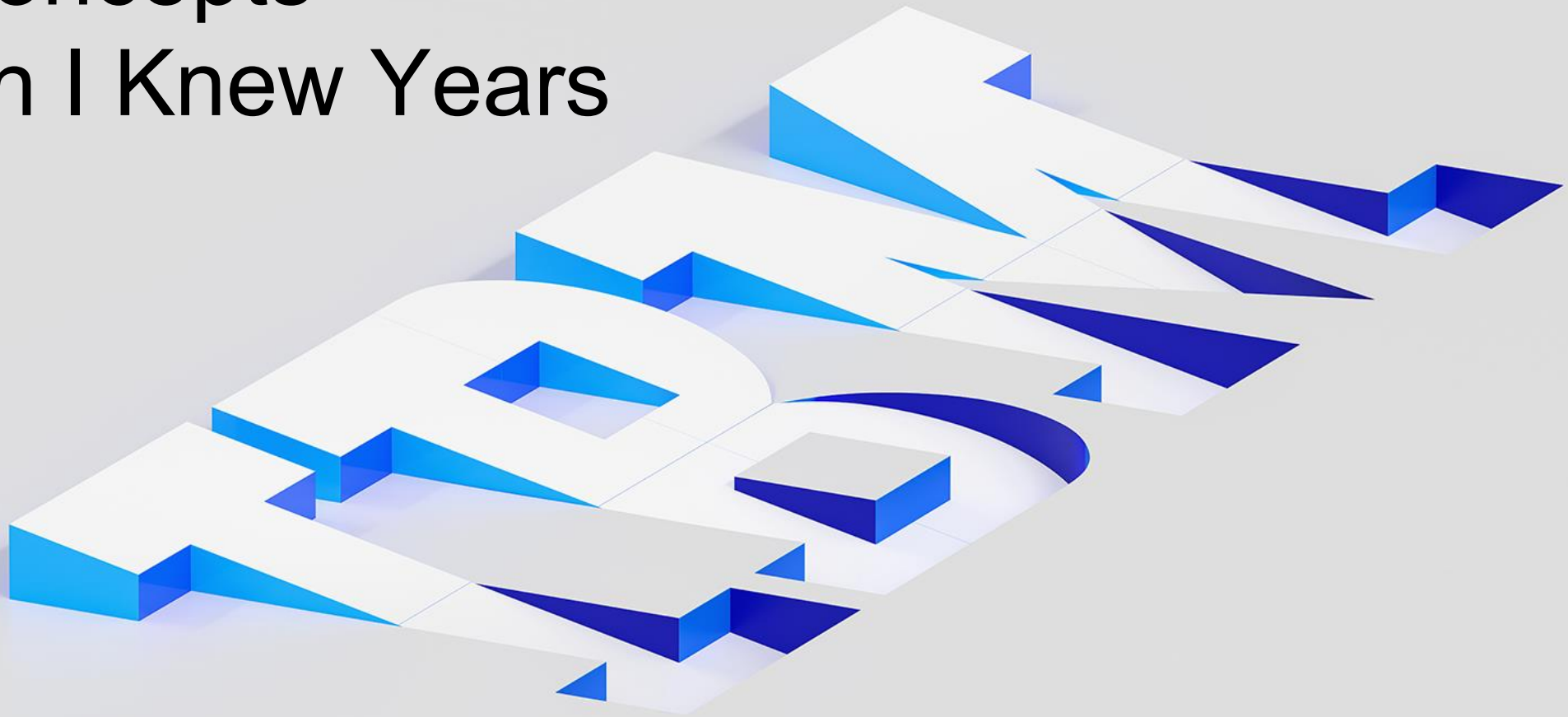
Git Concepts I Wish I Knew Years Ago



Sanjula Ganepola
Software Developer – IBM i App Dev & AI Toolchain
Sanjula.Ganepola@ibm.com
2025 February

© Copyright IBM Corporation 2025

Agenda

- Git Basics
- Using Git in a Team
- Branching and Merge Conflicts
- Merge Commits vs. Rebase Merging vs. Squash Merging
- Using Git on IBM i
- Tips and Tricks
- Git in VS Code

© Copyright IBM Corporation 2024

How well do you know Git?

- 🤩 **Never used it** I have not used Git before, but I am interested!
- 🧐 **Beginner** I just started learning the basics.
- 👨‍🎓 **Intermediate** I understand the basics, but I am not familiar with advanced concepts.
- 🕵️ **Expert** I have a deep understanding of Git.

© Copyright IBM Corporation 2024

Git Basics

© Copyright IBM Corporation 2024

Let’s get setup

Let’s do this once …

1. Install Git on your PC:
   - https://git-scm.com/downloads
2. Setup your name and email in your Git config:

```
MINGW64:/c/Users/SanjulaGanepola

AzureAD+SanjulaGanepola@IBM-PW0AC6C8 MINGW64 ~
$ git config --global user.name "Sanjula Ganepola"

AzureAD+SanjulaGanepola@IBM-PW0AC6C8 MINGW64 ~
$ git config --global user.email "Sanjula.Ganepola@ibm.com"

AzureAD+SanjulaGanepola@IBM-PW0AC6C8 MINGW64 ~
$
```


© Copyright IBM Corporation 2024

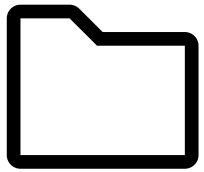
Setup a new project with Git

MINGW64:/c/Users/SanjulaGanepola/Desktop/my-awesome-project

```
AzureAD+SanjulaGanepola@IBM-PW0AC6C8 MINGW64 ~/Desktop
$ mkdir my-awesome-project

AzureAD+SanjulaGanepola@IBM-PW0AC6C8 MINGW64 ~/Desktop
$ cd my-awesome-project/

AzureAD+SanjulaGanepola@IBM-PW0AC6C8 MINGW64 ~/Desktop/my-awesome-project
$ touch awesomeness.js

AzureAD+SanjulaGanepola@IBM-PW0AC6C8 MINGW64 ~/Desktop/my-awesome-project
$ git init
Initialized empty Git repository in C:/Users/SanjulaGanepola/Desktop/my-awesome-project/.git/

AzureAD+SanjulaGanepola@IBM-PW0AC6C8 MINGW64 ~/Desktop/my-awesome-project (master)
$ ls .git/
HEAD  config  description  hooks/  info/  objects/  refs/

AzureAD+SanjulaGanepola@IBM-PW0AC6C8 MINGW64 ~/Desktop/my-awesome-project (master)
$
```


© Copyright IBM Corporation 2024

Where does your code live?



WORKING
DIRECTORY

Where we actively edit local files (our playground)

STAGING
AREA

Temporary location for changes before committing

LOCAL
REPO

Where we store committed changes locally

REMOTE
REPO

A sever (such as GitHub) for sharing and backing up code

© Copyright IBM Corporation 2024

Moving code to the staging area

WORKING DIRECTORY → STAGING AREA | LOCAL REPO | REMOTE REPO

MINGW64:/c/Users/SanjulaGanepola/Desktop/my-awesome-project

```
AzureAD+SanjulaGanepola@IBM-PW0AC6C8 MINGW64 ~/Desktop/my-awesome-project (master)
$ git status
On branch master

No commits yet

Untracked files:
  (use "git add <file>..." to include in what will be committed)
        awesomeness.js

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

AzureAD+SanjulaGanepola@IBM-PW0AC6C8 MINGW64 ~/Desktop/my-awesome-project (master)
$ git add awesomeness.js

AzureAD+SanjulaGanepola@IBM-PW0AC6C8 MINGW64 ~/Desktop/my-awesome-project (master)
$ git status
On branch master

No commits yet

Changes to be committed:
  (use "git rm --cached <file>..." to unstage)
        new file:   awesomeness.js


AzureAD+SanjulaGanepola@IBM-PW0AC6C8 MINGW64 ~/Desktop/my-awesome-project (master)
$
```


© Copyright IBM Corporation 2024

Excluding files from being tracked

WORKING DIRECTORY → STAGING AREA

LOCAL REPO

REMOTE REPO

MINGW64:/c/Users/SanjulaGanepola/Desktop/my-awesome-project

```
AzureAD+SanjulaGanepola@IBM-PW0AC6C8 MINGW64 ~/Desktop/my-awesome-project (master)
$ echo MY_API_KEY=NO-ONE-STEAL-THIS > .env

AzureAD+SanjulaGanepola@IBM-PW0AC6C8 MINGW64 ~/Desktop/my-awesome-project (master)
$ git status
On branch master

No commits yet

Changes to be committed:
  (use "git rm --cached <file>..." to unstage)
        new file:   awesomeness.js

Untracked files:
  (use "git add <file>..." to include in what will be committed)
        .env


AzureAD+SanjulaGanepola@IBM-PW0AC6C8 MINGW64 ~/Desktop/my-awesome-project (master)
$ echo .env > .gitignore

AzureAD+SanjulaGanepola@IBM-PW0AC6C8 MINGW64 ~/Desktop/my-awesome-project (master)
$ echo MY_API_KEY= > .env.sample

AzureAD+SanjulaGanepola@IBM-PW0AC6C8 MINGW64 ~/Desktop/my-awesome-project (master)
$ git status
On branch master

No commits yet

Changes to be committed:
  (use "git rm --cached <file>..." to unstage)
        new file:   awesomeness.js

Untracked files:
  (use "git add <file>..." to include in what will be committed)
        .env.sample
        .gitignore


AzureAD+SanjulaGanepola@IBM-PW0AC6C8 MINGW64 ~/Desktop/my-awesome-project (master)
$ git add .
warning: in the working copy of '.env.sample', LF will be replaced by CRLF the next time Git touches it
warning: in the working copy of '.gitignore', LF will be replaced by CRLF the next time Git touches it

AzureAD+SanjulaGanepola@IBM-PW0AC6C8 MINGW64 ~/Desktop/my-awesome-project (master)
$
```


© Copyright IBM Corporation 2024

Committing files to the local repository

WORKING DIRECTORY → STAGING AREA → LOCAL REPO → REMOTE REPO

MINGW64:/c/Users/SanjulaGanepola/Desktop/my-awesome-project

```
AzureAD+SanjulaGanepola@IBM-PW0AC6C8 MINGW64 ~/Desktop/my-awesome-project (master)
$ git commit -s -m "Initial commit"
[master (root-commit) fd28db6] Initial commit
 3 files changed, 2 insertions(+)
 create mode 100644 .env.sample
 create mode 100644 .gitignore
 create mode 100644 awesomeness.js

AzureAD+SanjulaGanepola@IBM-PW0AC6C8 MINGW64 ~/Desktop/my-awesome-project (master)
$ git status
On branch master
nothing to commit, working tree clean

AzureAD+SanjulaGanepola@IBM-PW0AC6C8 MINGW64 ~/Desktop/my-awesome-project (master)
$ git log
commit fd28db67cc90e5299aff7b1fb2c13be535c149f2 (HEAD -> master)
Author: Sanjula Ganepola <Sanjula.Ganepola@ibm.com>
Date:   Tue Apr 1 16:41:49 2025 -0400

    Initial commit

    Signed-off-by: Sanjula Ganepola <Sanjula.Ganepola@ibm.com>

AzureAD+SanjulaGanepola@IBM-PW0AC6C8 MINGW64 ~/Desktop/my-awesome-project (master)
$
```


© Copyright IBM Corporation 2024

Committing more files

WORKING DIRECTORY → STAGING AREA → LOCAL REPO REMOTE REPO

MINGW64:/c/Users/SanjulaGanepola/Desktop/my-awesome-project

```
AzureAD+SanjulaGanepola@IBM-PW0AC6C8 MINGW64 ~/Desktop/my-awesome-project (master)
$ echo "console.log('This code is so cool');" > awesomeness.js

AzureAD+SanjulaGanepola@IBM-PW0AC6C8 MINGW64 ~/Desktop/my-awesome-project (master)
$ git add awesomeness.js
warning: in the working copy of 'awesomeness.js', LF will be replaced by CRLF the next t
ime Git touches it

AzureAD+SanjulaGanepola@IBM-PW0AC6C8 MINGW64 ~/Desktop/my-awesome-project (master)
$ git commit -s -m "Add console log"
[master e7809ee] Add console log
 2 files changed, 1 insertion(+)
 create mode 100644 coolCode.js

AzureAD+SanjulaGanepola@IBM-PW0AC6C8 MINGW64 ~/Desktop/my-awesome-project (master)
$ git log --oneline
e7809ee (HEAD -> master) Add console log
fd28db6 Initial commit

AzureAD+SanjulaGanepola@IBM-PW0AC6C8 MINGW64 ~/Desktop/my-awesome-project (master)
$
```


© Copyright IBM Corporation 2024

Checking out a specific commit

MINGW64:/c/Users/SanjulaGanepola/Desktop/my-awesome-project

```
AzureAD+SanjulaGanepola@IBM-PW0AC6C8 MINGW64 ~/Desktop/my-awesome-project (master)
$ git checkout fd28db6
Note: switching to 'fd28db6'.

You are in 'detached HEAD' state. You can look around, make experimental
changes and commit them, and you can discard any commits you make in this
state without impacting any branches by switching back to a branch.
```

e7809ee Add console log

awesomeness.js

File Edit View

```
console.log('This code is so cool');
```

Ln 1, Col 1 | 37 characters | 100% | Windows (CRL | UTF-8

fd28db6 Initial commit

awesomeness.js

File Edit View

Ln 1, Col 1 | 0 characters | 100% | Windows (CRL | UTF-8

© Copyright IBM Corporation 2024

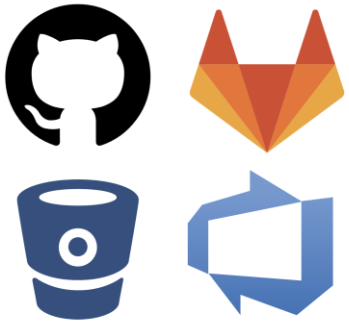
Comparing two commits

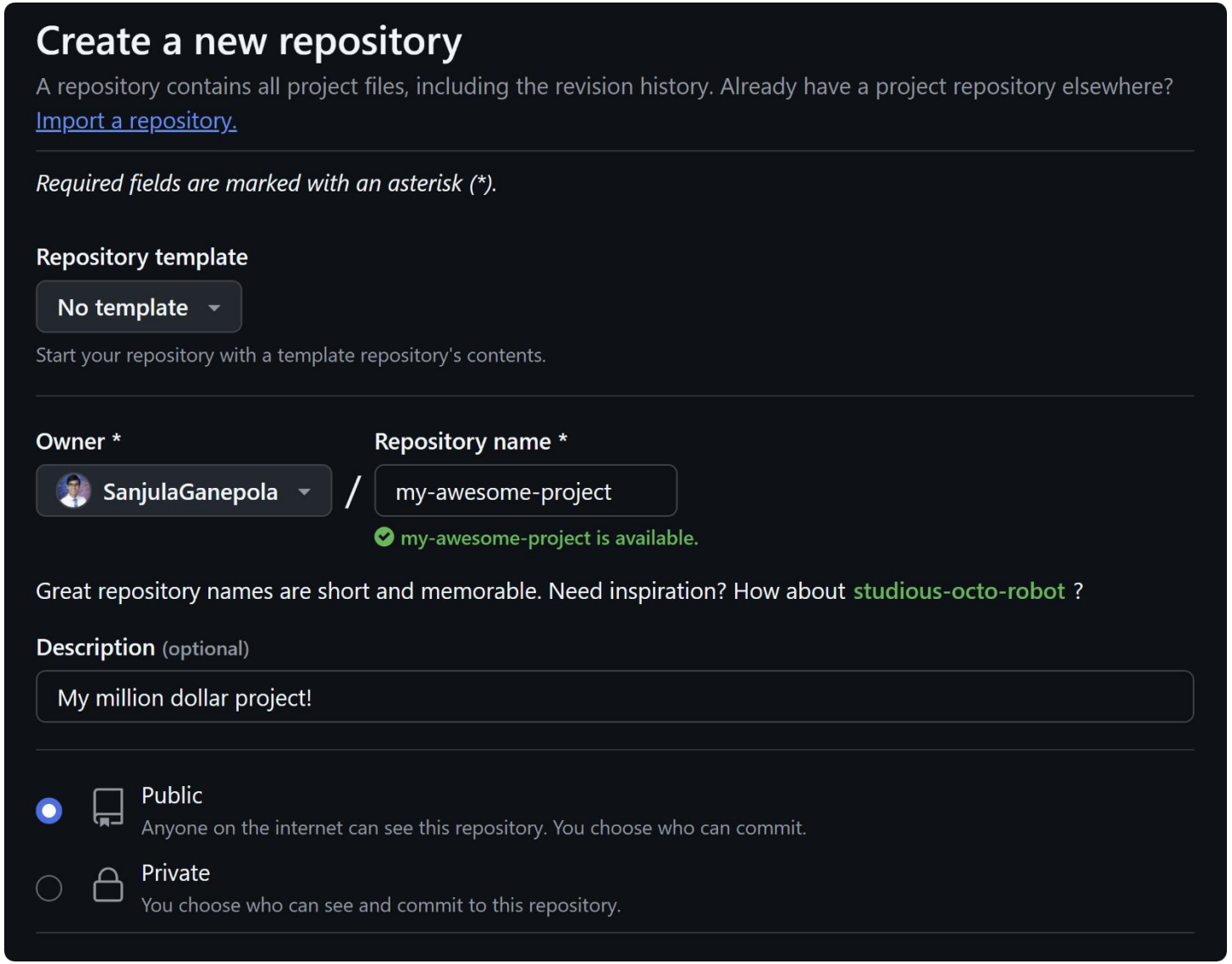
MINGW64:/c/Users/SanjulaGanepola/Desktop/my-awesome-project

```
AzureAD+SanjulaGanepola@IBM-PW0AC6C8 MINGW64 ~/Desktop/my-awesome-project ((fd28db6...))
$ git switch -
Previous HEAD position was fd28db6 Initial commit
Switched to branch 'master'

AzureAD+SanjulaGanepola@IBM-PW0AC6C8 MINGW64 ~/Desktop/my-awesome-project (master)
$ git diff fd28db6 e7809ee
diff --git a/awesomeness.js b/awesomeness.js
index e69de29..875aae3 100644
--- a/awesomeness.js
+++ b/awesomeness.js
@@ -0,0 +1 @@
+console.log('This code is so cool');
diff --git a/coolCode.js b/coolCode.js
new file mode 100644
index 0000000..e69de29

AzureAD+SanjulaGanepola@IBM-PW0AC6C8 MINGW64 ~/Desktop/my-awesome-project (master)
$
```


© Copyright IBM Corporation 2024

Creating a remote repository





© Copyright IBM Corporation 2024

Using the quick setup instructions

https://github.com/SanjulaGanepola/my-awesome-project

Quick setup — if you've done this kind of thing before

Set up in Desktop or HTTPS SSH https://github.com/SanjulaGanepola/my-awesome-project.git

Get started by creating a new file or uploading an existing file. We recommend every repository include a README, LICENSE, and .gitignore.

