NTNU
Norwegian University of
Science and Technology

Version control for researchers

Sigurd Hofsmo Jakobsen

Department of electric power engineering

June 21, 2018

Outline

Taking over some old code and fixing it

Version control using git

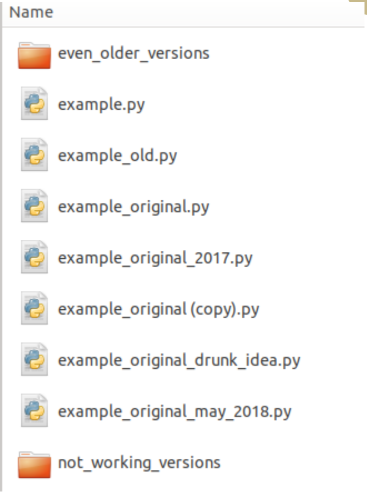
Making the code more readable

More topics to be considered

Discussion

Presentation of the case

— Some master students are going to continue the work of a PhD



Name

even_older_versions

example.py

example_old.py

example_original.py

example_original_2017.py

example_original (copy).py

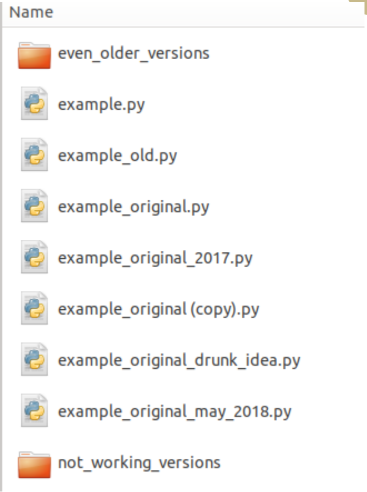
example_original_drunk_idea.py

example_original_may_2018.py

not_working_versions

Presentation of the case

- Some master students are going to continue the work of a PhD
- They received several files on a USB stick (don't tell the it guy)



Name

even_older_versions

example.py

example_old.py

example_original.py

example_original_2017.py

example_original (copy).py

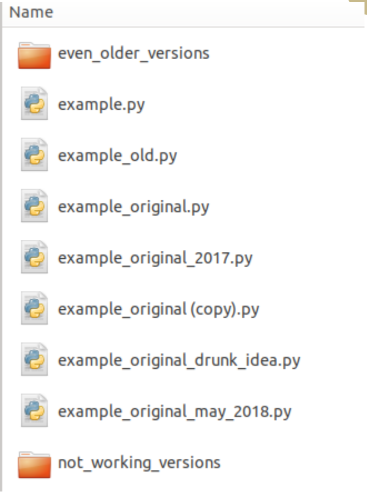
example_original_drunk_idea.py

example_original_may_2018.py

not_working_versions

Presentation of the case

- Some master students are going to continue the work of a PhD
- They received several files on a USB stick (don’t tell the it guy)
- There are no explanation on what the files do

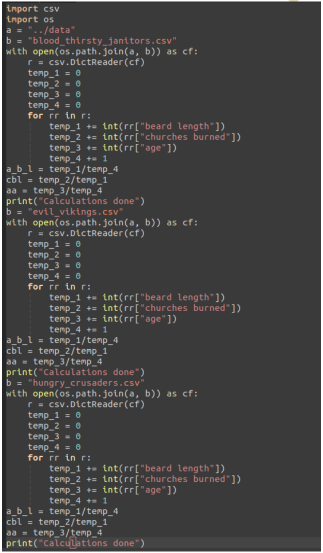


Name

even_older_versions
example.py
example_old.py
example_original.py
example_original_2017.py
example_original (copy).py
example_original_drunk_idea.py
example_original_may_2018.py
not_working_versions

Presentation of the case

- Some master students are going to continue the work of a PhD
- They received several files on a USB stick (don’t tell the it guy)
- There are no explanation on what the files do
- Looking at one of the files doesn’t help



```
import csv
import os
a = "../data"
b = "blood_thirsty_janitors.csv"
with open(os.path.join(a, b)) as cf:
    r = csv.DictReader(cf)
    temp_1 = 0
    temp_2 = 0
    temp_3 = 0
    temp_4 = 0
    for rr in r:
        temp_1 += int(rr["beard length"])
        temp_2 += int(rr["churches burned"])
        temp_3 += int(rr["age"])
        temp_4 += 1
a_b_l = temp_1/temp_4
cbl = temp_2/temp_1
aa = temp_3/temp_4
print("Calculations done")
b = "evil_vikings.csv"
with open(os.path.join(a, b)) as cf:
    r = csv.DictReader(cf)
    temp_1 = 0
    temp_2 = 0
    temp_3 = 0
    temp_4 = 0
    for rr in r:
        temp_1 += int(rr["beard length"])
        temp_2 += int(rr["churches burned"])
        temp_3 += int(rr["age"])
        temp_4 += 1
a_b_l = temp_1/temp_4
cbl = temp_2/temp_1
aa = temp_3/temp_4
print("Calculations done")
b = "hungry_crusaders.csv"
with open(os.path.join(a, b)) as cf:
    r = csv.DictReader(cf)
    temp_1 = 0
    temp_2 = 0
    temp_3 = 0
    temp_4 = 0
    for rr in r:
        temp_1 += int(rr["beard length"])
        temp_2 += int(rr["churches burned"])
        temp_3 += int(rr["age"])
        temp_4 += 1
a_b_l = temp_1/temp_4
cbl = temp_2/temp_1
aa = temp_3/temp_4
print("Calculations done")
```

Problems which have to be solved

- How do they share the code with each other?

Problems which have to be solved

- How do they share the code with each other?
  - With version control.

Problems which have to be solved

— How do they share the code with each other?
  - With version control.

— How do they keep track of the old versions?

Problems which have to be solved

- How do they share the code with each other?
  - With version control.
- How do they keep track of the old versions?
  - With version control.

Problems which have to be solved

- How do they share the code with each other?
  - With version control.
- How do they keep track of the old versions?
  - With version control.
- How do they make the code readable

Problems which have to be solved

— How do they share the code with each other?
  - With version control.

— How do they keep track of the old versions?
  - With version control.

— How do they make the code readable
  - Implement and follow code style guidelines

Problems which have to be solved

— How do they share the code with each other?
  - With version control.

— How do they keep track of the old versions?
  - With version control.

— How do they make the code readable
  - Implement and follow code style guidelines

— How do prevent copy paste?

Problems which have to be solved

— How do they share the code with each other?
  - With version control.

— How do they keep track of the old versions?
  - With version control.

— How do they make the code readable
  - Implement and follow code style guidelines

— How do prevent copy paste?
  - Create functions.

Outline

Taking over some old code and fixing it

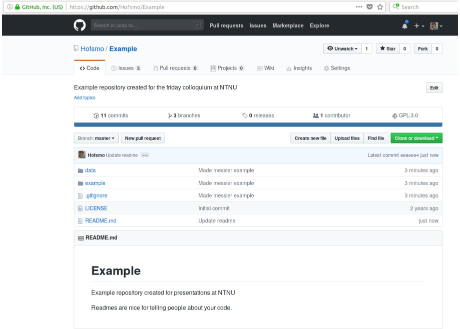
Version control using git

Making the code more readable

More topics to be considered

Discussion

- The code can easily be shared online on GitHub



Example repository created for the friday colloquium at NTNU

11 commits · 3 branches · 0 releases · 1 contributor · GPL-3.0

Hofsmo Update readme … Latest commit 068e654 just now

data	Made messier example	3 minutes ago
example	Made messier example	3 minutes ago
.gitignore	Made messier example	3 minutes ago
LICENSE	Initial commit	2 years ago
README.md	Update readme	just now

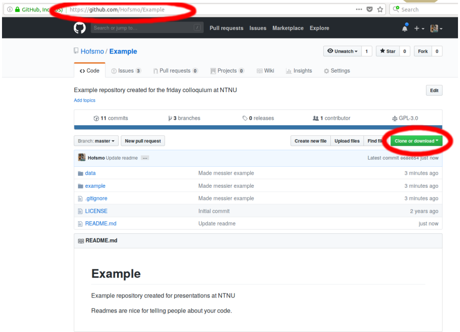
README.md

Example

Example repository created for presentations at NTNU

Readmes are nice for telling people about your code.

