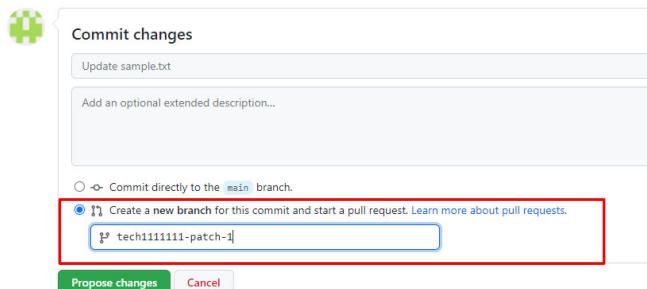
the knowledge academy

© 2021 The Knowledge Academy Ltd

72

## Editing Files from Other Users

**Step 6:** Choose to add a commitment to the current branch or to a new branch below the commit message fields. If the current branch is the default one, you should choose to create a new branch and create a pull request for your commit



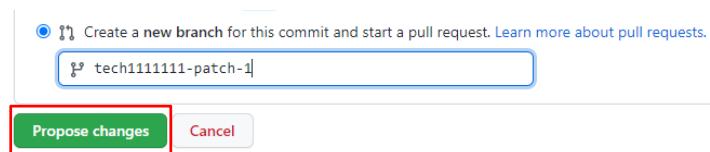
the knowledge academy

© 2021 The Knowledge Academy Ltd

73

## Editing Files from Other Users

**Step 7:** Click on the **Propose changes**



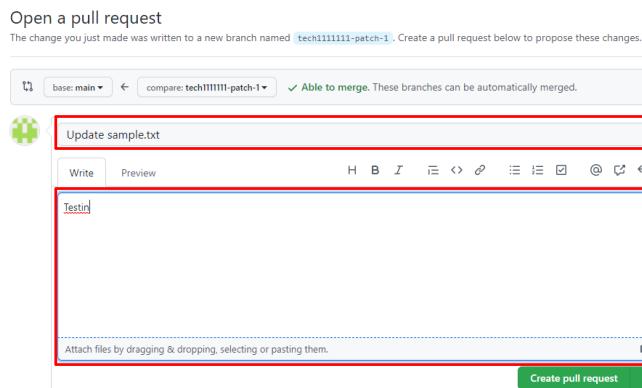
the knowledge academy

© 2021 The Knowledge Academy Ltd

74

# Editing Files from Other Users

**Step 8:** Enter a description and title for your pull request



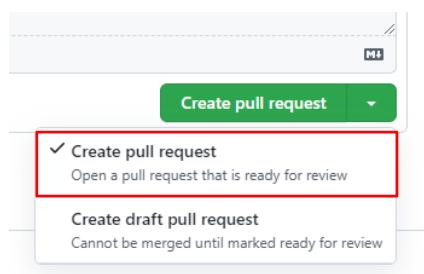
the knowledge academy

© 2021 The Knowledge Academy Ltd

75

# Editing Files from Other Users

**Step 9:** Click on the **Create pull request**



the knowledge academy

© 2021 The Knowledge Academy Ltd

76

# Tracking Changes

- The following are the steps for Tracking changes in a file:

## Step 1: Click on the Blame



23 lines (18 sloc) | 411 Bytes

```
1 <!DOCTYPE html>
2
3 <html>
4 <head>
5   <meta http-equiv="Content-Type" content="text/html; charset=utf-8"/>
6   <title>Spoon-Knife</title>
7   <LINK href="styles.css" rel="stylesheet" type="text/css">
8 </head>
9
10 <body>
11
12 
13
```

Raw **Blame** ⌂ ⌂ ⌂

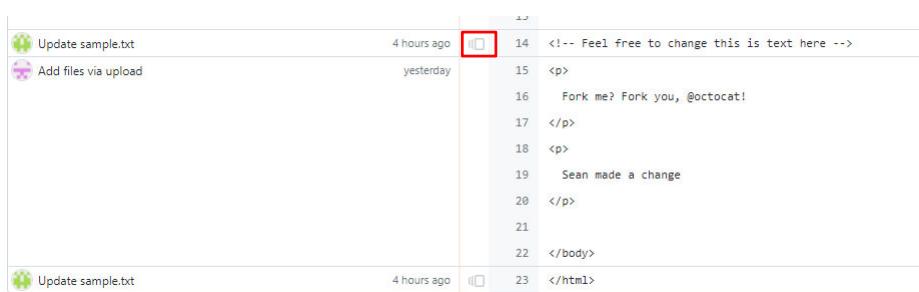
the knowledge academy

© 2021 The Knowledge Academy Ltd

77

# Tracking Changes

**Step 2:** To view earlier revisions of a specific line or reblame, click until you have found the changes you are interested in viewing



