

# Git:

## A distributed version control system

# Why a distributed version control system?

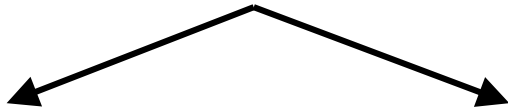
- Distributed – so that developers can **share** their work worldwide.
- Version control – to work on various aspects of a project **simultaneously** and combine the result; to revisit **previous versions** and track **changes**; to define **milestones** of the development cycle that developers and users can refer to.
- System – to establish a software project that is available on a wide range of platforms, has a command line and a Web interface and is **maintained by the community**.

# Why Git?

- Git automatically **tracks changes** and complains if a local copy gets out of sync with the master code.
- Community development platforms are available where **projects can be hosted and shared** (free of charge) such as Github.com and Gitlab.com.
- **Development platforms** are available under an Open Source license so they can be installed and deployed locally.

# Public vs. private

## Public platforms (e.g. github.com)



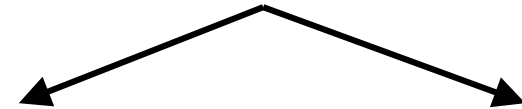
### Public repositories

- Can be seen by the world.
- More detailed rules for who can contribute (create issues, create pull requests, merge) can be set up.

### Private repositories

- Can be seen by the owner and the platform provider.

## Private platform (e.g. local gitlab setup)



### Public repositories

- Can be seen by anyone who has access to the platform.
- More detailed rules for who can contribute (create issues, create pull requests, merge) can be set up.

### Private repositories

- Can be seen by the owner and the platform provider (company IT).

# How does Git work? CLI | WebUI

## Initialize Git repository

<https://github.com/new>

### Create a new repository

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

*Required fields are marked with an asterisk (\*).*

Owner \*

Repository name \*

✔ gitdemo is available.

Great repository names are short and memorable. Need inspiration? How about

**cautious-fiesta** ?

Description (optional)

```
$ mkdir gitdemo
```

```
$ cd gitdemo
```

```
$ git init
```

```
Initialized empty Git repository in  
/home/user/gitdemo/.git/
```

Open Source compliance: Technical must-knows for legal experts

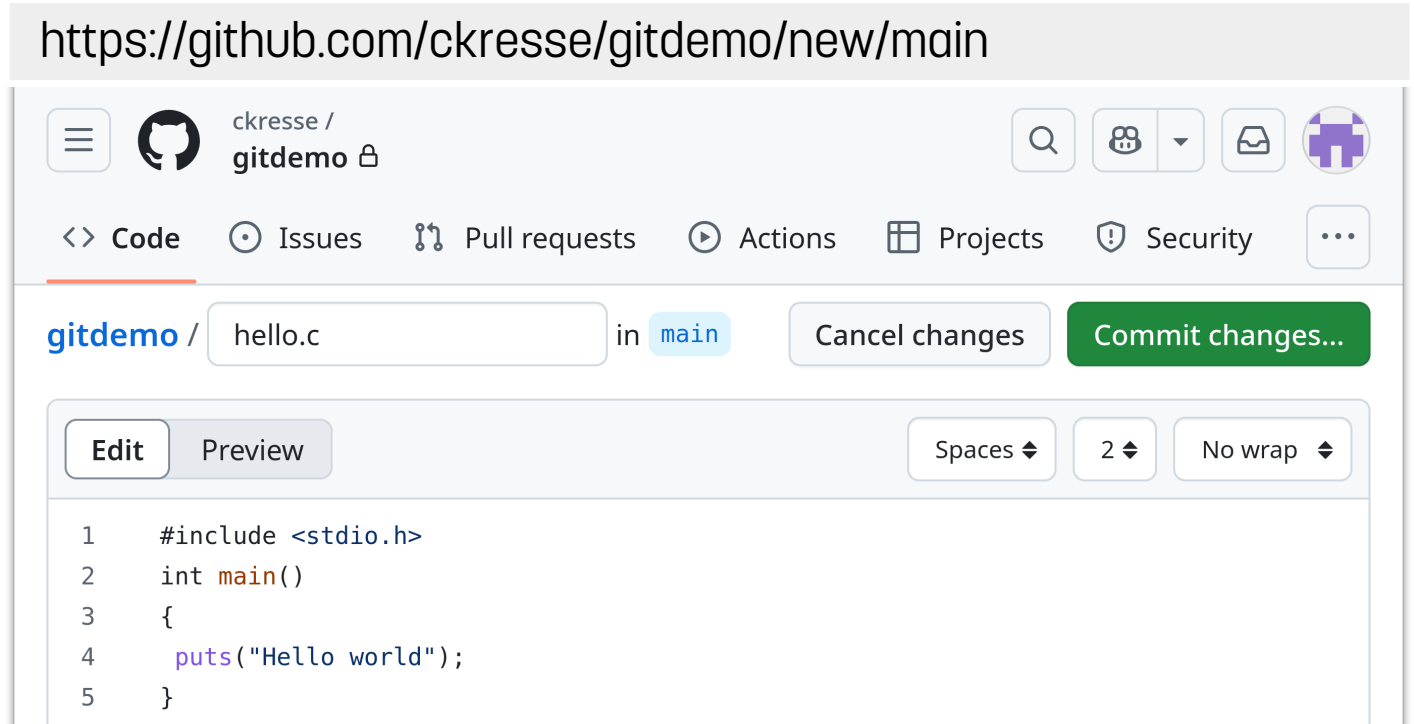
Git: A distributed version control system