- The code can easily be shared online on GitHub
- Everyone can access and download the code



Example repository created for the friday colloquium at NTNU

Hofsmo Update readme … Latest commit e68e654 just now

data	Made messier example	3 minutes ago
example	Made messier example	3 minutes ago
.gitignore	Made messier example	3 minutes ago
LICENSE	Initial commit	2 years ago
README.md	Update readme	just now

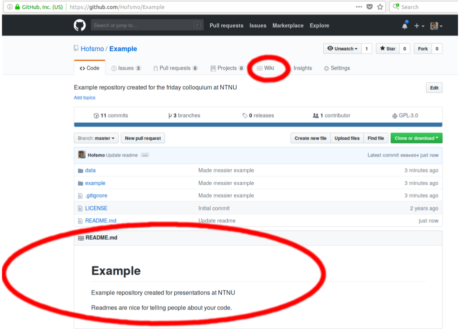
README.md

Example

Example repository created for presentations at NTNU

Readmes are nice for telling people about your code.

- The code can easily be shared online on GitHub
- Everyone can access and download the code
- Readme and wiki for documenting the code



Example repository created for the friday colloquium at NTNU

Hofsmo Update readme ... Latest commit 068e654 just now

data	Made messier example	3 minutes ago
example	Made messier example	3 minutes ago
.gitignore	Made messier example	3 minutes ago
LICENSE	Initial commit	2 years ago
README.md	Update readme	just now

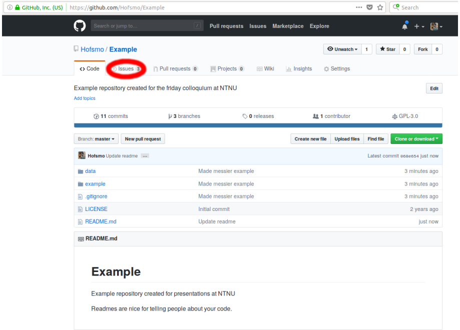
README.md

Example

Example repository created for presentations at NTNU

Readmes are nice for telling people about your code.

- The code can easily be shared online on GitHub
- Everyone can access and download the code
- Readme and wiki for documenting the code
- Issue tracker for reporting bugs and discussing the code



Example repository created for the friday colloquium at NTNU

Hofsmo Update readme … Latest commit e68e654 just now

data	Made messier example	3 minutes ago
example	Made messier example	3 minutes ago
.gitignore	Made messier example	3 minutes ago
LICENSE	Initial commit	2 years ago
README.md	Update readme	just now

README.md

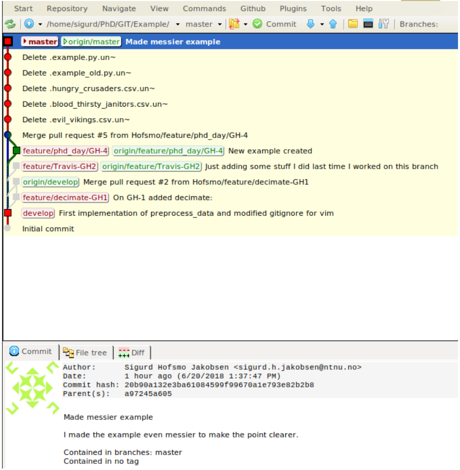
Example

Example repository created for presentations at NTNU

Readmes are nice for telling people about your code.

Getting rid of the old files using git

- Download the GitHub repository using a GUI or the terminal.



```
master origin/master Made messier example
Delete .example.py.un~
Delete .example_old.py.un~
Delete .hungry_crusaders.csv.un~
Delete .blood_thirsty_janitors.csv.un~
Delete .evil_vikings.csv.un~
Merge pull request #5 from Hofsmo/feature/phd_day/GH-4
feature/phd_day/GH-4 origin/feature/phd_day/GH-4 New example created
feature/Travis-GH2 origin/feature/Travis-GH2 Just adding some stuff I did last time I worked on this branch
origin/develop Merge pull request #2 from Hofsmo/feature/decimate-GH1
feature/decimate-GH1 On GH-1 added decimate:
develop First implementation of preprocess_data and modified gitignore for vim
Initial commit
```

```
Author:       Sigurd Hofsmo Jakobsen <sigurd.h.jakobsen@ntnu.no>
Date:         1 hour ago (6/20/2018 1:37:47 PM)
Commit hash:  20b90a132e3ba61084599f99670a1e793e82b2b8
Parent(s):    a97245a605
```

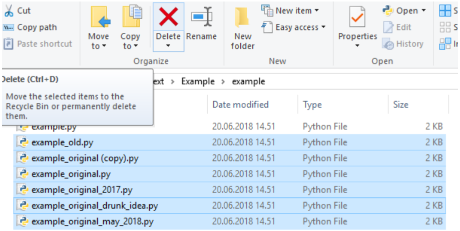
Made messier example

I made the example even messier to make the point clearer.

Contained in branches: master
Contained in no tag

Getting rid of the old files using git

- Download the GitHub repository using a GUI or the terminal.
- Find the files and delete them.



Delete (Ctrl+D)

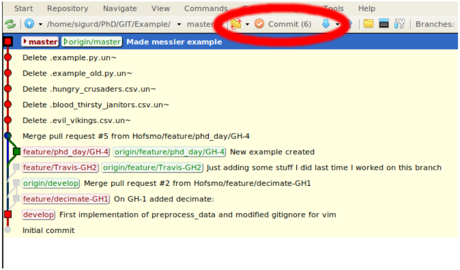
Move the selected items to the Recycle Bin or permanently delete them.

ext › Example › example

	Date modified	Type	Size
example.py	20.06.2018 14.51	Python File	2 KB
example_old.py	20.06.2018 14.51	Python File	2 KB
example_original (copy).py	20.06.2018 14.51	Python File	2 KB
example_original.py	20.06.2018 14.51	Python File	2 KB
example_original_2017.py	20.06.2018 14.51	Python File	2 KB
example_original_drunk_idea.py	20.06.2018 14.51	Python File	2 KB
example_original_may_2018.py	20.06.2018 14.51	Python File	2 KB

Getting rid of the old files using git

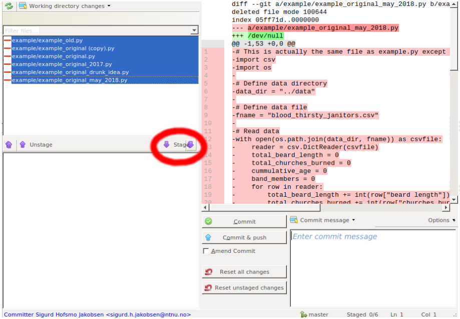
- Download the GitHub repository using a GUI or the terminal.
- Find the files and delete them.
- The GUI informs you that changes have been made.



master origin/master Made messier example
Delete .example.py.un~
Delete .example_old.py.un~
Delete .hungry_crusaders.csv.un~
Delete .blood_thirsty_janitors.csv.un~
Delete .evil_vikings.csv.un~
Merge pull request #5 from Hofsmo/feature/phd_day/GH-4
feature/phd_day/GH-4 origin/feature/phd_day/GH-4 New example created
feature/Travis-GH2 origin/feature/Travis-GH2 Just adding some stuff I did last time I worked on this branch
origin/develop Merge pull request #2 from Hofsmo/feature/decimate-GH1
feature/decimate-GH1 On GH-1 added decimate:
develop First implementation of preprocess_data and modified gitignore for vim
Initial commit

Getting rid of the old files using git

- Download the GitHub repository using a GUI or the terminal.
- Find the files and delete them.
- The GUI informs you that changes have been made.
- The changes have to be staged.



