



Society of Actuaries in Ireland

What could possibly go wrong?
How version control systems can help actuaries

14th December 2020

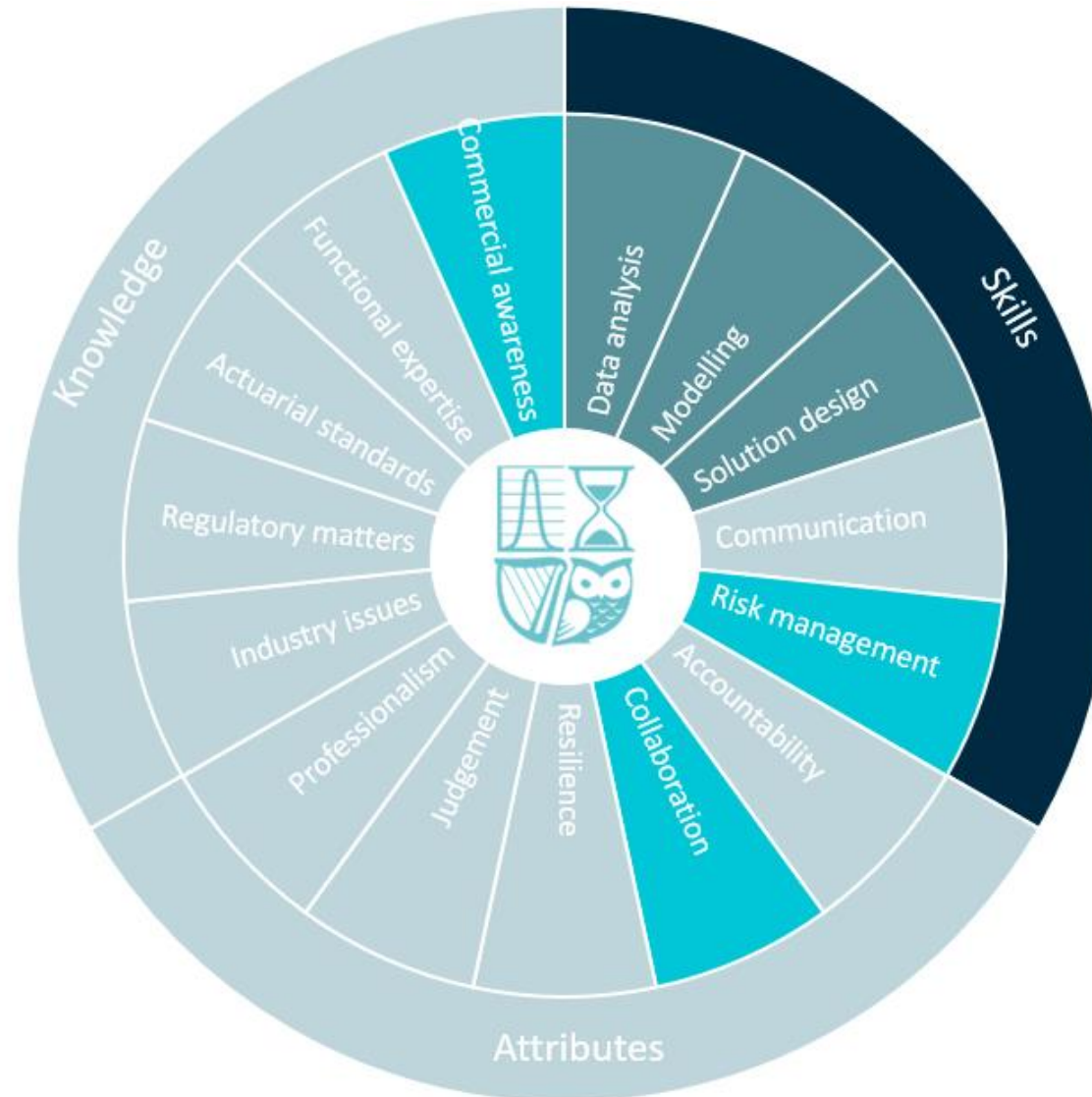
Disclaimer

The views expressed in this presentation are those of the presenter(s) and not necessarily those of the Society of Actuaries in Ireland or their employers.

Who am I

- **Shane Kennedy FSAI, FIA**
- **Associate Director of Actuarial Service, Optum Ireland**
- **13 yrs experience across Life and Healthcare industries**
- **Member of SAI Data Analytics committee**

Competency Framework Wheel





What could possibly go wrong?

➤ Nasa Mars climate orbiter

➤ Irish Leaving Cert 2020



What could possibly go wrong?

➤ Excel is a great tool but ...



expense_assumptions_2019_v29_final



expense_assumptions_2019_v29_final_v2



expense_assumptions_2019_v29_final_v2_final



expense_assumptions_2019_v29_final_v2_updated

➤ And so is word but ...



term_pricing_report_final



term_pricing_report_final_13_10_2020



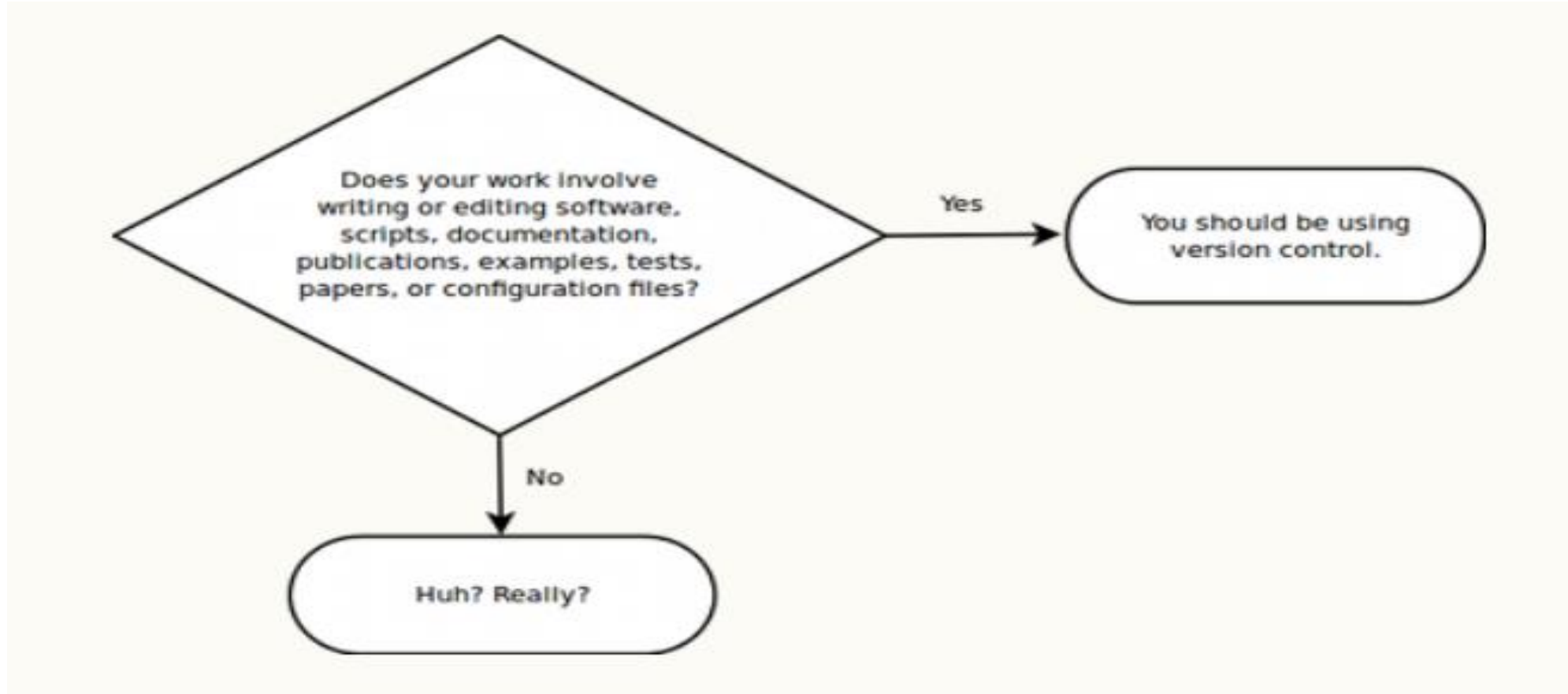
term_pricing_report_final_sk

WOMBAT: Waste Of Money Brains and Time!



