

Introduction to Git and GitHub

Natasha Watkins

January 23, 2018

Version control

git and GitHub are tools used for **version control**

Version control:

- ▶ tracks changes made to your work
- ▶ prevents you from losing work (for example, accidentally deleting a file)
- ▶ allows you to go back to an old version of you work (undo changes)

What is Git and GitHub?

Git is a widely used, open source **version control system**


- ▶ Does not require an internet connection to use
- ▶ Usually used with the **command prompt** - there are also desktop applications (such as GitHub Desktop)


GitHub is a (mostly) free hosting service for your Git repositories


- ▶ Makes it simple to collaborate and share code/documents on the internet
- ▶ Acts as an online backup of your project


Projects using GitHub


`https://github.com/bitcoin/bitcoin`


 bitcoin / bitcoin


 Watch ▾ 2,740


 ★ Star 25,581


 🍴 Fork 14,916

 <> Code

 ⓘ Issues 577


 🔄 Pull requests 279


 📁 Projects 7


 📊 Insights


Bitcoin Core integration/staging tree <https://bitcoin.org/en/download>


bitcoin c-plus-plus p2p cryptocurrency cryptography

 15,877 commits

 9 branches

 186 releases

 503 contributors

 MIT

Branch: master ▾



New pull request

Create new file







Upload files

Find file

Clone or download ▾

 **MarcoFalke** Merge #12177: trivial: fix address_type help text of getnewaddress an... 

Latest commit cad504b 16 hours ago

 .github	Make default issue text all comments to make issues more readable	2 months ago
 .tx	qt: Set transifex slug to 0.14	a year ago
 build-aux/m4	Explicitly search for bdb5.3.	7 months ago
 contrib	Merge #12063: [Trivial] Update license year range to 2018	13 days ago
 depends	Merge #11903: [trivial] Add required package dependencies for depends...	25 days ago
 doc	Merge #12112: Docs: Remove the ending slashes from RPC URI format.	6 days ago

Projects using GitHub

<https://github.com/python/cpython>

python / cpython

Watch

782

★ Star

15,422

Fork

3,977

Code

Pull requests 593

Insights

The Python programming language <https://www.python.org/>

100,784 commits

6 branches

342 releases

462 contributors

Branch: master

New pull request

Create new file

Upload files

Find file

Clone or download



tiran bpo-32549: Compile OpenSSL 1.1.0 on Travis CI (#5180)

Latest commit ced9cbs 9 hours ago

.github

Fix version in AppVeyor config (GH-5124)

9 days ago

Doc

bpo-32410: Implement loop.sock_sendfile() (#4976)

11 hours ago

Grammar

bpo-30406: Make async and await proper keywords (#1669)

3 months ago

Include

bpo-32544: Speed up hasattr() and getattr() (GH-5173)

17 hours ago

Lib

bpo-32403: Faster date and datetime constructors (#4993)

11 hours ago

Mac

advance copyright years to 2018 (#5094)

12 days ago


Misc




bpo-32549: Compile OpenSSL 1.1.0 on Travis CI (#5180)







9 hours ago

Projects using GitHub






<https://github.com/QuantEcon/QuantEcon.py>

 **QuantEcon / QuantEcon.py**


 Unwatch ▾ 138  Unstar 573  Fork 745







 Code  Issues 53  Pull requests 5  Projects 0  Wiki  Insights

A community based Python library for quantitative economics http://quantecon.org/python_index.html