...or create a new repository on the command line

```
echo "# my-awesome-project" >> README.md
git init
git add README.md
git commit -m "first commit"
git branch -M main
git remote add origin https://github.com/SanjulaGanepola/my-awesome-project.git
git push -u origin main
```

...or push an existing repository from the command line

```
git remote add origin https://github.com/SanjulaGanepola/my-awesome-project.git
git branch -M main
git push -u origin main
```


© Copyright IBM Corporation 2024

Pushing to the remote repository

WORKING DIRECTORY | STAGING AREA | LOCAL REPO → REMOTE REPO

MINGW64:/c/Users/SanjulaGanepola/Desktop/my-awesome-project

```
AzureAD+SanjulaGanepola@IBM-PW0AC6C8 MINGW64 ~/Desktop/my-awesome-project (master)
$ git remote add origin https://github.com/SanjulaGanepola/my-awesome-project.git

AzureAD+SanjulaGanepola@IBM-PW0AC6C8 MINGW64 ~/Desktop/my-awesome-project (master)
$ git branch -M main

AzureAD+SanjulaGanepola@IBM-PW0AC6C8 MINGW64 ~/Desktop/my-awesome-project (main)
$ git push -u origin main
Enumerating objects: 8, done.
Counting objects: 100% (8/8), done.
Delta compression using up to 20 threads
Compressing objects: 100% (4/4), done.
Writing objects: 100% (8/8), 591 bytes | 295.00 KiB/s, done.
Total 8 (delta 2), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (2/2), done.
To https://github.com/SanjulaGanepola/my-awesome-project.git
 * [new branch]      main -> main
branch 'main' set up to track 'origin/main'.

AzureAD+SanjulaGanepola@IBM-PW0AC6C8 MINGW64 ~/Desktop/my-awesome-project (main)
$
```


© Copyright IBM Corporation 2024

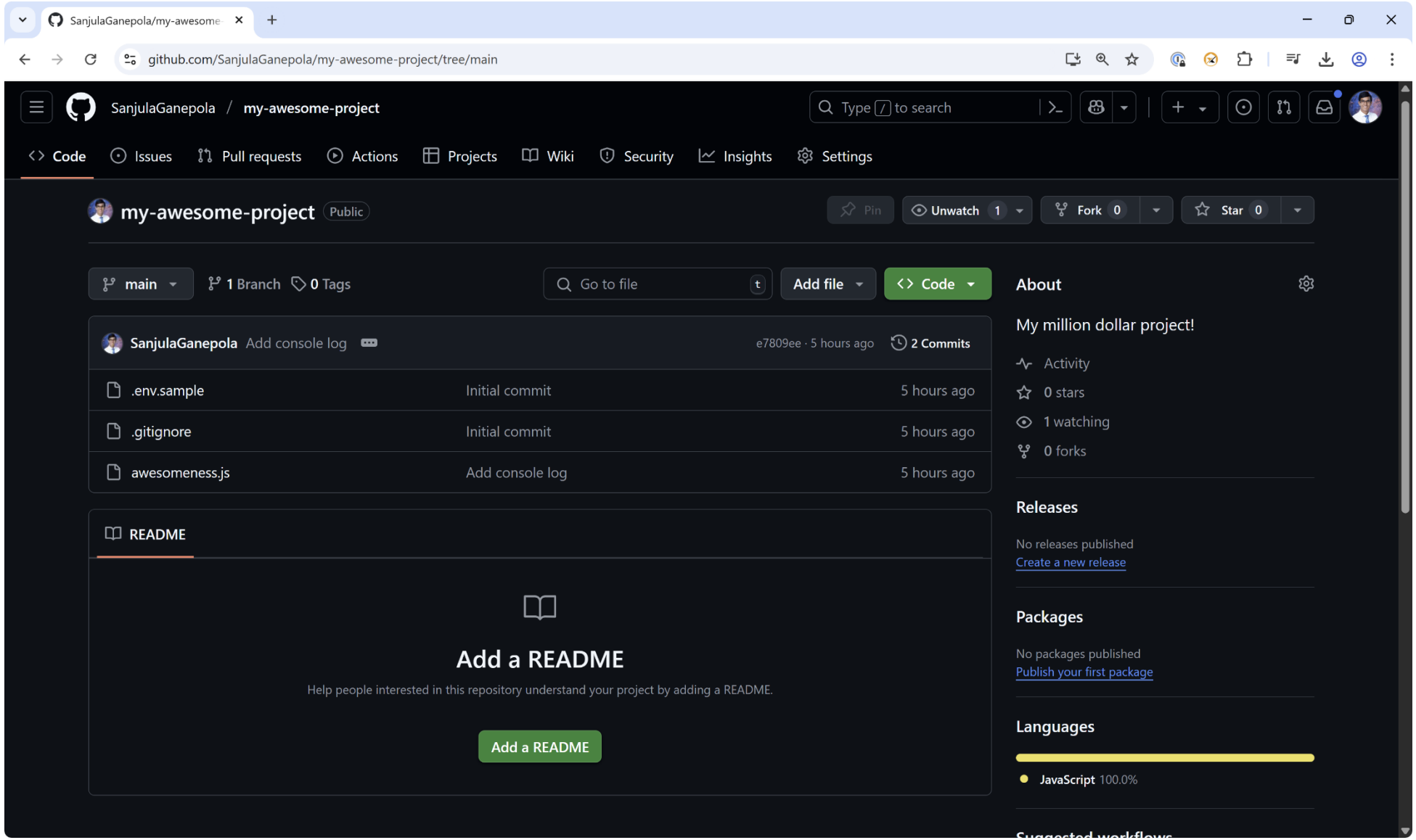
Listing the remote repositories

MINGW64:/c/Users/SanjulaGanepola/Desktop/my-awesome-project

```
AzureAD+SanjulaGanepola@IBM-PW0AC6C8 MINGW64 ~/Desktop/my-awesome-project (main)
$ git remote -v
origin  https://github.com/SanjulaGanepola/my-awesome-project.git (fetch)
origin  https://github.com/SanjulaGanepola/my-awesome-project.git (push)

AzureAD+SanjulaGanepola@IBM-PW0AC6C8 MINGW64 ~/Desktop/my-awesome-project (main)
$
```


© Copyright IBM Corporation 2024

Viewing the remote repository

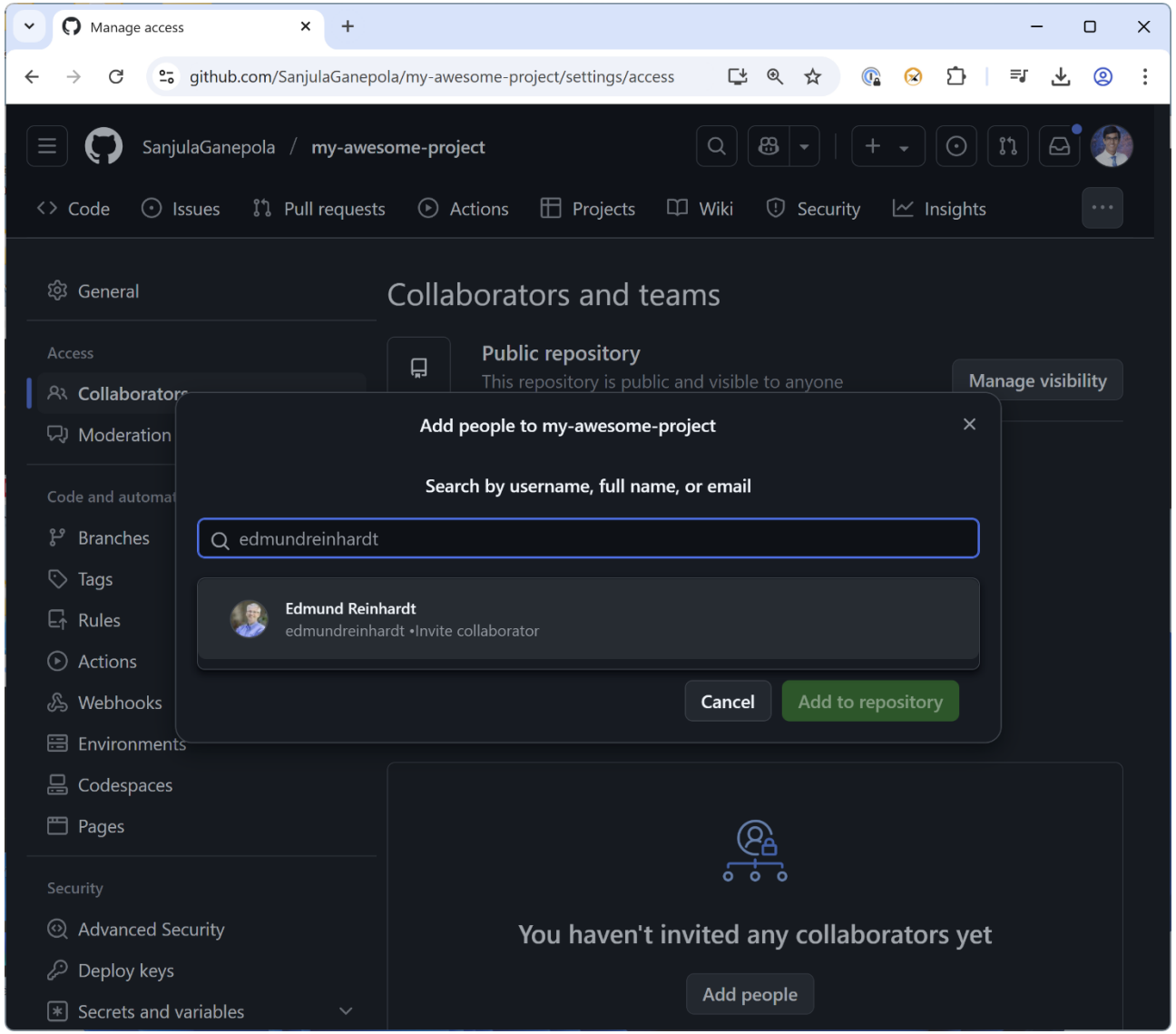


© Copyright IBM Corporation 2024

Using Git in a Team

© Copyright IBM Corporation 2024

Adding collaborators



© Copyright IBM Corporation 2024

Cloning the remote repository

WORKING DIRECTORY | STAGING AREA | LOCAL REPO ← REMOTE REPO

Edmund
(Team Member)

MINGW64:/c/Users/SanjulaGanepola/Desktop/developer2

```
AzureAD+SanjulaGanepola@IBM-PW0AC6C8 MINGW64 ~/Desktop/developer2
$ git clone https://github.com/SanjulaGanepola/my-awesome-project.git
Cloning into 'my-awesome-project'...
remote: Enumerating objects: 10, done.
remote: Counting objects: 100% (10/10), done.
remote: Compressing objects: 100% (3/3), done.
remote: Total 10 (delta 3), reused 10 (delta 3), pack-reused 0 (from 0)
Receiving objects: 100% (10/10), done.
Resolving deltas: 100% (3/3), done.

AzureAD+SanjulaGanepola@IBM-PW0AC6C8 MINGW64 ~/Desktop/developer2
$
```


© Copyright IBM Corporation 2024

Working → Staging → Committing → Pushing

WORKING DIRECTORY → STAGING AREA → LOCAL REPO → REMOTE REPO

Edmund
(Team Member)

MINGW64:/c/Users/SanjulaGanepola/Desktop/developer2/my-awesome-project

```
AzureAD+SanjulaGanepola@IBM-PW0AC6C8 MINGW64 ~/Desktop/developer2/my-awesome-project (main)
$ touch helper.js

AzureAD+SanjulaGanepola@IBM-PW0AC6C8 MINGW64 ~/Desktop/developer2/my-awesome-project (main)
$ git add helper.js

AzureAD+SanjulaGanepola@IBM-PW0AC6C8 MINGW64 ~/Desktop/developer2/my-awesome-project (main)
$ git commit -s -m "Add helper functions"
[main 6bbc0e6] Add helper functions
 1 file changed, 0 insertions(+), 0 deletions(-)
 create mode 100644 helper.js

AzureAD+SanjulaGanepola@IBM-PW0AC6C8 MINGW64 ~/Desktop/developer2/my-awesome-project (main)
$ git push
Enumerating objects: 3, done.
Counting objects: 100% (3/3), done.
Delta compression using up to 20 threads
Compressing objects: 100% (2/2), done.
Writing objects: 100% (2/2), 291 bytes | 291.00 KiB/s, done.
Total 2 (delta 1), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (1/1), completed with 1 local object.
To https://github.com/SanjulaGanepola/my-awesome-project.git
   e7809ee..6bbc0e6  main -> main

AzureAD+SanjulaGanepola@IBM-PW0AC6C8 MINGW64 ~/Desktop/developer2/my-awesome-project (main)
$
```


© Copyright IBM Corporation 2024

Option 1: Fetching and merging from the remote repository

WORKING DIRECTORY | STAGING AREA | LOCAL REPO | REMOTE REPO

← *MERGE* ← *FETCH*

Sanjula (Owner)

MINGW64:/c/Users/SanjulaGanepola/Desktop/my-awesome-project

```
AzureAD+SanjulaGanepola@IBM-PW0AC6C8 MINGW64 ~/Desktop/my-awesome-project (main)
$ git fetch
remote: Enumerating objects: 3, done.
remote: Counting objects: 100% (3/3), done.
remote: Compressing objects: 100% (1/1), done.
remote: Total 2 (delta 1), reused 2 (delta 1), pack-reused 0 (from 0)
Unpacking objects: 100% (2/2), 271 bytes | 67.00 KiB/s, done.
From https://github.com/SanjulaGanepola/my-awesome-project
   e7809ee..6bbc0e6  main       -> origin/main

AzureAD+SanjulaGanepola@IBM-PW0AC6C8 MINGW64 ~/Desktop/my-awesome-project (main)
$ git merge
Updating e7809ee..6bbc0e6
Fast-forward
 helper.js | 0
 1 file changed, 0 insertions(+), 0 deletions(-)
 create mode 100644 helper.js

AzureAD+SanjulaGanepola@IBM-PW0AC6C8 MINGW64 ~/Desktop/my-awesome-project (main)
$
```


© Copyright IBM Corporation 2024

Option 2: Pulling from the remote repository

WORKING DIRECTORY	STAGING AREA	LOCAL REPO	REMOTE REPO

PULL

Sanjula (Owner)

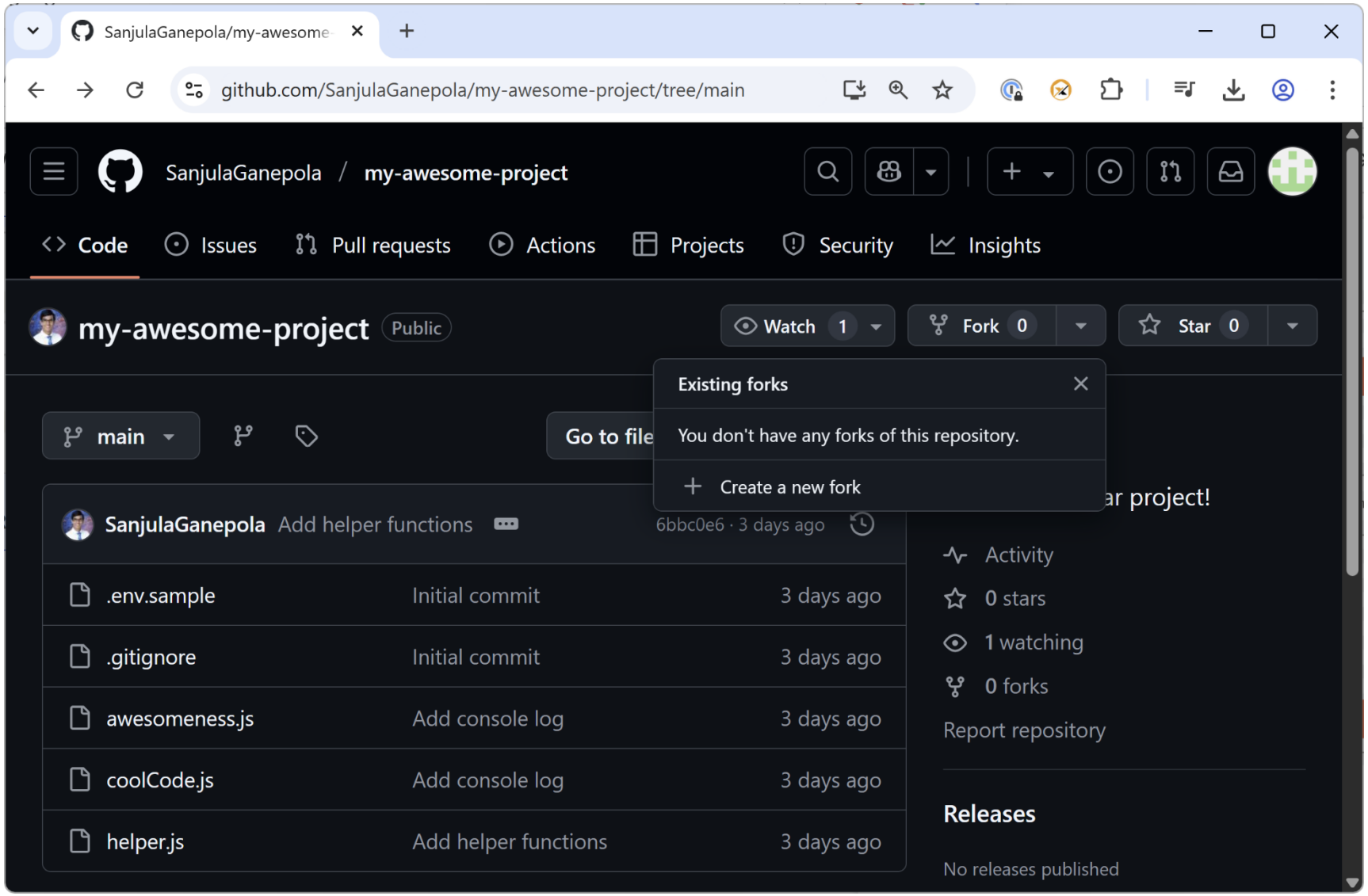
MINGW64:/c/Users/SanjulaGanepola/Desktop/my-awesome-project

```
AzureAD+SanjulaGanepola@IBM-PW0AC6C8 MINGW64 ~/Desktop/my-awesome-project (main)
$ git pull
Already up to date.

AzureAD+SanjulaGanepola@IBM-PW0AC6C8 MINGW64 ~/Desktop/my-awesome-project (main)
$
```


© Copyright IBM Corporation 2024

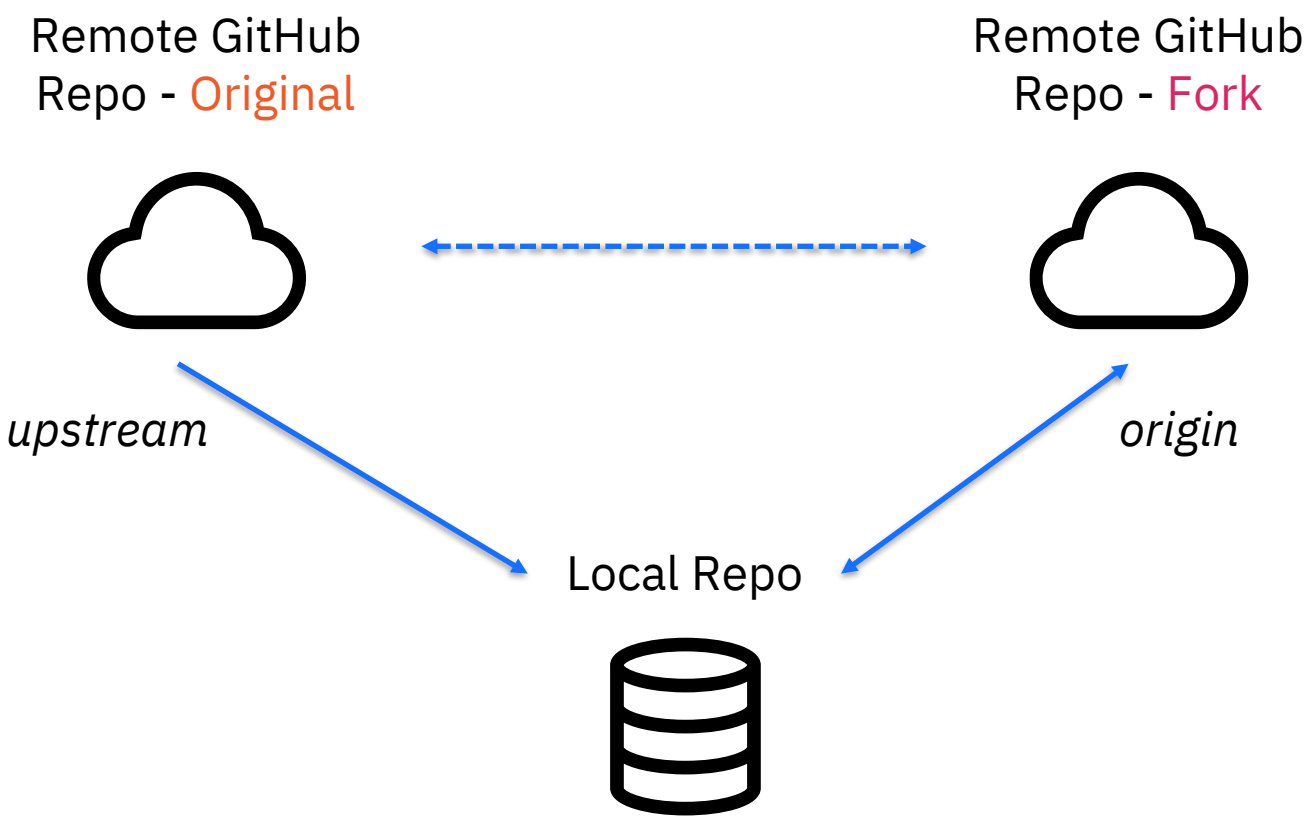
Creating a fork

John
(Outside
Collaborator)



© Copyright IBM Corporation 2024

Upstream vs. Origin



© Copyright IBM Corporation 2024

Listing Remote Repositories

John
(Outside
Collaborator)

MINGW64:/c/Users/SanjulaGanepola/Desktop/fork/my-awesome-project

```
AzureAD+SanjulaGanepola@IBM-PW0AC6C8 MINGW64 ~/Desktop/fork/my-awesome-project (main)
$ git remote add upstream https://github.com/SanjulaGanepola/my-awesome-project.git

AzureAD+SanjulaGanepola@IBM-PW0AC6C8 MINGW64 ~/Desktop/fork/my-awesome-project (main)
$ git remote -v
origin  https://github.com/JohnDoeDemo/my-awesome-project.git (fetch)
origin  https://github.com/JohnDoeDemo/my-awesome-project.git (push)
upstream        https://github.com/SanjulaGanepola/my-awesome-project.git (fetch)
upstream        https://github.com/SanjulaGanepola/my-awesome-project.git (push)

AzureAD+SanjulaGanepola@IBM-PW0AC6C8 MINGW64 ~/Desktop/fork/my-awesome-project (main)
$
```

Edmund
(Team Member)

MINGW64:/c/Users/SanjulaGanepola/Desktop/developer2/my-awesome-project

```
AzureAD+SanjulaGanepola@IBM-PW0AC6C8 MINGW64 ~/Desktop/developer2/my-awesome-project (main)
$ git remote -v
origin  https://github.com/SanjulaGanepola/my-awesome-project.git (fetch)
origin  https://github.com/SanjulaGanepola/my-awesome-project.git (push)

AzureAD+SanjulaGanepola@IBM-PW0AC6C8 MINGW64 ~/Desktop/developer2/my-awesome-project (main)
$
```

Sanjula
(Owner)