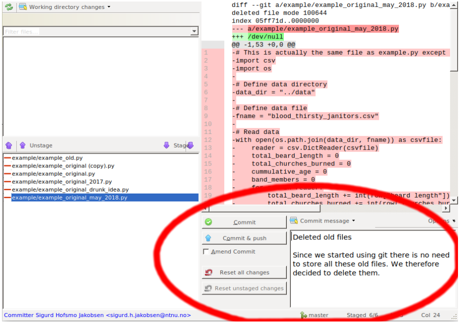
```
example/example_old.py
example/example_original (copy).py
example/example_original.py
example/example_original_2017.py
example/example_original_drunk_idea.py
example/example_original_may_2018.py
```

```
diff --git a/example/example_original_may_2018.py b/exa
deleted file mode 100644
index 05ff71d..0000000
--- a/example/example_original_may_2018.py
+++ /dev/null
@@ -1,53 +0,0 @@
-# This is actually the same file as example.py except
-import csv
-import os
-
-# Define data directory
-data_dir = "../data"
-
-# Define data file
-fname = "blood_thirsty_janitors.csv"
-
-# Read data
-with open(os.path.join(data_dir, fname)) as csvfile:
-    reader = csv.DictReader(csvfile)
-    total_beard_length = 0
-    total_churches_burned = 0
-    cummulative_age = 0
-    band_members = 0
-    for row in reader:
-        total_beard_length += int(row["beard length"])
-        total_churches_burned += int(row["churches_bur
```

Committer Sigurd Hofsmo Jakobsen <sigurd.h.jakobsen@ntnu.no>   master   Staged 0/6   Ln 1   Col 1

Getting rid of the old files using git

- Download the GitHub repository using a GUI or the terminal.
- Find the files and delete them.
- The GUI informs you that changes have been made.
- The changes have to be staged.
- Write a commit message and commit the change.



example/example_old.py
example/example_original (copy).py
example/example_original.py
example/example_original_2017.py
example/example_original_drunk_idea.py
example/example_original_may_2018.py

```
diff --git a/example/example_original_may_2018.py b/exa
deleted file mode 100644
index 05ff71d..0000000
--- a/example/example_original_may_2018.py
+++ /dev/null
@@ -1,53 +0,0 @@
-# This is actually the same file as example.py except
-import csv
-import os
-
-# Define data directory
-data_dir = "../data"
-
-# Define data file
-fname = "blood_thirsty_janitors.csv"
-
-# Read data
-with open(os.path.join(data_dir, fname)) as csvfile:
-    reader = csv.DictReader(csvfile)
-    total_beard_length = 0
-    total_churches_burned = 0
-    cummulative_age = 0
-    band_members = 0
-    for row in reader:
-        total_beard_length += int(row["beard length"])
-        total_churches_burned += int(row["churches bur
```

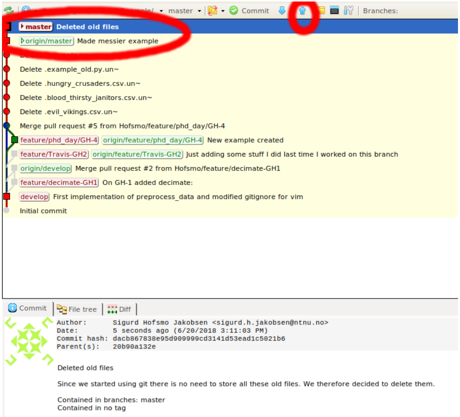
Deleted old files

Since we started using git there is no need to store all these old files. We therefore decided to delete them.

Committer Sigurd Hofsmo Jakobsen <sigurd.h.jakobsen@ntnu.no>   master   Staged 6/6   Col 24

Getting rid of the old files using git

- Download the GitHub repository using a GUI or the terminal.
- Find the files and delete them.
- The GUI informs you that changes have been made.
- The changes have to be staged.
- Write a commit message and commit the change.
- Our local version is now ahead of the GitHub version and the changes can be pushed.

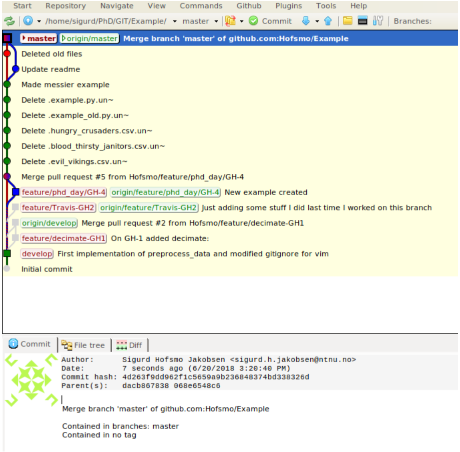


```
master Deleted old files
origin/master Made messier example
Delete .example_old.py.un~
Delete .hungry_crusaders.csv.un~
Delete .blood_thirsty_janitors.csv.un~
Delete .evil_vikings.csv.un~
Merge pull request #5 from Hofsmo/feature/phd_day/GH-4
feature/phd_day/GH-4 origin/feature/phd_day/GH-4 New example created
feature/Travis-GH2 origin/feature/Travis-GH2 Just adding some stuff I did last time I worked on this branch
origin/develop Merge pull request #2 from Hofsmo/feature/decimate-GH1
feature/decimate-GH1 On GH-1 added decimate:
develop First implementation of preprocess_data and modified gitignore for vim
Initial commit
```

```
Author:      Sigurd Hofsmo Jakobsen <sigurd.h.jakobsen@ntnu.no>
Date:        5 seconds ago (6/20/2018 3:11:03 PM)
Commit hash: dacb867838e95d909999cd3141d53ead1c5021b6
Parent(s):   20b90a132e

Deleted old files

Since we started using git there is no need to store all these old files. We therefore decided to delete them.

Contained in branches: master
Contained in no tag
```

Getting rid of the old files using git

- Download the GitHub repository using a GUI or the terminal.
- Find the files and delete them.
- The GUI informs you that changes have been made.
- The changes have to be staged.
- Write a commit message and commit the change.
- Our local version is now ahead of the GitHub version and the changes can be pushed.
- The versions are now identical



```
master  origin/master  Merge branch 'master' of github.com:Hofsmo/Example
Deleted old files
Update readme
Made messier example
Delete .example.py.un~
Delete .example_old.py.un~
Delete .hungry_crusaders.csv.un~
Delete .blood_thirsty_janitors.csv.un~
Delete .evil_vikings.csv.un~
Merge pull request #5 from Hofsmo/feature/phd_day/GH-4
feature/phd_day/GH-4  origin/feature/phd_day/GH-4  New example created
feature/Travis-GH2  origin/feature/Travis-GH2  Just adding some stuff I did last time I worked on this branch
origin/develop  Merge pull request #2 from Hofsmo/feature/decimate-GH1
feature/decimate-GH1  On GH-1 added decimate:
develop  First implementation of preprocess_data and modified gitignore for vim
Initial commit
```

```
Author:       Sigurd Hofsmo Jakobsen <sigurd.h.jakobsen@ntnu.no>
Date:         7 seconds ago (6/20/2018 3:20:40 PM)
Commit hash:  4d263f9dd962f1c5659a9b236848374bd338326d
Parent(s):    dacb867838 068e6548c6

Merge branch 'master' of github.com:Hofsmo/Example

Contained in branches: master
Contained in no tag
```

Outline

Taking over some old code and fixing it

Version control using git

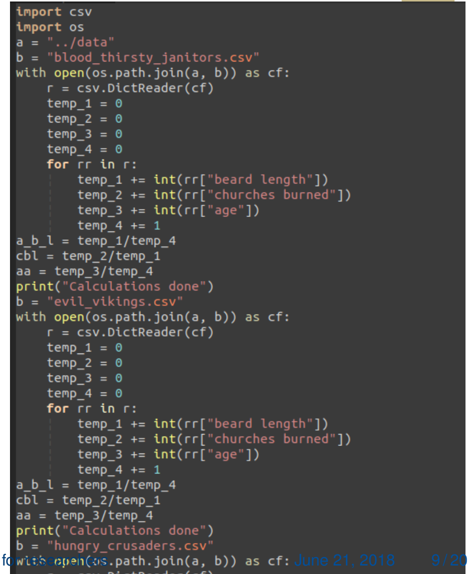
Making the code more readable

More topics to be considered

Discussion

General tips on making code better

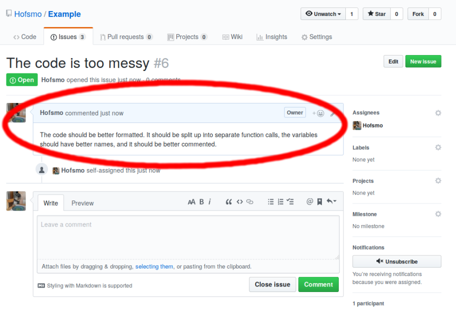
- Use descriptive variable names
- Have some space in between lines
- Write comments
- Write docstrings
- Don't copy and paste code



```
import csv
import os
a = "../data"
b = "blood_thirsty_janitors.csv"
with open(os.path.join(a, b)) as cf:
    r = csv.DictReader(cf)
    temp_1 = 0
    temp_2 = 0
    temp_3 = 0
    temp_4 = 0
    for rr in r:
        temp_1 += int(rr["beard length"])
        temp_2 += int(rr["churches burned"])
        temp_3 += int(rr["age"])
        temp_4 += 1
a_b_l = temp_1/temp_4
cbl = temp_2/temp_1
aa = temp_3/temp_4
print("Calculations done")
b = "evil_vikings.csv"
with open(os.path.join(a, b)) as cf:
    r = csv.DictReader(cf)
    temp_1 = 0
    temp_2 = 0
    temp_3 = 0
    temp_4 = 0
    for rr in r:
        temp_1 += int(rr["beard length"])
        temp_2 += int(rr["churches burned"])
        temp_3 += int(rr["age"])
        temp_4 += 1
a_b_l = temp_1/temp_4
cbl = temp_2/temp_1
aa = temp_3/temp_4
print("Calculations done")
b = "hungry_crusaders.csv"
with open(os.path.join(a, b)) as cf:
```