 1,177 commits  10 branches  25 releases  18 contributors  BSD-3-Clause

Branch: master ▾

 **mmcky** Merge pull request #384 from QuantEcon/payoff_arrays ... Latest commit f8dc498 2 days ago

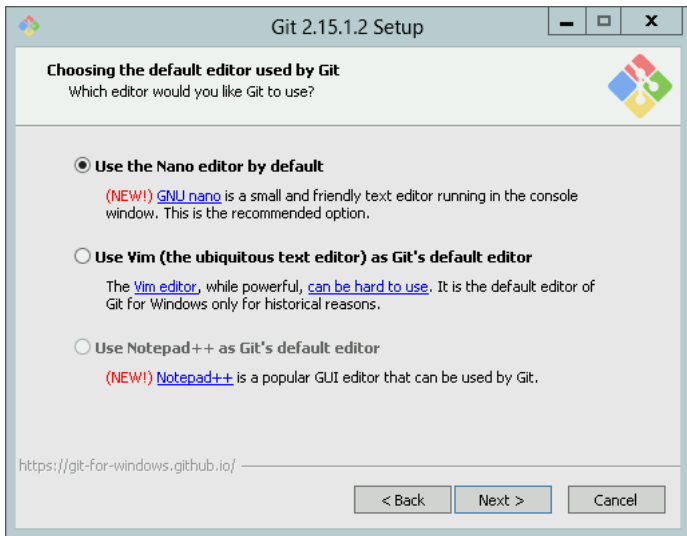
 data	Major reorganization, nearly complete	4 years ago
 docs	remove models subpackage from QuantEcon.py	7 days ago
 quantecon	Merge pull request #384 from QuantEcon/payoff_arrays	2 days ago
 .coveragerc	Add simplex_grid and simplex_index (#344)	4 months ago
 .coveralls.yml	DOC: fixed readthedocs	3 years ago
 .gitignore	.gitignore: Add Numba cache extensions	a year ago

What will we use Git for?

- ▶ Git will be used to access course materials and submit assignments during the Shenzhen Winter Camp
- ▶ Learning Git and GitHub will also be very useful for your own projects, such as a thesis or group assignment

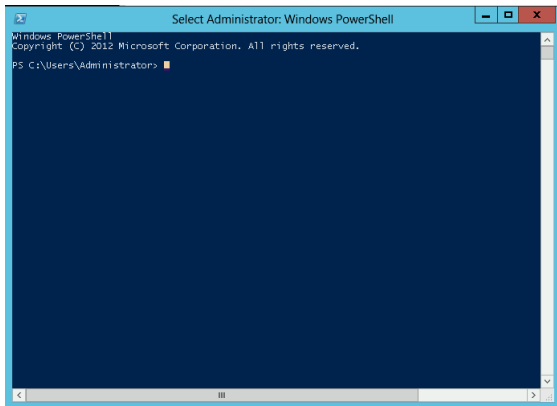
Let's install Git for Windows

1. Go to <https://git-scm.com/download/win>
2. The download should start automatically
3. Open the .exe file and start setup
4. Follow the setup instructions and leave the default selections
5. **Recommended:** Change the text editor to **nano** if you are unfamiliar with **vim**