What could possibly go wrong?

- As actuaries embrace tools such as R and Python version control systems become even more important



Source: <http://soundsoftware.ac.uk/why-version-control>

- What is a version control system?
- Relevance for Actuaries
- Demonstration



What is a version control system?

“Version control systems are a category of software tools that help a software team manage changes to source code over time.”

Source: Atlassian.com

- Manages the evolution of a set of files (a repository) in a structured way
- Complete history of content is tracked and available
- This content may include model source code, validation testing code and documentation
- Examples include Git, SVN, CVS, Mercurial



What is a *Distributed* version control system (DVCS)?

- Each user has a local project history called a repository
- A single remote repository exists which is the source of truth (master)
- Users can work offline since they have their own local copy of the project history
- DVCS allow users to 'pull' content from and 'push' content to the master copy

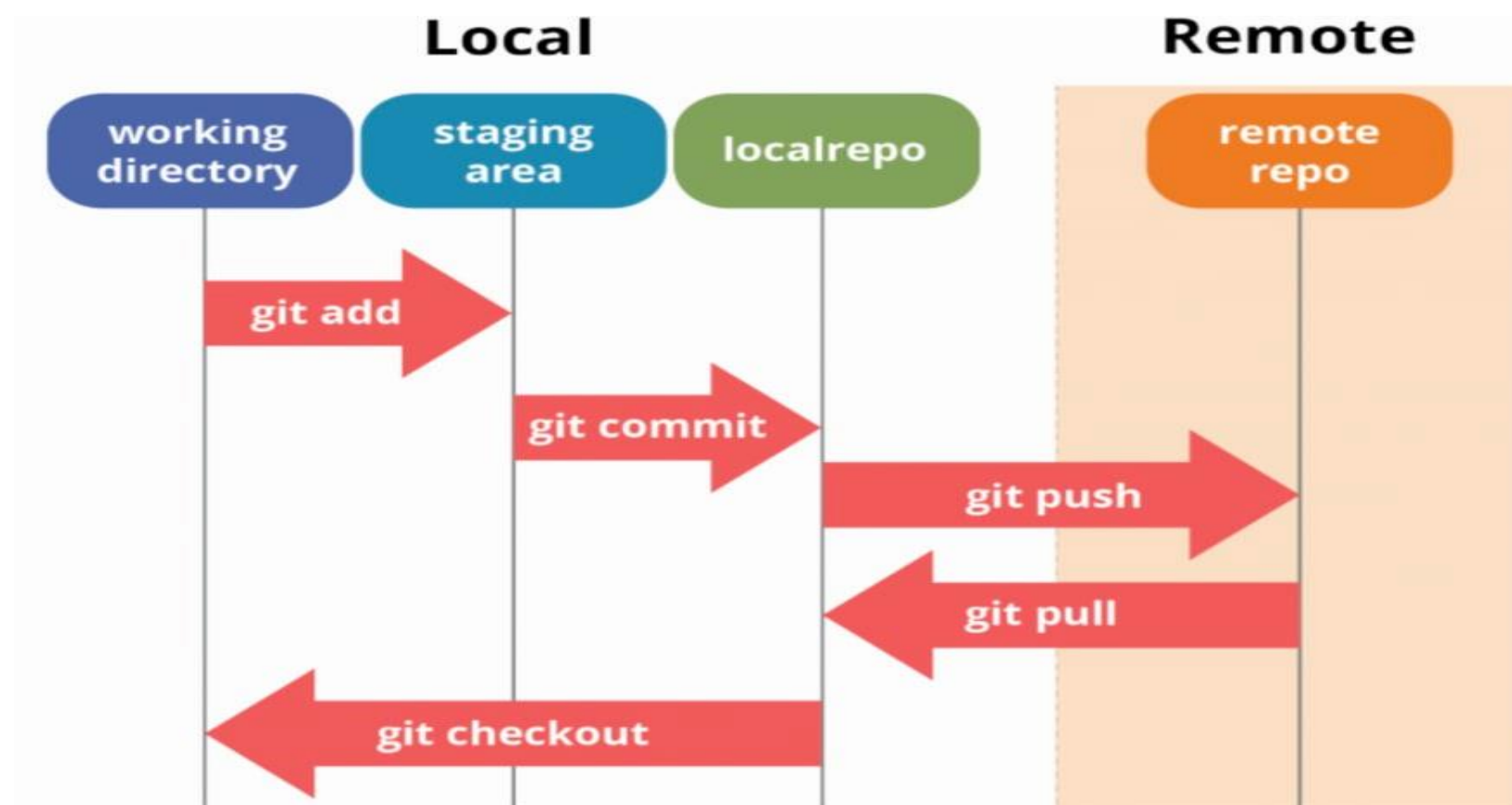


The Lingo!





Local and remote hosting



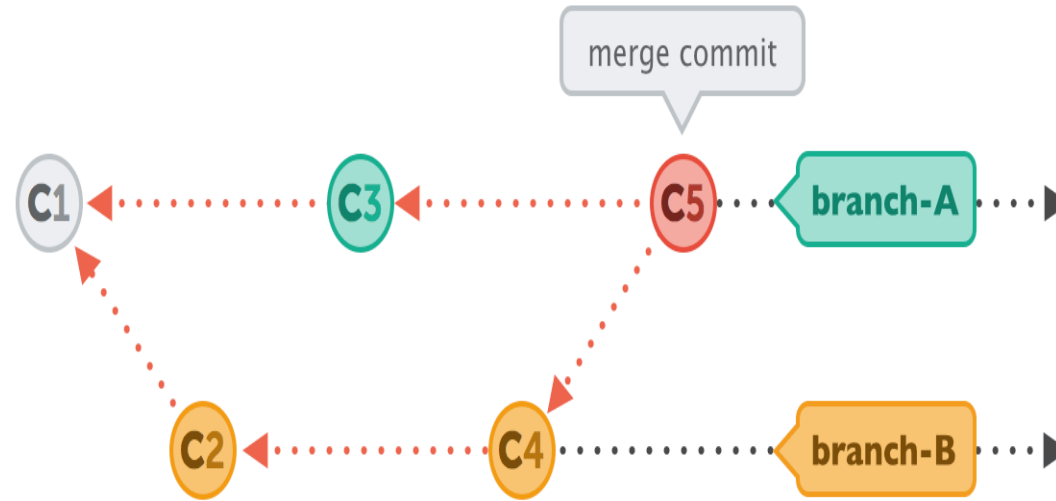
source: <https://dev.to/mollynem/git-github--workflow-fundamentals-5496>

- Git used for demonstration purposes throughout this presentation
- User 'Pulls' project from Remote repo (master) onto their Local machine
- Local machine has three main components: working directory, staging area, local repo
- User can add to staging area, commit to local repo, push to remote repo



Health warning – some technical stuff!

Directed Acyclic Graph (DAG)



source: <https://i.stack.imgur.com/yE1kA.gif>

- Arrows point to the preceding **commit**, not the subsequent one
- **Branches** are labels on commits
- Commits can be **merged** to incorporate changes from more than one branch



Lets design an outdoor play gym!





Lets design an outdoor play gym!