Using the GitHub issue tracker

— Write a descriptive issue



Hofsmo / Example

The code is too messy #6

Open Hofsmo opened this issue just now · 0 comments

Hofsmo commented just now

The code should be better formatted. It should be split up into separate function calls, the variables should have better names, and it should be better commented.

Hofsmo self-assigned this just now

Assignees: Hofsmo

Labels: None yet

Projects: None yet

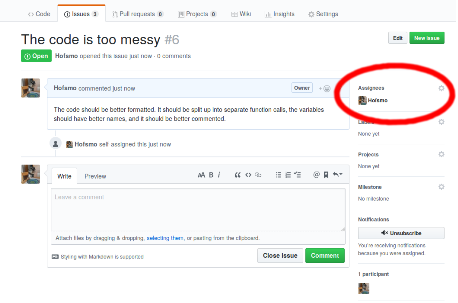
Milestone: No milestone

Notifications: You're receiving notifications because you were assigned.

1 participant

Using the GitHub issue tracker

- Write a descriptive issue
- You can make someone responsible for fixing the issue



The code is too messy #6

Open Hofsmo opened this issue just now · 0 comments

Hofsmo commented just now

The code should be better formatted. It should be split up into separate function calls, the variables should have better names, and it should be better commented.

Hofsmo self-assigned this just now

Leave a comment

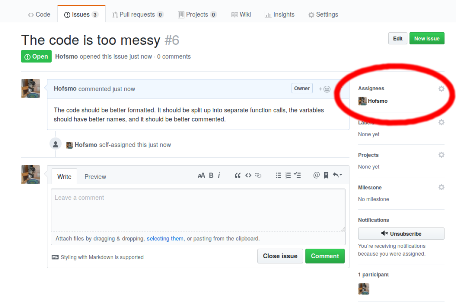
Attach files by dragging & dropping, selecting them, or pasting from the clipboard.

Styling with Markdown is supported

You're receiving notifications because you were assigned.

Using the GitHub issue tracker

- Write a descriptive issue
- You can make someone responsible for fixing the issue
- The issue get a number you can use for referencing it



The code is too messy #6

Hofsmo opened this issue just now · 0 comments

Hofsmo commented just now

The code should be better formatted. It should be split up into separate function calls, the variables should have better names, and it should be better commented.

Hofsmo self-assigned this just now

Leave a comment

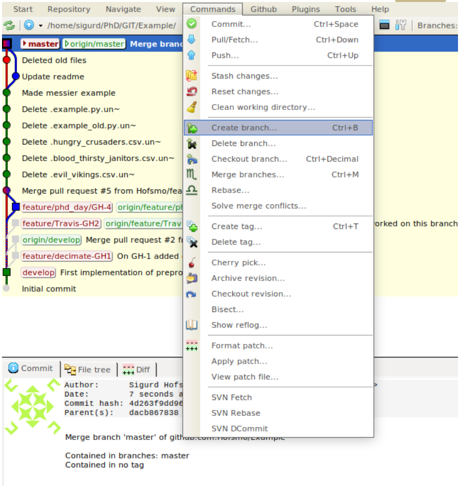
Attach files by dragging & dropping, selecting them, or pasting from the clipboard.

Styling with Markdown is supported

You're receiving notifications because you were assigned.

Using git for fixing the code

— Create a branch for the issue



master origin/master Merge branc
Deleted old files
Update readme
Made messier example
Delete .example.py.un~
Delete .example_old.py.un~
Delete .hungry_crusaders.csv.un~
Delete .blood_thirsty_janitors.csv.un~
Delete .evil_vikings.csv.un~
Merge pull request #5 from Hofsmo/fea
feature/phd_day/GH-4 origin/feature/ph
feature/Travis-GH2 origin/feature/Trav orked on this branch
origin/develop Merge pull request #2 fi
feature/decimate-GH1 On GH-1 added
develop First implementation of prepro
Initial commit

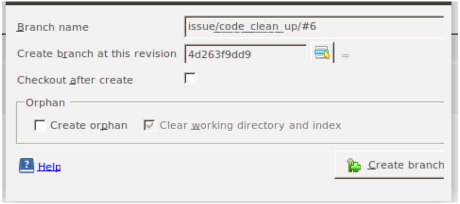
```
Author:       Sigurd Hofs
Date:         7 seconds a
Commit hash:  4d263f9dd96
Parent(s):    dacb867838
```

Merge branch 'master' of github.com:Hofsmo/Example

Contained in branches: master
Contained in no tag

Using git for fixing the code

- Create a branch for the issue
- Give the branch a good name



Branch name: issue/code_clean_up/#6

Create branch at this revision: 4d263f9dd9

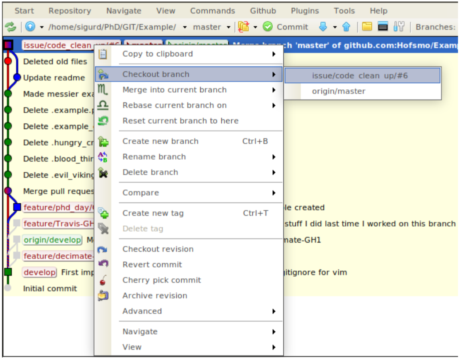
Checkout after create □

Orphan

□ Create orphan ☑ Clear working directory and index

Using git for fixing the code

- Create a branch for the issue
- Give the branch a good name
- Checkout the branch

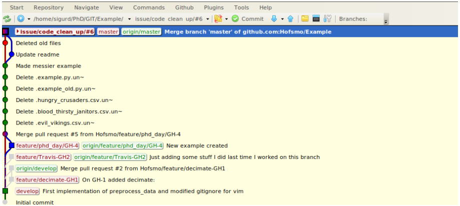


```
issue/code_clean   master   origin/master   Merge branch 'master' of github.com:Hofsmo/Examp
Deleted old files
Update readme
Made messier exa
Delete .example.
Delete .example_
Delete .hungry_cr
Delete .blood_thir
Delete .evil_viking
Merge pull reques
feature/phd_day/
feature/Travis-GH
origin/develop  M
feature/decimate-
develop  First imp
Initial commit
```

```
le created
stuff I did last time I worked on this branch
mate-GH1
itignore for vim
```

Using git for fixing the code

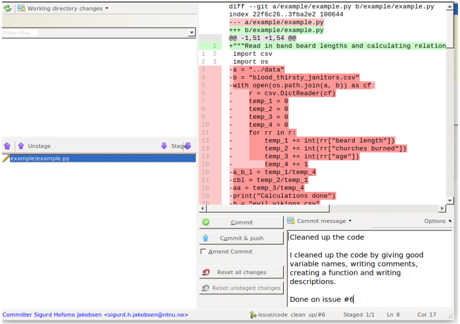
- Create a branch for the issue
- Give the branch a good name
- Checkout the branch
- We are now ready to clean up the code



```
issue/code_clean_up/#6 master origin/master Merge branch 'master' of github.com:Hofsmo/Example
Deleted old files
Update readme
Made messier example
Delete .example.py.un~
Delete .example_old.py.un~
Delete .hungry_crusaders.csv.un~
Delete .blood_thirsty_janitors.csv.un~
Delete .evil_vikings.csv.un~
Merge pull request #5 from Hofsmo/feature/phd_day/GH-4
feature/phd_day/GH-4 origin/feature/phd_day/GH-4 New example created
feature/Travis-GH2 origin/feature/Travis-GH2 Just adding some stuff I did last time I worked on this branch
origin/develop Merge pull request #2 from Hofsmo/feature/decimate-GH1
feature/decimate-GH1 On GH-1 added decimate:
develop First implementation of preprocess_data and modified gitignore for vim
Initial commit
```