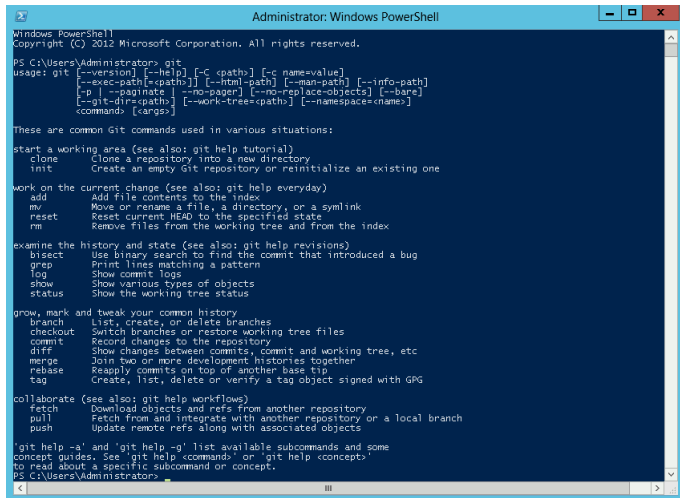
Setting up Git

Now that we have Git installed, open up **Windows PowerShell**



Setting up Git

Check that Git is working by typing `git` and hitting enter



```
Administrator: Windows PowerShell
Windows PowerShell
Copyright (C) 2012 Microsoft Corporation. All rights reserved.

PS C:\Users\Administrator> git
usage: git [--version] [--help] [-C <path>] [-c name=value]
       [--exec-path<=path>] [--html-path] [--man-path] [--info-path]
       [-p | --paginate] [--no-pager] [--no-replace-objects] [--bare]
       [--git-dir=<path>] [--work-tree=<path>] [--namespace=<name>]
       <command> [<args>]

These are common Git commands used in various situations:

start a working area (see also: git help tutorial)
  clone      Clone a repository into a new directory
  init       Create an empty Git repository or reinitialize an existing one

work on the current change (see also: git help everyday)
  add        Add file contents to the index
  mv         Move or rename a file, a directory, or a symlink
  reset      Reset current HEAD to the specified state
  rm         Remove files from the working tree and from the index

examine the history and state (see also: git help revisions)
  bisect     Use binary search to find the commit that introduced a bug
  grep       Print lines matching a pattern
  log        Show commit logs
  show       Show various types of objects
  status     Show the working tree status

grow, mark and tweak your common history
  branch     List, create, or delete branches
  checkout   Switch branches or restore working tree files
  commit     Record changes to the repository
  diff       Show changes between commits, commit and working tree, etc
  merge      Join two or more development histories together
  rebase     Reapply commits on top of another base tip
  tag        Create, list, delete or verify a tag object signed with GPG

collaborate (see also: git help workflows)
  fetch      Download objects and refs from another repository
  pull       Fetch from and integrate with another repository or a local branch
  push       Update remote refs along with associated objects

'git help -a' and 'git help -g' list available subcommands and some
concept guides. See 'git help <command>' or 'git help <concept>'
to read about a specific subcommand or concept.
PS C:\Users\Administrator>
```

Setting up Git

We need to configure Git to recognize us as the user

In the **prompt**, type:

- ▶ `git config --global user.name "Your Name"`
- ▶ `git config --global user.email "email@web.com"`

Note: Please make sure you have access to this email!

Creating a GitHub account

We will also need to set up an account on GitHub

1. Open `https://github.com/`
2. Sign up for GitHub using the same email you entered into the prompt

Using a Git repository

A **repository** is like a folder that contains your project's files, as well as the history of changes to the files

Instead of creating a new repository, we will be downloading (or **cloning**) one from QuantEcon

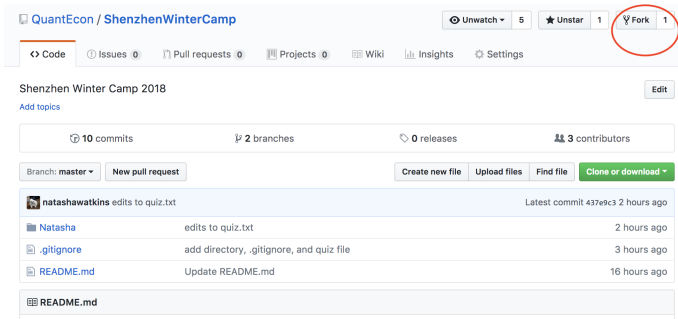
The repository is located at

`https://github.com/QuantEcon/ShenzhenWinterCamp`

Forking a repository

You will first need to **fork** the ShenzhenWinterCamp repository

- ▶ **Forking** means to copy a repository into your own GitHub account
- ▶ Go to the ShenzhenWinterCamp repository and click the fork button on the upper right



Forking a repository

You should now have a repository called `ShenzhenWinterCamp` in your **own** GitHub account

The screenshot shows a GitHub repository page for 'natashawatkins / ShenzhenWinterCamp'. The repository is a fork of 'QuantEcon/ShenzhenWinterCamp'. The page includes navigation tabs for Code, Pull requests (0), Projects (0), Wiki, Insights, and Settings. Below the repository name, it says 'Shenzhen Winter Camp 2018' with an 'Add topics' link and an 'Edit' button. A summary bar shows 10 commits, 2 branches, 0 releases, and 3 contributors. Below this, there are buttons for 'Branch: master', 'New pull request', 'Create new file', 'Upload files', 'Find file', and 'Clone or download'. The commit history section shows three commits: 'natashawatkins edits to quiz.txt' (2 hours ago), 'Natasha edits to quiz.txt' (2 hours ago), and '.gitignore add directory, .gitignore, and quiz file' (3 hours ago). The README.md file was updated 16 hours ago.