MINGW64:/c/Users/SanjulaGanepola/Desktop/my-awesome-project

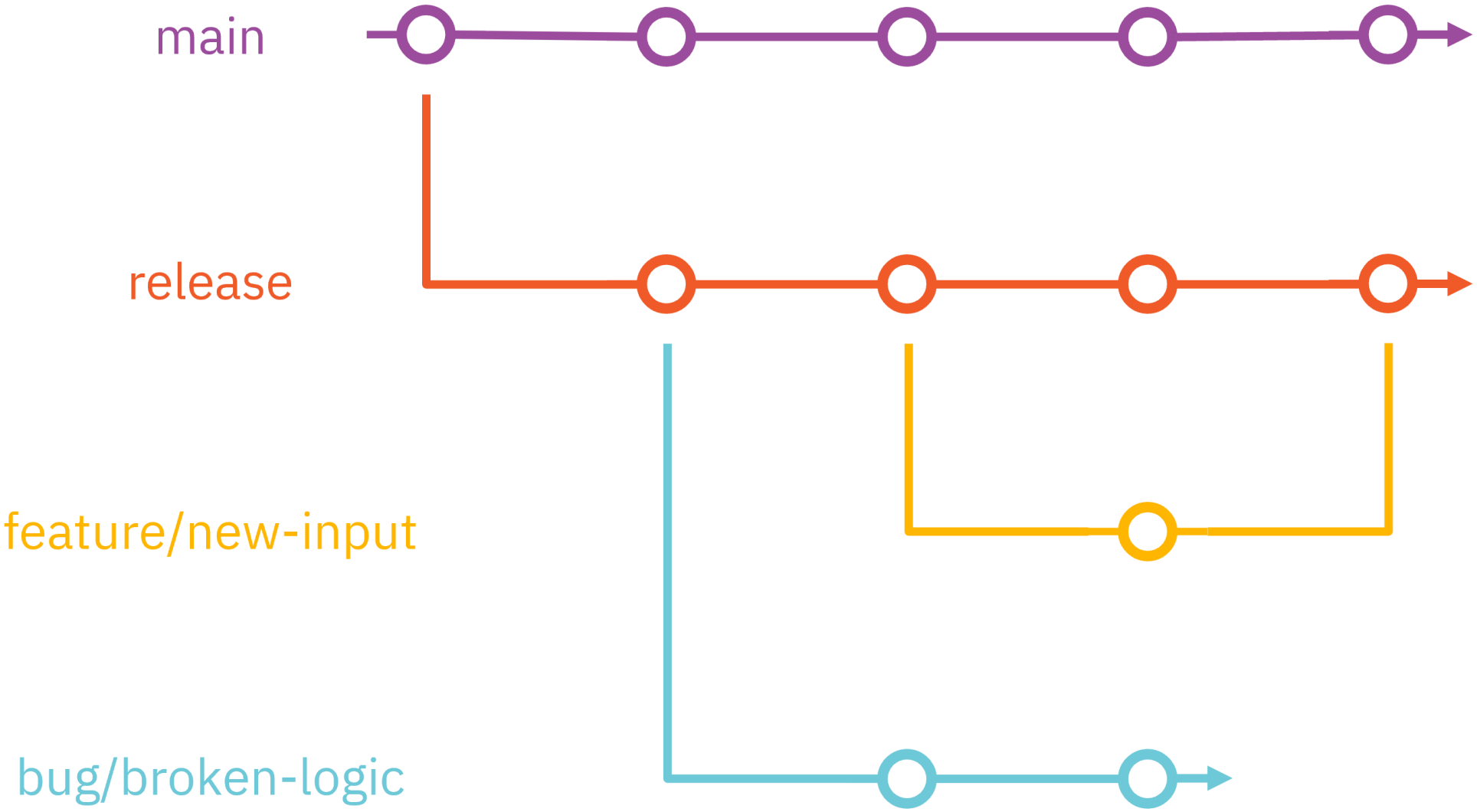
```
AzureAD+SanjulaGanepola@IBM-PW0AC6C8 MINGW64 ~/Desktop/my-awesome-project (main)
$ git remote add JohnDoeDemo https://github.com/JohnDoeDemo/my-awesome-project.git

AzureAD+SanjulaGanepola@IBM-PW0AC6C8 MINGW64 ~/Desktop/my-awesome-project (main)
$ git remote -v
JohnDoeDemo     https://github.com/JohnDoeDemo/my-awesome-project.git (fetch)
JohnDoeDemo     https://github.com/JohnDoeDemo/my-awesome-project.git (push)
origin  https://github.com/SanjulaGanepola/my-awesome-project.git (fetch)
origin  https://github.com/SanjulaGanepola/my-awesome-project.git (push)

AzureAD+SanjulaGanepola@IBM-PW0AC6C8 MINGW64 ~/Desktop/my-awesome-project (main)
$
```


© Copyright IBM Corporation 2024

Branching and Merge Conflicts

© Copyright IBM Corporation 2024

Branching in Git



© Copyright IBM Corporation 2024

Creating and checking out a branch

MINGW64:/c/Users/SanjulaGanepola/Desktop/developer2/my-awesome-project

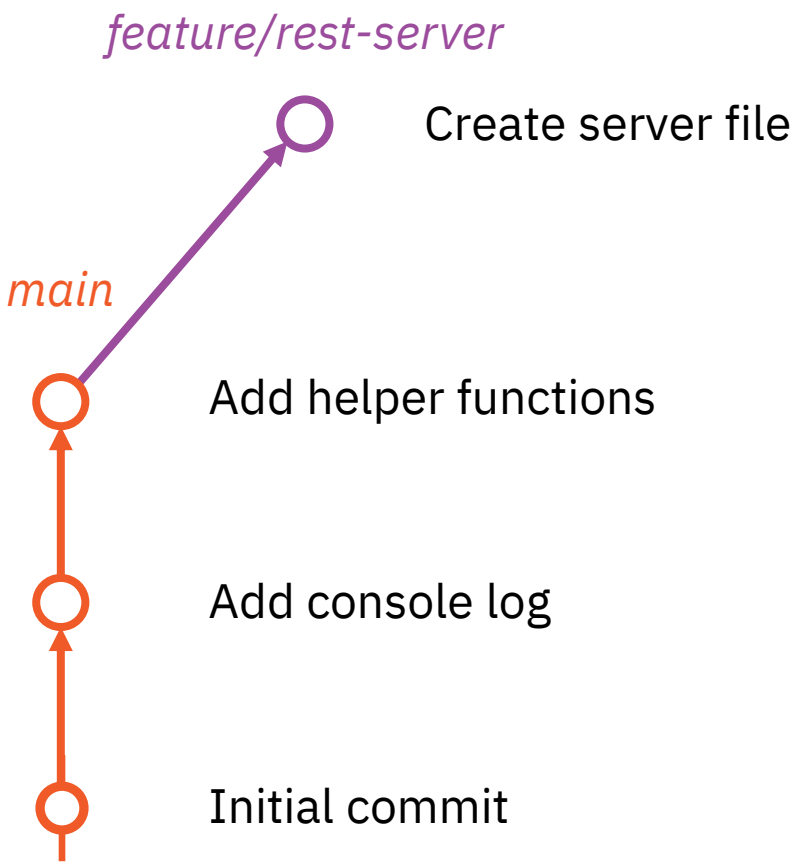
```
AzureAD+SanjulaGanepola@IBM-PW0AC6C8 MINGW64 ~/Desktop/developer2/my-awesome-project (main)
$ git branch feature/rest-server

AzureAD+SanjulaGanepola@IBM-PW0AC6C8 MINGW64 ~/Desktop/developer2/my-awesome-project (main)
$ git branch
  feature/rest-server
* main

AzureAD+SanjulaGanepola@IBM-PW0AC6C8 MINGW64 ~/Desktop/developer2/my-awesome-project (main)
$ git checkout feature/rest-server
Switched to branch 'feature/rest-server'

AzureAD+SanjulaGanepola@IBM-PW0AC6C8 MINGW64 ~/Desktop/developer2/my-awesome-project (feature/rest-server
)
$
```


© Copyright IBM Corporation 2024

Pushing to the branch

MINGW64:/c/Users/SanjulaGanepola/Desktop/developer2/my-awesome-project

```
AzureAD+SanjulaGanepola@IBM-PW0AC6C8 MINGW64 ~/Desktop/developer2/my-awesome-project (feature/rest-server)
$ echo "console.log('Starting server...');" > server.js

AzureAD+SanjulaGanepola@IBM-PW0AC6C8 MINGW64 ~/Desktop/developer2/my-awesome-project (feature/rest-server)
$ git add server.js
warning: in the working copy of 'server.js', LF will be replaced by CRLF the next time Git touches it

AzureAD+SanjulaGanepola@IBM-PW0AC6C8 MINGW64 ~/Desktop/developer2/my-awesome-project (feature/rest-server)
$ git commit -s -m "Create server file"
[feature/rest-server 6c4a7fb] Create server file
 1 file changed, 1 insertion(+)
 create mode 100644 server.js

AzureAD+SanjulaGanepola@IBM-PW0AC6C8 MINGW64 ~/Desktop/developer2/my-awesome-project (feature/rest-server)
$ git push
Enumerating objects: 4, done.
Counting objects: 100% (4/4), done.
Delta compression using up to 20 threads
Compressing objects: 100% (2/2), done.
Writing objects: 100% (3/3), 330 bytes | 330.00 KiB/s, done.
Total 3 (delta 1), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (1/1), completed with 1 local object.
remote:
remote: Create a pull request for 'feature/rest-server' on GitHub by visiting:
remote:      https://github.com/SanjulaGanepola/my-awesome-project/pull/new/feature/rest-server
remote:
To https://github.com/SanjulaGanepola/my-awesome-project.git
 * [new branch]      feature/rest-server -> feature/rest-server
branch 'feature/rest-server' set up to track 'origin/feature/rest-server'.

AzureAD+SanjulaGanepola@IBM-PW0AC6C8 MINGW64 ~/Desktop/developer2/my-awesome-project (feature/rest-server)
$ git log --oneline
6c4a7fb (HEAD -> feature/rest-server, origin/feature/rest-server) Create server file
6bbc0e6 (origin/main, origin/HEAD, main) Add helper functions
e7809ee Add console log
fd28db6 Initial commit

AzureAD+SanjulaGanepola@IBM-PW0AC6C8 MINGW64 ~/Desktop/developer2/my-awesome-project (feature/rest-server)
$
```



© Copyright IBM Corporation 2024

Merging branches

MINGW64:/c/Users/SanjulaGanepola/Desktop/developer2/my-awesome-project

```
AzureAD+SanjulaGanepola@IBM-PW0AC6C8 MINGW64 ~/Desktop/developer2/my-awesome-project (feature/rest-server)
$ git checkout main
Switched to branch 'main'
Your branch is up to date with 'origin/main'.

AzureAD+SanjulaGanepola@IBM-PW0AC6C8 MINGW64 ~/Desktop/developer2/my-awesome-project (main)
$ git merge feature/rest-server
Updating 6bbc0e6..6c4a7fb
Fast-forward
 server.js | 1 +
 1 file changed, 1 insertion(+)
 create mode 100644 server.js

AzureAD+SanjulaGanepola@IBM-PW0AC6C8 MINGW64 ~/Desktop/developer2/my-awesome-project (main)
$ git push
Total 0 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
To https://github.com/SanjulaGanepola/my-awesome-project.git
   6bbc0e6..6c4a7fb  main -> main

AzureAD+SanjulaGanepola@IBM-PW0AC6C8 MINGW64 ~/Desktop/developer2/my-awesome-project (main)
$ git log --oneline
6c4a7fb (HEAD -> main, origin/main, origin/feature/rest-server, origin/HEAD, feature/rest-server) Create serve
r file
6bbc0e6 Add helper functions
e7809ee Add console log
fd28db6 Initial commit

AzureAD+SanjulaGanepola@IBM-PW0AC6C8 MINGW64 ~/Desktop/developer2/my-awesome-project (main)
$
```


© Copyright IBM Corporation 2024

Fast forward merge vs non-fast forward merge

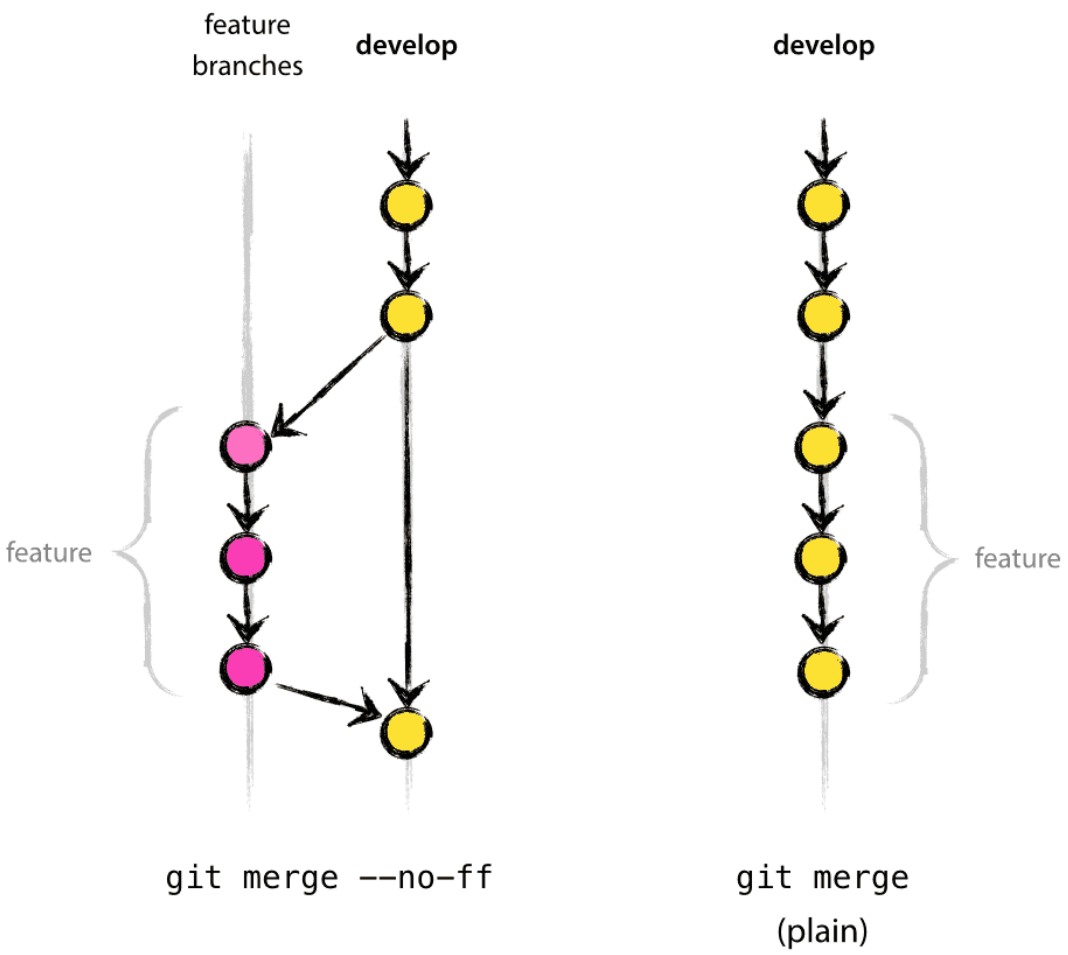
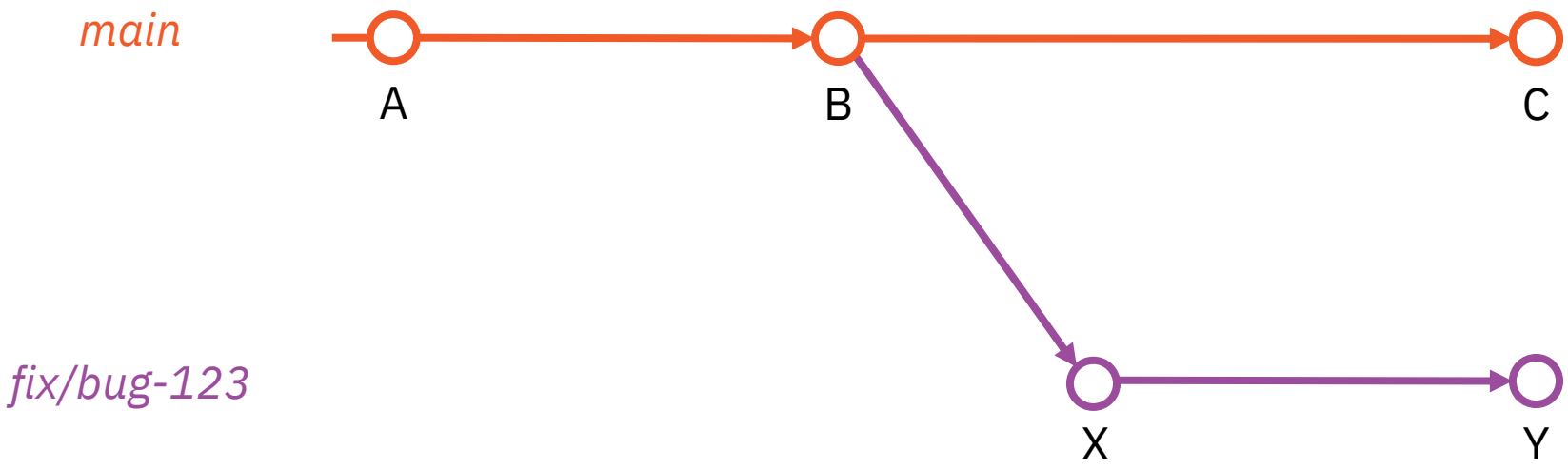


Image from nvie.com/posts/a-successful-git-branching-model/
by Vincent Driessen

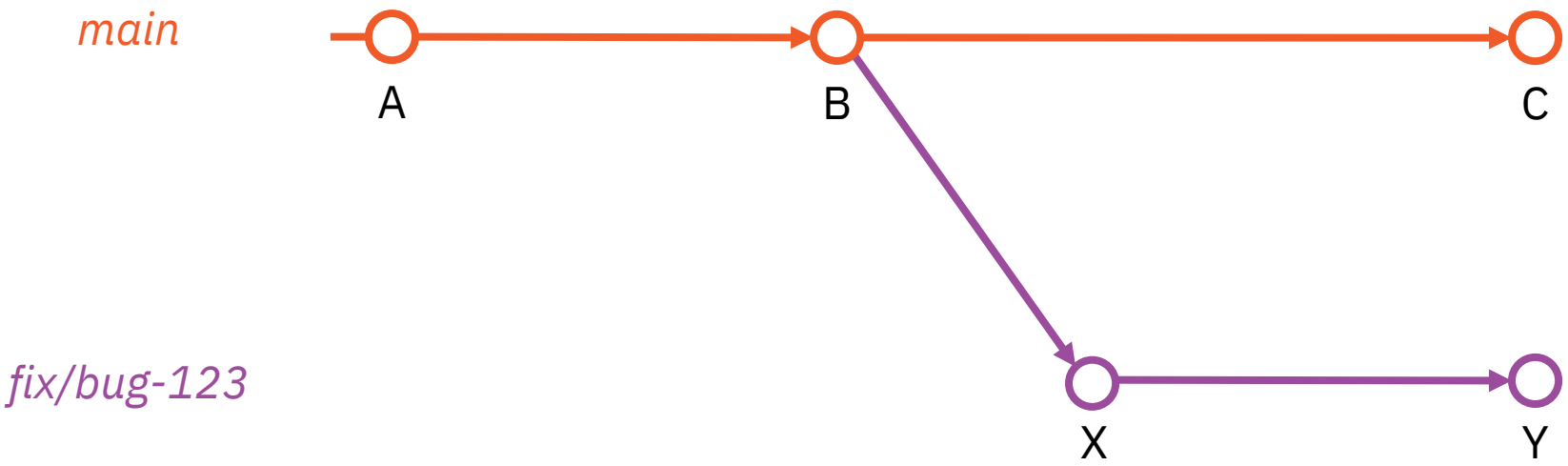
© Copyright IBM Corporation 2024

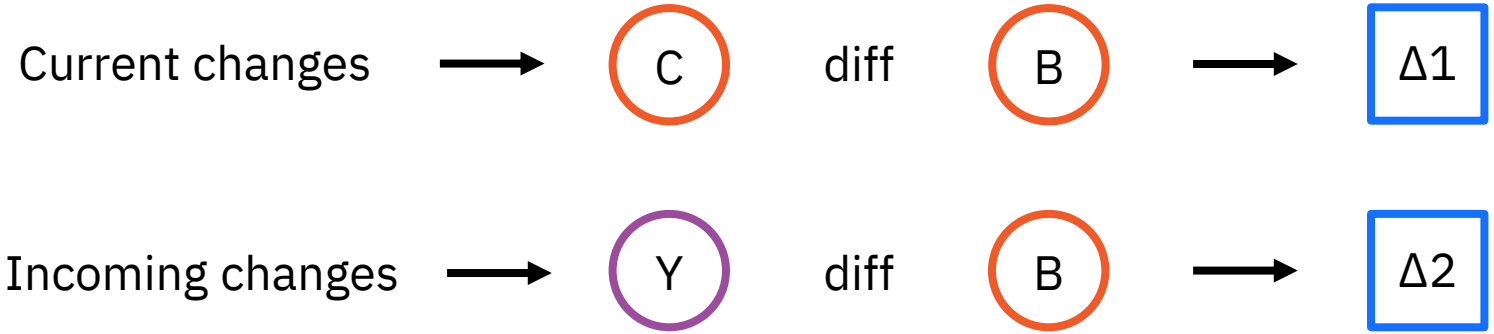
When can merge conflicts happen?