COOL January 29, 2025

# How does Git work? CLI | WebUI

## Create the source code file "hello.c"

```
$ cat <<EOF >hello.c
> #include <stdio.h>
> int main()
> {
> puts("Hello world");
> }
> EOF
```



The screenshot shows the GitHub web interface for creating a new file. The URL is `https://github.com/ckresse/gitdemo/new/main`. The repository name is `ckresse / gitdemo`. The file name is `hello.c` in the `main` branch. The interface includes a navigation bar with options like `Code`, `Issues`, `Pull requests`, `Actions`, `Projects`, and `Security`. Below the navigation bar, there are buttons for `Cancel changes` and `Commit changes...`. The main content area shows the file content in a code editor with tabs for `Edit` and `Preview`. The code is as follows:

```
1 #include <stdio.h>
2 int main()
3 {
4     puts("Hello world");
5 }
```

# How does Git work? Only CLI

## Get status of the repository

```
$ git status
```

```
On branch main
```

```
No commits yet
```

```
Untracked files:
```

```
(use "git add <file>..." to include in what will be committed)
```

```
hello.c
```

```
nothing added to commit but untracked files present (use "git add" to track)
```

# How does Git work? Only CLI

## Get status of the repository

```
$ git status
```


```
On branch main
```

```
No commits yet
```

```
Untracked files:
```

```
(use "git add <file>..." to include in what will be committed)
```

```
hello.c
```



Git detected that a yet  
**unknown file** was added

```
nothing added to commit but untracked files present (use "git add" to track)
```



# How does Git work? CLI

## Make the new material known to the repository

```
$ git add hello.c
$ git commit -m "Add a new file to greet the world."
[main (root-commit) 14dadcd] Add a new file to greet the world.
1 file changed, 5 insertions(+)
create mode 100644 hello.c
```

A meaningful  
commit message  
can be added

# How does Git work? CLI

Make the new material known to the repository

```
$ git add hello.c
$ git commit -m "Add a new file to greet the world.
Signed-off-by: Caren Kresse, OSADL, ckresse@osadl.org"
[main (root-commit) 14dadcd] Add a new file to greet the world. Signed-off-by...
1 file changed, 5 insertions(+)
create mode 100644 hello.c
```

Many FOSS projects require a "signed-off-by" statement that indicates **acceptance of the contribution rules**, e.g. confirming that the commiter has **the right** to commit the code and **to license** it under the project's license.

# How does Git work? CLI

Make the new material known to the repository  
and commit it to the upstream repository

```
$ git add hello.c
$ git commit -m "Add a new file to greet the world.
Signed-off-by: Caren Kresse, OSADL, ckresse@osadl.org"
[main (root-commit) 14dadcd] Add a new file to greet the world. Signed-off-by...
 1 file changed, 5 insertions(+)
 create mode 100644 hello.c
```

```
$ git branch -M main
$ git remote add origin https://github.com/ckresse/gitdemo.git
$ git push -u origin main
```

- Creates a branch named "main"
- Adds an upstream origin to the local repository
- Pushes the local changes to the upstream repository