Using git for fixing the code

- Create a branch for the issue
- Give the branch a good name
- Checkout the branch
- We are now ready to clean up the code
- After the code is clean reference the issue in the commit message



```
diff --git a/example/example.py b/example/example.py
index 22f6c26..3fba2e2 100644
--- a/example/example.py
+++ b/example/example.py
@@ -1,51 +1,54 @@
+"""Read in band beard lengths and calculating relation
 import csv
 import os
-a = "../data"
-b = "blood_thirsty_janitors.csv"
-with open(os.path.join(a, b)) as cf:
-    r = csv.DictReader(cf)
-    temp_1 = 0
-    temp_2 = 0
-    temp_3 = 0
-    temp_4 = 0
-    for rr in r:
-        temp_1 += int(rr["beard length"])
-        temp_2 += int(rr["churches burned"])
-        temp_3 += int(rr["age"])
-        temp_4 += 1
-a_b_l = temp_1/temp_4
-cbl = temp_2/temp_1
-aa = temp_3/temp_4
-print("Calculations done")
-b = "evil_vikings.csv"
```

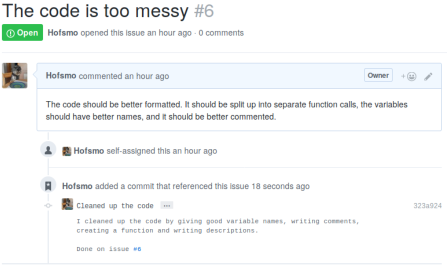
Commit message:

```
Cleaned up the code

I cleaned up the code by giving good
variable names, writing comments,
creating a function and writing
descriptions.

Done on issue #6
```

Committer Sigurd Hofsmo Jakobsen <sigurd.h.jakobsen@ntnu.no>   issue/code clean up/#6   Staged 1/1   Ln 8   Col 17

Using git for fixing the code

- Create a branch for the issue
- Give the branch a good name
- Checkout the branch
- We are now ready to clean up the code
- After the code is clean reference the issue in the commit message
- We can now see the commit message in GitHub



The code is too messy #6

Open Hofsmo opened this issue an hour ago · 0 comments

Hofsmo commented an hour ago

The code should be better formatted. It should be split up into separate function calls, the variables should have better names, and it should be better commented.

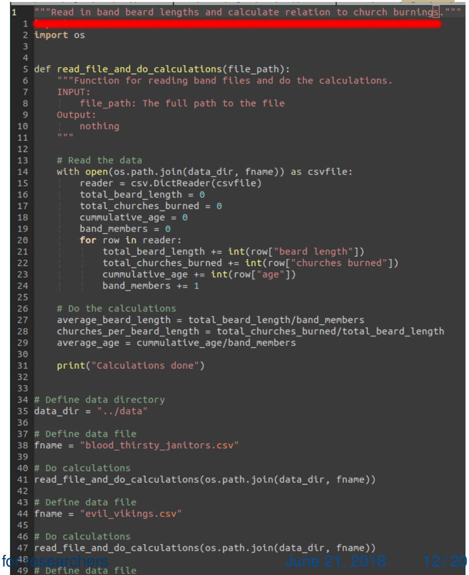
Hofsmo self-assigned this an hour ago

Hofsmo added a commit that referenced this issue 18 seconds ago

```
Cleaned up the code   ...                                   323a924

I cleaned up the code by giving good variable names, writing comments,
creating a function and writing descriptions.

Done on issue #6
```

What clean up did I do on the code

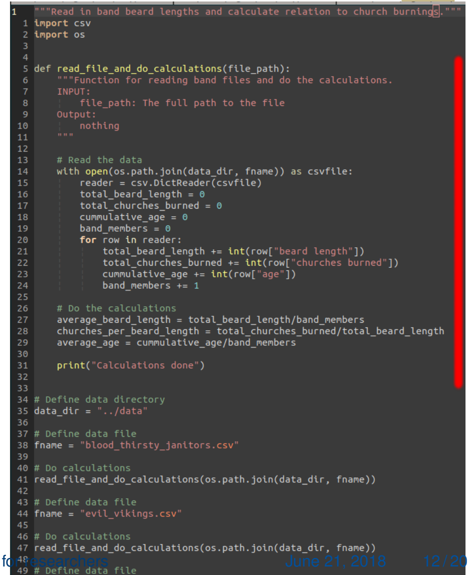
— Added a docstring to the file



```
"""Read in band beard lengths and calculate relation to church burnings."""
[redacted]
import os


def read_file_and_do_calculations(file_path):
    """Function for reading band files and do the calculations.
    INPUT:
        file_path: The full path to the file
    Output:
        nothing
    """

    # Read the data
    with open(os.path.join(data_dir, fname)) as csvfile:
        reader = csv.DictReader(csvfile)
        total_beard_length = 0
        total_churches_burned = 0
        cummulative_age = 0
        band_members = 0
        for row in reader:
            total_beard_length += int(row["beard length"])
            total_churches_burned += int(row["churches burned"])
            cummulative_age += int(row["age"])
            band_members += 1

    # Do the calculations
    average_beard_length = total_beard_length/band_members
    churches_per_beard_length = total_churches_burned/total_beard_length
    average_age = cummulative_age/band_members

    print("Calculations done")


# Define data directory
data_dir = "../data"

# Define data file
fname = "blood_thirsty_janitors.csv"

# Do calculations
read_file_and_do_calculations(os.path.join(data_dir, fname))

# Define data file
fname = "evil_vikings.csv"

# Do calculations
read_file_and_do_calculations(os.path.join(data_dir, fname))

# Define data file
```

What clean up did I do on the code

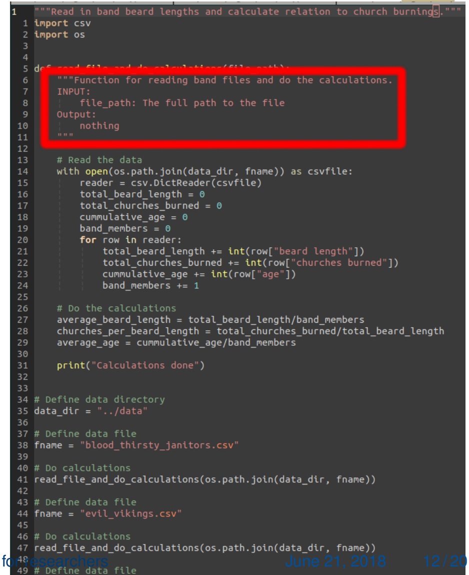
- Added a docstring to the file
- Created a function



```
"""Read in band beard lengths and calculate relation to church burnings."""
import csv
import os


def read_file_and_do_calculations(file_path):
    """Function for reading band files and do the calculations.
    INPUT:
        file_path: The full path to the file
    Output:
        nothing
    """

    # Read the data
    with open(os.path.join(data_dir, fname)) as csvfile:
        reader = csv.DictReader(csvfile)
        total_beard_length = 0
        total_churches_burned = 0
        cummulative_age = 0
        band_members = 0
        for row in reader:
            total_beard_length += int(row["beard length"])
            total_churches_burned += int(row["churches burned"])
            cummulative_age += int(row["age"])
            band_members += 1

    # Do the calculations
    average_beard_length = total_beard_length/band_members
    churches_per_beard_length = total_churches_burned/total_beard_length
    average_age = cummulative_age/band_members

    print("Calculations done")


# Define data directory
data_dir = "../data"

# Define data file
fname = "blood_thirsty_janitors.csv"

# Do calculations
read_file_and_do_calculations(os.path.join(data_dir, fname))

# Define data file
fname = "evil_vikings.csv"

# Do calculations
read_file_and_do_calculations(os.path.join(data_dir, fname))

# Define data file
```