© Copyright IBM Corporation 2024

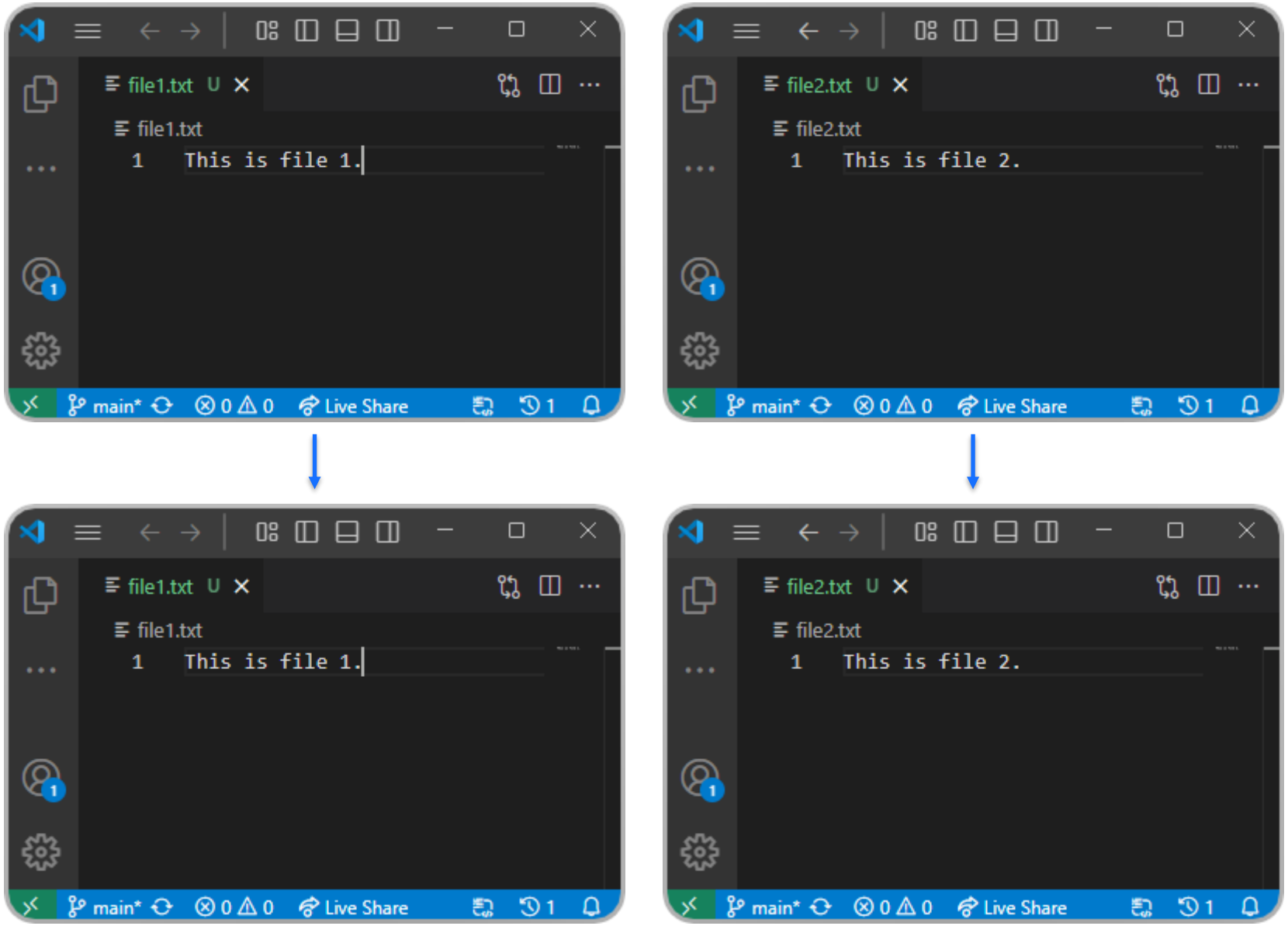
3-way merge





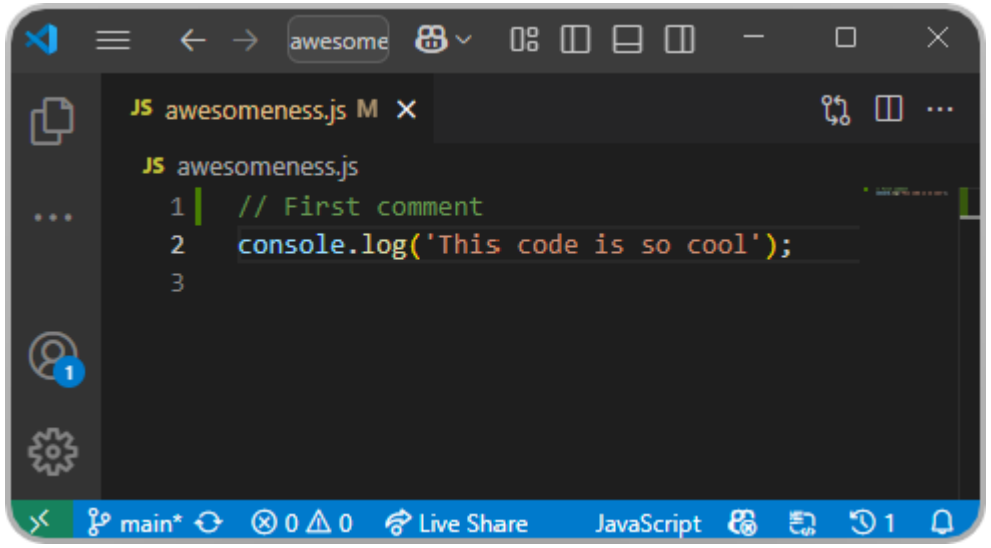
© Copyright IBM Corporation 2024

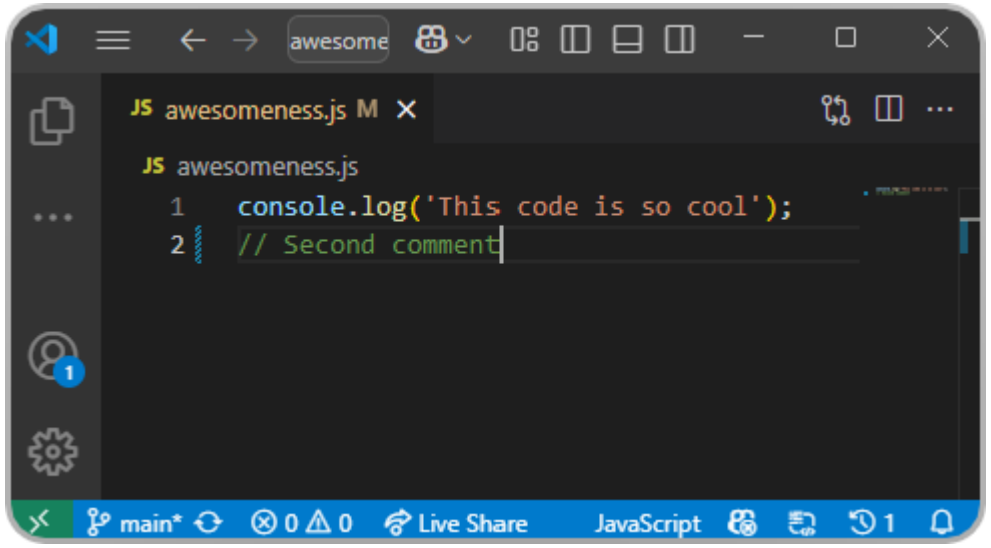
Scenario 1: No file overlap

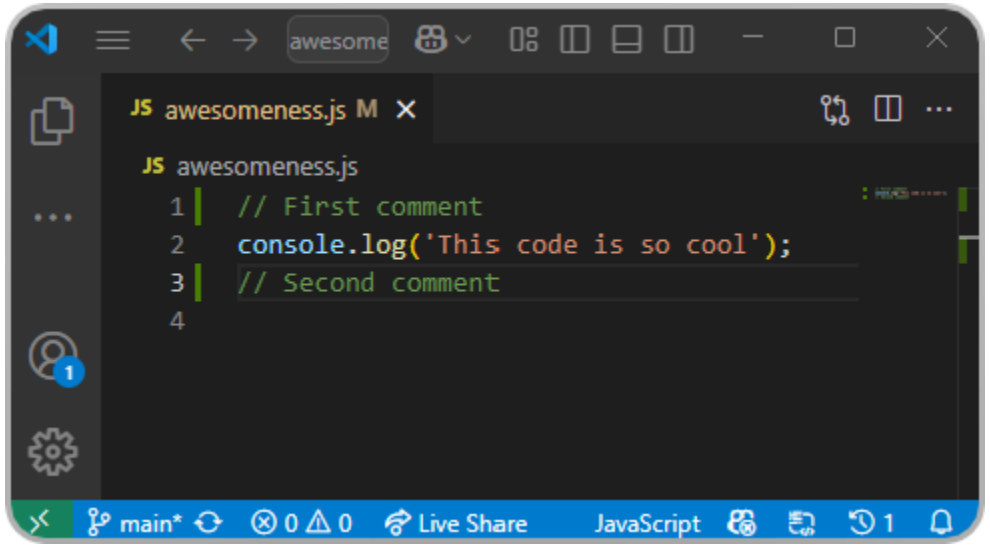


© Copyright IBM Corporation 2024

Scenario 2: No line overlap

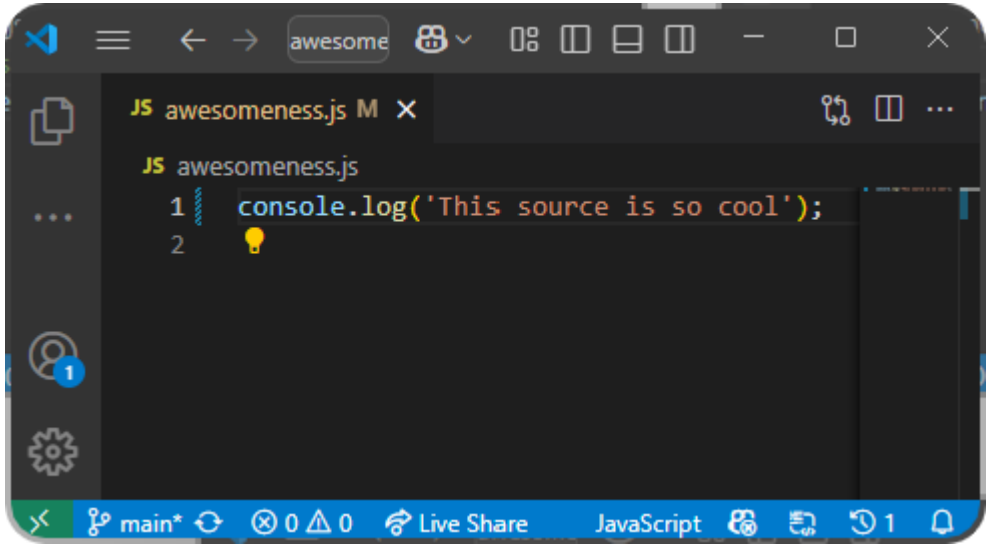


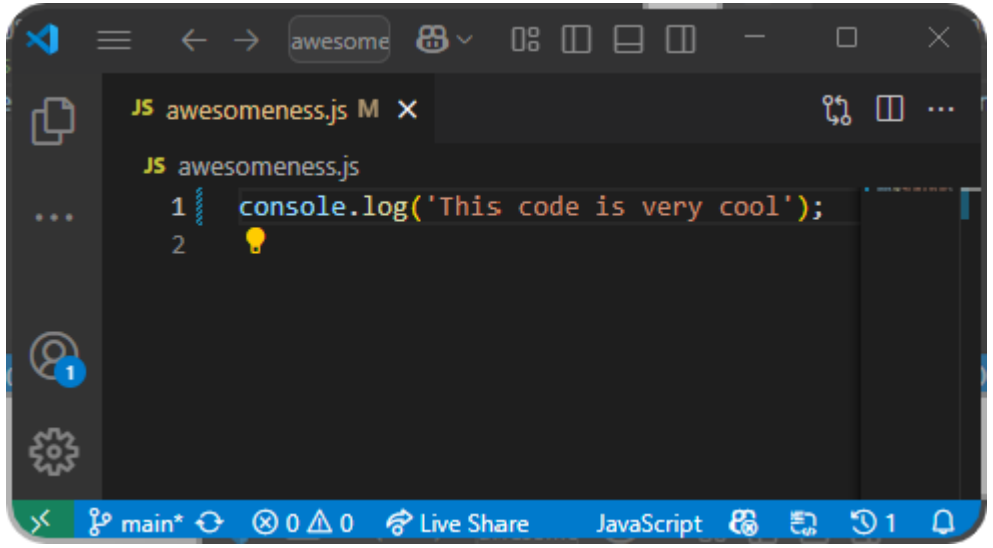


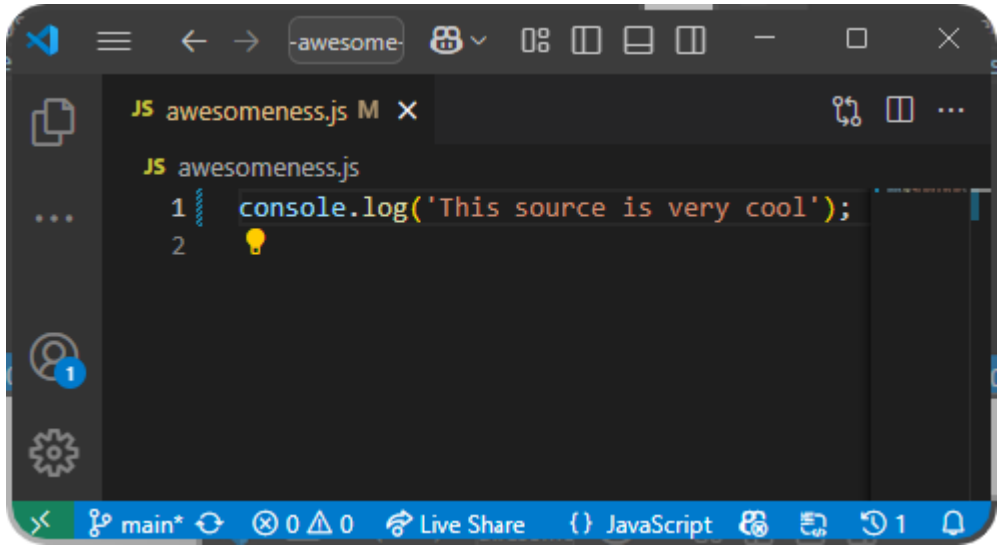


© Copyright IBM Corporation 2024

Scenario 3: Line overlap

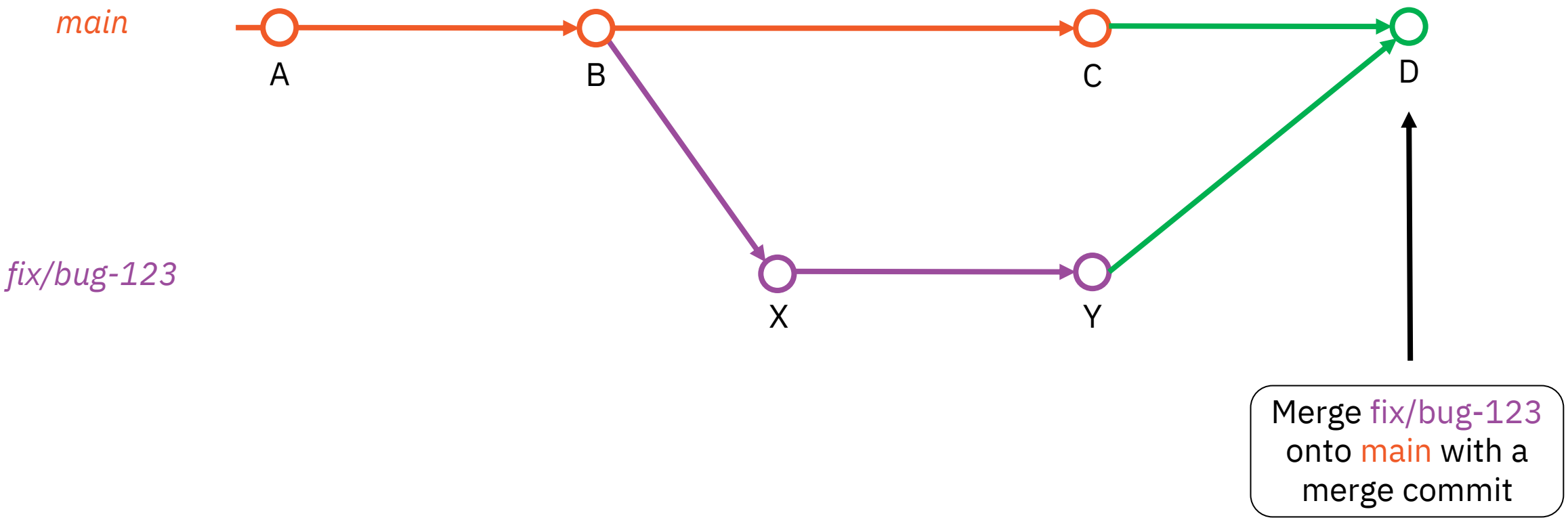






© Copyright IBM Corporation 2024

Merge commit

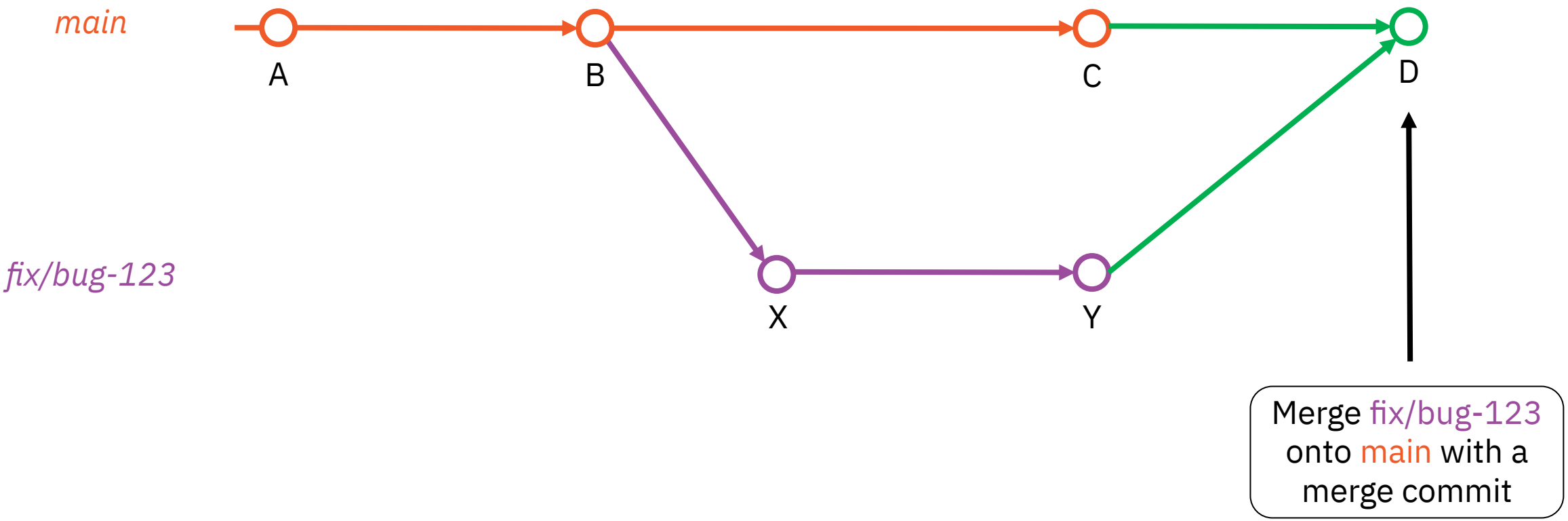


© Copyright IBM Corporation 2024

Merge Commits vs. Rebase Merging vs. Squash Merging

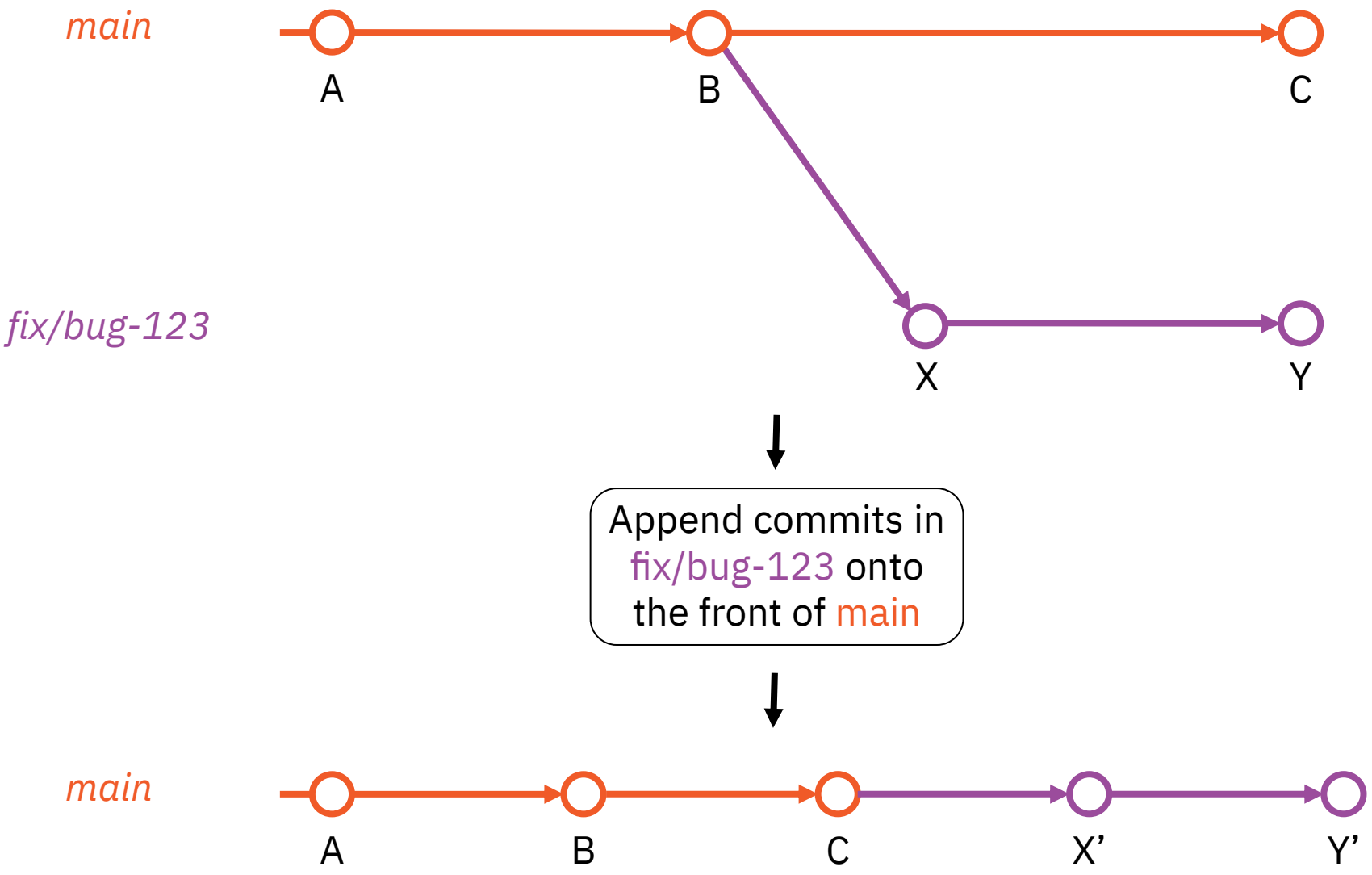
© Copyright IBM Corporation 2024

Merge commits



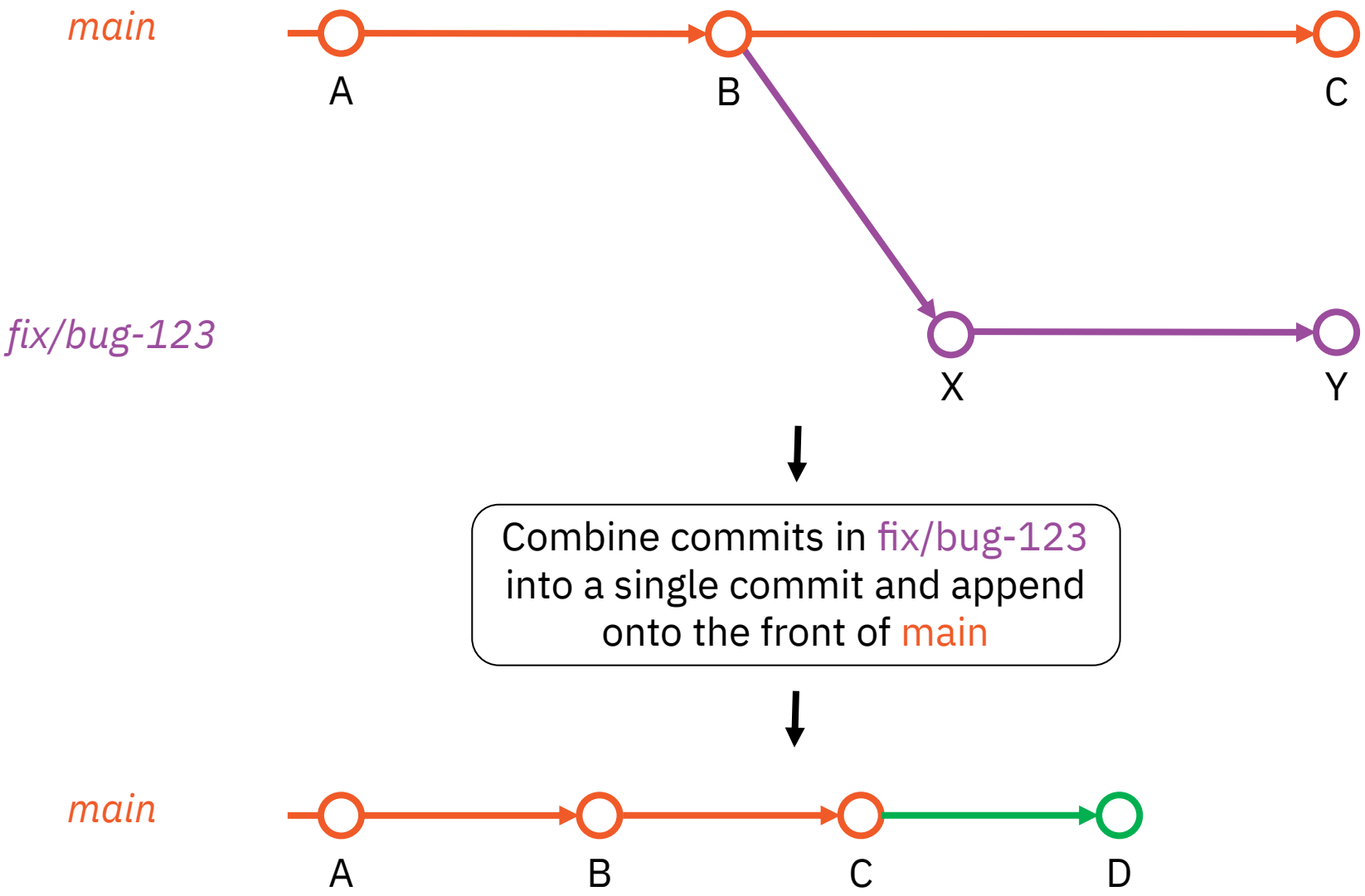
© Copyright IBM Corporation 2024

Rebase merging



© Copyright IBM Corporation 2024

Squash merging



© Copyright IBM Corporation 2024

Pull request settings

Pull Requests

When merging pull requests, you can allow any combination of merge commits, squashing, or rebasing. At least one option must be enabled. If you have linear history requirement enabled on any protected branch, you must enable squashing or rebasing.

- [x] **Allow merge commits**
  Add all commits from the head branch to the base branch with a merge commit.

  Default commit message
  Presented when merging a pull request with merge.

  Default message

- [x] **Allow squash merging**
  Combine all commits from the head branch into a single commit in the base branch.

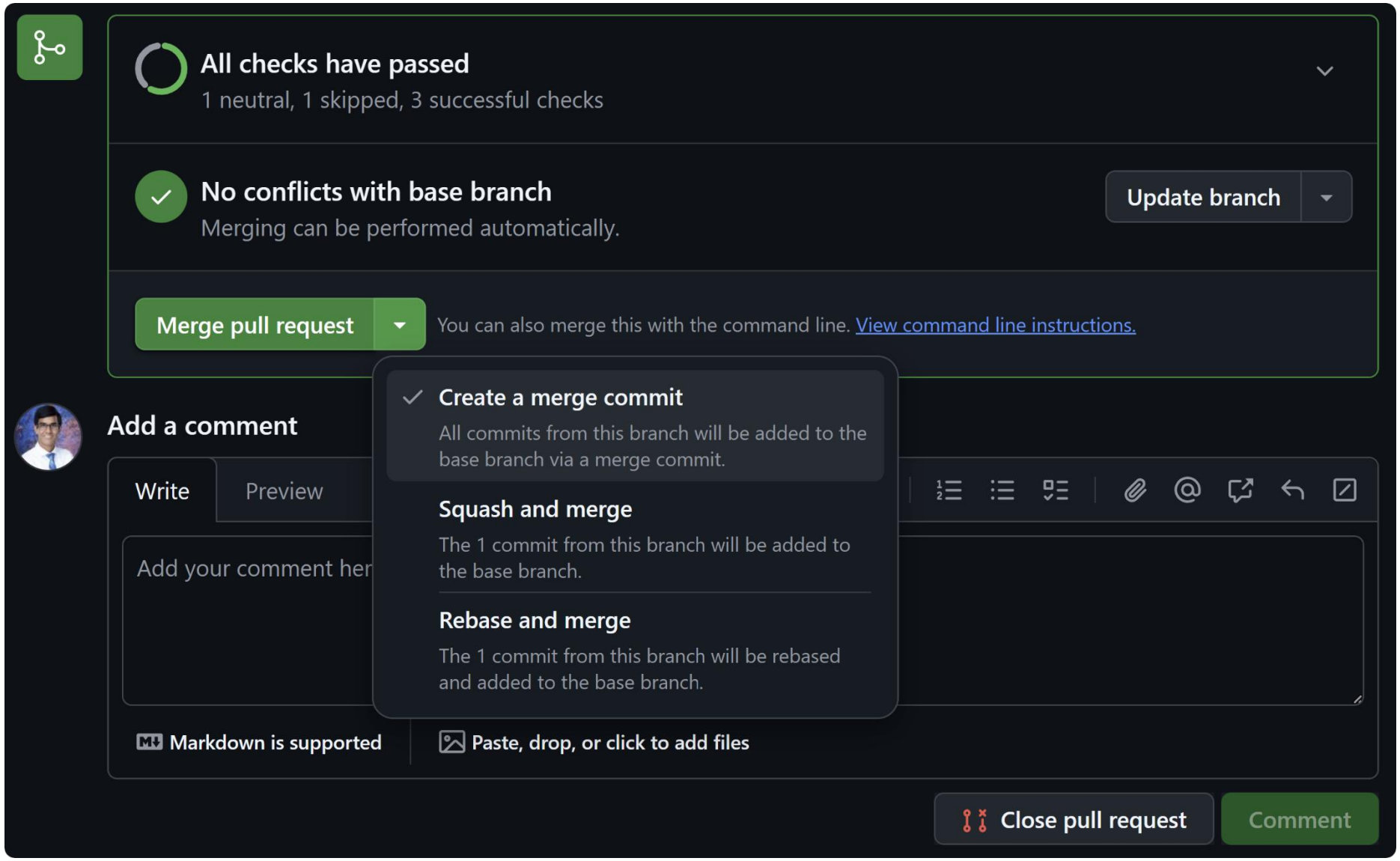
  Default commit message
  Presented when merging a pull request with squash.

  Default message

- [x] **Allow rebase merging**
  Add all commits from the head branch onto the base branch individually.

© Copyright IBM Corporation 2024

Pull request merge options

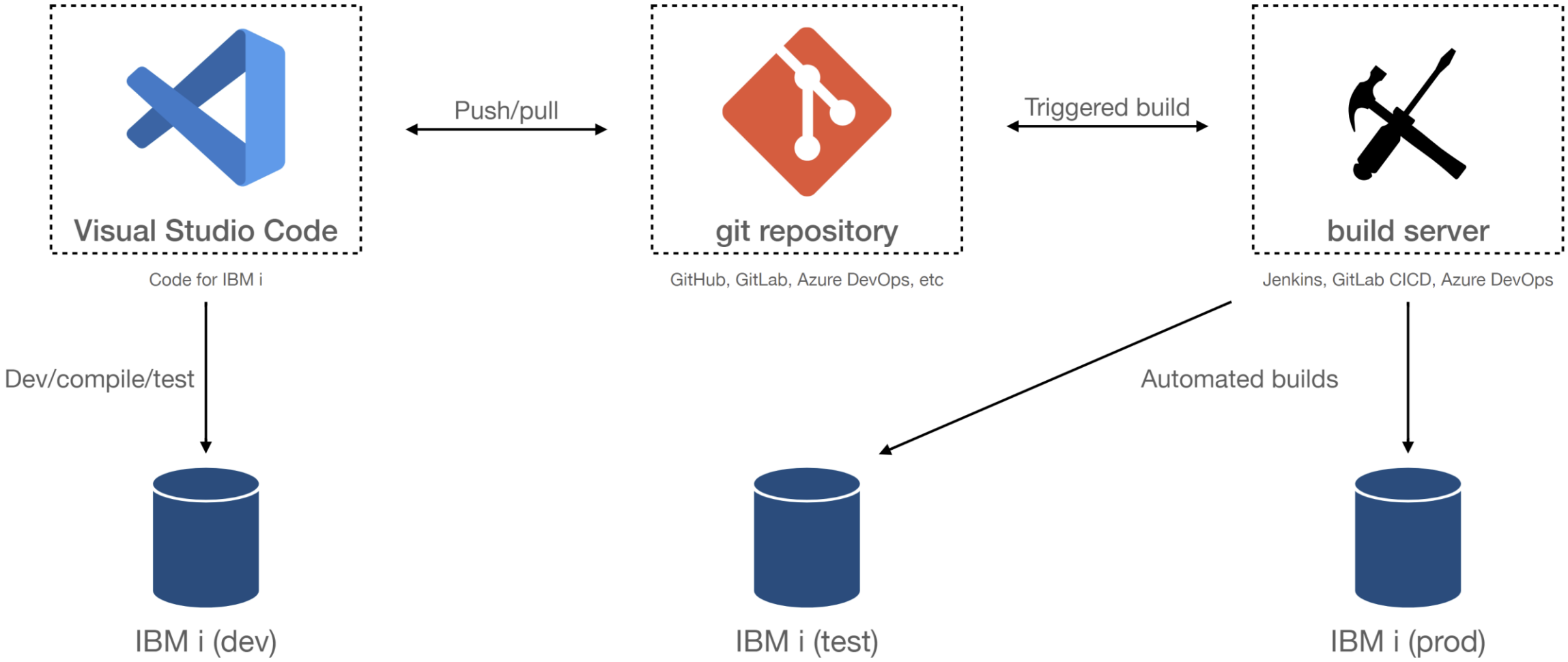


© Copyright IBM Corporation 2024

Using Git on IBM i