[natashawatkins / ShenzhenWinterCamp](#)
forked from QuantEcon/ShenzhenWinterCamp

Watch 0 Star 0 Fork 2

<> Code Pull requests 0 Projects 0 Wiki Insights Settings

Shenzhen Winter Camp 2018 [Add topics](#) [Edit](#)

10 commits 2 branches 0 releases 3 contributors

Branch: master New pull request Create new file Upload files Find file Clone or download

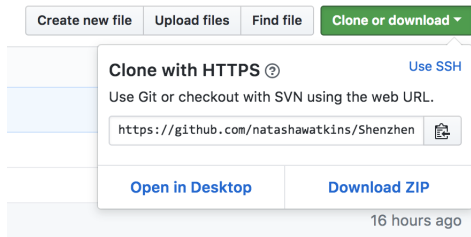
This branch is even with QuantEcon:master. [Pull request](#) [Compare](#)

natashawatkins	edits to quiz.txt	Latest commit 437e9c3 2 hours ago
Natasha	edits to quiz.txt	2 hours ago
.gitignore	add directory, .gitignore, and quiz file	3 hours ago
README.md	Update README.md	16 hours ago

Cloning a Git repository

Next we need to copy (or **clone**) the repository to your local computer

1. On your repository's GitHub page, click the **Clone or download** button and copy the url



2. Navigate to somewhere on your computer (maybe the Desktop) where you would like to save the folder

```
cd ~/Desktop
```

Using a Git repository

3. In the **prompt**, type

```
git clone
```

and paste the url (to paste, right click in the PowerShell window)

4. You should now see a folder called ShenzhenWinterCamp on your Desktop

Using a Git repository

5. Navigate to the repository using the **prompt**

```
cd ~/Desktop/ShenzhenWinterCamp
```

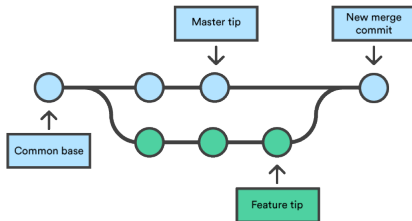
Helpful prompt commands:

- ▶ `cd` - change directory
- ▶ `cd ..` - move back one level in directory
- ▶ `pwd` - print working directory
- ▶ `ls` - list files in working directory

Creating a branch

Before we start editing files in the repository, I want you to set up a **branch** in the repository

- ▶ A **branch** allows you to make changes to your files without having to worry about the main version
- ▶ Used when adding new features to your project
- ▶ When you are finished, changes in the branch are **merged** into the main version¹



¹ Image source: <https://www.atlassian.com/git/tutorials/using-branches/git-merge>

Creating a branch

Create a branch called `your-name` by typing the following command in the prompt

```
git checkout -b "YOUR-NAME"
```

- ▶ `-b` is used to create a new branch

Exercise: Quiz!