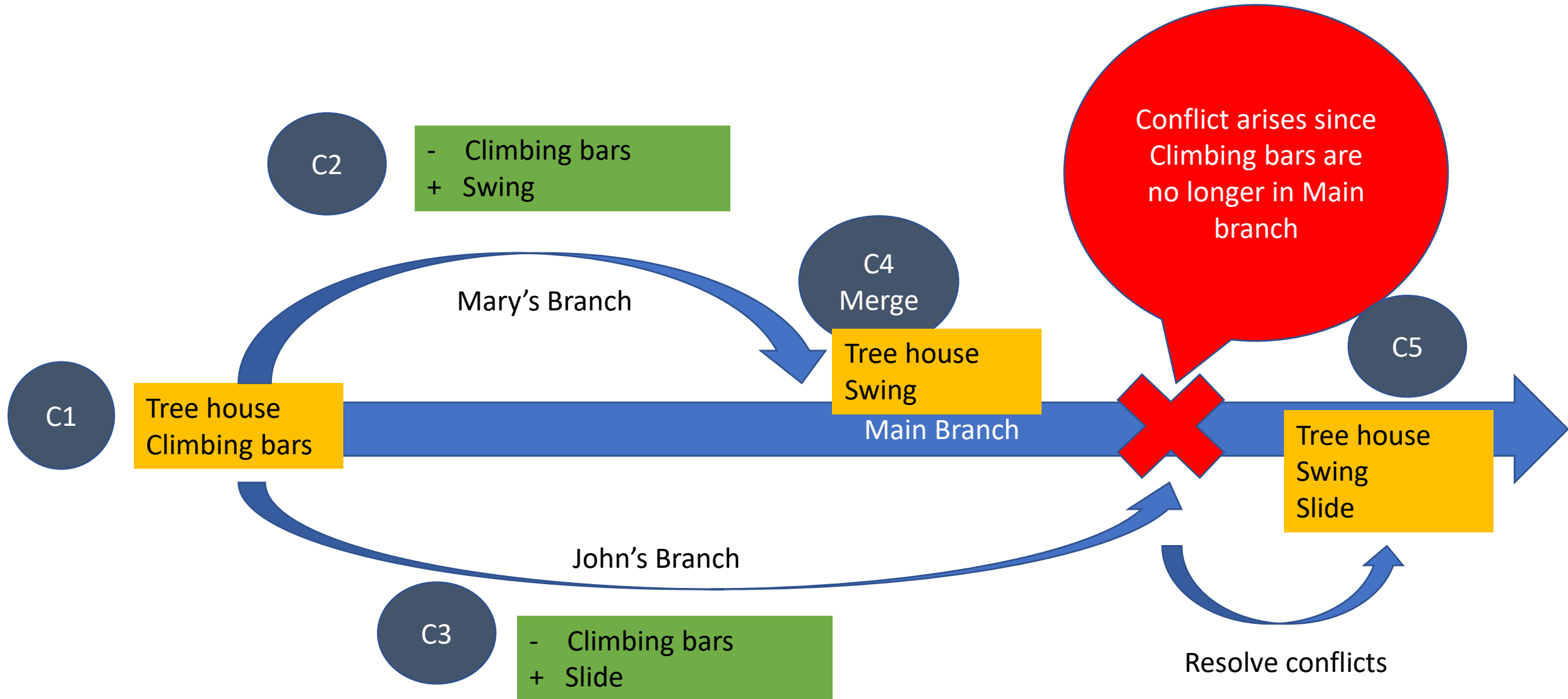
- *Standard play gyms include a Tree house and climbing bars so we start with this*
- *Mary and John have been tasked with enhancing the design*
- *Mary wants to remove the climbing bars and replace with a swing*
- *John wants to remove the climbing bars and add a slide*
- *In the end, they agree to include both!!*

Conflict!



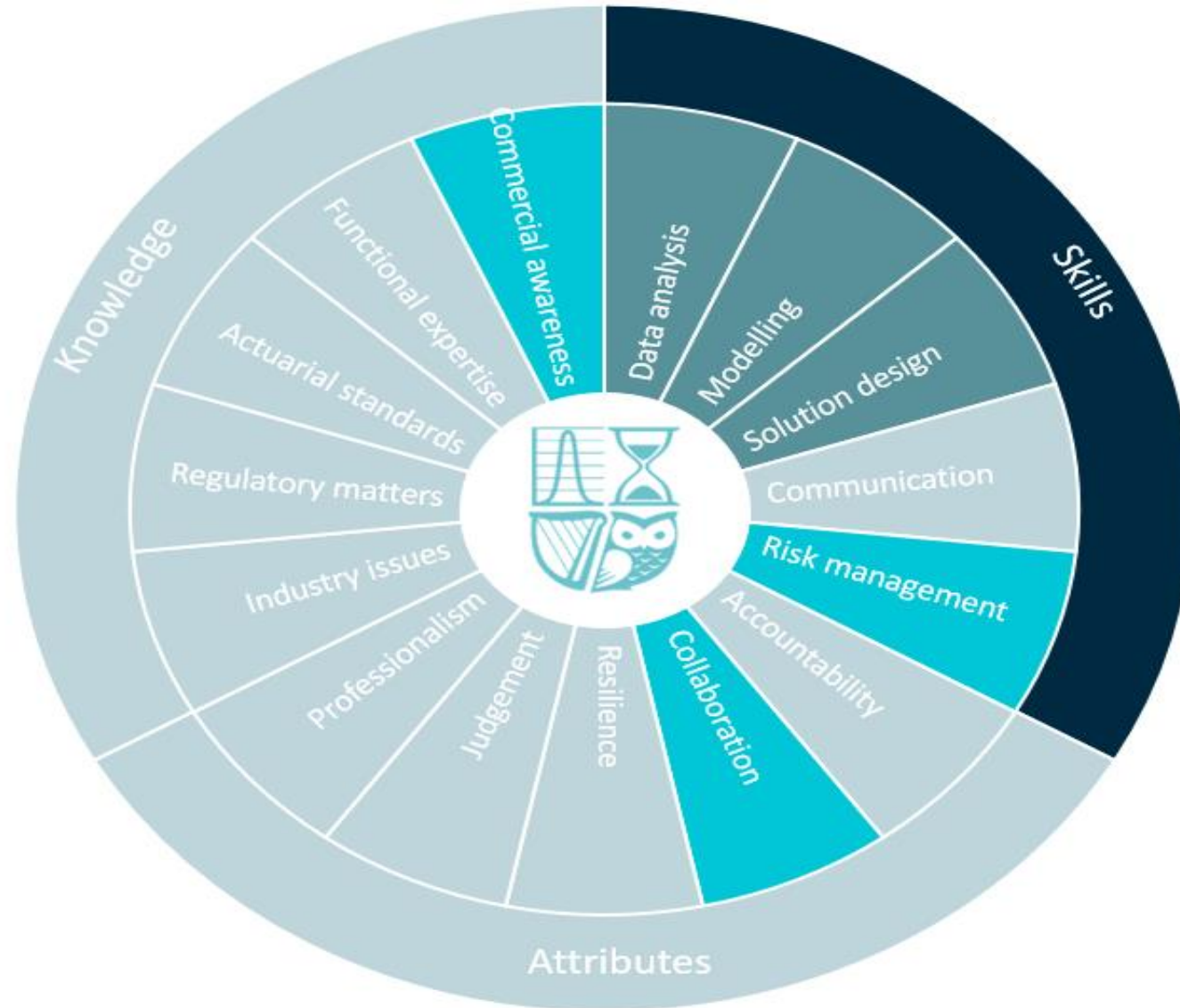
Conflicts

Conflicts can arise when changes are made that don't gel



-
- What is a version control system?
 - Relevance for Actuaries?
 - Demonstration

Competency Framework Wheel





Relevance for Actuaries

- Risk Management
 - Tracked changes, maintains history of all changes made
 - Can have different permissions depending on seniority levels
- Collaboration
 - Facilitates multiple people working at the same time
 - Actuaries can work with other professionals (Data scientists, Software developers) on projects



Commercial considerations

- Choice of version control system
 - Distributed (e.g. Git) more popular in recent times than Centralized (e.g. SVN)
- Choice of repository hosting service
 - Examples include Github, Bitbucket, Gitlab, Assembla, Beanstalk, Sourcetree
 - Some offer free private repositories to individuals and organisations
 - Typically cloud based
 - Need to ensure security and data privacy features are appropriate
- Interacting with version control systems
 - Can use command line however User Interfaces (UIs) are more user friendly
 - Many applications facilitate easy interaction with version control systems
 - Examples include R Studio, SAS Enterprise, Visual Studio, Tortoise, Github desktop (Git only)

-
- What is a version control system?
 - Relevance for Actuaries?
 - Demonstration



Demonstration

For the purposes of the demonstration we will use the following

- Version Control System => Git
- Repository hosting service => Github
- User Interface => R Studio

Items to be covered include:

- Getting set-up on a version control system (Git) and hosting service (Github)
- Integrating with User Interface (R Studio)
- Useful functions (create branch, stage, commit, push, pull request, resolve conflicts, merge)

Some advice: Just get set up and get started!!