What clean up did I do on the code

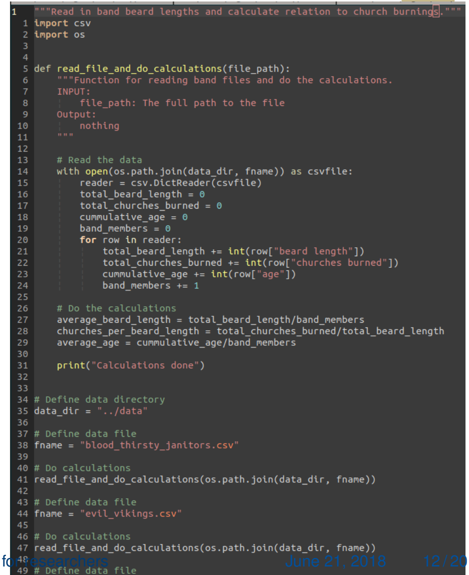
- Added a docstring to the file
- Created a function
- Wrote a docstring for the function



```
"""Read in band beard lengths and calculate relation to church burnings."""
import csv
import os


def read_file_and_do_calculations(file_path):
    """Function for reading band files and do the calculations.
    INPUT:
        file_path: The full path to the file
    Output:
        nothing
    """

    # Read the data
    with open(os.path.join(data_dir, fname)) as csvfile:
        reader = csv.DictReader(csvfile)
        total_beard_length = 0
        total_churches_burned = 0
        cummulative_age = 0
        band_members = 0
        for row in reader:
            total_beard_length += int(row["beard length"])
            total_churches_burned += int(row["churches burned"])
            cummulative_age += int(row["age"])
            band_members += 1

    # Do the calculations
    average_beard_length = total_beard_length/band_members
    churches_per_beard_length = total_churches_burned/total_beard_length
    average_age = cummulative_age/band_members

    print("Calculations done")


# Define data directory
data_dir = "../data"

# Define data file
fname = "blood_thirsty_janitors.csv"

# Do calculations
read_file_and_do_calculations(os.path.join(data_dir, fname))

# Define data file
fname = "evil_vikings.csv"

# Do calculations
read_file_and_do_calculations(os.path.join(data_dir, fname))

# Define data file
```

What clean up did I do on the code

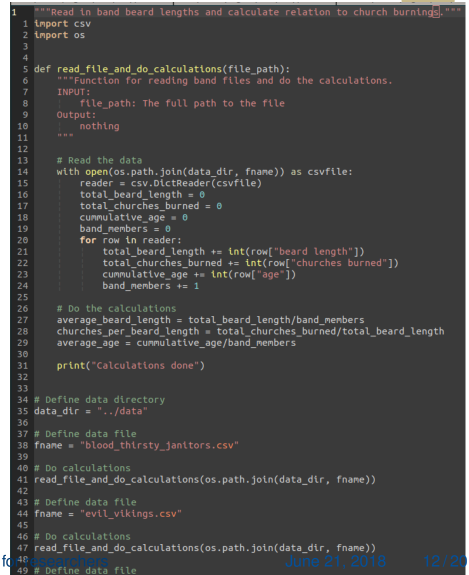
- Added a docstring to the file
- Created a function
- Wrote a docstring for the function
- Added proper spacing



```
"""Read in band beard lengths and calculate relation to church burnings."""
import csv
import os


def read_file_and_do_calculations(file_path):
    """Function for reading band files and do the calculations.
    INPUT:
        file_path: The full path to the file
    Output:
        nothing
    """

    # Read the data
    with open(os.path.join(data_dir, fname)) as csvfile:
        reader = csv.DictReader(csvfile)
        total_beard_length = 0
        total_churches_burned = 0
        cummulative_age = 0
        band_members = 0
        for row in reader:
            total_beard_length += int(row["beard length"])
            total_churches_burned += int(row["churches burned"])
            cummulative_age += int(row["age"])
            band_members += 1

    # Do the calculations
    average_beard_length = total_beard_length/band_members
    churches_per_beard_length = total_churches_burned/total_beard_length
    average_age = cummulative_age/band_members

    print("Calculations done")


# Define data directory
data_dir = "../data"

# Define data file
fname = "blood_thirsty_janitors.csv"

# Do calculations
read_file_and_do_calculations(os.path.join(data_dir, fname))

# Define data file
fname = "evil_vikings.csv"

# Do calculations
read_file_and_do_calculations(os.path.join(data_dir, fname))

# Define data file
```

What clean up did I do on the code

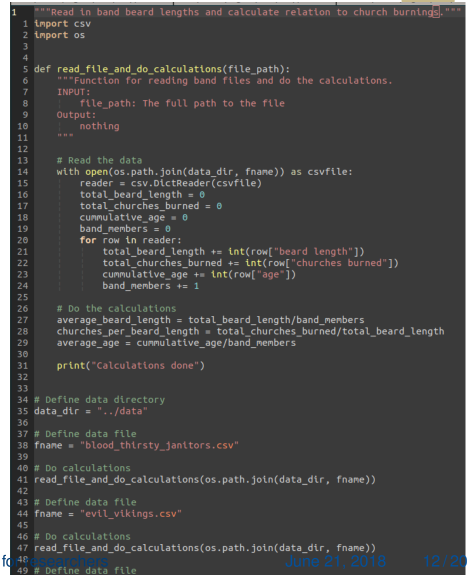
- Added a docstring to the file
- Created a function
- Wrote a docstring for the function
- Added proper spacing
- Added comments



```
"""Read in band beard lengths and calculate relation to church burnings."""
import csv
import os


def read_file_and_do_calculations(file_path):
    """Function for reading band files and do the calculations.
    INPUT:
        file_path: The full path to the file
    Output:
        nothing
    """

    # Read the data
    with open(os.path.join(data_dir, fname)) as csvfile:
        reader = csv.DictReader(csvfile)
        total_beard_length = 0
        total_churches_burned = 0
        cummulative_age = 0
        band_members = 0
        for row in reader:
            total_beard_length += int(row["beard length"])
            total_churches_burned += int(row["churches burned"])
            cummulative_age += int(row["age"])
            band_members += 1

    # Do the calculations
    average_beard_length = total_beard_length/band_members
    churches_per_beard_length = total_churches_burned/total_beard_length
    average_age = cummulative_age/band_members

    print("Calculations done")


# Define data directory
data_dir = "../data"

# Define data file
fname = "blood_thirsty_janitors.csv"

# Do calculations
read_file_and_do_calculations(os.path.join(data_dir, fname))

# Define data file
fname = "evil_vikings.csv"

# Do calculations
read_file_and_do_calculations(os.path.join(data_dir, fname))

# Define data file
```

What clean up did I do on the code

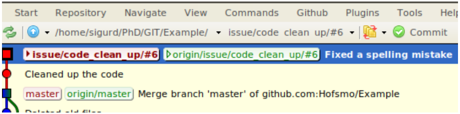
- Added a docstring to the file
- Created a function
- Wrote a docstring for the function
- Added proper spacing
- Added comments
- Made reasonable variable names



```
"""Read in band beard lengths and calculate relation to church burnings."""
import csv
import os


def read_file_and_do_calculations(file_path):
    """Function for reading band files and do the calculations.
    INPUT:
        file_path: The full path to the file
    Output:
        nothing
    """

    # Read the data
    with open(os.path.join(data_dir, fname)) as csvfile:
        reader = csv.DictReader(csvfile)
        total_beard_length = 0
        total_churches_burned = 0
        cummulative_age = 0
        band_members = 0
        for row in reader:
            total_beard_length += int(row["beard length"])
            total_churches_burned += int(row["churches burned"])
            cummulative_age += int(row["age"])
            band_members += 1

    # Do the calculations
    average_beard_length = total_beard_length/band_members
    churches_per_beard_length = total_churches_burned/total_beard_length
    average_age = cummulative_age/band_members

    print("Calculations done")


# Define data directory
data_dir = "../data"

# Define data file
fname = "blood_thirsty_janitors.csv"

# Do calculations
read_file_and_do_calculations(os.path.join(data_dir, fname))

# Define data file
fname = "evil_vikings.csv"

# Do calculations
read_file_and_do_calculations(os.path.join(data_dir, fname))

# Define data file
```

Resolving the issue

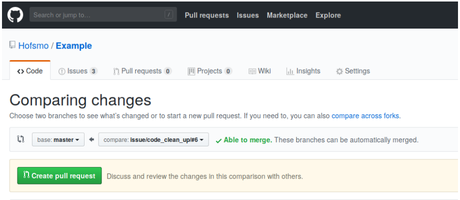
— The code is now clean, and we want the changes in our master branch.



```
issue/code_clean_up/#6  origin/issue/code_clean_up/#6  Fixed a spelling mistake
Cleaned up the code
master  origin/master  Merge branch 'master' of github.com:Hofsmo/Example
Deleted old files
```

Resolving the issue

- The code is now clean, and we want the changes in our master branch.
- GitHub automatically checks if the changes are compatible.