...and adding/editing files in a Git repository

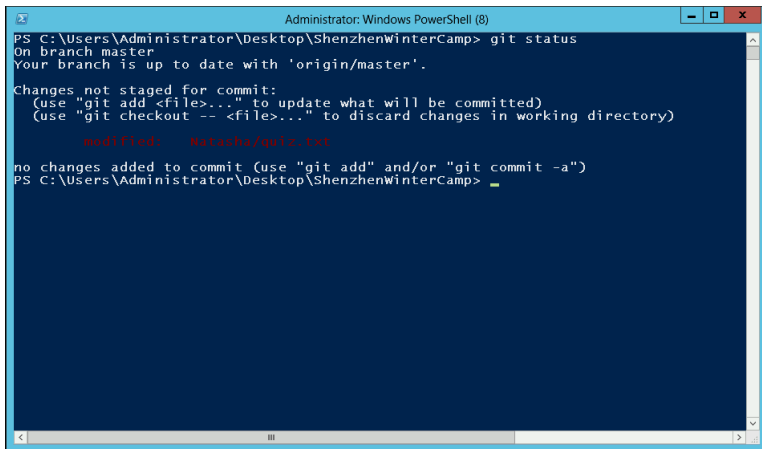
Task: Open `Natasha/quiz.txt` and complete the quiz

- ▶ You may use the internet to help you answer the questions
- ▶ When you are finished, save the file

Using a Git repository

Git is aware that you have modified a file in the repository

Try typing `git status`



```
Administrator: Windows PowerShell (8)
PS C:\Users\Administrator\Desktop\ShenzhenWinterCamp> git status
On branch master
Your branch is up to date with 'origin/master'.

Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git checkout -- <file>..." to discard changes in working directory)

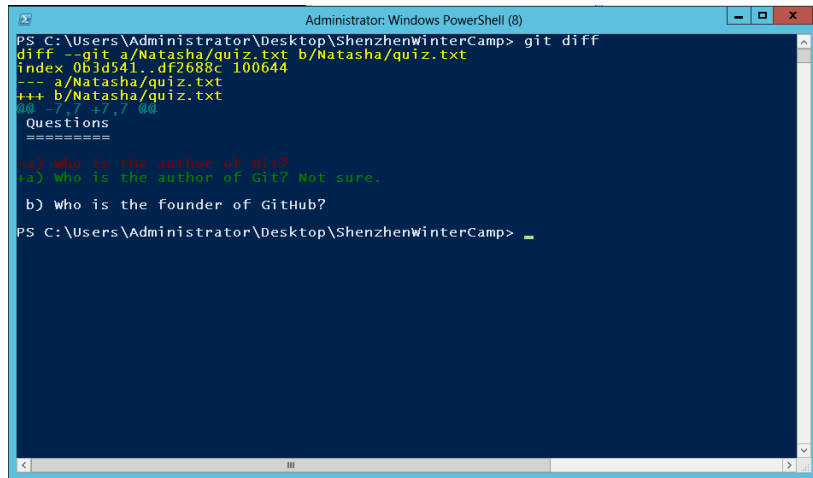
        modified:   Natasha/guiz.txt

no changes added to commit (use "git add" and/or "git commit -a")
PS C:\Users\Administrator\Desktop\ShenzhenWinterCamp> 
```

Using a Git repository

You can view the changes you made to quiz.txt in the prompt

Try typing `git diff`



```
Administrator: Windows PowerShell (8)
PS C:\Users\Administrator\Desktop\ShenzhenWinterCamp> git diff
diff --git a/Natasha/quiz.txt b/Natasha/quiz.txt
index 0b3d541..df2688c 100644
--- a/Natasha/quiz.txt
+++ b/Natasha/quiz.txt
@@ -7,7 +7,7 @@
 Questions
 =====
-a) Who is the author of Git?
+a) Who is the author of Git? Not sure.

 b) Who is the founder of GitHub?
PS C:\Users\Administrator\Desktop\ShenzhenWinterCamp> _
```