Getting set-up

- Downloading Git: <https://git-scm.com/downloads>

The screenshot shows the Git website's Downloads page. The header features the Git logo and the tagline "--distributed-even-if-your-workflow-isnt". A search bar is located in the top right. The left sidebar contains links for "About", "Documentation", "Downloads" (highlighted), "GUI Clients", "Logos", and "Community". The main content area is titled "Downloads" and lists operating systems: "Mac OS X", "Windows", and "Linux/Unix". A monitor graphic displays the "Latest source Release 2.29.2" with a link to "Release Notes (2020-10-29)" and a button to "Download 2.29.2 for Windows". Below the OS links, it states "Older releases are available and the Git source repository is on GitHub." At the bottom, there are sections for "GUI Clients" and "Logos", each with a "View" link.

git --distributed-even-if-your-workflow-isnt

Search entire site...

About
Documentation
Downloads
GUI Clients
Logos
Community

The entire **Pro Git book** written by Scott Chacon and Ben Straub is available to [read online for free](#). Dead tree versions are available on [Amazon.com](#).

Downloads

Mac OS X Windows Linux/Unix

Older releases are available and the Git source repository is on GitHub.

Latest source Release
2.29.2
[Release Notes \(2020-10-29\)](#)
Download 2.29.2 for Windows

GUI Clients

Git comes with built-in GUI tools (**git-gui**, **gitk**), but there are several third-party tools for users looking for a platform-specific experience.

[View GUI Clients →](#)

Logos

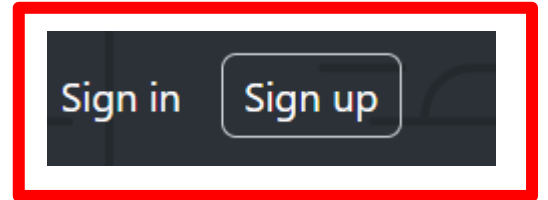
Various Git logos in PNG (bitmap) and EPS (vector) formats are available for use in online and print projects.

[View Logos →](#)



Getting set-up

- Creating a Github account: github.com



Join GitHub

Create your account

Username *

Email address *

Password *

Make sure it's **at least 15 characters** OR **at least 8 characters including a number and a lowercase letter**. [Learn more](#).

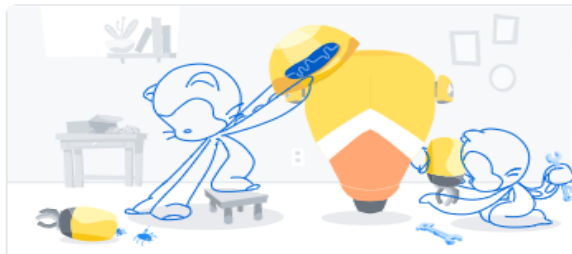


Getting set-up

- Starting a new project

What do you want to do first?

Every developer needs to configure their environment, so let's get your GitHub experience optimized for you.



Start a new project

Start a new repository or bring over an existing repository to keep contributing to it.

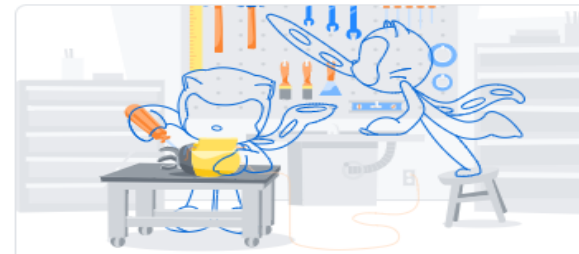
Create a repository



Collaborate with your team

Improve the way your team works together and get access to more features with an organization.

Create an organization



Learn how to use GitHub

Get started with an "Introduction to GitHub" course in our Learning Lab.

Start Learning

[Skip this for now >](#)



Getting set-up

- Creating a new 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 *

 testactuarialsaianalytics ▾

Repository name *

assumptionsetting ✓

Great repository names are short and memorable. Need inspiration? How about **musical-fortnight**?

Description (optional)

Data analysis as part of annual assumption setting process



Public

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



Private

You choose who can see and commit to this repository.

Initialize this repository with:

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

☒ **Add a README file**


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

☐ **Add .gitignore**

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

☐ **Choose a license**

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

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


Create repository






Collaborating


27

- Inviting Collaborators



[Pull requests](#) [Issues](#) [Marketplace](#) [Explore](#)


 [testactuarialsaianalytics / assumptionsetting](#)


[Unwatch](#) 1 [Star](#) 0 [Fork](#) 0

[Code](#) [Issues](#) [Pull requests](#) [Actions](#) [Projects](#) [Wiki](#) [Security](#) [Insights](#) [Settings](#)

[main](#) 1 branch 0 tags

[Go to file](#) [Add file](#) [Code](#)

 [testactuarialsaianalytics](#) Initial commit c83ac71 now 1 commits

 [README.md](#) Initial commit now

README.md

assumptionsetting

Data analysis as part of annual assumption setting process

About

Data analysis as part of annual assumption setting process

[Readme](#)

Releases

No releases published

[Create a new release](#)

Packages

No packages published

[Publish your first package](#)



Collaborating

- Inviting Collaborators

testactuarialsaianalytics / assumptionsetting

<> Code

! Issues

🔗 Pull requests

▶ Actions

📁 Projects

📖 Wiki

🛡 Security

📈 Insights

⚙ Settings

Options

Manage access

Security & analysis

Branches

Webhooks

Notifications

Integrations

Deploy keys

Actions

Secrets

Moderation settings

Settings

Repository name

assumptionsetting

Rename

☐ Template repository

Template repositories let users generate new repositories with the same directory structure and files. [Learn more.](#)

Social preview

Upload an image to customize your repository's social media preview.

Images should be at least 640×320px (1280×640px for best display).


[Download template](#)




Collaborating

29

- Inviting Collaborators



[Pull requests](#) [Issues](#) [Marketplace](#) [Explore](#)

 [testactuarialsaianalytics / assumptionsetting](#)

[Code](#) [Issues](#) [Pull requests](#) [Actions](#) [Projects](#) [Wiki](#) [Security](#) [Insights](#) [Settings](#)

Options

Manage access

Security & analysis

Branches

Webhooks

Notifications

Integrations

Deploy keys

Actions

Secrets

Moderation settings

Who has access

PUBLIC REPOSITORY


This repository is public and visible to anyone.

[Manage](#)

DIRECT ACCESS

0 collaborators have access to this repository. Only you can contribute to this repository.