Comparing changes

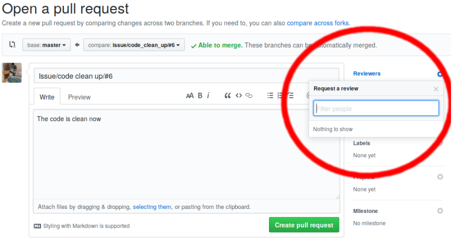
Choose two branches to see what's changed or to start a new pull request. If you need to, you can also compare across forks.

base: master ← compare: issue/code_clean_up/#6 ✓ Able to merge. These branches can be automatically merged.

Create pull request Discuss and review the changes in this comparison with others.

Resolving the issue

- The code is now clean, and we want the changes in our master branch.
- GitHub automatically checks if the changes are compatible.
- You can ask someone to review your code.



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: master ▾ ← compare: Issue/code_clean_up/#6 ▾ ✓ Able to merge. These branches can be automatically merged.

Issue/code clean up/#6

The code is clean now

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

Styling with Markdown is supported

Reviewers

Request a review

Filter people

Nothing to show

Labels

None yet

Projects

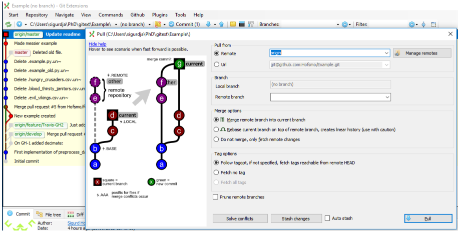
None yet

Milestone

No milestone

Another student downloading the changes

- Other students can now get the changes using pull



Example (no branch) - Git Extensions

C:\Users\sigurdja\PhD\gitext\Example\ (no branch)

origin/master Update readme
Made messier example
master Deleted old file.
Delete .example.py.un~
Delete .example_old.py.un~
Delete .hungry_crusaders.csv.un~
Delete .blood_thirsty_janitors.csv.un
Delete .evil_vikings.csv.un~
Merge pull request #5 from Hofsmo/f
New example created
origin/feature/Travis-GH2 Just add
origin/develop Merge pull request #
On GH-1 added decimate:
First implementation of preprocess_d
Initial commit

Author: Sigurd Ho
Date: 4 hours a

Pull (C:\Users\sigurdja\PhD\gitext\Example\)

Hide help
Hover to see scenario when fast forward is possible.

Pull from
● Remote: origin
○ Url: git@github.com:Hofsmo/Example.git

Branch
Local branch: (no branch)
Remote branch:

Merge options
● Merge remote branch into current branch
○ Rebase current branch on top of remote branch, creates linear history (use with caution)
○ Do not merge, only fetch remote changes

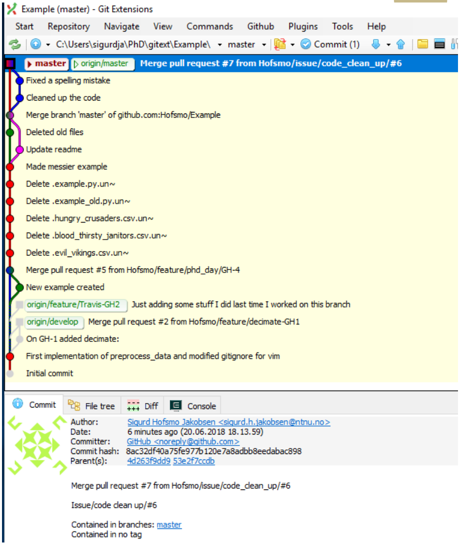
Tag options
● Follow tagopt, if not specified, fetch tags reachable from remote HEAD
○ Fetch no tag
○ Fetch all tags

□ Prune remote branches

□ Auto stash

Another student downloading the changes

- Other students can now get the changes using pull
- Afterwards the changes will be merged and available



master origin/master Merge pull request #7 from Hofsmo/issue/code_clean_up/#6
Fixed a spelling mistake
Cleaned up the code
Merge branch 'master' of github.com:Hofsmo/Example
Deleted old files
Update readme
Made messier example
Delete .example.py.un~
Delete .example_old.py.un~
Delete .hungry_crusaders.csv.un~
Delete .blood_thirsty_janitors.csv.un~
Delete .evil_vikings.csv.un~
Merge pull request #5 from Hofsmo/feature/phd_day/GH-4
New example created
origin/feature/Travis-GH2 Just adding some stuff I did last time I worked on this branch
origin/develop Merge pull request #2 from Hofsmo/feature/decimate-GH1
On GH-1 added decimate:
First implementation of preprocess_data and modified gitignore for vim
Initial commit

Author: Sigurd Hofsmo Jakobsen <sigurd.h.jakobsen@ntnu.no>
Date: 6 minutes ago (20.06.2018 18.13.59)
Committer: GitHub <noreply@github.com>
Commit hash: 8ac32df40a75fe977b120e7a8adbb8eedabac898
Parent(s): 4d263f9dd9 53e2f7ccdb

Merge pull request #7 from Hofsmo/issue/code_clean_up/#6

Issue/code clean up/#6

Contained in branches: master
Contained in no tag

Outline

Taking over some old code and fixing it

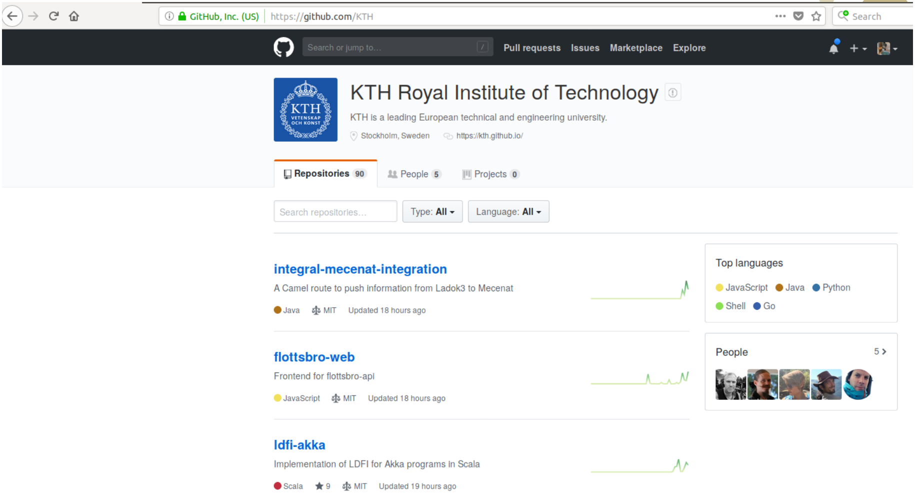
Version control using git

Making the code more readable

More topics to be considered

Discussion

GitHub provides organizational pages



GitHub, Inc. (US) https://github.com/KTH

KTH Royal Institute of Technology

KTH is a leading European technical and engineering university.