© Copyright IBM Corporation 2024

What Does Modern Development Look Like?



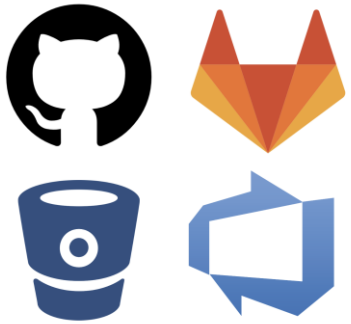
© Copyright IBM Corporation 2024

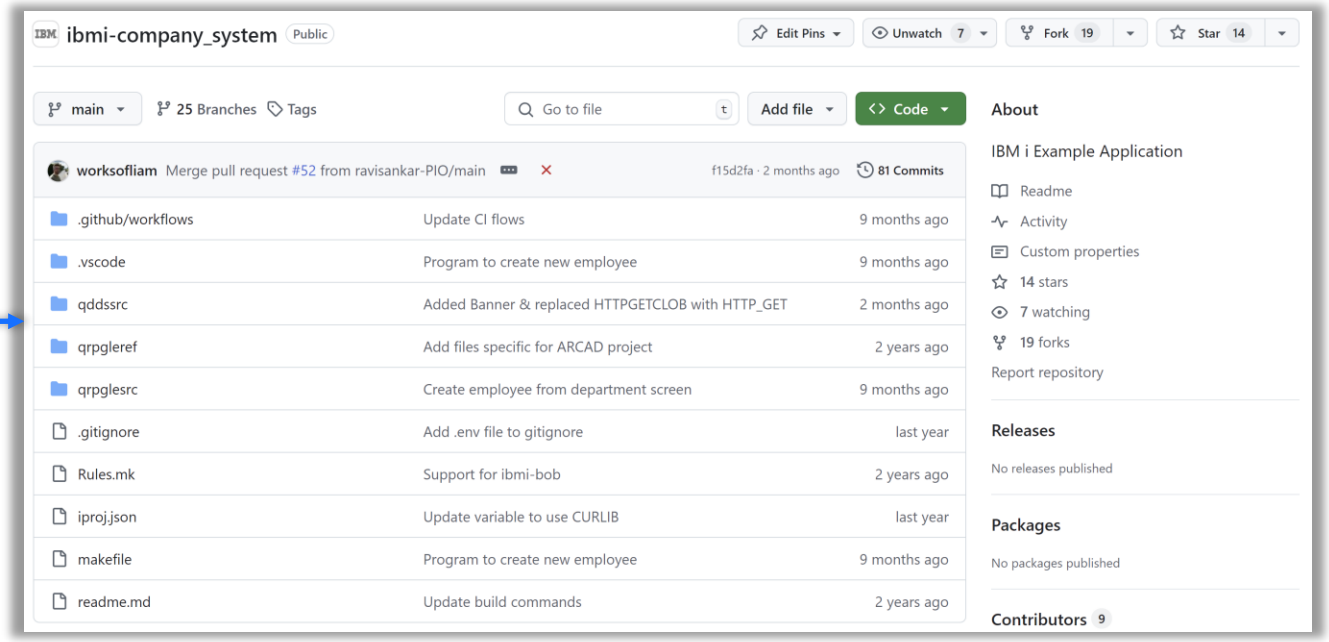
Development with Git

- Source is managed by Git
  - Complete change history
  - Branching and merging capabilities
  - Traceability

- Variety of options for Git hosting services
  - GitHub, GitLab, Azure DevOps, etc

GitHub, GitLab, Azure DevOps, etc





© Copyright IBM Corporation 2024

Local Development Experience

- IBM i
  - IBM i integration via open-source extensions
  - Support for RPGLE, COBOL, CL, SQL, and more!

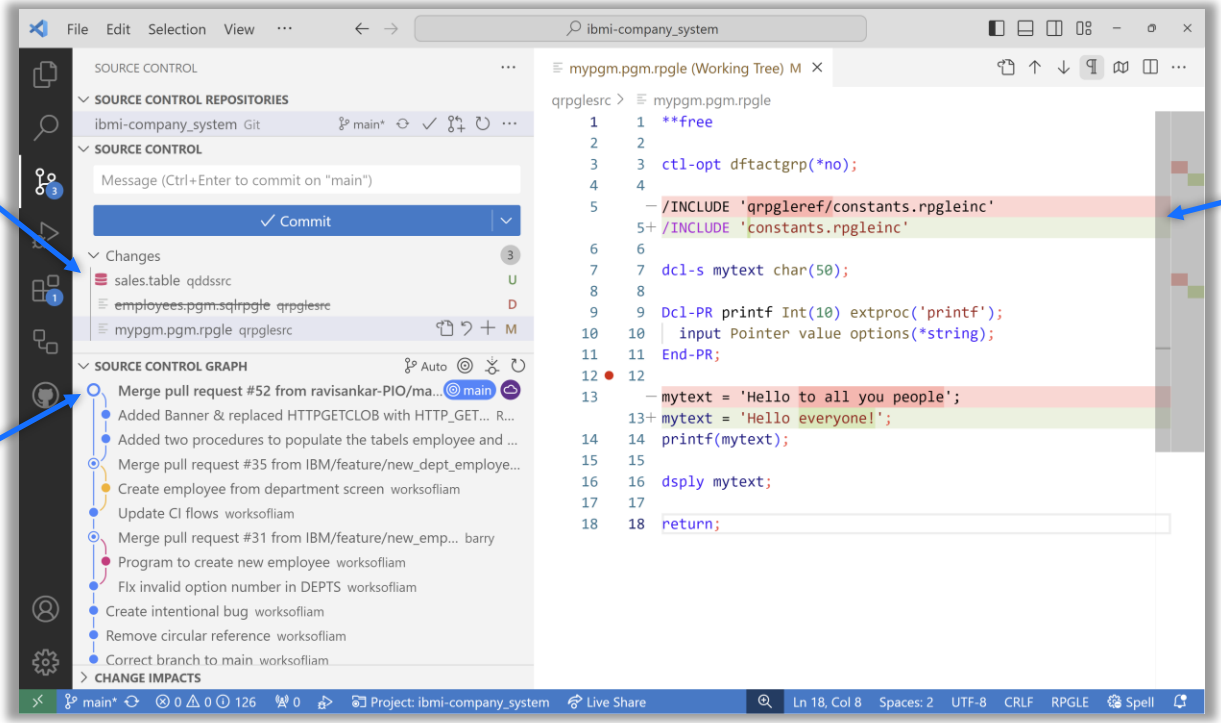
- Git
  - First class Git support
  - Hundreds of Git tools

Visual Studio Code

Code for IBM i

View current repository changes

View a graphical representation of incoming and outgoing commits



View changes line by line

© Copyright IBM Corporation 2024

Let's use a different (but similar) file system

MYPROJECT
- QRPGLESRC
  - CUSTOMER.RPGLE
  - INVENTORY.RPGLE
  - STARTJOB.RPGLE
- QSQLSRC
  - CUSTOMERS.SQL
  - INVENTORY.SQL
- QCLLESRC
  - START.CLLE
- QCMDSRC
  - STARTJOB.CMD

QSYS.LIB Library

No more character name restrictions

Now usable with Git and Make

/my-project
- /.git
- qrpglesrc
  - customer.rpgle
  - inventory.rpgle
  - startjob.rpgle
- qsqlsrc
  - customer.table
  - inventory.table
- qcllesrc
  - start.clle
- qcmdsrc
  - startjob.cmd

IFS/Local File System

© Copyright IBM Corporation 2024

Let’s go one step further!

/my-project
- /.git
- qrpglesrc
  - customer.rpgle
  - inventory.rpgle
  - startjob.rpgle
- qsqlsrc
  - customer.table
  - inventory.table
- qcllesrc
  - start.clle
- qcmdsrc
  - startjob.cmd

IFS/Local File System

Flexible directory structure

/my-project
- /.git
- customer-management
  - customer.rpgle
  - customer.table
- inventory-service
  - inventory.rpgle
  - inventory.table
- start-command
  - startjob.pgm.rpgle
  - start.clle
  - startjob.cmd

Logical Organization

© Copyright IBM Corporation 2024

How to migrate source from QSYS?