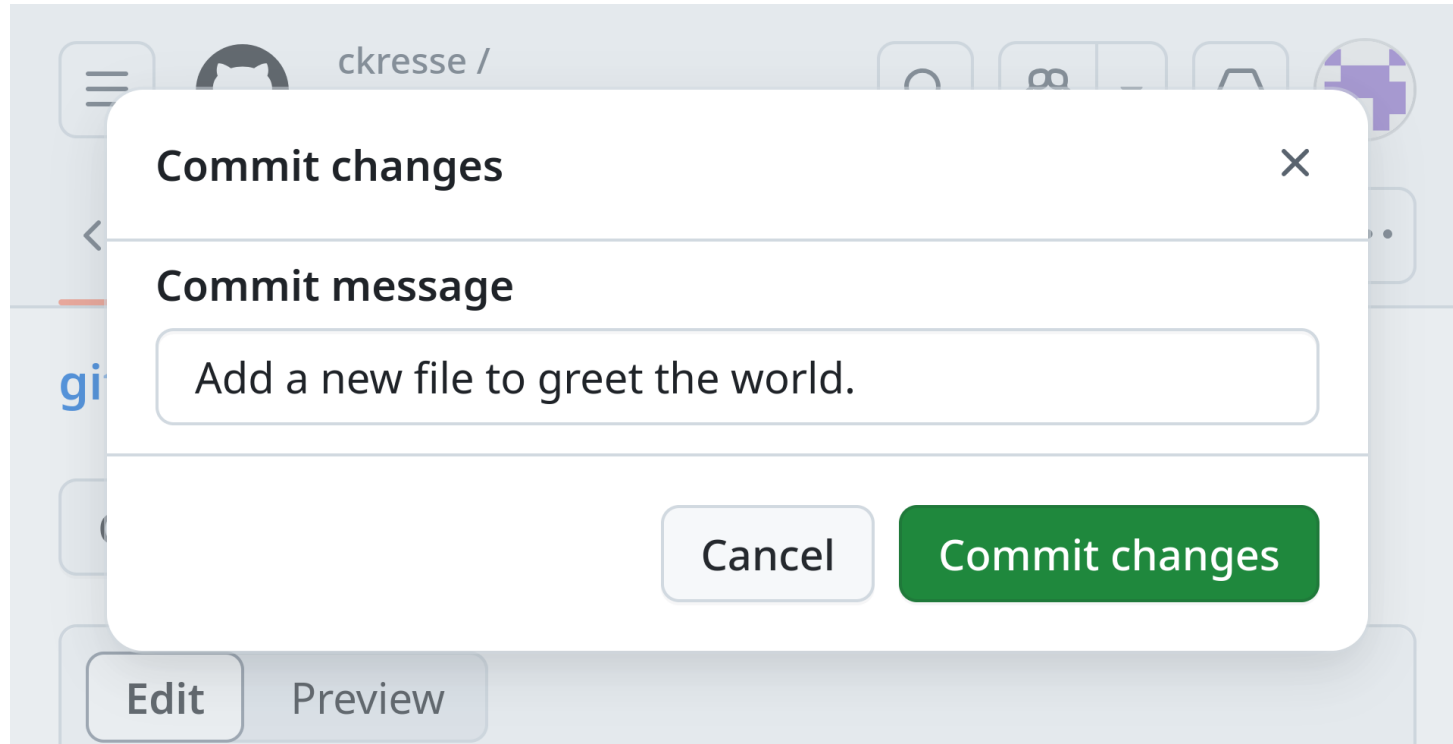
Open Source compliance: Technical must-knows for legal experts

Git: A distributed version control system

COOL January 29, 2025

# How does Git work? WebUI

## Commit the new material



# How does Git work? CLI | WebUI

## Show the logged history

```
$ git log
```

```
commit 14dadcdb51ba8973fcbe2986aa537d777 (HEAD->master, origin/main, origin/HEAD)
```

```
Author: Caren Kresse <ckresse@osadl.org>
```

```
Date: Thu Nov 28 16:33:58 2024 +0100
```

```
    Add a new file to greet the world.
```

```
    Signed-off-by: Caren Kresse, OSADL, ckresse@osadl.org
```

### Commits

main

All users

All time

Commits on Jan 14, 2025

Add a new file to greet the world.

Verified

d38c0d5



ckresse authored 2 minutes ago

# How does Git work? CLI | WebUI

## Show changes in a file

Edit the file (replace "world" with "OSADL") and show the difference between the old and the new file

```
$ git diff
@ hello.c:2 @
#include <stdio.h>
int main()
{
    puts("Hello world");
    puts("Hello OSADL");
}
```

```
< Files gitdemo / hello.c in main Cancel changes Commit changes...
Edit Preview
... @@ -1,5 +1,5 @@
1 #include <stdio.h>
2 int main()
3 {
4 puts("Hello world");
5 }
...
1 #include <stdio.h>
2 int main()
3 {
4 puts("Hello OSADL");
5 }
```

# How does Git work? Only CLI

## Get new status of the repository

```
$ git status
```

```
On branch master
```

```
Changes not staged for commit:
```

```
(use "git add <file>..." to update what will be committed)
```

```
(use "git restore <file>..." to discard changes in working directory)
```

```
    modified:   hello.c
```

```
no changes added to commit (use "git add" and/or "git commit -a")
```

# How does Git work? Only CLI

## Get new status of the repository

```
$ git status
```

```
On branch master
```

```
Changes not staged for commit:
```

```
(use "git add <file>..." to update what will be committed)
```

```
(use "git restore <file>..." to discard changes in working directory)
```

```
modified: hello.c
```

Git detected that a file  
was **modified** and  
not yet committed

```
no changes added to commit (use "git add" and/or "git commit -a")
```