Manage access



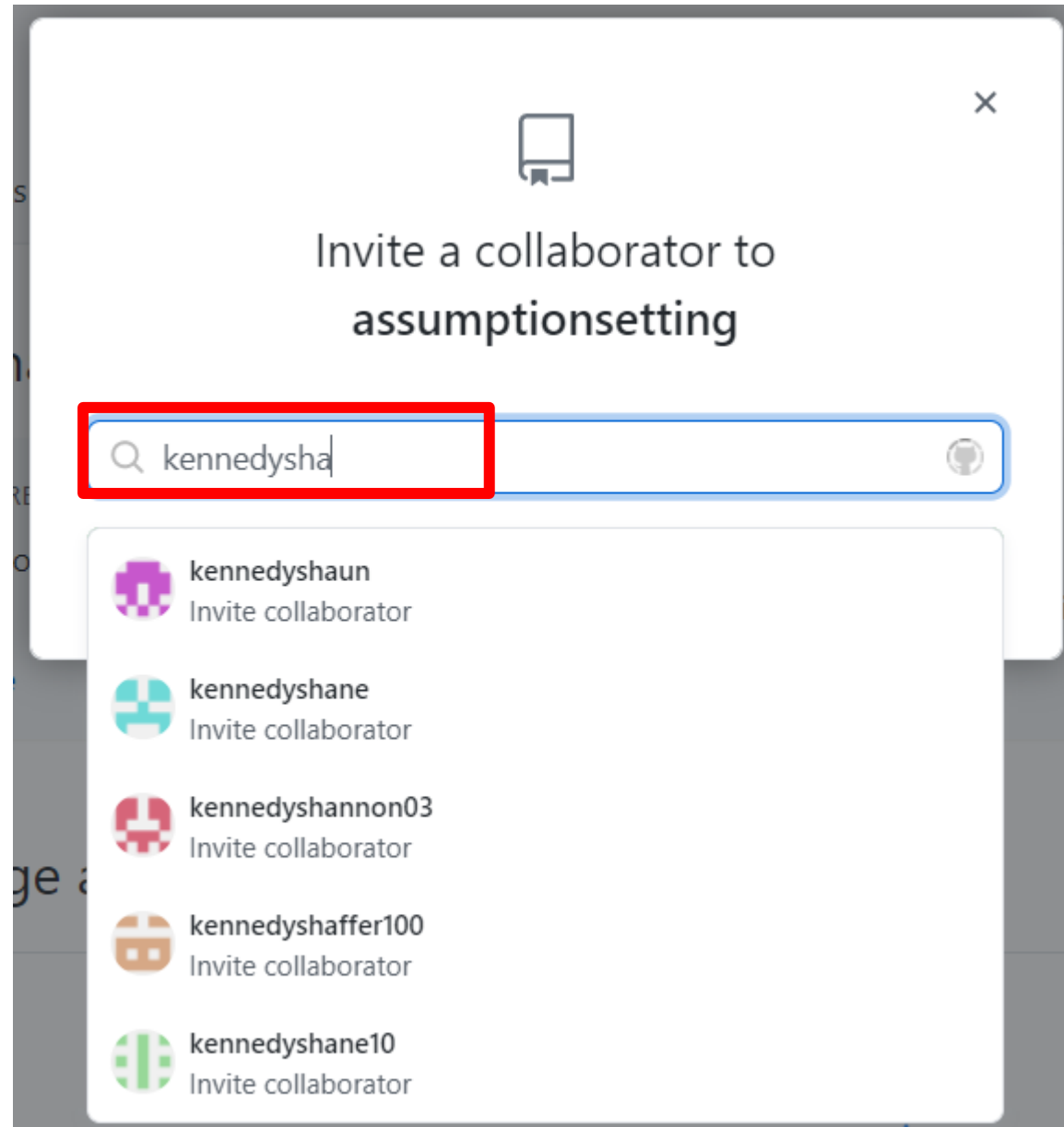
You haven't invited any collaborators yet

[Invite a collaborator](#)



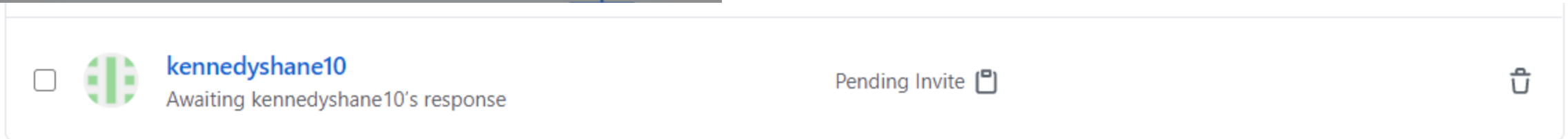
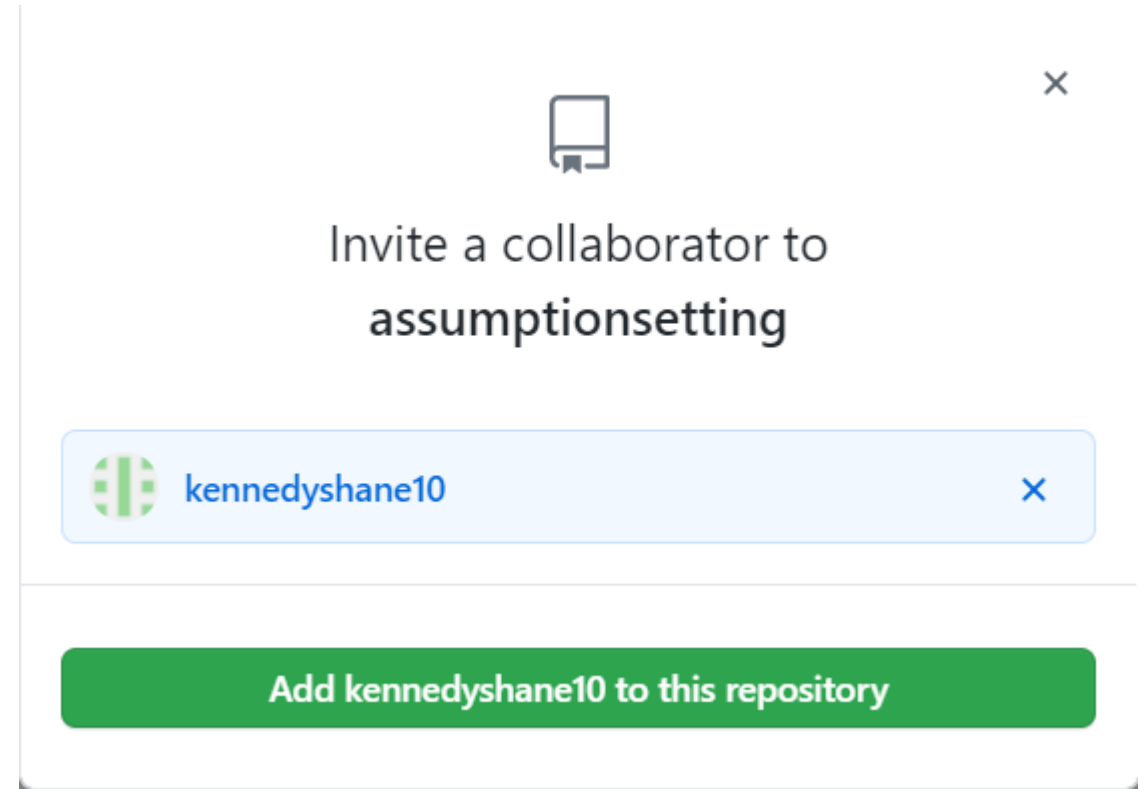
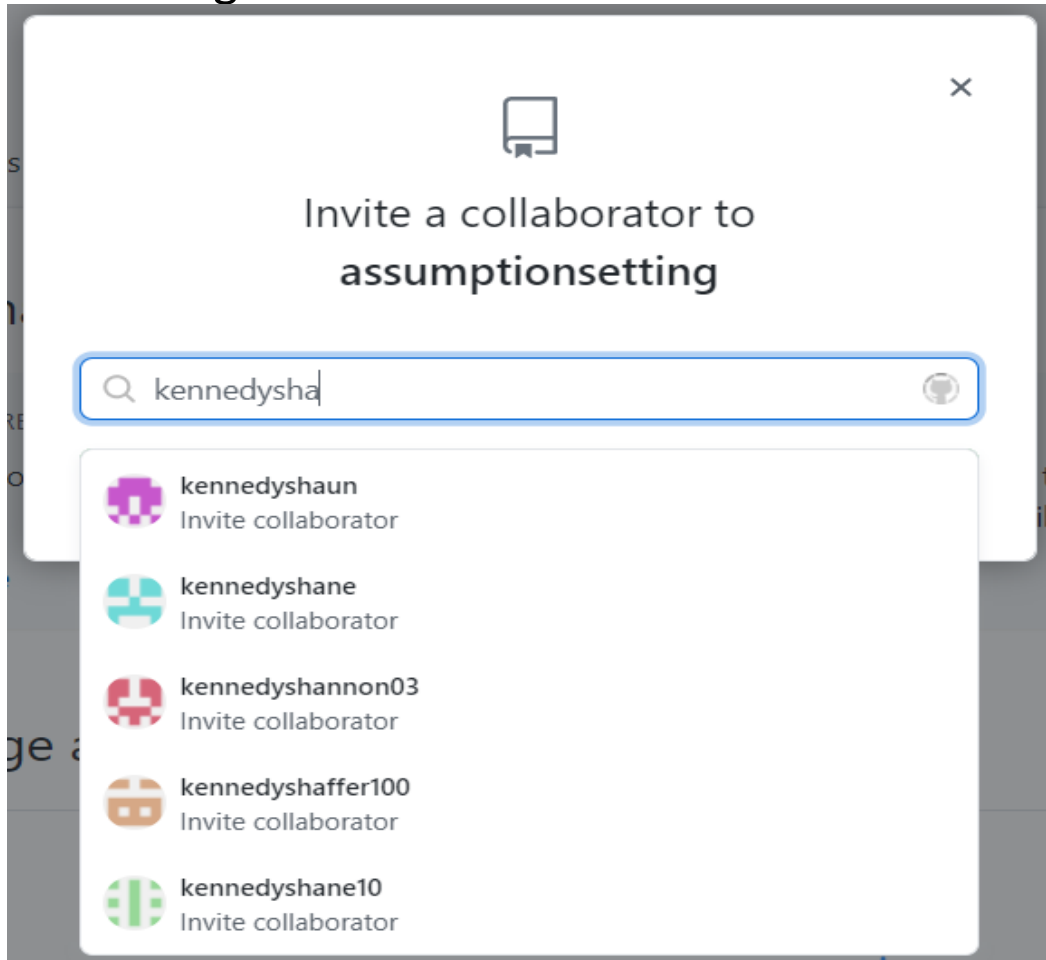
Collaborating