CVTSRCPF from BOB

QSYS members in source physical files

↓

Properly encoded, terminated, and named source files in an IFS directory

↓

Download to local project

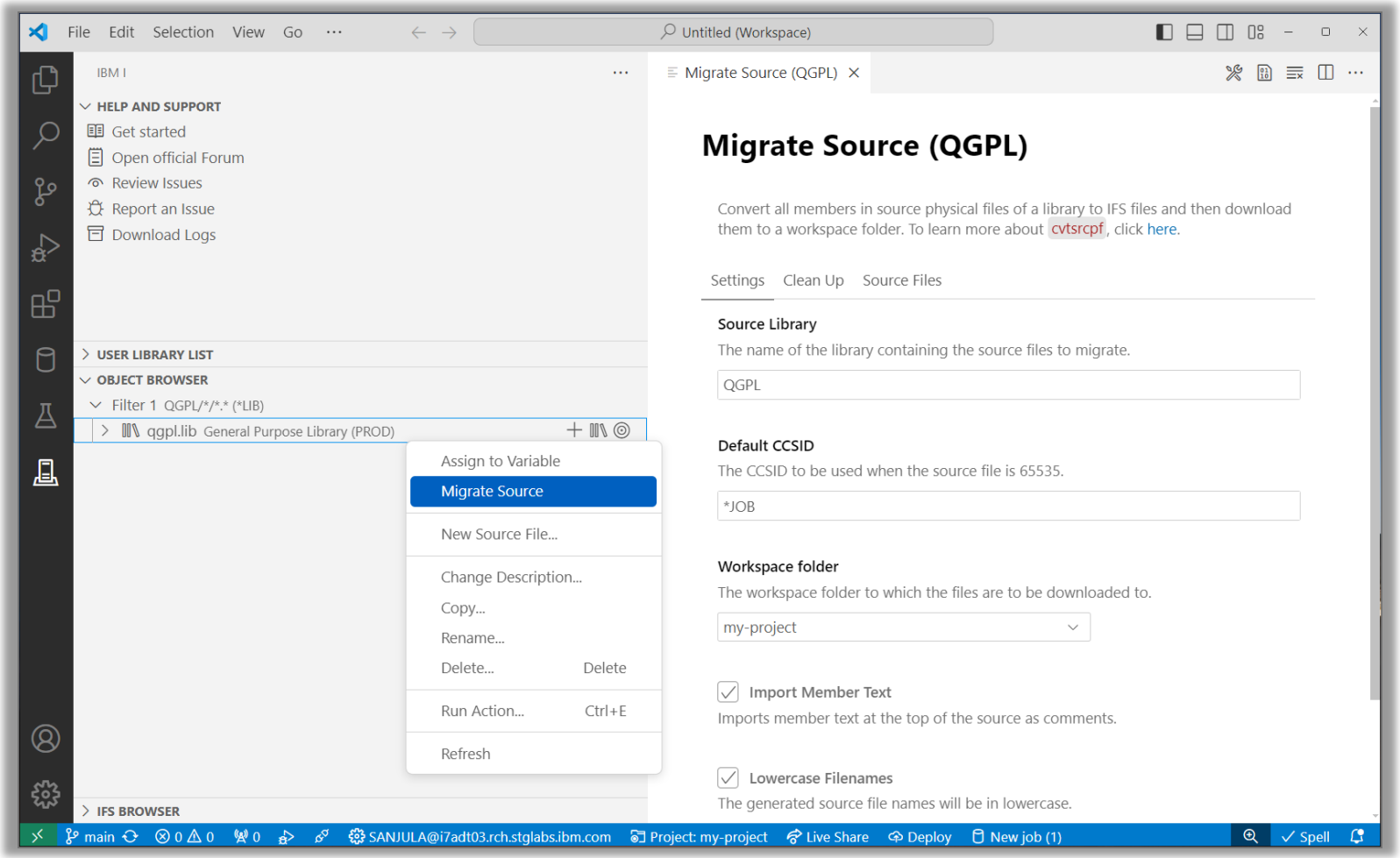
↓

Source Orbit

Rename extensions

↓

Convert includes/copy directives to Unix style paths



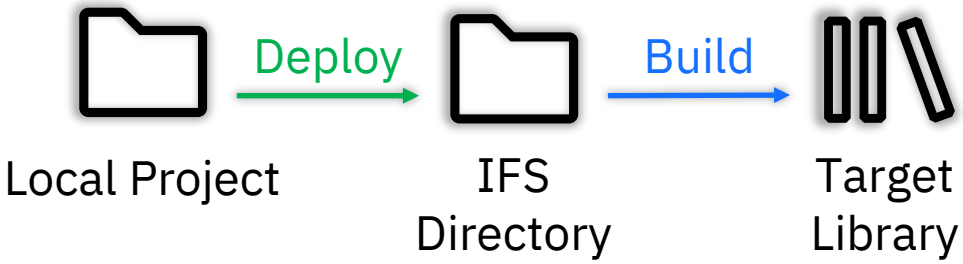
© Copyright IBM Corporation 2024

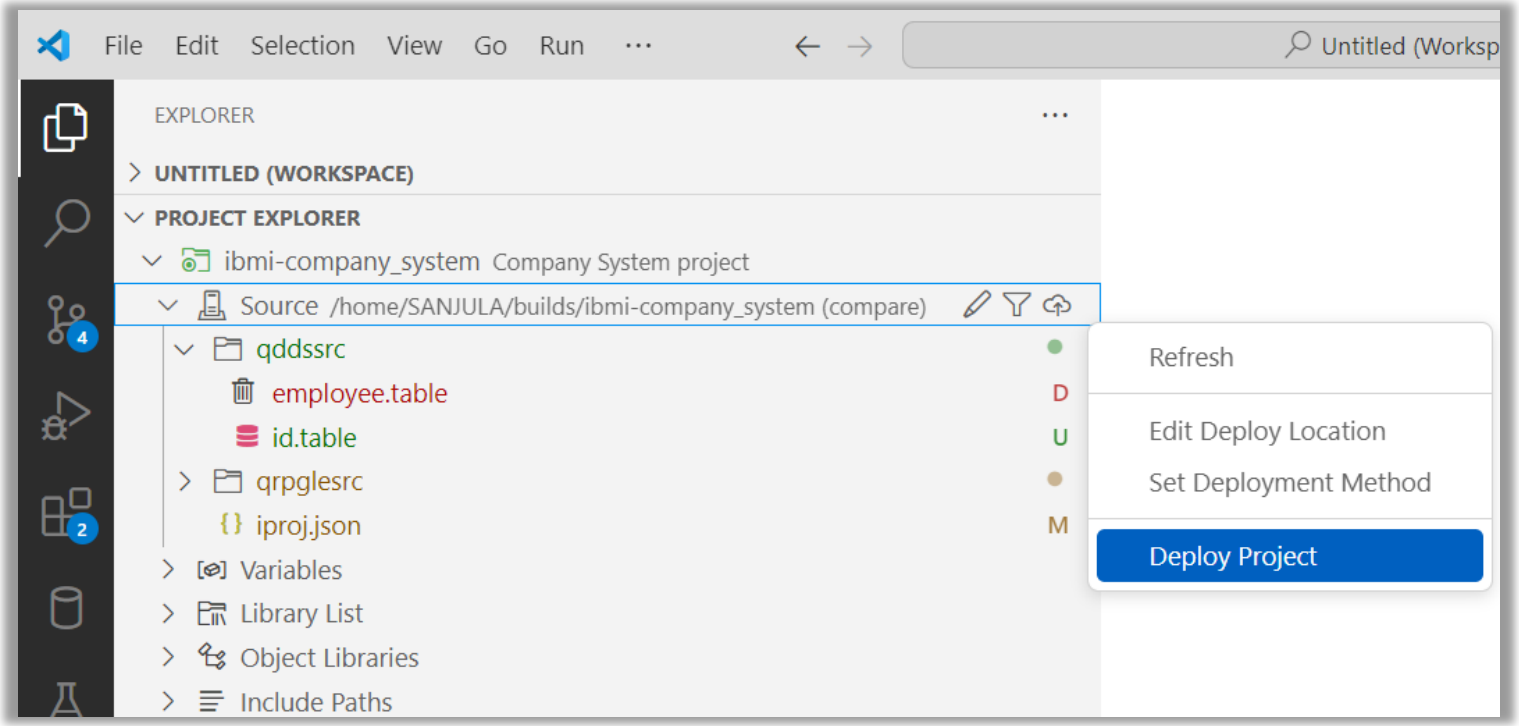
How does source get built from local files?

- Set deploy location
  - Where source gets uploaded to
  - Typically set one
  - Each developer gets a unique location
  - Each repository gets a unique location

- Set deployment method
  - Compare (typically the safest)
  - Changes (typically the fastest)
  - Working Changes
  - Staged Changes
  - All

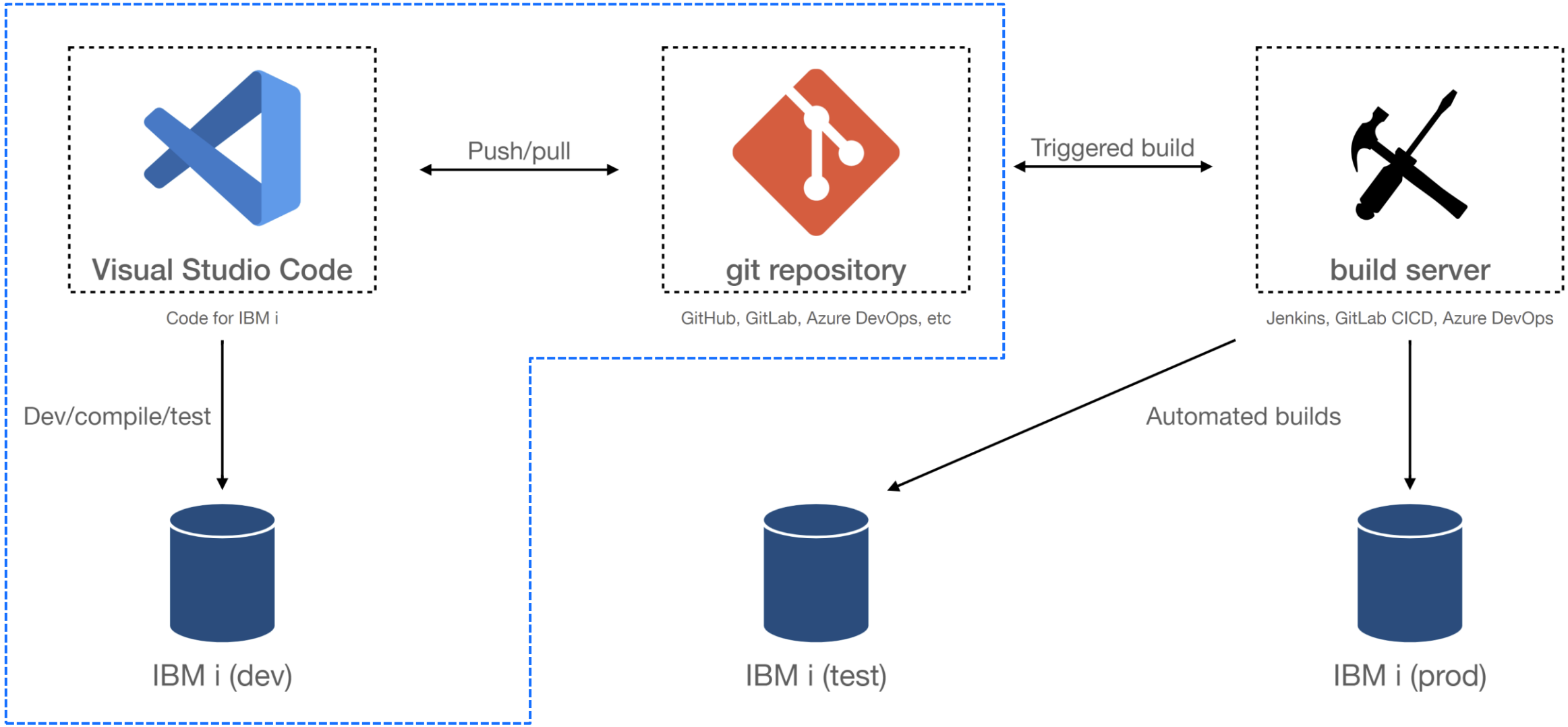
- Deploy project
  - Moves files to deploy location based on deployment method





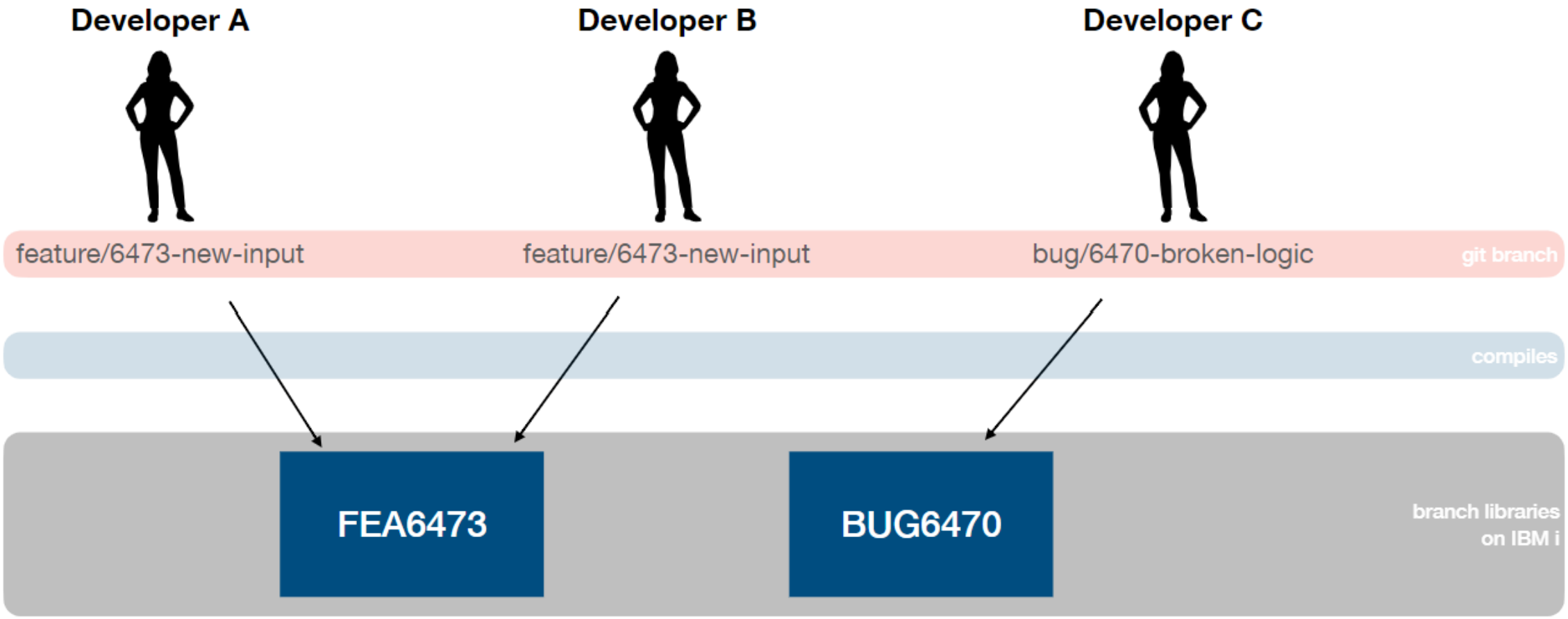
© Copyright IBM Corporation 2024

How to work efficiently with these 3 components?



© Copyright IBM Corporation 2024

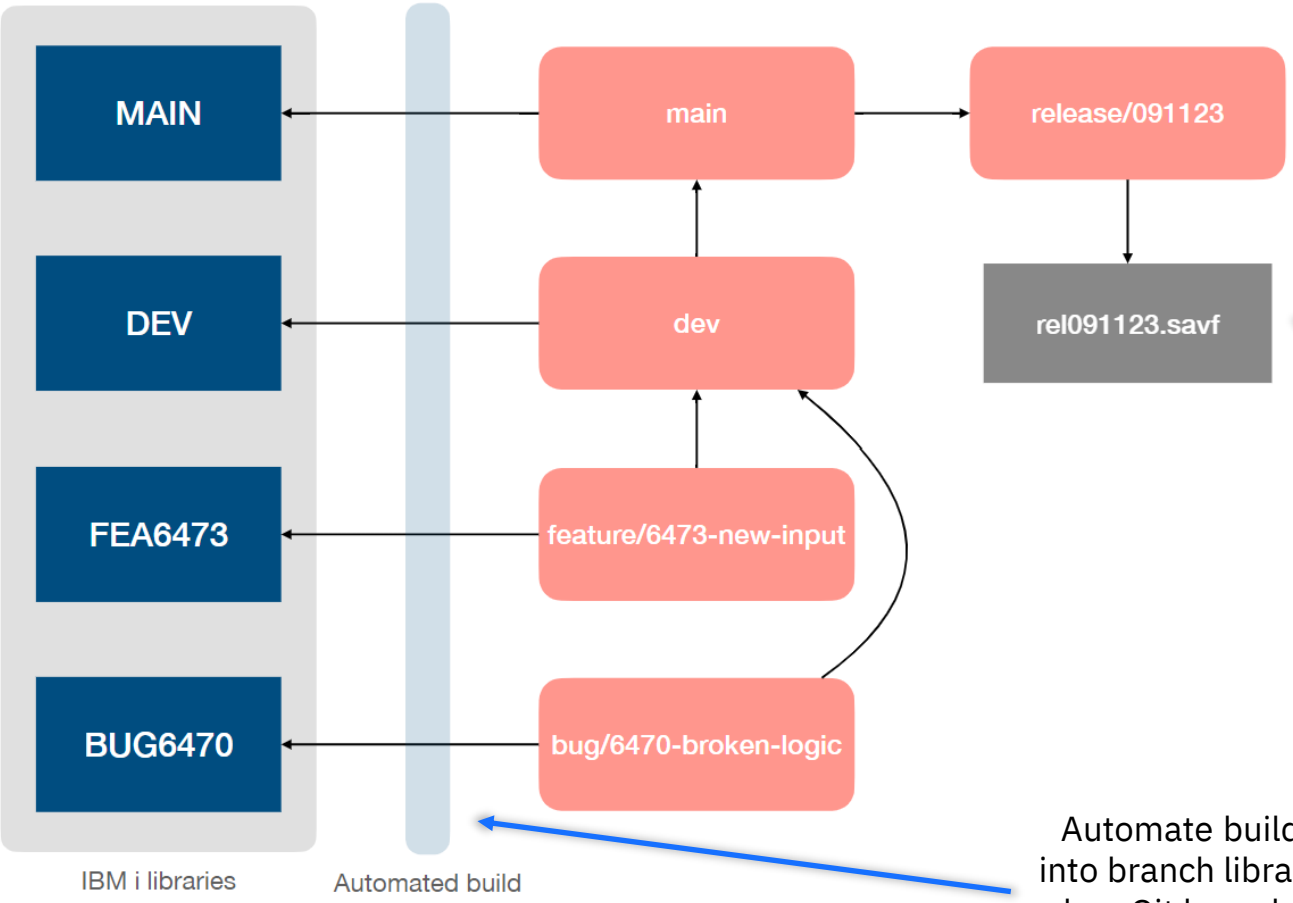
Map Git branches to libraries on IBM i



© Copyright IBM Corporation 2024

Build Server

Let's go one step further and automate this process



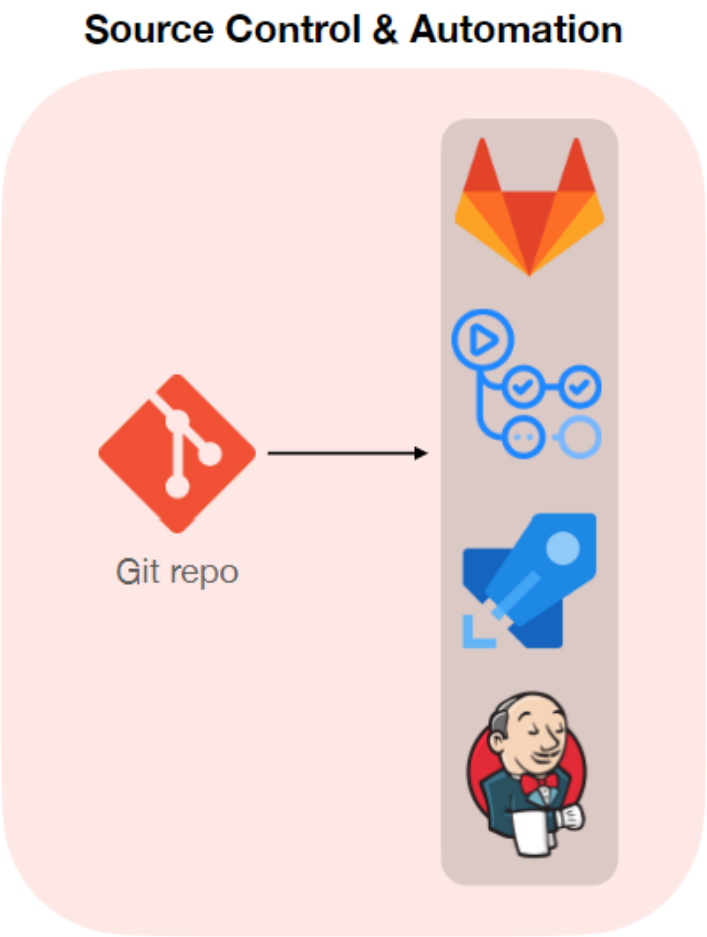
Jenkins, GitLab CICD, Azure DevOps

Generate save files when releases are created

Automate build into branch library when Git branch is updated

© Copyright IBM Corporation 2024

How to create workflows?

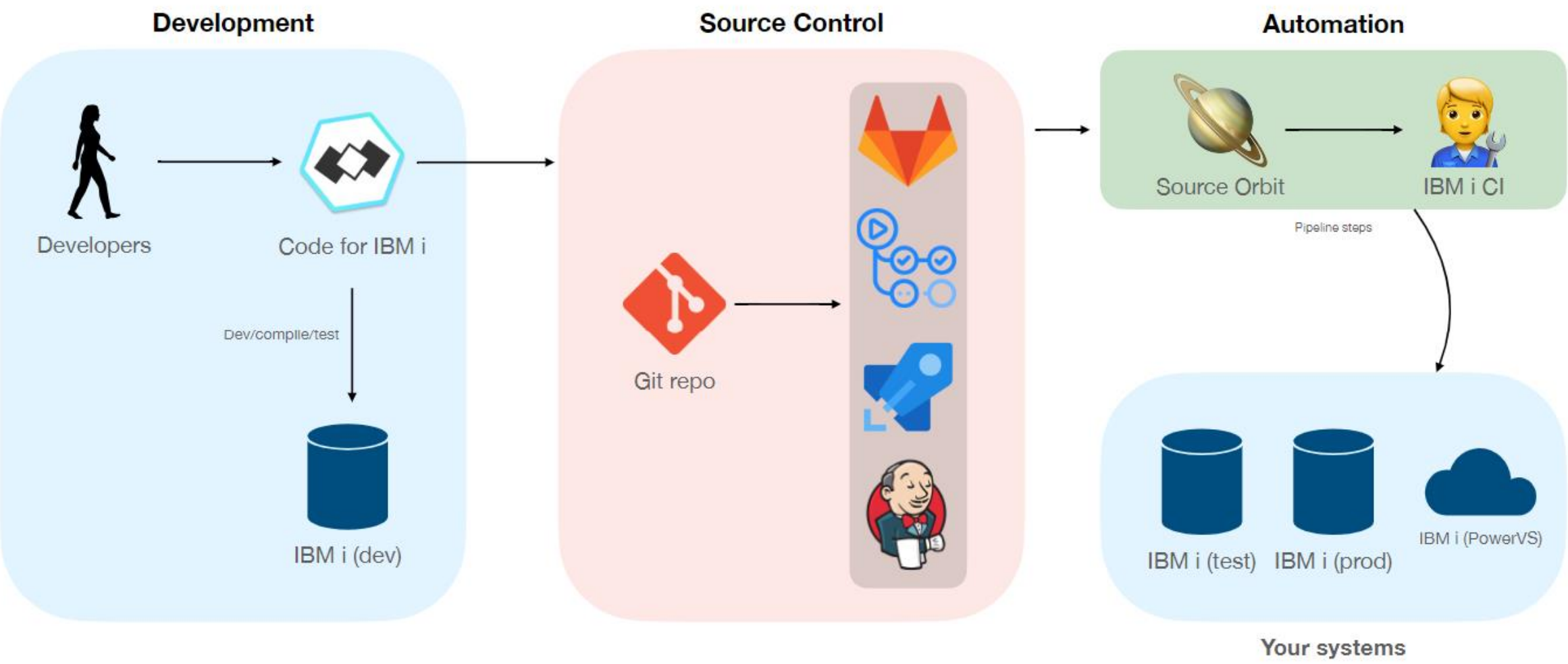


GitLab	`.gitlab-ci.yml`
Azure DevOps	`azure-pipeline.yml`
GitHub	`anything.yml`
Jenkins	`Jenkinsfile`

yml can also be yaml

© Copyright IBM Corporation 2024

How to achieve automation with IBM i?