# How does Git work? CLI | WebUI

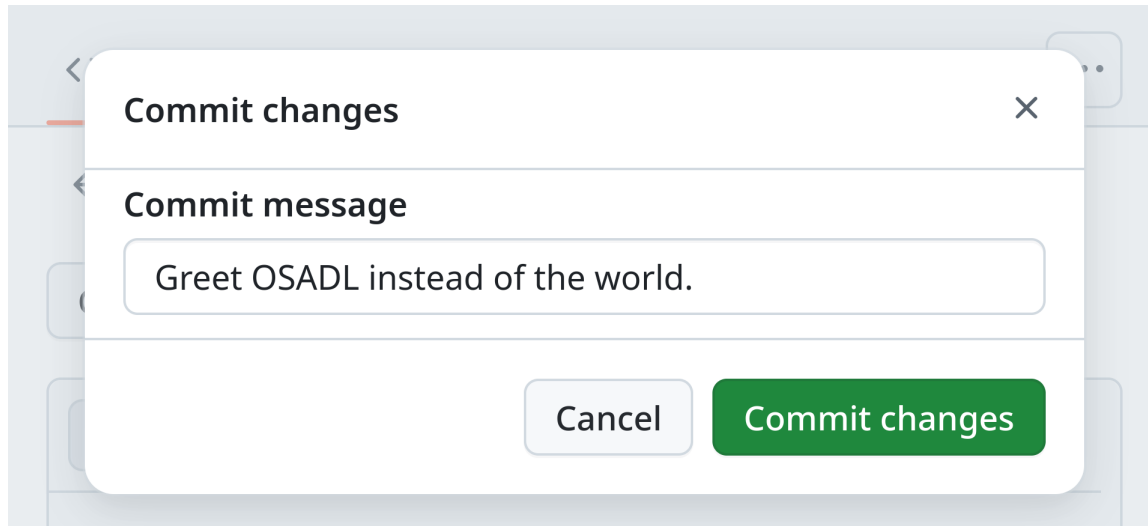
## Commit change to the repository

```
$ git add .
```

```
$ git commit -m "Greet OSADL instead of the world."
```

```
[master 6da4f27] Greet OSADL instead of the world.
```

```
1 file changed, 1 insertion(+), 1 deletion(-)
```



# How does Git work? CLI

## Show the updated logged history

```
$ git log
```

```
commit 6da4f2771dac8f3900199a762fb8c578cb2f6638 (HEAD -> master)
```

```
Author: Carsten Emde <cbe@osadl.org>
```

```
Date: Thu Nov 28 17:42:01 2024 +0100
```

**Greet OSADL instead of the world.**

```
commit 14dadcdb51ba8973fcbe2986aa537d7770edff49 (origin/main, origin/HEAD)
```

```
Author: Carsten Emde <cbe@osadl.org>
```

```
Date: Thu Nov 28 16:33:58 2024 +0100
```

**Add a new file to greet the world.**

The change is not yet pushed to the upstream repository.

Open Source compliance: Technical must-knows for legal experts

Git: A distributed version control system

COOL January 29, 2025

# How does Git work? WebUI

## Show the updated logged history

History for [gitedemo](#) / `hello.c` on `main` All users All time

-o- Commits on Jan 15, 2025

**Greet OSADL instead of the world.** Verified 9362da1

ckresse authored 1 hour ago

-o- Commits on Jan 14, 2025

**Add a new file to greet the world.** Verified d38c0d5

ckresse authored 18 hours ago

-o- End of commit history for this file

# Collaborating on Github

- Issues
- Branches
- Pull requests
- Merging

# Github issues



<> Code **Issues** Pull requests Actions Projects Security Insights Settings



## Create new issue

### Add a title \*

### Add a description

Write Preview H B I | ☰ <> 🔗 | ☰ ☰ ☰ ☰ ⏪ ☑

I noticed that greeting only OSADL is not polite.

Negative results can be reproduced by entering a room full of people and greeting only one person.

I suggest greeting all COOL participants. Can anyone help with this?

Assignees No one - [Assign yourself](#)