- Inviting Collaborators



Collaborating

- Inviting Collaborators





Collaborating

- Creating an organization

Choose a plan

Pick a plan for your team

Free

The basics of GitHub for every team

- ∞ Unlimited public/private repositories
- ∞ Unlimited collaborators
- ✓ 2,000 Actions minutes/month
Free for public repositories
- ✓ 500MB of GitHub Packages
- ✓ Community Support

\$0

Join for free

Team

Advanced collaboration and support for teams

- ← Everything in Free
- ✓ Required reviewers
- ✓ 3,000 Actions minutes/month
Free for public repositories
- ✓ 2GB of GitHub Packages
- ✓ Code owners

\$4 per user/month

Continue with Team

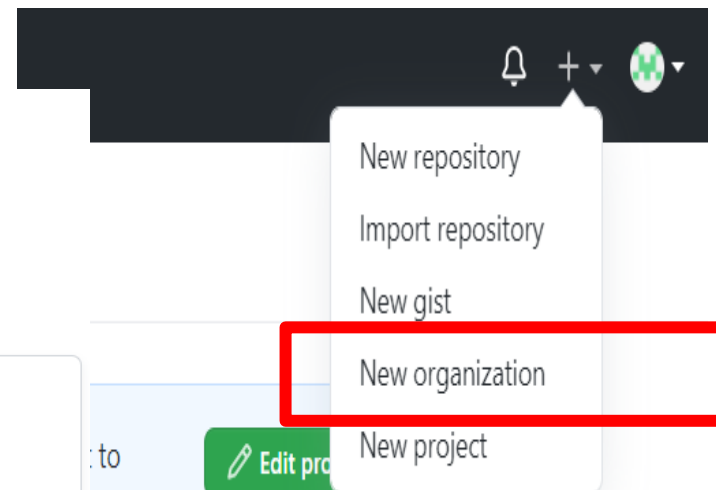
Enterprise

Security, compliance, and flexible deployment for enterprises

- ← Everything in Team
- ✓ SAML single sign-on
- ✓ 50,000 Actions minutes/month
Free for public repositories
- ✓ 50GB of GitHub Packages
- ✓ Advanced auditing

\$21 per user/month

Start Enterprise trial





Collaborating

- Creating an organization

Tell us about your organization

Set up your team

Organization account name *

saiactuarial



This will be the name of your account on GitHub.

Your URL will be: <https://github.com/saiactuarial>.

Contact email *

kennedyshane10@gmail.com

This organization belongs to: *

☒ My personal account

I.e., kennedyshane10

☐ A business or institution

For example: GitHub, Inc., Example Institute, American Red Cross

Next

By creating an account, you agree to the [Terms of Service](#). For more information about GitHub's privacy practices, see the [GitHub Privacy Statement](#). We'll occasionally send you account-related emails.



Collaborating

- Creating an organization

Start collaborating

Welcome to saiactuarial

Add organization members

Organization members will be able to view repositories, organize into teams, review code, and tag other members using @mentions.

[Learn more about permissions for organizations →](#)

Search by username, full name or email address



actuarialdataanalytics
actuarialdataanalytics



Complete setup

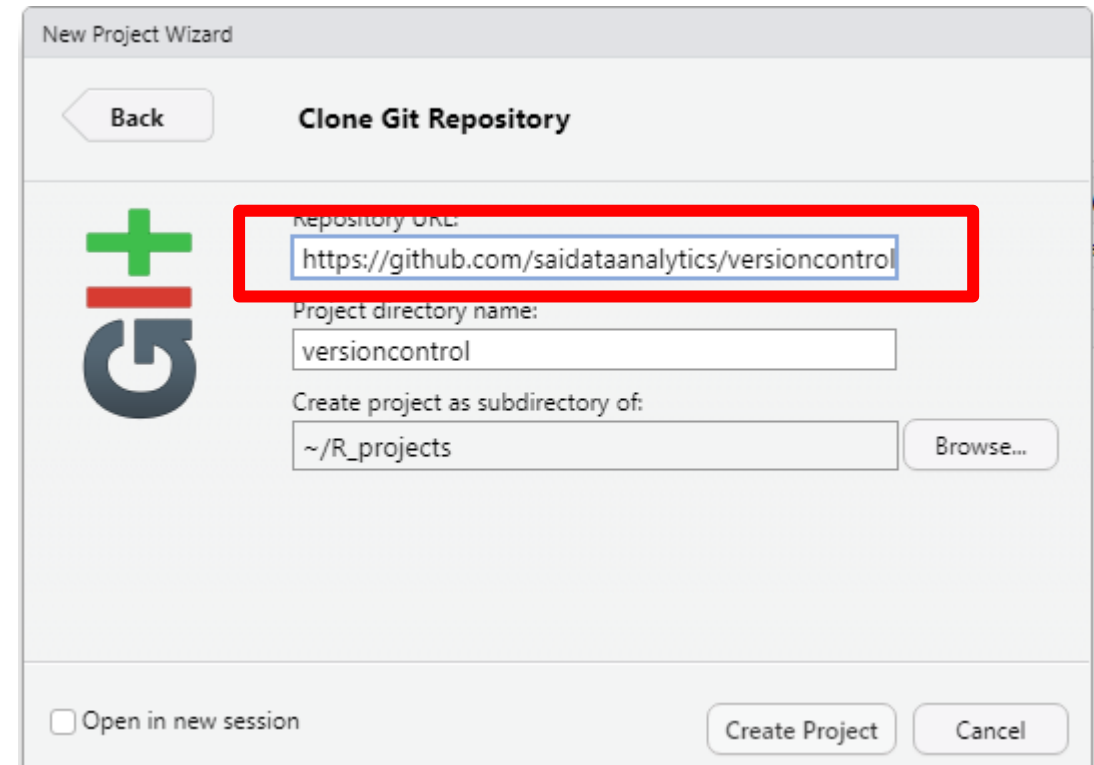
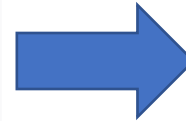
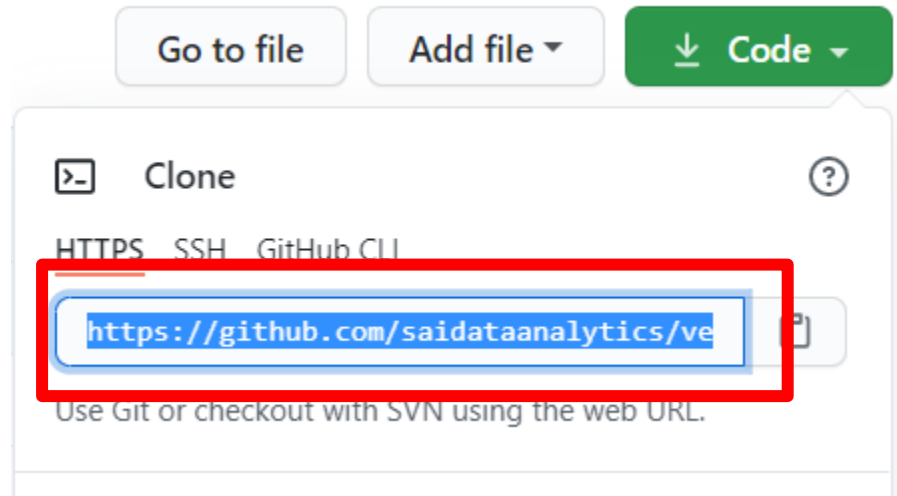
[Skip this step](#)