Update sample.txt 4 hours ago

Add files via upload yesterday

Update sample.txt 4 hours ago

14 <!-- Feel free to change this is text here -->

15 <p>

16 Fork me? Fork you, @octocat!

17 </p>

18 <p>

19 Sean made a change

20 </p>

21

22 </body>

23 </html>

the knowledge academy

© 2021 The Knowledge Academy Ltd

78

# Deleting Files

- The following are the steps for Deleting a file:

**Step 1:** Click on  the top of the file



23 lines (18 sloc) | 411 Bytes

```
1 <!DOCTYPE html>
2
3 <html>
4 <head>
5   <meta http-equiv="Content-Type" content="text/html; charset=utf-8"/>
6   <title>Spoon-Knife</title>
7   <LINK href="styles.css" rel="stylesheet" type="text/css">
8 </head>
9
10 <body>
11   <h1>Spoon-Knife</h1>
12   <p>A simple web application for managing spoons and knives.</p>
13   <ul>
14     <li>Spoons:</li>
15     <li>Knives:</li>
16   </ul>
17
18   <form action="/add" method="post">
19     <label>Name:</label>
20     <input type="text" name="name">
21     <label>Type:</label>
22     <input type="text" name="type">
23     <input type="submit" value="Add">
24   </form>
25
26 </body>
```

Raw Blame   

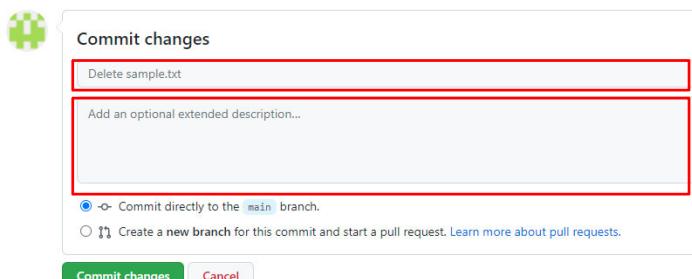
*the*knowledgeacademy

© 2021 The Knowledge Academy Ltd

79

# Deleting Files

**Step 2:** Enter a short commit message at the bottom of the page that explains the change you made to the file. You can attribute the commit to one or more authors in the commit message



*the*knowledgeacademy

© 2021 The Knowledge Academy Ltd

80

# Deleting Files

**Step 3:** Click on the **Commit changes**

- > Commit directly to the `main` branch.  
 ⚡ Create a new branch for this commit and start a pull request. Learn more about pull requests.

**Commit changes**

Cancel

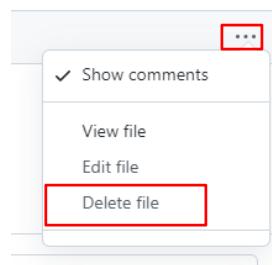
*the*knowledgeacademy

© 2021 The Knowledge Academy Ltd

81

# Deleting Files

**Step 4:** Click on the More options..., and select **Delete file**



*the*knowledgeacademy

© 2021 The Knowledge Academy Ltd

82

# Module 9: Securing GitHub

*theknowledgeacademy*

## User Roles and Access Levels

- Following are some security features built into the appliance, along with information about GitHub's development practices for application security

### User Roles and Access Levels

- GitHub Enterprise renders a Linux user administration account and two types of application users

**01**  
Standard User

**02**  
VM  
Administrator

**03**  
Site  
Administrators

*theknowledgeacademy*

# Organisations and Teams

## Organisations and Teams

- In the GitHub Enterprise, an organisation is the core concept
- Organisations enable you to generate a number of logical containers for the projects and the business units
- Every organisation account functions as the owner of one or more repositories and the organisation owners can add users to the teams they create



*the*knowledgeacademy

# Organisations and Teams

(Continued)

- You can manage access to the repository's working teams inside the organisations
- Teams consist of repositories and members
- Select any of these three permission levels while adding a repository to a team, to grant team members:
  - **Write:** close issues, merge pull requests and edit code
  - **Read:** open issues, view and fork code and pull requests
  - **Admin:** change repository settings

*the*knowledgeacademy

# Authentication

- GitHub Enterprise provides four types of authentication methods:
  - SAML, External LDAP or CAS authentication by utilising SAML Identity Provider, Active Directory, or other compatible service
  - SSH for both Git protocol access and OS level systems administration
  - Username/password and HTTP cookies for the session management and web application authentication
  - Personal Access Tokens and OAuth for API and external service authentication

# Encryption

- GitHub enterprise run behind the corporate firewall
- It helps to secure the communications over the wire, if we run the GitHub enterprise over SSL



# Audit and Access Logging

- Keeping a correct record of all the system activity and the users is a central requirement for several clients
- GitHub Enterprises comprise detailed audit records which are accessible to the site administrators