---

Labels No labels

---

Type No type

---

Projects No projects

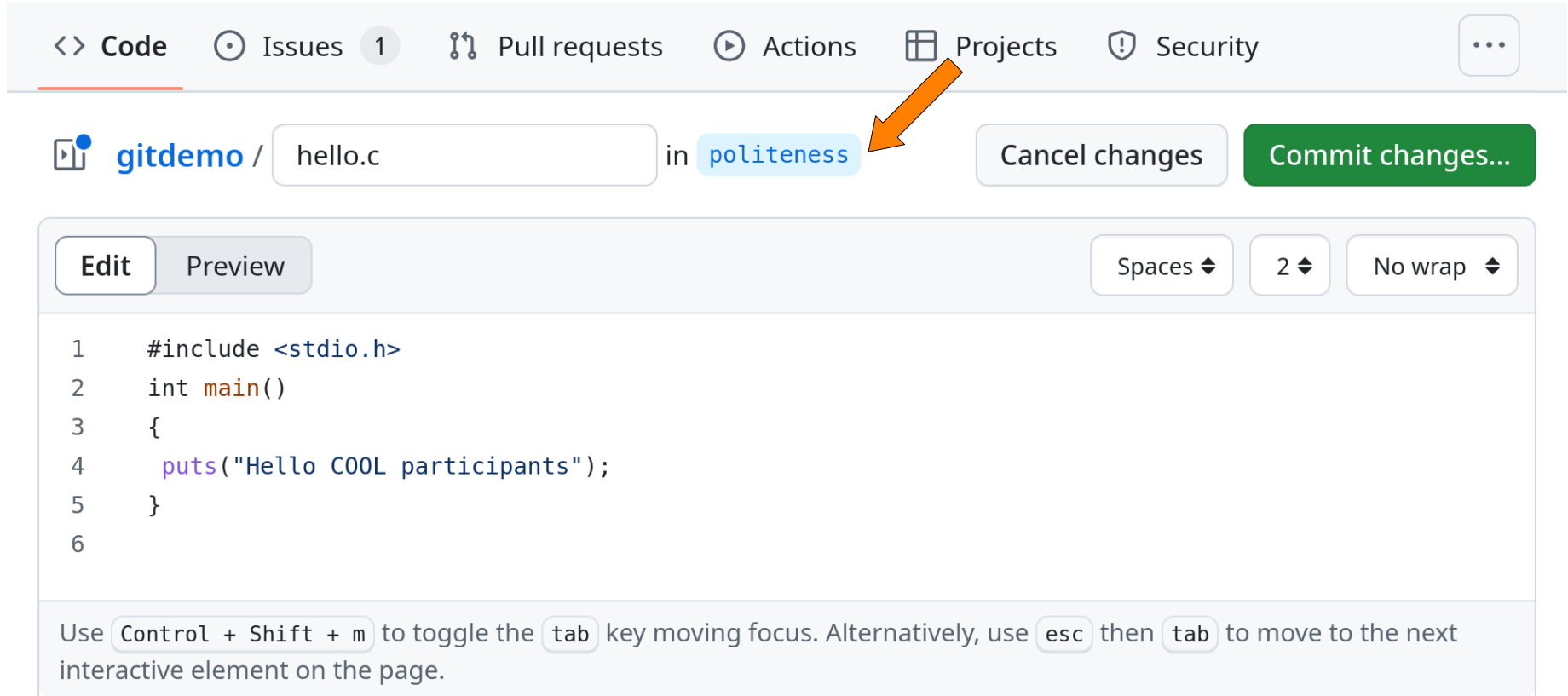
---

Milestone No milestone

# Creating a branch

The screenshot shows the GitHub interface for a repository named 'gitdemo'. At the top, there are navigation tabs for 'Code', 'Issues', 'Pull requests', 'Actions', 'Projects', and 'Security'. Below this, the repository name 'gitdemo' is displayed as 'Private', along with 'Unwatch' (1), 'Fork' (0), and 'Star' (0) buttons. The main content area shows the 'main' branch selected in a dropdown menu. An orange arrow points to this dropdown. A modal window titled 'Switch branches/tags' is open, showing a search for 'politeness'. The modal has tabs for 'Branches' and 'Tags'. A button labeled 'Create branch politeness from main' is highlighted. To the right of the modal, there is a list of recent commits, including one with hash '362da1' from '2 days ago'. Further right, there is an 'About' section with a gear icon, and a 'Releases' section showing 'No releases published'.