Collaborating

- Clone an existing repository

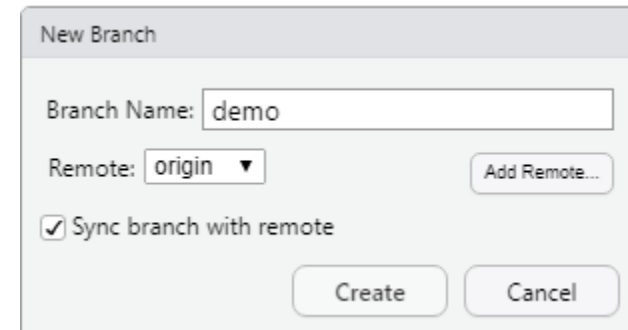
R Studio: File -> New Project -> Version Control -> Git





Collaborating

- Creating a new branch from R Studio



```
Git Branch
Close

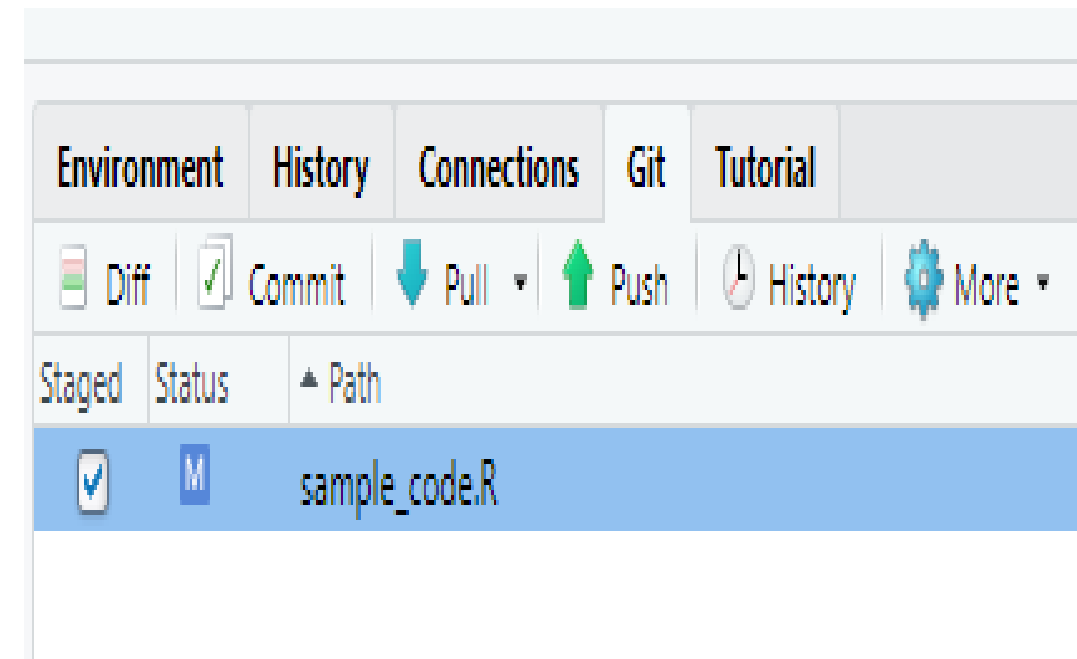
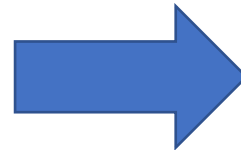
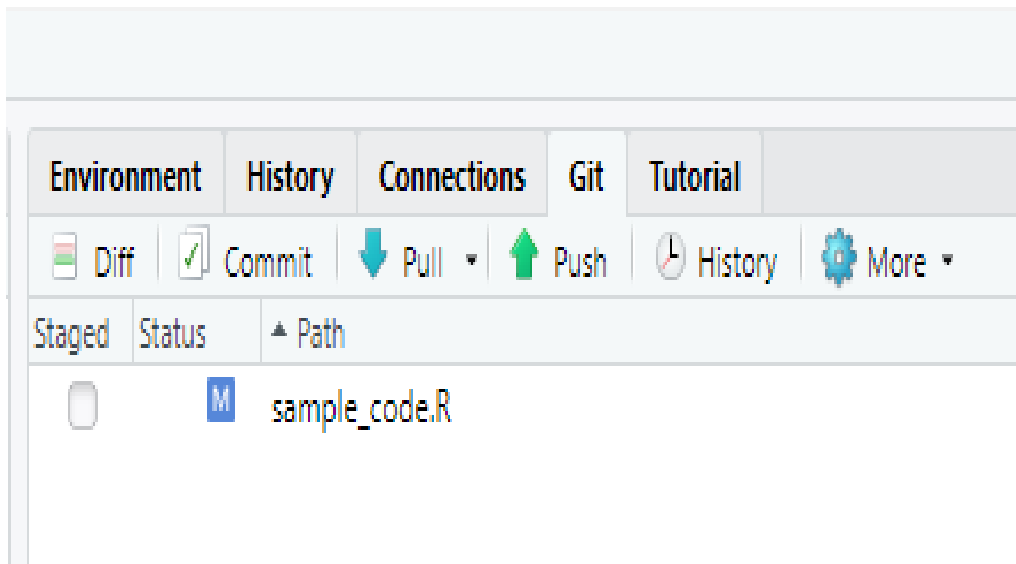
>>> C:/Program Files/Git/bin/git.exe checkout -B demo
Switched to a new branch 'demo'
>>> C:/Program Files/Git/bin/git.exe push -u origin demo
remote:
remote: Create a pull request for 'demo' on GitHub by visiting:
remote:   https://github.com/saidataanalytics/versioncontrol/pull/new/demo
remote:
remote: Branch 'demo' set up to track remote branch 'demo' from 'origin'.
To https://github.com/saidataanalytics/versioncontrol.git
* [new branch]      demo -> demo
|
```