Staging changes in the repository

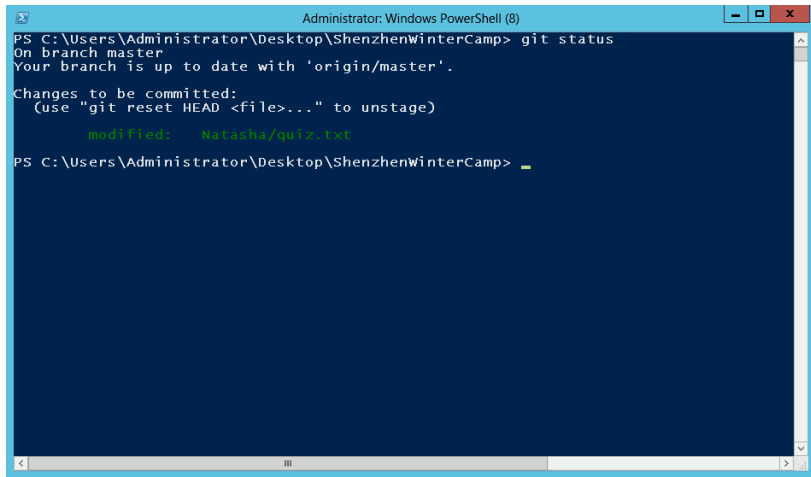
- ▶ Once you have finished an edit, you want to update Git's latest “snapshot” of your repository
- ▶ First you should move changes into the **staging area**, where edits are organized before they are **committed**
- ▶ Type

```
git add Natasha/quiz.txt
```

to add the file to the staging area

Staging changes in the repository

- ▶ Type `git status` to check you have added the file you want to the staging area



```
Administrator: Windows PowerShell (8)
PS C:\Users\Administrator\Desktop\ShenzhenWinterCamp> git status
On branch master
Your branch is up to date with 'origin/master'.

Changes to be committed:
  (use "git reset HEAD <file>..." to unstage)

    modified:   Natasha/quiz.txt

PS C:\Users\Administrator\Desktop\ShenzhenWinterCamp> 
```

Committing changes to your repository

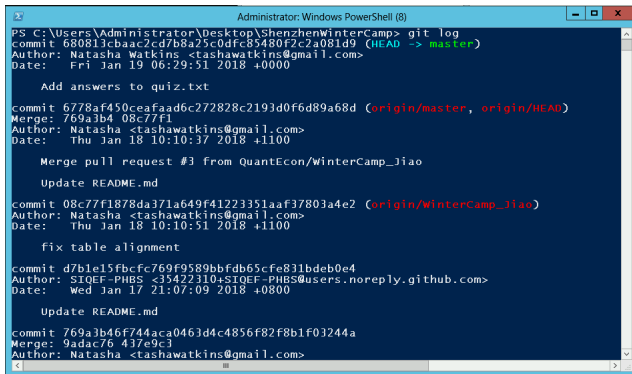
- ▶ If you are happy with the changes that are staged, you should now **commit** them to your repository
- ▶ **Commits** are permanent snapshots of your repository
- ▶ A descriptive message of the changes made should accompany a commit

```
git commit -m "Add answers to quiz.txt"
```

Viewing the commit history

- ▶ Commit messages are useful when looking through the history of commits to the repository
- ▶ Type the following command to view the log

git log



```
Administrator: Windows PowerShell (8)
PS C:\Users\Administrator\Desktop\ShenzhenWinterCamp> git log
commit 680813cbaac2cd7b8a25c0dfc85480f2c2a081d9 (HEAD -> master)
Author: Natasha Watkins <tashawatkins@gmail.com>
Date:   Fri Jan 19 06:29:51 2018 +0000

    Add answers to quiz.txt

commit 6778af450ceafaad6c272828c2193d0f6d89a68d (origin/master, origin/HEAD)
Merge: 769a1b4 08c77f1
Author: Natasha <tashawatkins@gmail.com>
Date:   Thu Jan 18 10:10:37 2018 +1100

    Merge pull request #3 from QuantEcon/WinterCamp_jiao

    Update README.md

commit 08c77f1878da371a649f41223351aaf37803a4e2 (origin/WinterCamp_jiao)
Author: Natasha <tashawatkins@gmail.com>
Date:   Thu Jan 18 10:10:51 2018 +1100

    fix table alignment

commit d7b1e15fbcfc769f9589bbfdb65cfe831bdeb0e4
Author: SIQEF-PHBS <35422310+SIQEF-PHBS@users.noreply.github.com>
Date:   Wed Jan 17 21:07:09 2018 +0800

    Update README.md

commit 769a1b46f744aca0463d4c4856f82f8b1f03244a
Merge: 9adac76 437e9c3
Author: Natasha <tashawatkins@gmail.com>
```

Pushing changes to GitHub

Now we're going to correct each other's quiz answers!