# Modifying a file in the new branch

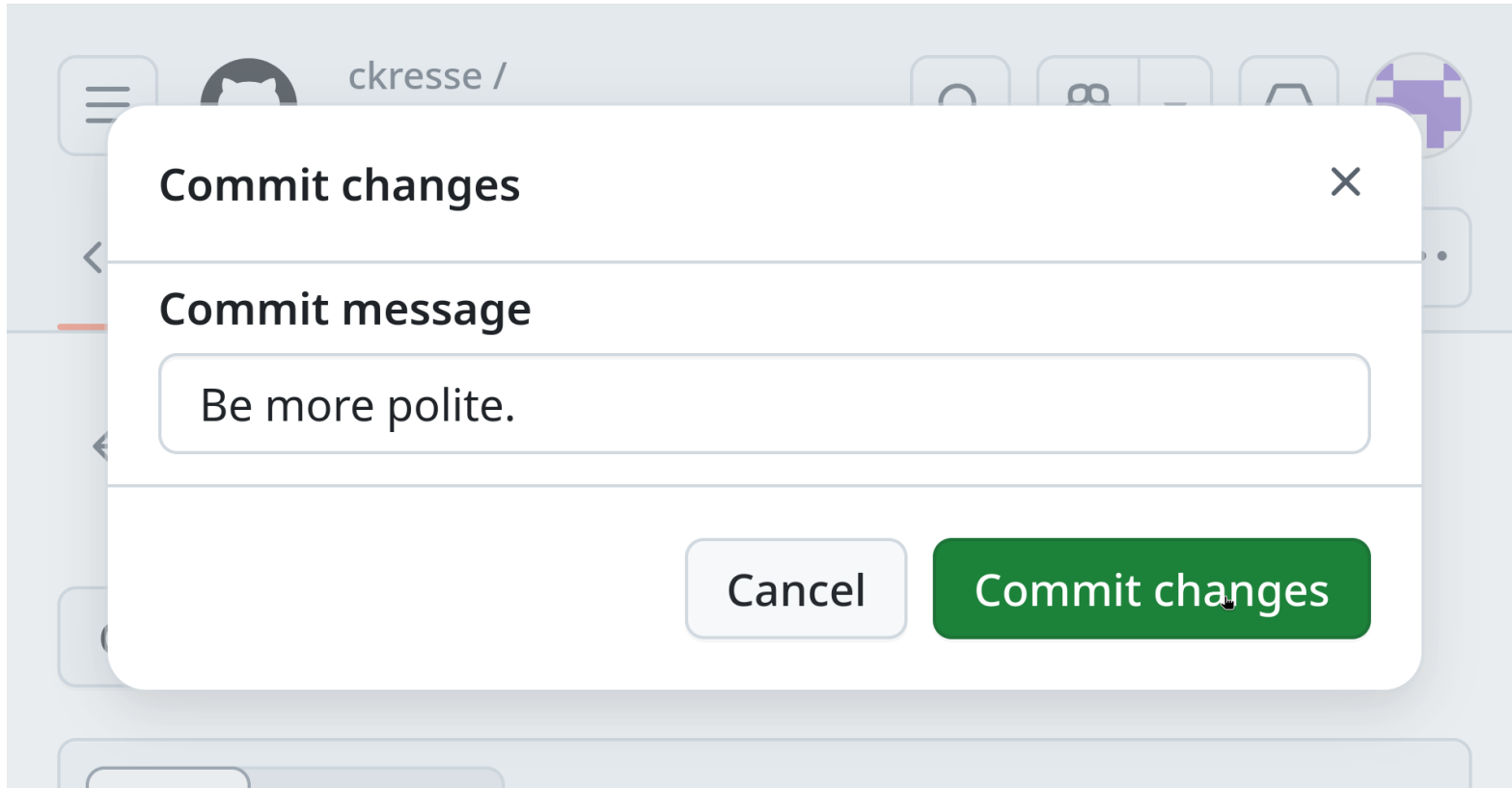


The screenshot shows the GitHub web interface. At the top, there is a navigation bar with links for Code, Issues (1), Pull requests, Actions, Projects, and Security. Below this, the repository path is shown as `gitdemo / hello.c` in the `politeness` branch. An orange arrow points to the `politeness` branch name. To the right of the path are buttons for "Cancel changes" and "Commit changes...". Below the path, there are tabs for "Edit" and "Preview", and settings for "Spaces", "2" (line width), and "No wrap". The main area contains the code for `hello.c`:

```
1  #include <stdio.h>
2  int main()
3  {
4      puts("Hello COOL participants");
5  }
6
```

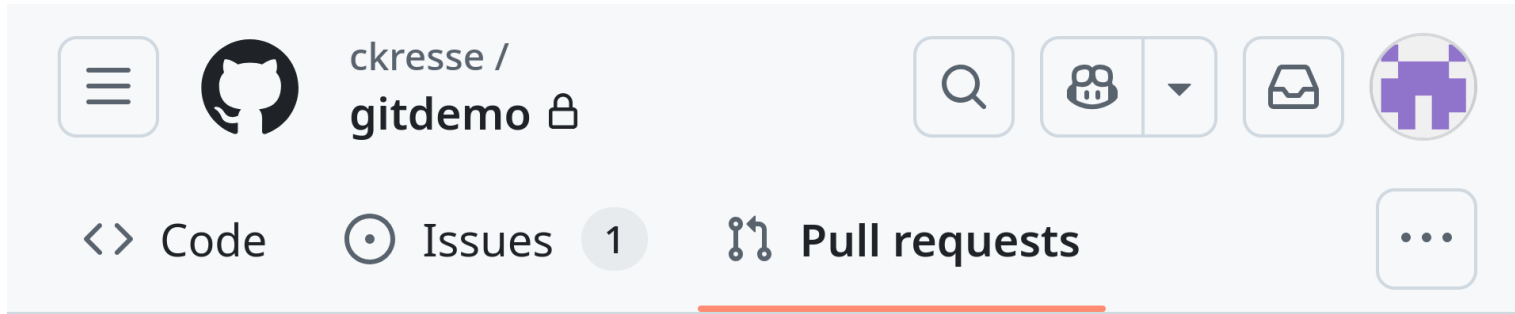
At the bottom, there is a note: "Use `Control + Shift + m` to toggle the `tab` key moving focus. Alternatively, use `esc` then `tab` to move to the next interactive element on the page."