Collaborating

- Edit code locally then stage, commit and push to remote host

STAGE





Collaborating

- Edit code locally then stage, commit and push to remote host

COMMIT

RStudio: Review Changes

Changes History demo Stage Revert Ignore Pull Push

staged Status Path

sample_code.R

Commit message

Including boxplot| 17 characters

Amend previous commit

Commit

Show Staged Unstaged Context 5 line Ignore Whitespace Unstage All

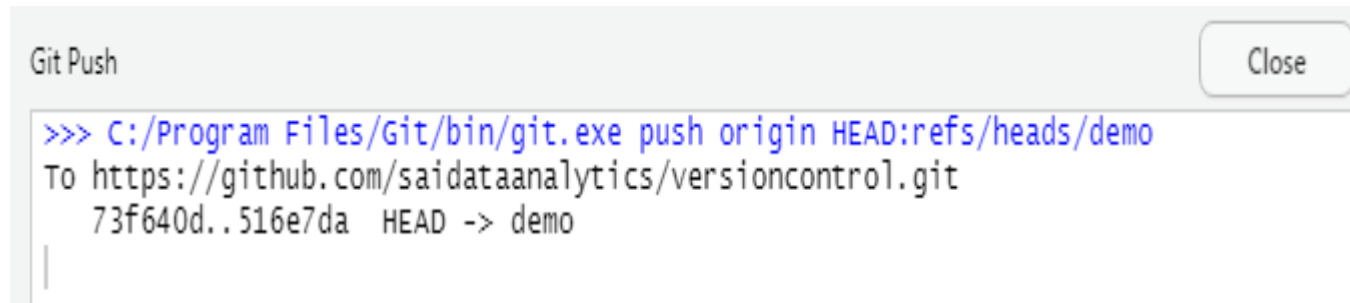
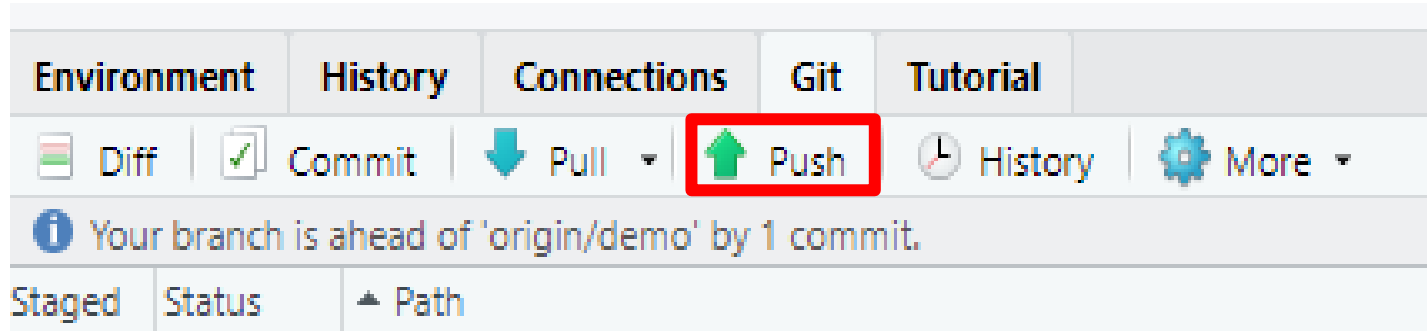
```
@@ -10,7 +10,7 @@ str(mtcars)
10 10 #create a scatterplot of mpg versus hp
11 11 my_scatterplot<-ggplot(mtcars,aes(x = mpg, y = hp)) + geom_point()
12 12 my_scatterplot
13 13
14 14 #create a box and whiskers plot of cylinders versus mpg - updated
15 15 #my_boxplot<-ggplot(mtcars,aes(x = as.factor(cyl), y =mpg )) + geom_boxplot()
16 16 #my_boxplot
    No newline at end of file
15 15 my_boxplot<-ggplot(mtcars,aes(x = as.factor(cyl), y =mpg )) + geom_boxplot()
16 16 my_boxplot
    No newline at end of file
```



Collaborating

- Edit code locally then stage, commit and push to remote host

PUSH





Collaborating

- Edit code locally then stage, commit and push to remote host

Back to Github: <https://github.com/saidataanalytics/versioncontrol/branches>

Overview

Yours

Active

Stale

All branches

Search branches...

Default branch

main

Updated 4 days ago by kennedyshane10

Default

Change default branch

Active branches

demo

Updated 6 minutes ago by kennedyshane10

0 | 1

New pull request



Collaborating

- 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: main ▼



compare: demo ▼

✓ **Able to merge.** These branches can be automatically merged.

Showing 1 changed file with 2 additions and 2 deletions.

Create pull request

▼ 4 ■■■ sample_code.R

↑...

@@ -12,5 +12,5 @@ my_scatterplot<-ggplot(mtcars,aes(x = mpg, y = hp)) + geom_point()

12 12 my_scatterplot

13 13

14 14 #create a box and whiskers plot of cylinders versus mpg - updated

15 - #my_boxplot<-ggplot(mtcars,aes(x = as.factor(cyl), y =mpg)) + geom_boxplot()

16 - #my_boxplot ⊖

15 + my_boxplot<-ggplot(mtcars,aes(x = as.factor(cyl), y =mpg)) + geom_boxplot()

16 + my_boxplot ⊖



Collaborating

- Merge Pull request into base branch



This branch has no conflicts with the base branch

Merging can be performed automatically.

Merge pull request



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



Collaborating

- Create pull request with conflicts



base: main ▾



compare: demo ▾

✗ **Can't automatically merge.** Don't worry, you can still create the pull request.

[Create pull request](#) ▾

```
5 - #load built in mtcars dataset and view sample
6 - data(mtcars)
7 - head(mtcars)
8 - str(mtcars)
9 -
10 - #create a scatterplot of mpg versus hp
11 - my_scatterplot<-ggplot(mtcars,aes(x = mpg, y = hp)) + geom_point()
12 - my_scatterplot
13 -
14 - #create a box and whiskers plot of cylinders versus mpg - updated
15 - my_boxplot<-ggplot(mtcars,aes(x = as.factor(cyl), y =mpg )) + geom_boxplot()
16 - my_boxplot ⊖
```

```
5 + x <- VADeaths[1:3, "Rural Male"]
6 + barplot(x)
```



Collaborating

- Resolving conflicts

Add more commits by pushing to the **demo** branch on **saidataanalytics/versioncontrol**.



This branch has conflicts that must be resolved

Use the [web editor](#) or the [command line](#) to resolve conflicts.

Resolve conflicts

Conflicting files

sample_code.R

```
5 <<<<<<< demo
6 x <- VADeaths[1:3, "Rural Male"]
7 barplot(x)
8 |=====
9 #load built in mtcars dataset and view sample
10 data(mtcars)
11 head(mtcars)
12 str(mtcars)
13
14 #create a scatterplot of mpg versus hp
15 my_scatterplot<-ggplot(mtcars,aes(x = mpg, y = hp)) + geom_point()
16 my_scatterplot
17
18 #create a box and whiskers plot of cylinders versus mpg
19 my_boxplot<-ggplot(mtcars,aes(x = as.factor(cyl), y =mpg )) + geom_boxplot()
20 my_boxplot
21 >>>>>>> main
22
```



Collaborating

- Resolving conflicts

1 conflict

Prev ^

Next v



Mark as resolved

```
5 #load built in mtcars dataset and view sample
6 data(mtcars)
7 head(mtcars)
8 str(mtcars)
9
10 #create a scatterplot of mpg versus hp
11 my_scatterplot<-ggplot(mtcars,aes(x = mpg, y = hp)) + geom_point()
12 my_scatterplot
13
14 #create a box and whiskers plot of cylinders versus mpg
15 my_boxplot<-ggplot(mtcars,aes(x = as.factor(cyl), y =mpg )) + geom_boxplot()
16 my_boxplot
17
```

Edit code in Github to resolve conflicts, then commit and merge

[Commit merge](#)[Merge pull request](#)

- What is a version control system?
 - Directed Acyclic Graph
 - Local and Remote Hosting
- Relevance for Actuaries
 - Risk Management
 - Collaboration
 - Commercial considerations
- Demonstration
 - Getting set-up on a version control system (Git) and hosting service (Github)
 - Integrating with User Interface (R Studio)
 - Useful functions (create branch, stage, commit, push, pull request, resolve conflicts, merge)

Presentation and additional information available on github page:

<https://github.com/saidataanalytics/versioncontrol>



Society of Actuaries in Ireland

Thank you

© Society of Actuaries in Ireland