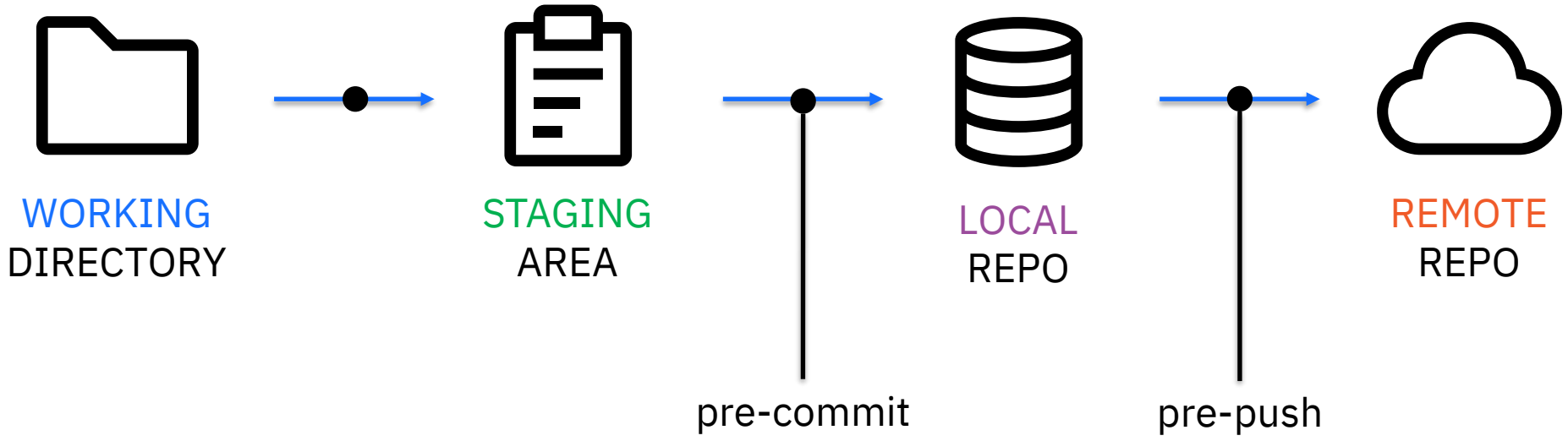
© Copyright IBM Corporation 2024

Tips and Tricks

© Copyright IBM Corporation 2024

Git hooks

https://git-scm.com/book/en/v2/Customizing-Git-Git-Hooks



© Copyright IBM Corporation 2024

Pre-commit to run RPG linter

File Edit Selection View ··· ibmi-company_system

EXPLORER

OPEN EDITORS

IBMI-COMPANY_SYSTEM

- .evfevent
- .git
  - hooks
    - applypatch-msg.sample
    - commit-msg.sample
    - fsmonitor-watchman.sample
    - post-update.sample
    - pre-applypatch.sample
    - pre-commit
    - pre-merge-commit.sample
    - pre-push.sample
    - pre-rebase.sample
    - pre-receive.sample
    - prepare-commit-msg.sample
    - push-to-checkout.sample
    - sendemail-validate.sample
    - update.sample

PROJECT EXPLORER

$ pre-commit

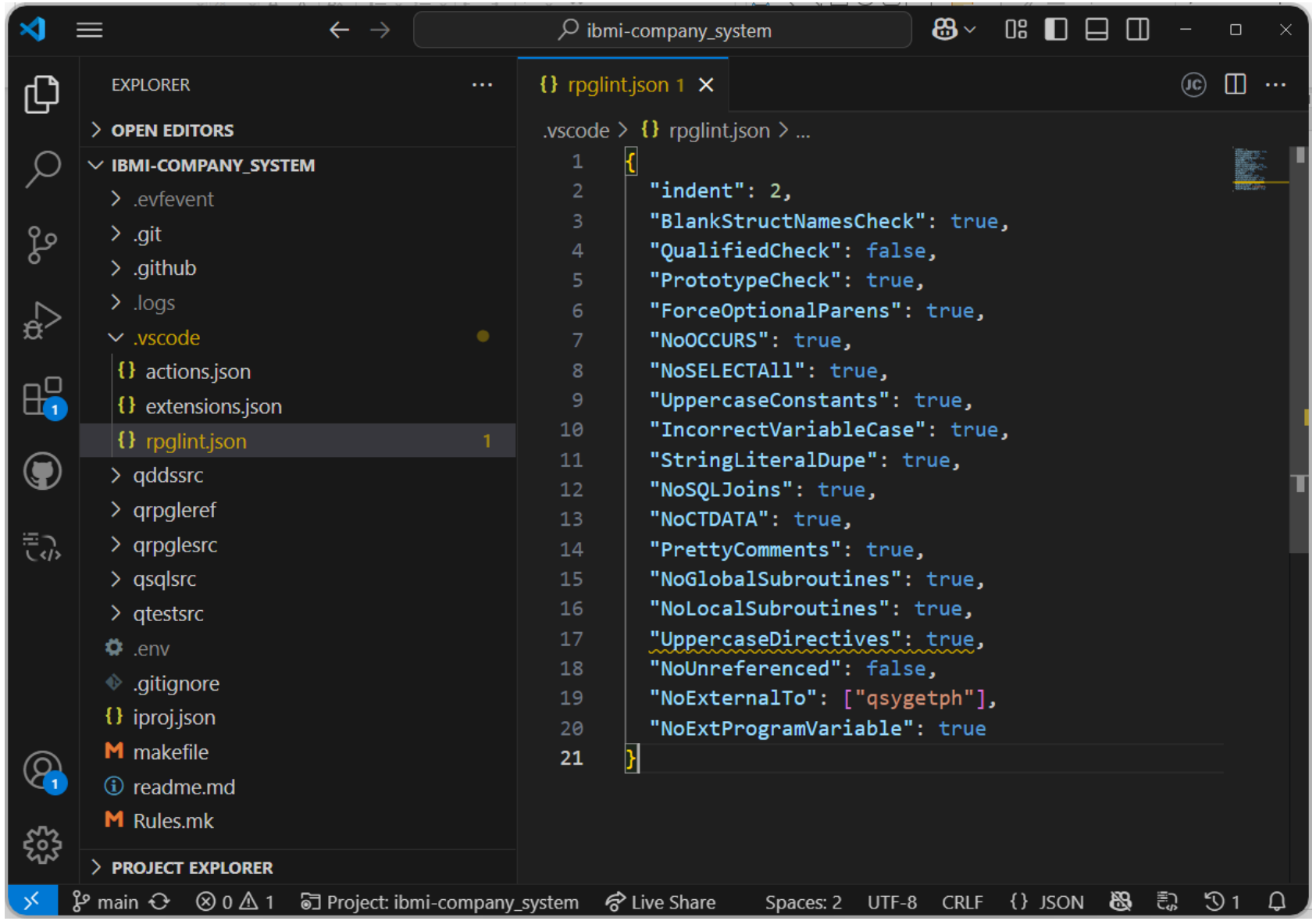
.git > hooks > $ pre-commit

```
#!/bin/sh

if command -v rpglint >/dev/null 2>&1; then
    # Run the linter
    echo "✍️ Running rpglint..."
    rpglint
else
    # Warn the user, but don't block the commit
    echo "⚠️  Skiping lint as 'rpglint' is not installed."
    echo "👉 You can install it with: npm install -g @halcyontech/rpglint"
fi
```

main+ 0 5 Project: ibmi-company_system Live Share Ln 6, Col 12 Spaces: 4 UTF-8 LF {} Shell Script 1

© Copyright IBM Corporation 2024

Setup lint config



© Copyright IBM Corporation 2024

Git hooks

ibmi-company_system

empdet.sqlrpgle 5, M

qrpglesrc > empdet.sqlrpgle > getEmployeeDetail

```
7   dcl-proc getEmployeeDetail export;
24        empno = :empno;
25
26        if (sqlcode = 0);
27          employee_detail.found = *on;
28        else;
29          employee_detail.found = *off;
30        endif;
31
```

PROBLEMS 5 COMMENTS DEBUG CONSOLE TERMINAL ··· bash

```
AzureAD+SanjulaGanepola@IBM-PW0AC6C8 MINGW64 ~/Git/ibmi-company_system (main)
$ git commit -s -m "Add error handling"
✍Running rpglint...
Linting 6 files.

qrpglesrc\empdet.sqlrpgle: 5 errors.
        Line 26: expected indent of 2, got 6
        Line 27: expected indent of 4, got 8
        Line 28: expected indent of 2, got 6
        Line 29: expected indent of 4, got 8
        Line 30: expected indent of 2, got 6
```

main+ ⊗ 0 ⚠ 5 Project: ibmi-company_system Live Share {} RPGLE 1

© Copyright IBM Corporation 2024

Git blame

MINGW64:/c/Users/SanjulaGanepola/Git/ibmi-company_system

```
AzureAD+SanjulaGanepola@IBM-PW0AC6C8 MINGW64 ~/Git/ibmi-company_system (main)
$ git blame qtestsrc/empdet.test.sqlrpgle
c92a0dd5 qtestsrc/empdet.test.sqlrpgle  (Sanjula Ganepola 2025-05-08 22:00:55 -0400  1) **free
c92a0dd5 qtestsrc/empdet.test.sqlrpgle  (Sanjula Ganepola 2025-05-08 22:00:55 -0400  2)
5f80e7e8 qtestsrc/empdett.test.sqlrpgle (Sanjula Ganepola 2025-05-12 19:43:01 -0400  3) ctl-opt nomain BNDDIR('APP');
c92a0dd5 qtestsrc/empdet.test.sqlrpgle  (Sanjula Ganepola 2025-05-08 22:00:55 -0400  4)
c92a0dd5 qtestsrc/empdet.test.sqlrpgle  (Sanjula Ganepola 2025-05-08 22:00:55 -0400  5) /include 'qrpgleref/empdet.rpgleinc'
c92a0dd5 qtestsrc/empdet.test.sqlrpgle  (Sanjula Ganepola 2025-05-08 22:00:55 -0400  6) /include qinclude,TESTCASE
c92a0dd5 qtestsrc/empdet.test.sqlrpgle  (Sanjula Ganepola 2025-05-08 22:00:55 -0400  7)
1d7fcfc3 qtestsrc/empdet.test.sqlrpgle  (Sanjula Ganepola 2025-05-09 20:44:06 -0400  8) exec sql
1d7fcfc3 qtestsrc/empdet.test.sqlrpgle  (Sanjula Ganepola 2025-05-09 20:44:06 -0400  9)   set option commit = *none;
1d7fcfc3 qtestsrc/empdet.test.sqlrpgle  (Sanjula Ganepola 2025-05-09 20:44:06 -0400 10)
c92a0dd5 qtestsrc/empdet.test.sqlrpgle  (Sanjula Ganepola 2025-05-08 22:00:55 -0400 11) dcl-proc setUpSuite export;
c92a0dd5 qtestsrc/empdet.test.sqlrpgle  (Sanjula Ganepola 2025-05-08 22:00:55 -0400 12)   // Insert sample data into employe
e
c92a0dd5 qtestsrc/empdet.test.sqlrpgle  (Sanjula Ganepola 2025-05-08 22:00:55 -0400 13)   exec sql
c92a0dd5 qtestsrc/empdet.test.sqlrpgle  (Sanjula Ganepola 2025-05-08 22:00:55 -0400 14)     insert into employee (
c92a0dd5 qtestsrc/empdet.test.sqlrpgle  (Sanjula Ganepola 2025-05-08 22:00:55 -0400 15)       empno, firstnme, midinit, last
name, workdept, phoneno,
c92a0dd5 qtestsrc/empdet.test.sqlrpgle  (Sanjula Ganepola 2025-05-08 22:00:55 -0400 16)       hiredate, job, edlevel, sex, b
irthdate, salary, bonus, comm
c92a0dd5 qtestsrc/empdet.test.sqlrpgle  (Sanjula Ganepola 2025-05-08 22:00:55 -0400 17)     ) values
c92a0dd5 qtestsrc/empdet.test.sqlrpgle  (Sanjula Ganepola 2025-05-08 22:00:55 -0400 18)       ('000010', 'CHRISTINE', 'I', '
HAAS', 'A00', '3978', '01/01/65', 'PRES', 18, 'F', null, 52750, 1000, 4220),
```


© Copyright IBM Corporation 2024

Stashing changes temporarily

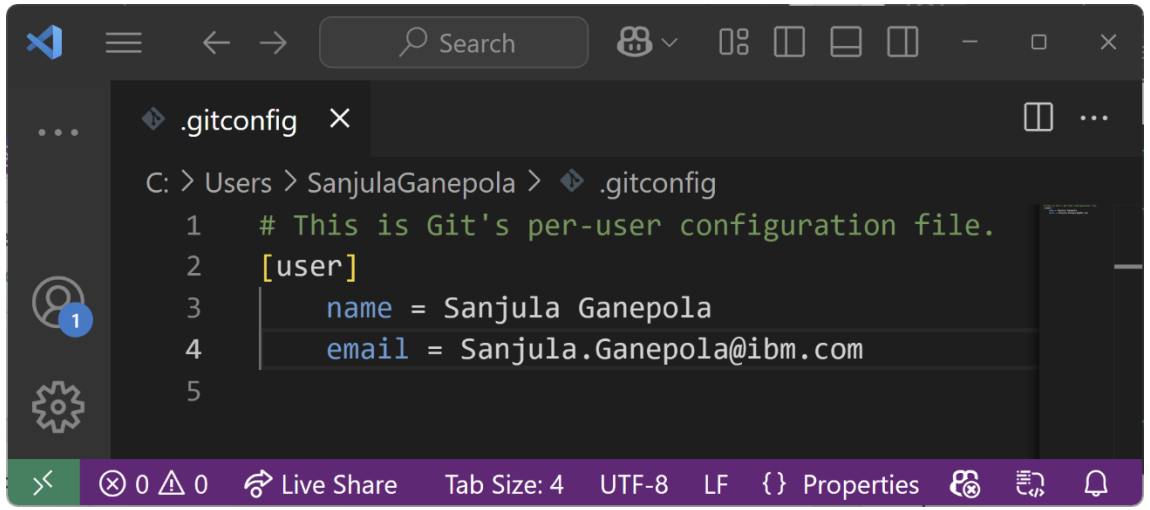
MINGW64:/c/Users/SanjulaGanepola/Desktop/my-awesome-project

```
AzureAD+SanjulaGanepola@IBM-PW0AC6C8 MINGW64 ~/Desktop/my-awesome-project (main)
$ echo "console.log('My second log')" >> awesomeness.js

AzureAD+SanjulaGanepola@IBM-PW0AC6C8 MINGW64 ~/Desktop/my-awesome-project (main)
$ git stash
warning: in the working copy of 'awesomeness.js', LF will be replaced by CRLF the next time Git touches i
Saved working directory and index state WIP on main: e7809ee Add console log

AzureAD+SanjulaGanepola@IBM-PW0AC6C8 MINGW64 ~/Desktop/my-awesome-project (main)
$ git stash list
stash@{0}: WIP on main: e7809ee Add console log

AzureAD+SanjulaGanepola@IBM-PW0AC6C8 MINGW64 ~/Desktop/my-awesome-project (main)
$ git stash pop
On branch main
Your branch is up to date with 'origin/main'.

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:   awesomeness.js

no changes added to commit (use "git add" and/or "git commit -a")
Dropped refs/stash@{0} (bb9499bf426fe6d52133190d089c741e4820a17a)

AzureAD+SanjulaGanepola@IBM-PW0AC6C8 MINGW64 ~/Desktop/my-awesome-project (main)
$
```


© Copyright IBM Corporation 2024

Git Configs

- Scopes of configuration
  - System: System-wide configuration
    - `C:\Program Files\Git\etc\gitconfig`
    - `/etc/gitconfig`
  - Global: User specific configuration
    - `C:\Users\<user>\.gitconfig`
    - `/home/<user>/.gitconfig`
  - Local: Repository specific configuration
    - `<git_repo>/.git/config`
- Git Config Docs: https://git-scm.com/docs/git-config



© Copyright IBM Corporation 2024

Without auto setup remote branch config

MINGW64:/c/Users/SanjulaGanepola/Desktop/developer2/my-awesome-project

```
AzureAD+SanjulaGanepola@IBM-PW0AC6C8 MINGW64 ~/Desktop/developer2/my-awesome-project (main)
$ git checkout -b feature/ui-buttons
Switched to a new branch 'feature/ui-buttons'

AzureAD+SanjulaGanepola@IBM-PW0AC6C8 MINGW64 ~/Desktop/developer2/my-awesome-project (feature/ui-buttons)
$ git push
fatal: The current branch feature/ui-buttons has no upstream branch.
To push the current branch and set the remote as upstream, use

    git push --set-upstream origin feature/ui-buttons

To have this happen automatically for branches without a tracking
upstream, see 'push.autoSetupRemote' in 'git help config'.


AzureAD+SanjulaGanepola@IBM-PW0AC6C8 MINGW64 ~/Desktop/developer2/my-awesome-project (feature/ui-buttons)
$ git push --set-upstream origin feature/ui-buttons
Total 0 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
remote:
remote: Create a pull request for 'feature/ui-buttons' on GitHub by visiting:
remote:      https://github.com/SanjulaGanepola/my-awesome-project/pull/new/feature/ui-buttons
remote:
To https://github.com/SanjulaGanepola/my-awesome-project.git
 * [new branch]      feature/ui-buttons -> feature/ui-buttons
branch 'feature/ui-buttons' set up to track 'origin/feature/ui-buttons'.

AzureAD+SanjulaGanepola@IBM-PW0AC6C8 MINGW64 ~/Desktop/developer2/my-awesome-project (feature/ui-buttons)
$
```


© Copyright IBM Corporation 2024

With auto setup remote branch config

MINGW64:/c/Users/SanjulaGanepola/Desktop/developer2/my-awesome-project

```
AzureAD+SanjulaGanepola@IBM-PW0AC6C8 MINGW64 ~/Desktop/developer2/my-awesome-project (feature/ui-text-boxes)
$ git config --global --bool push.autoSetupRemote true

AzureAD+SanjulaGanepola@IBM-PW0AC6C8 MINGW64 ~/Desktop/developer2/my-awesome-project (feature/ui-text-boxes)
$ git push
Total 0 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
remote:
remote: Create a pull request for 'feature/ui-text-boxes' on GitHub by visiting:
remote:      https://github.com/SanjulaGanepola/my-awesome-project/pull/new/feature/ui-text-boxes
remote:
To https://github.com/SanjulaGanepola/my-awesome-project.git
 * [new branch]      feature/ui-text-boxes -> feature/ui-text-boxes
branch 'feature/ui-text-boxes' set up to track 'origin/feature/ui-text-boxes'.

AzureAD+SanjulaGanepola@IBM-PW0AC6C8 MINGW64 ~/Desktop/developer2/my-awesome-project (feature/ui-text-boxes)
$
```

.gitconfig

C: > Users > SanjulaGanepola > .gitconfig

```
# This is Git's per-user configuration file.
[user]
    name = Sanjula Ganepola
    email = Sanjula.Ganepola@ibm.com
[push]
    autoSetupRemote = true
```


© Copyright IBM Corporation 2024

Alias to undo commits

MINGW64:/c/Users/SanjulaGanepola/Desktop/developer2/my-awesome-project

```
AzureAD+SanjulaGanepola@IBM-PW0AC6C8 MINGW64 ~/Desktop/developer2/my-awesome-project (main)
$ git config --global alias.undo 'reset --soft HEAD^'

AzureAD+SanjulaGanepola@IBM-PW0AC6C8 MINGW64 ~/Desktop/developer2/my-awesome-project (main)
$ touch README.md

AzureAD+SanjulaGanepola@IBM-PW0AC6C8 MINGW64 ~/Desktop/developer2/my-awesome-project (main)
$ git add README.md

AzureAD+SanjulaGanepola@IBM-PW0AC6C8 MINGW64 ~/Desktop/developer2/my-awesome-project (main)
$ git commit -s -m "Add README"
[main ad11aed] Add README
 1 file changed, 0 insertions(+), 0 deletions(-)
 create mode 100644 README.md

AzureAD+SanjulaGanepola@IBM-PW0AC6C8 MINGW64 ~/Desktop/developer2/my-awesome-project (main)
$ git status
On branch main
Your branch is ahead of 'origin/main' by 1 commit.
  (use "git push" to publish your local commits)

nothing to commit, working tree clean

AzureAD+SanjulaGanepola@IBM-PW0AC6C8 MINGW64 ~/Desktop/developer2/my-awesome-project (main)
$ git undo

AzureAD+SanjulaGanepola@IBM-PW0AC6C8 MINGW64 ~/Desktop/developer2/my-awesome-project (main)
$ git status
On branch main
Your branch is up to date with 'origin/main'.

Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
        new file:   README.md


AzureAD+SanjulaGanepola@IBM-PW0AC6C8 MINGW64 ~/Desktop/developer2/my-awesome-project (main)
$
```

.gitconfig

C: > Users > SanjulaGanepola > .gitconfig

```
# This is Git's per-user configuration file.
[user]
    name = Sanjula Ganepola
    email = Sanjula.Ganepola@ibm.com
[push]
    autoSetupRemote = true
[alias]
    undo = reset --soft HEAD^
```


© Copyright IBM Corporation 2024

Alias to search all commits for a message

MINGW64:/c/Users/SanjulaGanepola/Desktop/developer2/my-awesome-project

```
AzureAD+SanjulaGanepola@IBM-PW0AC6C8 MINGW64 ~/Desktop/developer2/my-awesome-project (main)
$ git config --global alias.searchCommits '!f() { git log --all --grep="$1" --oneline; }; f'

AzureAD+SanjulaGanepola@IBM-PW0AC6C8 MINGW64 ~/Desktop/developer2/my-awesome-project (main)
$ git searchCommits "Add"
6bbc0e6 Add helper functions
e7809ee Add console log

AzureAD+SanjulaGanepola@IBM-PW0AC6C8 MINGW64 ~/Desktop/developer2/my-awesome-project (main)
$
```

.gitconfig

C: > Users > SanjulaGanepola > .gitconfig

```
# This is Git's per-user configuration file.
[user]
    name = Sanjula Ganepola
    email = Sanjula.Ganepola@ibm.com
[push]
    autoSetupRemote = true
[alias]
    undo = reset --soft HEAD^
    searchCommits = "!f() { git log --all --grep=\"$1\" --oneline; }; f"
```

Live Share | Ln 1, Col 1 | Tab Size: 4 | UTF-8 | LF | Properties

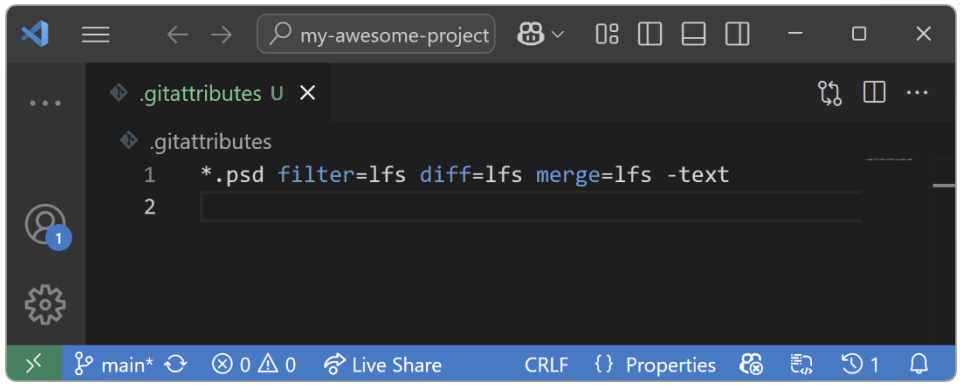
© Copyright IBM Corporation 2024

Git Large File Storage (LFS)

Open-source Git extension for versioning large files!
↪ Git LFS handles large files by storing references to the file in the repository, but not the actual file itself.