# Committing changes





# Git detects commits in a branch and differences to the main branch




 politeness had recent pushes 1 minute ago



[Compare & pull request](#)

# Opening a pull request (PR)

## 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](#). [Learn more about diff comparisons here](#).


 base: main ← compare: politeness ✓ **Able to merge.** These branches can be automatically merged.

 **Add a title** Reviewers   
 No reviews 

Be more polite.

1 commit 1 file changed 1 contributor

Commits on Jan 17, 2025

Be more polite. Verified  8c11f76 <>

 ckresse authored 4 minutes ago

Showing 1 changed file with 1 addition and 1 deletion. Split Unified

```
2 hello.c
... @@ -1,5 +1,5 @@
1 1 #include <stdio.h>
2 2 int main()
3 3 {
4 - puts("Hello OSADL");
4 + puts("Hello COOL participants");
5 5 }
```

# Adding PR to the open issue

<> Code Issues 1 Pull requests 1 Actions Projects Security Insights

## Politeness problems #1

Open



ckresse opened 17 minutes ago

I noticed that greeting only OSADL is not polite.

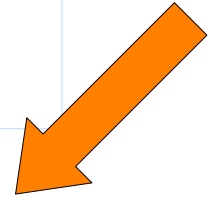
Negative results can be reproduced by entering a room full of people and greeting only one person.

I suggest greeting all COOL participants. Can anyone help with this?

Create sub-issue



ckresse linked a pull request that will close this issue [Be more polite. #2](#) 1 minute ago



# Merging the PR with the suggested change into the main branch

Open

Be more polite. #2

ckresse wants to merge 1 commit into `main` from `politeness`



Continuous integration has not been set up

[GitHub Actions](#) and [several other apps](#) can be used to automatically catch bugs and enforce style.

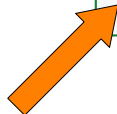


This branch has no conflicts with the base branch

Merging can be performed automatically.

Merge pull request

or view [command line instructions](#).



Projects

None yet

Milestone

No milestone

Development

Successfully merging this pull request may close these issues.

Politeness problems

Corresponding issue

# Merge messages

The screenshot shows a GitHub pull request interface. At the top, a purple 'Merged' badge is visible. The main content area shows a commit merge by 'ckresse' into the 'main' branch from the 'politeness' branch, with a 'Revert' button. Below this is a comment from 'ckresse' stating 'Your fix has been merged. Thank you for the contribution.' At the bottom, a notification box says 'Pull request successfully merged and closed' and 'You're all set—the politeness branch can be safely deleted.' On the right side, there are sections for 'None yet', 'Milestone', 'Development', and 'Notifications'. A callout bubble points to the 'Politeness problems' section, indicating that merging the pull request may close these issues.

**Merged** Be more polite. #2  
ckresse merged 1 commit into `main` from `politeness` 1 minute ago

ckresse merged commit `8201f07` into `main` 1 minute ago Revert

ckresse commented now Author ...

Your fix has been merged. Thank you for the contribution.

**Pull request successfully merged and closed** Delete branch

You're all set—the `politeness` branch can be safely deleted.

None yet

Milestone  
No milestone

Development  
Successfully merging this pull request may close these issues.

**Politeness problems**

Notifications Customize

Unsubscribe

You're receiving notifications because you're watching this repository.

Closes the corresponding issue

# Repository history (all commits)

The screenshot shows the commit history for the repository 'ckresse / gitdemo'. The interface includes a navigation bar with options like Code, Issues, Pull requests, Actions, Projects, Security, and Insights. The main content area is titled 'Commits' and shows a list of commits on the 'main' branch, filtered by 'All users' and 'All time'. The commits are grouped by date:

- Commits on Jan 17, 2025**
  - Merge pull request #2 from ckresse/politeness** (Verified, 8201f07) - ckresse authored 2 minutes ago
  - Be more polite.** (Verified, 8c11f76) - ckresse authored 15 minutes ago
- Commits on Jan 15, 2025**
  - Greet OSADL instead of the world.** (Verified, 9362da1) - ckresse authored 2 days ago
- Commits on Jan 14, 2025**
  - Add a new file to greet the world.** (Verified, d38c0d5) - ckresse authored 3 days ago

Open Source compliance: Technical must-knows for legal experts  
Git: A distributed version control system  
COOL January 29, 2025

# Git blame

📄 main [gitdemo / hello.c](#) 🔖  t ...