Stockholm, Sweden https://kth.github.io/

Repositories 90 People 5 Projects 0

integral-mecenat-integration

A Camel route to push information from Ladok3 to Mecenat

Java MIT Updated 18 hours ago

flottsbro-web

Frontend for flottsbro-api

JavaScript MIT Updated 18 hours ago

ldfi-akka

Implementation of LDFI for Akka programs in Scala

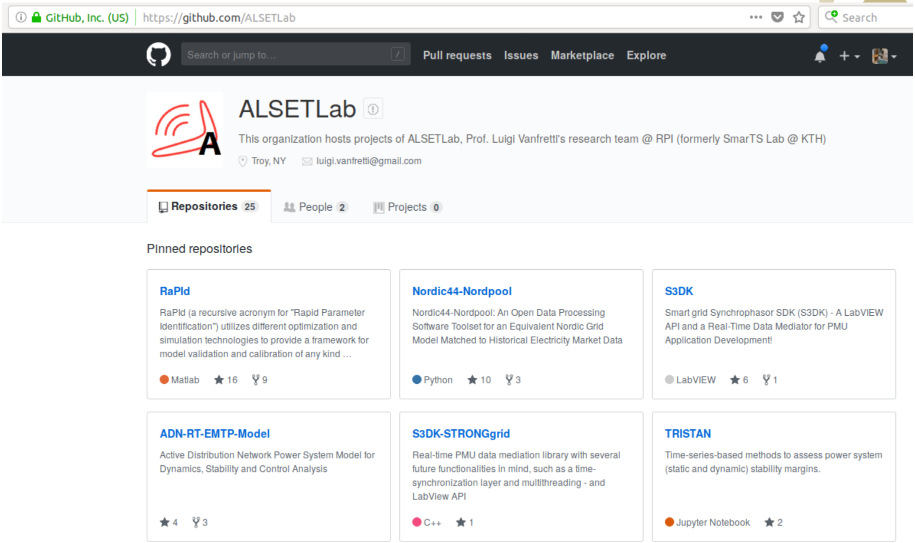
Scala ★ 9 MIT Updated 19 hours ago

Top languages

JavaScript Java Python Shell Go

People 5

GitHub provides organizational pages



GitHub, Inc. (US) https://github.com/ALSETLab

ALSETLab

This organization hosts projects of ALSETLab, Prof. Luigi Vanfretti's research team @ RPI (formerly SmarTS Lab @ KTH)

Troy, NY luigi.vanfretti@gmail.com

Pinned repositories

RaPId
RaPId (a recursive acronym for "Rapid Parameter Identification") utilizes different optimization and simulation technologies to provide a framework for model validation and calibration of any kind ...
● Matlab ★ 16 ⑂ 9

Nordic44-Nordpool
Nordic44-Nordpool: An Open Data Processing Software Toolset for an Equivalent Nordic Grid Model Matched to Historical Electricity Market Data
● Python ★ 10 ⑂ 3

S3DK
Smart grid Synchrophasor SDK (S3DK) - A LabVIEW API and a Real-Time Data Mediator for PMU Application Development!
● LabVIEW ★ 6 ⑂ 1

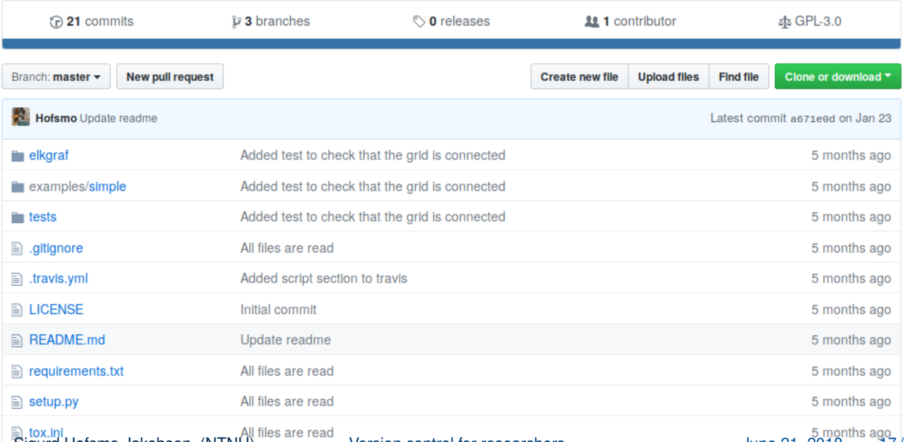
ADN-RT-EMTP-Model
Active Distribution Network Power System Model for Dynamics, Stability and Control Analysis
★ 4 ⑂ 3

S3DK-STRONGgrid
Real-time PMU data mediation library with several future functionalities in mind, such as a time-synchronization layer and multithreading - and LabView API
● C++ ★ 1

TRISTAN
Time-series-based methods to assess power system (static and dynamic) stability margins.
● Jupyter Notebook ★ 2

Structuring of packages and toolboxes

- Properly structured packages can easily be installed by others.
- For Python check out setuptools
- For Matlab check out their help pages

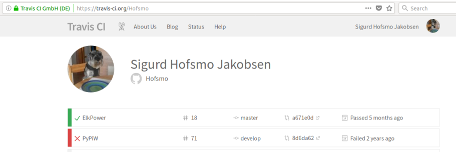


Hofsmo Update readme — Latest commit a671e0d on Jan 23

elkgraf	Added test to check that the grid is connected	5 months ago
examples/simple	Added test to check that the grid is connected	5 months ago
tests	Added test to check that the grid is connected	5 months ago
.gitignore	All files are read	5 months ago
.travis.yml	Added script section to travis	5 months ago
LICENSE	Initial commit	5 months ago
README.md	Update readme	5 months ago
requirements.txt	All files are read	5 months ago
setup.py	All files are read	5 months ago
tox.ini	All files are read	5 months ago

Automating code sanity tests

- Code can be tested for guidelines (PEP8)
- The code installation can be tested (tox)
- Unit tests can be done (pytest)
- Documentation can be generated and tested (sphinx)
- Many more options



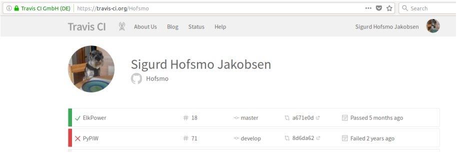
Sigurd Hofsmo Jakobsen

Hofsmo

✓ ElkPower	# 18	master	a671e0d	Passed 5 months ago
✗ PyPiW	# 71	develop	8d6da62	Failed 2 years ago

Automating code sanity tests

- Code can be tested for guidelines (PEP8)
- The code installation can be tested (tox)
- Unit tests can be done (pytest)
- Documentation can be generated and tested (sphinx)
- Many more options



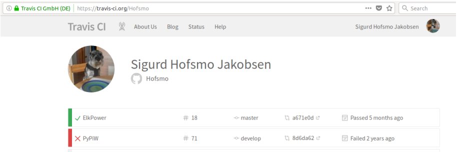
Sigurd Hofsmo Jakobsen

Hofsmo

✓ ElkPower	# 18	master	a671e0d	Passed 5 months ago
✗ PyPiW	# 71	develop	8d6da62	Failed 2 years ago

Automating code sanity tests

- Code can be tested for guidelines (PEP8)
- The code installation can be tested (tox)
- Unit tests can be done (pytest)
- Documentation can be generated and tested (sphinx)
- Many more options



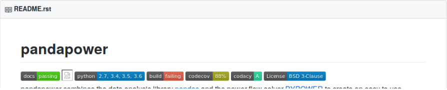
Sigurd Hofsmo Jakobsen

Hofsmo

✓ ElkPower	# 18	master	a671e0d	Passed 5 months ago
✗ PyPiW	# 71	develop	8d6da62	Failed 2 years ago

Automating code sanity tests

- Code can be tested for guidelines (PEP8)
- The code installation can be tested (tox)
- Unit tests can be done (pytest)
- Documentation can be generated and tested (sphinx)
- Many more options



README.rst

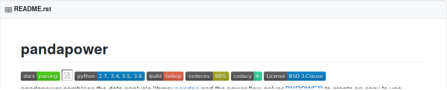
pandapower

docs passing | python 2.7, 3.4, 3.5, 3.6 | build failing | codecov 88% | codacy A | License BSD 3-Clause

pandapower combines the data analysis library pandas and the power flow solver PYPOWER to create an easy to use

Automating code sanity tests

- Code can be tested for guidelines (PEP8)
- The code installation can be tested (tox)
- Unit tests can be done (pytest)
- Documentation can be generated and tested (sphinx)
- Many more options



README.rst

pandapower

docs passing | python 2.7, 3.4, 3.5, 3.6 | build failing | codecov 88% | codacy A | License BSD 3-Clause

pandapower combines the data analysis library pandas and the power flow solver PYPOWER to create an easy to use

Outline

Taking over some old code and fixing it

Version control using git

Making the code more readable

More topics to be considered

Discussion

Questions for discussion

- How does my code look?
- How do we work with code today? Cooperation? Version control?
- Could my code be useful for others?
- Should the department have a more active approach to the topic?
  - Education?
  - Infrastructure?
  - Who has the responsibility?
- Actions?