- ▶ You will make a **pull request** on GitHub with your quiz answers committed to your branch
- ▶ A pull request asks the maintainer of the repository to **merge** the changes into the main version (`master`)
- ▶ First we need to **push** (upload) the changes to GitHub

```
git push origin <branch-name>
```

Making a pull request (PR)


- ▶ Navigate to the **original** repository:
`https://github.com/QuantEcon/ShenzhenWinterCamp`
- ▶ Click **Compare & pull request**

The screenshot shows the GitHub repository page for `QuantEcon / ShenzhenWinterCamp`. The repository has 5 unwatchers, 1 unstar, and 2 forks. The navigation bar includes links for Code, Issues (0), Pull requests (0), Projects (0), Wiki, Insights, and Settings. The repository name is `Shenzhen Winter Camp 2018`, with an `Edit` button. Below the repository name, it shows 12 commits, 2 branches, 0 releases, and 3 contributors. A section titled 'Your recently pushed branches:' lists a branch `natashawatkins:natasha-watkins` pushed 1 minute ago. A green button labeled `Compare & pull request` is circled in red. Below this, there are buttons for `Branch: master`, `New pull request`, `Create new file`, `Upload files`, `Find file`, and `Clone or download`. The pull request list shows a merge of branch `'master'` of `https://github.com/QuantEcon/ShenzhenWinterCamp` by `natashawatkins` 7 minutes ago. The list also includes files `.gitignore` (updated 4 hours ago) and `README.md` (updated 17 hours ago).

Making 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](#).

 base fork: QuantEcon/ShenzhenWinter... ▾

base: master ▾

 ←

head fork: natashawatkins/ShenzhenW... ▾

compare: natasha-watkins ▾



Add Natasha's Quiz Answers



Write

Preview

AA ▾ B i



- This PR adds my quiz answers to `quiz.txt`

Attach files by dragging & dropping, [selecting them](#), or pasting from the clipboard.

☒ Allow edits from maintainers. [Learn more](#)

Create pull request

Reviewers



No reviews—request one

Assignees



No one—assign yourself

Labels



None yet

Projects



None yet

Milestone

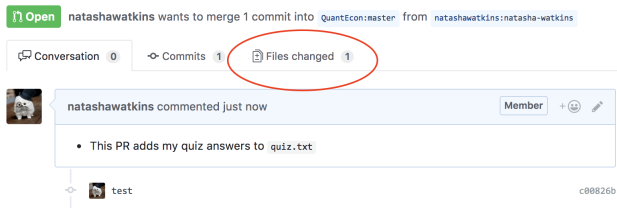


No milestone

Exercise: Make a PR and comment on other PRs

1. Make a pull request to the original repository with your quiz answers
2. Give a score to the quiz of the person sitting next to you - go to their PR and view changes

Add Natasha's Quiz Answers #2



The screenshot shows a GitHub Pull Request interface. At the top, it says 'natashawatkins wants to merge 1 commit into QuantEcon:master from natashawatkins:natasha-watkins'. Below this, there are three tabs: 'Conversation' (0), 'Commits' (1), and 'Files changed' (1). The 'Files changed' tab is selected and circled in red. Below the tabs, there is a comment from 'natashawatkins' posted 'just now'. The comment text is 'This PR adds my quiz answers to quiz.txt'. The comment is from a 'Member' and has icons for reactions and editing. At the bottom of the comment, there is a 'test' label and a commit hash 'c00826b'.

3. Add the score in a comment on the pull request

More resources

- ▶ Version Control with Git
(<https://swcarpentry.github.io/git-novice/>)
- ▶ Git cheatsheet from Atlassian (<https://www.atlassian.com/git/tutorials/atlassian-git-cheatsheet>)