 **ckresse** Be more polite. 8c11f76 · 17 minutes ago 🕒

5 lines (5 loc) · 68 Bytes

Code **Blame** Raw 📄 ⬇️ ✎ ⌵

Older  Newer  👤 Contributors 1

3 days ago	 Add a new file to greet the wo...	1	<code>#include &lt;stdio.h&gt;</code>
		2	<code>int main()</code>
		3	<code>{</code>
17 minutes ago	 Be more polite. 📄	4	<code>puts("Hello COOL participants");</code> 
3 days ago	 Add a new file to greet the wo...	5	<code>}</code>

# Git blame

9362da1 gitdemo / hello.c

ckresse Greet OSADL instead of the world. 9362da1 · 2 days ago

5 lines (5 loc) · 56 Bytes

Code **Blame** Raw Copy Download Edit

Older                 Newer

Contributors 1

Time	Author	Commit Message	Line	Code
3 days ago	ckresse	Add a new file to greet the wo...	1	<code>#include &lt;stdio.h&gt;</code>
			2	<code>int main()</code>
			3	<code>{</code>
2 days ago	ckresse	Greet OSADL instead of the w...	4	<code>puts("Hello OSADL");</code>
3 days ago	ckresse	Add a new file to greet the wo...	5	<code>}</code>



# Git blame

📄 d38c0d5 gitdemo / hello.c 🔍 Go to file

**ckresse** Add a new file to greet the world. d38c0d5 · 3 days ago

5 lines (5 loc) · 56 Bytes

Code **Blame** Raw Copy Download Edit

Older Newer Contributors 1

```
3 days ago  Add a new file to greet the wo...  
1 #include <stdio.h>  
2 int main()  
3 {  
4     puts("Hello world");  
5 }
```

# open62541 Github repository

The screenshot shows the GitHub repository page for `open62541`. The repository is public and has 173 watchers, 1.3k forks, and 2.7k stars. It is a C project with 43 branches and 92 tags. The repository is maintained by `tomitkl` and `Jpfr`, with 11,055 commits. The repository is licensed under MPL-2.0 and CC0-1.0. The repository is a source implementation of OPC UA (OPC Unified Architecture) aka IEC 62541 licensed under Mozilla Public License v2.0. The repository is categorized under `client`, `server`, `pubsub`, `industrial-automation`, `publish-subscribe`, `opc`, `opc-ua`, `iec-62541`, `opcua`, and `tsn`. The repository has a README, Code of conduct, Security policy, Activity, Custom properties, 2.7k stars, 173 watching, and 1.3k forks. The repository has 79 releases, with the latest release being `v1.4.8` on Nov 25, 2024. The repository has no packages published. The repository has 291 contributors and 277 contributors. The repository is written in C (93.4%), Python (2.8%), CMake (2.1%), Shell (0.9%), C++ (0.6%), PowerShell (0.1%), and Other (0.1%).

File	Description	Last Commit
<code>.github</code>	refactor(ci): replace use of deprecated set-output ...	last week
<code>arch</code>	fix(arch): Make the POSIX EventLoop compile agai...	last month
<code>deps</code>	fix(plugins): Rename dirent.h to tr_dirent.h	3 days ago
<code>doc</code>	feat(core): Re-add the legacy JSON encoding (pre ...	2 weeks ago
<code>examples</code>	feat(examples): Add examples for the RT offset ta...	2 days ago
<code>include/open62541</code>	feat(pubsub): Offset table generation for PubSub ...	2 days ago
<code>plugins</code>	fix(plugin): Reset only the specified part of the tru...	2 days ago
<code>src</code>	fix(server): Remove declaration of 'I' hides previo...	2 days ago
<code>tests</code>	feat(pubsub): Offset table generation for PubSub ...	2 days ago
<code>tools</code>	refactor(tools): Use pylint suggestions to refactor ...	2 days ago
<code>.clang-format</code>	ClangFormat: Correct order of includes	6 years ago
<code>.clang-tidy</code>	CMake: add static analyzer during build process (...	6 years ago
<code>.cquery</code>	Add .cquery file for code completion in vim, VS C...	7 years ago
<code>.dockerignore</code>	Create multi-stage Dockerfile to minimize final D...	4 years ago
<code>.editorconfig</code>	Add clang-format and editorconfig file	8 years ago
<code>.gitignore</code>	fix(core): Remove /examples and /docs from the ....	2 days ago
<code>.gitmodules</code>	feat(plugin): Integration of nodesetLoader functi...	3 years ago
<code>CHANGES.md</code>	feat(core): Re-add the legacy JSON encoding (pre ...	2 weeks ago
<code>CMakeLists.txt</code>	fix(plugins): Rename dirent.h to tr_dirent.h	3 days ago
<code>CODE_OF_CONDUCT.md</code>	Doc: Fix various Codacy formatting issues	7 years ago
<code>CONTRIBUTING.md</code>	refactor(core): Use memset to zero out instead of ...	3 years ago
<code>CPPLINT.cfg</code>	Linting: Additional fixes and cleanup	6 years ago
<code>FEATURES.md</code>	refactor(core): Update FEATURES.md	2 years ago
<code>ISSUE_TEMPLATE.md</code>	Add more help for cmake options	6 years ago
<code>LICENSE</code>	Update LICENSE	9 years ago
<code>LICENSE-CCO</code>	clean up include structure	11 years ago
<code>README.md</code>	refactor(docs): Clean up and update the README	2 months ago

[https://github.com/  
open62541/open62541](https://github.com/open62541/open62541)