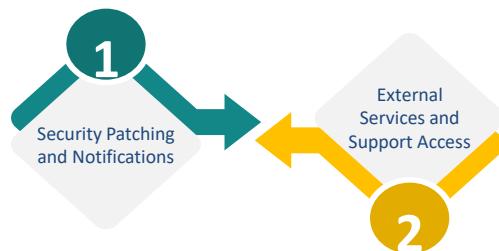
the knowledge academy

© 2021 The Knowledge Academy Ltd

89

# Application Security

- The Application security team of GitHub concentrate full-time on penetration testing, vulnerability assessment, and code review for GitHub products
- GitHub also engages with external security firms to render point-in-time security assessments of GitHub products



the knowledge academy

© 2021 The Knowledge Academy Ltd

90

# Application Security

(Continued)

## Security Patching and Notifications

- Patching the running services and core operating system to address security concerns, GitHub managed a part of its standard product release cycle
- This involves patches for stability, functionality, and non-critical security issues for GitHub applications
- Critical security patches are given as required outside of the regular release cycle to improve time to resolution and also limit modifications to the system

# Application Security

## External Services and Support Access

- By design, GitHub Enterprise can operate from your network to outside services without any egress access
- The system administrator can optionally enable the integration of external services, including Syslog, SMTP, and Gravatar
- The system does not attempt to communicate with GitHub's servers; your system administrator can collect useful data to troubleshoot any issues and deliver that data manually to the GitHub Enterprise Support Team

# Module 10: Automating GitHub

theknowledgeacademy

## Automatic Workflows

- The following are the steps For repository project board

**Step 1:** Go to the repository's main page on GitHub

**Step 2:** Click on the **Project** Tab

A screenshot of a GitHub repository page. At the top, there are tabs for Code, Issues, Pull requests, Actions, Projects (which is highlighted with a red box), Wiki, Security, Insights, and Settings. Below the tabs, there is a search bar with the text 'isopen' and a 'New project' button. A table lists one project: 'Demo-project' with status 'Project testing' and a note 'Updated 4 minutes ago'. There are also 'Sort' and '...' buttons.

theknowledgeacademy

# Automatic Workflows

**Step 3:** Click on the project board you need to automate in the list of projects

The screenshot shows a GitHub interface with the URL 'Delegateone / Demo'. The top navigation bar includes 'Code', 'Issues', 'Pull requests', 'Actions', 'Projects 1', 'Wiki', 'Security', 'Insights', and 'Settings'. A search bar contains the query 'is:open'. A green 'New project' button is visible. Below the search bar, a card for 'Demo-project' is shown, indicating 1 Open and 0 Closed issues. The card has a timestamp: 'Updated 4 minutes ago'. A red box highlights this entire card.

the knowledge academy

© 2021 The Knowledge Academy Ltd

95

# Automatic Workflows

**Step 4:** Click on the **more option...**, then select **Manage automation**

The screenshot shows a project board for 'Demo-project'. A context menu is open over a column labeled 'Project Column'. The menu options are: 'Edit column' (highlighted with a red box), 'Manage automation' (also highlighted with a red box), 'Archive all cards', 'Copy column link', and 'Delete column'. A 'Filter cards' search bar is visible at the top right of the board area.

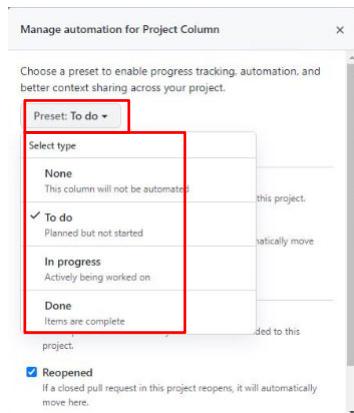
the knowledge academy

© 2021 The Knowledge Academy Ltd

96

# Automatic Workflows

**Step 5:** Choose an automation preset utilising the **Preset** drop-down menu



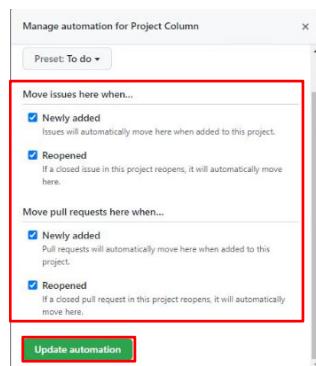
the knowledge academy

© 2021 The Knowledge Academy Ltd

97

# Automatic Workflows

**Step 6:** Choose the workflow automations you would like to configure for the column and then click on the **Update automation**



the knowledge academy

© 2021 The Knowledge Academy Ltd

98

# Congratulations



Congratulations on completing this course!

Keep in touch

[info@theknowledgeacademy.com](mailto:info@theknowledgeacademy.com)

Thank you

---

# *the*knowledgeacademy

*empowering the advancement of knowledge*

---



/The.Knowledge.Academy.Ltd



/TKA\_Training



/the-knowledge-academy



/+TheKnowledgeAcademyWinkfield



/TheKnowledgeAcademy

The world's largest global training provider

---

| info@theknowledgeacademy.com

| theknowledgeacademy.com