1. **Install**: Run `git lfs install` on your PC.

2. **Track File Types**: Run `git lfs track "*.psd"`



3. **Track .gitattributes:** `git add .gitattributes`

4. **Use Git as normal!**

Git LFS Docs: https://git-lfs.com/

© Copyright IBM Corporation 2024

Git bisect

ibmi-company_system

mypgm.pgm.rpgle

qrpglesrc > mypgm.pgm.rpgle > ...

```
11    End-PR;
12
13    mytext = 'Hello to all
14    printf(mytext);
15
16    dsply mytext;
17
18    return;
```

PROBLEMS 1 COMMENTS TERMINAL

```
4457ed2 (HEAD -> demo) Fix binding directory
abf352f Add error handling
c831812 This is the bug
9c38621 Update targets in Rules.mk
d571b18 Add info to README
a28d080 Update iproj.json
:
```

bash

git

demo 0 1 Project: ibmi-company_system Live Share {} RPGLE

© Copyright IBM Corporation 2024

Git bisect

ibmi-company_system

mypgm.pgm.rpgle

qrpglesrc > mypgm.pgm.rpgle > ...

```
12
13    mytext = 'Hello to all
14    printf(mytext);
15
```

PROBLEMS 1 COMMENTS DEBUG CONSOLE OUTPUT TERMINAL IBM I JOB LOG 1

bash
bash

```
4457ed2 (HEAD -> demo) Fix binding directory
abf352f Add error handling
c831812 This is the bug
9c38621 Update targets in Rules.mk
d571b18 Add info to README
a28d080 Update iproj.json

AzureAD+SanjulaGanepola@IBM-PW0AC6C8 MINGW64 ~/Git/ibmi-company_system (demo)
$ git bisect start
status: waiting for both good and bad commits

AzureAD+SanjulaGanepola@IBM-PW0AC6C8 MINGW64 ~/Git/ibmi-company_system (demoo
o|BISECTING)
$ git bisect bad
status: waiting for good commit(s), bad commit known
```

demo 0 1 Project: ibmi-company_system Live Share UTF-8 CRLF RPGLE 1

© Copyright IBM Corporation 2024

Git bisect

ibmi-company_system

mypgm.pgm.rpgle

qrpglesrc > mypgm.pgm.rpgle > ...

```
12
13    mytext = 'Hello to all you people';
14    printf(mytext);
15
16    dsply mytext;
```

PROBLEMS 1 | COMMENTS | DEBUG CONSOLE | OUTPUT | TERMINAL | IBM I JOB LOG 1

bash
bash

```
4457ed2 (HEAD -> demo) Fix binding directory
abf352f Add error handling
c831812 This is the bug
9c38621 Update targets in Rules.mk
d571b18 Add info to README
a28d080 Update iproj.json

AzureAD+SanjulaGanepola@IBM-PW0AC6C8 MINGW64 ~/Git/ibmi-company_system (demo|BISECTING)
$ git checkout a28d080
Note: switching to 'a28d080'.

You are in 'detached HEAD' state. You can look around, make experimental
changes and commit them, and you can discard any commits you make in this
```

a28d080b | 0 1 | Project: ibmi-company_system | Live Share | CRLF | RPGLE | 1

© Copyright IBM Corporation 2024

Git bisect

ibmi-company_system

mypgm.pgm.rpgle

qrpglesrc > mypgm.pgm.rpgle > ...

```
11    End-PR;
12
13    mytext = 'Hello to all you people';
14    printf(mytext);
15
16    dsply mytext;
17
```

PROBLEMS 1 COMMENTS DEBUG CONSOLE OUTPUT TERMINAL IBM I JOB LOG 1

bash
bash

```
AzureAD+SanjulaGanepola@IBM-PW0AC6C8 MINGW64 ~/Git/ibmi-company_system ((a28d080...)
|BISECTING)
$ git bisect good
```

a28d080b 0 1 Project: ibmi-company_system Live Share CRLF RPGLE 1

© Copyright IBM Corporation 2024

Git bisect

ibmi-company_system

mypgm.pgm.rpgle

qrpglesrc > mypgm.pgm.rpgle > ...

```
11    End-PR;
12
13    mytext = 'Hello to all you people';
14    printf(mytext);
15
16    dsply mytext;
17
```

PROBLEMS 1 | COMMENTS | DEBUG CONSOLE | OUTPUT | TERMINAL | IBM I JOB LOG 1

bash
bash

```
Bisecting: 2 revisions left to test after this (roughly 1 step)
[9c386211f362990da0bb46aad077dc3503fc4186] Update targets in Rules.mk

AzureAD+SanjulaGanepola@IBM-PW0AC6C8 MINGW64 ~/Git/ibmi-company_system ((9c38621...)
|BISECTING)
$ git bisect good
```

9c386211 | 0 1 | Project: ibmi-company_system | Live Share | CRLF | RPGLE | 1

© Copyright IBM Corporation 2024

Git bisect

ibmi-company_system

mypgm.pgm.rpgle

qrpglesrc > mypgm.pgm.rpgle > ...

```
11    End-PR;
12
13    mytext = 'Hello to all
14    printf(mytext);
15
16    dsply mytext;
17
```

PROBLEMS 1 | COMMENTS | DEBUG CONSOLE | OUTPUT | TERMINAL | IBM I JOB LOG 1

bash
bash

```
Bisecting: 0 revisions left to test after this (roughly 1 step)
[abf352fba9bc82d8c4ac93037dee1de8dd6b7755] Add error handling

AzureAD+SanjulaGanepola@IBM-PW0AC6C8 MINGW64 ~/Git/ibmi-company_system ((abf352f...)
|BISECTING)
$ git bisect bad
```

abf352fb | 0 1 | Project: ibmi-company_system | Live Share | CRLF | RPGLE | 1

© Copyright IBM Corporation 2024

Git bisect

ibmi-company_system

mypgm.pgm.rpgle

qrpglesrc > mypgm.pgm.rpgle > ...

```
12
13    mytext = 'Hello to all
14    printf(mytext);
15
16    dsply mytext;
```

PROBLEMS 1 | COMMENTS | DEBUG CONSOLE | OUTPUT | TERMINAL | IBM I JOB LOG 1

```
AzureAD+SanjulaGanepola@IBM-PW0AC6C8 MINGW64 ~/Git/ibmi-company_system ((c831812...)
|BISECTING)
$ git bisect bad
c8318129c9446abf4f1ee55846699660385d9142 is the first bad commit
commit c8318129c9446abf4f1ee55846699660385d9142 (HEAD)
Author: Sanjula Ganepola <Sanjula.Ganepola@ibm.com>
Date:   Fri May 16 19:45:13 2025 -0400

    This is the bug

    Signed-off-by: Sanjula Ganepola <Sanjula.Ganepola@ibm.com>

 qrpglesrc/mypgm.pgm.rpgle | 2 +-
 1 file changed, 1 insertion(+), 1 deletion(-)
```

bash
bash

c8318129 | 0 1 | Project: ibmi-company_system | Live Share | CRLF | RPGLE | 1

© Copyright IBM Corporation 2024

Git in VS Code

© Copyright IBM Corporation 2024

Viewing changes and in-editor diff

File Edit Selection View Go Run ibmi-company_system

SOURCE CONTROL

REPOSITORIES

ibmi-c... fix/sa

CHANGES

Message (Ctrl+Enter to commi...

✓ Commit

Changes

account.table qddssrc

employee.table qddssrc

~~popdept.sql qddssrc~~

constants.rpgleinc qrpgleref

depts.pgm.sqlrpgle qrpglesrc

mypgm.pgm.rpgle qrpglesrc

View & Sort

Pull

Push

Clone

Checkout to...

Fetch

Commit

Changes

Pull, Push

Branch

Remote

Stash

Tags

Show Git Output

GRAPH

employee.table (Working Tree) M

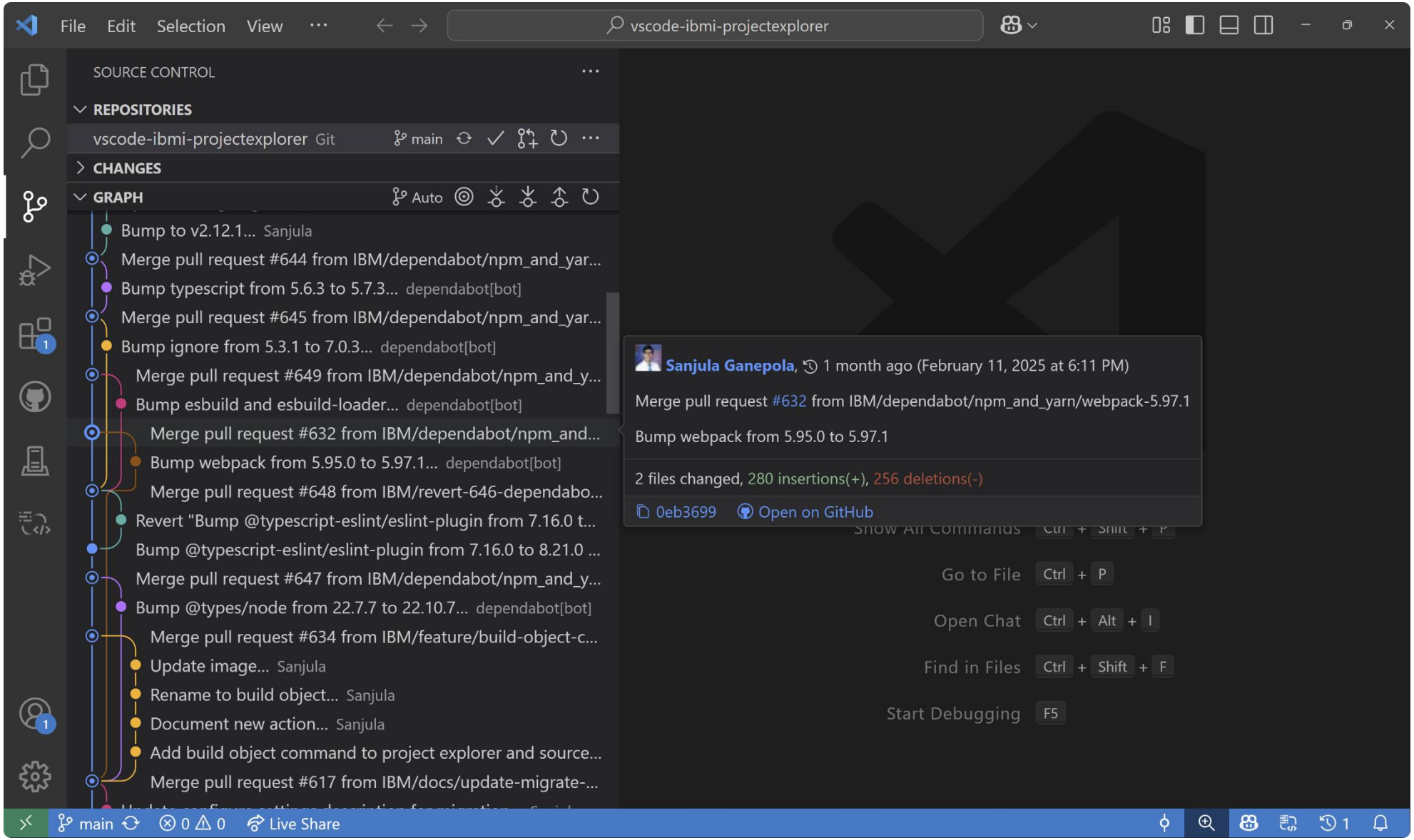
qddssrc > employee.table

```
3   3  CREATE OR REPLACE TABLE EMPLOYEE
4   4      (EMPNO       CHAR(6)         NOT NULL,
           FIRSTNME     VARCHAR(12)     NOT NULL,
           MIDINIT      CHAR(1)         NOT NULL,
           LASTNAME     VARCHAR(15)     NOT NULL,
           WORKDEPT     CHAR(3)                 ,
           PHONENO      CHAR(4)                 ,
           HIREDATE     DATE                    ,
           JOB          CHAR(8)                 ,
           JOB          CHAR(9)                 ,
           EDLEVEL      SMALLINT        NOT NULL,
           SEX          CHAR(1)                 ,
           BIRTHDATE    DATE                    ,
           SALARY       DECIMAL(9,2)            ,
           BONUS        DECIMAL(9,2)            ,
           COMM         DECIMAL(9,2)            ,
           PRIMARY KEY (EMPNO));

           move circular reference
           TER TABLE EMPLOYEE
               ADD FOREIGN KEY RED (WORKDEPT)
               REFERENCES DEPARTMENT
               ON DELETE SET NULL;

26  25  ALTER TABLE EMPLOYEE
27  26      ADD CONSTRAINT NUMBER
28   -      CHECK (PHONENO >= '0000' AND PHONENO <= '9998');
    27+     CHECK (PHONENO >= '0001' AND PHONENO <= '9998');
29  28
30  29  -- CREATE UNIQUE INDEX XEMP1
31  30  --        ON EMPLOYEE (EMPNO);
32  31
33  32  -- CREATE INDEX XEMP2
34  33  --        ON EMPLOYEE (WORKDEPT);
```

fix/sample-demo* 0 1 Project: ibmi-company_system Live Share aecspades (2 years ago) Ln 19, Col 1 Spaces: 6 UTF-8 CRLF {} sql 1

© Copyright IBM Corporation 2024

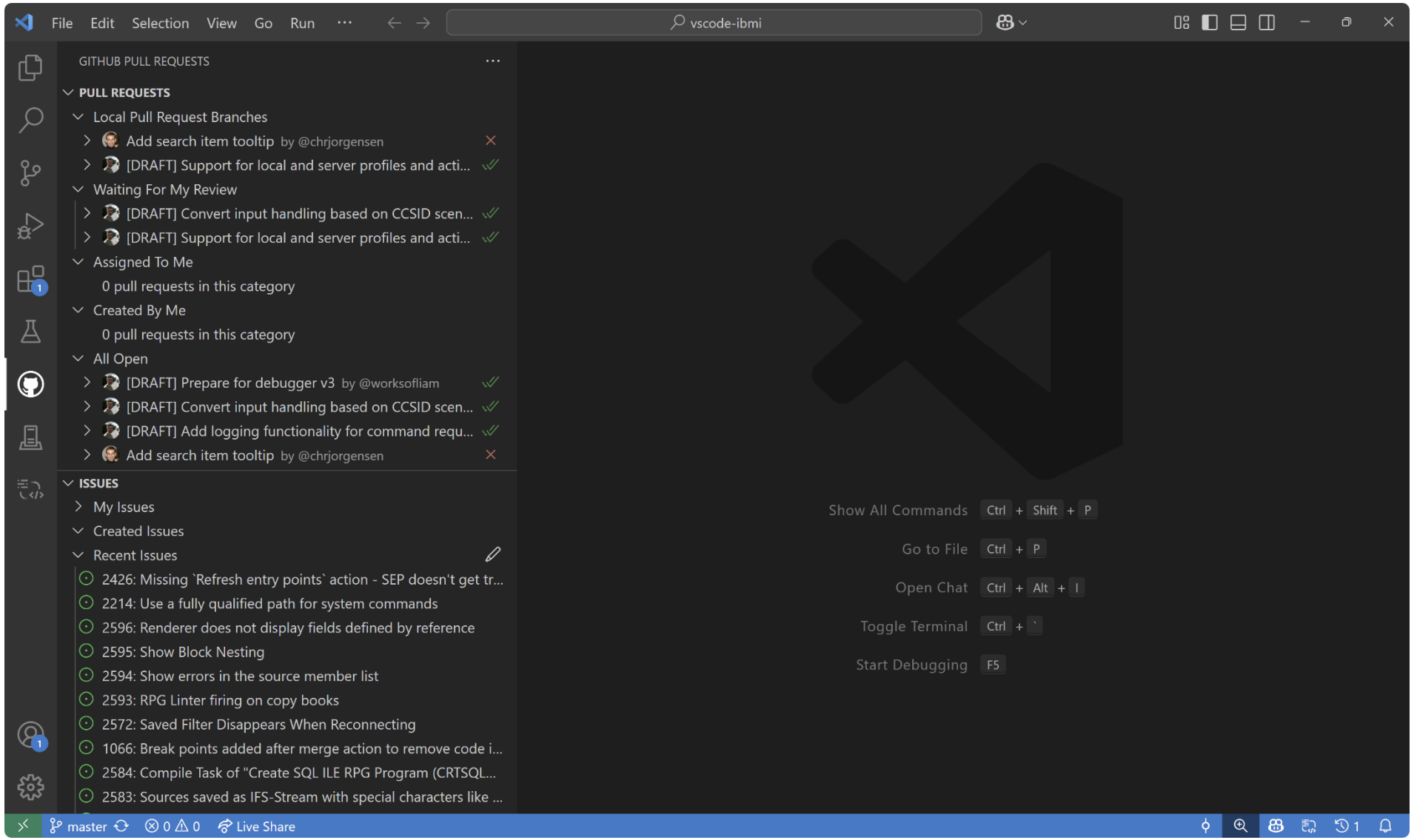
Viewing Git history as a graph



© Copyright IBM Corporation 2024

Viewing pull requests

https://marketplace.visualstudio.com/items?itemName=GitHub.vscode-pull-request-github



© Copyright IBM Corporation 2024

Viewing changes in pull requests

https://marketplace.visualstudio.com/items?itemName=GitHub.vscode-pull-request-github

File Edit Selection View Go Run ··· vscode-ibmi

GITHUB PULL REQUEST

CHANGES IN PULL REQUEST #2556

Convert input handling based on CCSID scenarios

- Files
  - src/api
    - tests
      - suites
        - encoding.test.ts M
        - setup.ts M
        - CompileTools.ts M
        - IBMi.ts M
        - IBMiContent.ts M
        - Search.ts M
        - Tools.ts M
- Commits

REVIEW PULL REQUEST #2556

Leave a comment

Approve | Comment | Request Changes

This branch has conflicts that must be resolved.

Checkout 'master'

Changes in Pull Request #2556 (7 files)

TS CompileTools.ts src\api

146 hidden lines | {} CompileTools > runCommand

```
147 147   command: [
148 148     ...options.noLibList? [] : buildLiblistCommands(connection, ileSet
149 149     ...commands.map(command =>
150 —         `${`system "${IBMi.escapeForShell(command)}"`}`,
    150+      `${`system "${IBMi.escapeForShell(connection.sysNameInAmerican(c
151 151     )
152 152   ].join(` && `),
153 153   directory: cwd,
```

36 hidden lines

TS IBMi.ts src\api

17 hidden lines

```
18 18 import { EventEmitter } from 'stream';
19 19 import { ConnectionConfig } from './configuration/config/types';
20 20 import { EditorPath } from '../typings';
   21+
   22+ export interface VariantInfo {
   23+   american: string,
   24+   local: string,
   25+   qsysNameRegex?: RegExp
   26+ };
21 27
22 28 export interface MemberParts extends IBMiMember {
23 29   basename: string
```

100 hidden lines | IBMi

```
124 130
125 131   remoteFeatures: { [name: string]: string | undefined };
126 132
127 —     variantChars: {
128 —       american: string,
129 —       local: string,
```

fix/convert_basis_by_basis ⊗ 0 ⚠ 0 Live Share Pull Request #2556

© Copyright IBM Corporation 2024

Any Questions?

© Copyright IBM Corporation 2024

Important Links

Installation

- https://git-scm.com/downloads
- https://git-lfs.com/
- https://marketplace.visualstudio.com/items?itemName=GitHub.vscode-pull-request-github

Documentation

- https://git-scm.com/docs

Recommended Git Tutorials

- https://www.youtube.com/watch?v=-iWaarLI7zI
- https://www.youtube.com/watch?v=e9lnsKot_SQ
- https://www.youtube.com/watch?v=0chZFIZLR_0&t=66s

© Copyright IBM Corporation 2024

Sanjula Ganepola - Git Concepts I Wish I Knew Years Ago

Please take the last minute of this session to complete the evaluation. A direct link to the evaluation can be found using the QR code to the right.

IBM i

IBM