Week 3: Assignment 1
Help us analyse feedback of participants in eYRC

[*Last Updated on:* ***21st April 2021, 19:00 Hrs***]

Aim

e-Yantra Robotics Competition (eYRC) **is a unique annual competition for students** in Engineering/ Science/ Polytechnic colleges. The entire competition is divided into small tasks which all the participating teams have to complete within a deadline.

In order to improve the theme, we take regular feedback from the students.

One such question in the feedback form is to **judge the understanding/knowledge (on a scale of 1-10) of participants** before and after completing the task.

To make sense of the data, we need your help to find the

- average knowledge of participants before & after completing the task
- percentage increase in knowledge between the number of participants who rated `<= 2` before attempting the task and `>=5` after attempting the task

Now, write an **AWK (.awk)** script to scan through the data in the given CSV file and help us achieve this goal.

DISCLAIMER:

The data provided is to be solely used to solve this assignment. It does not relate to the actual figures of the competition.

Given

- Two files are provided to solve this assignment.
  - Skeleton program file: **`week3_assignment1.awk`** (Contents of this file on **`master`** and **`feature`** branch are different)
  - Sample CSV file: **`week3_assignment1_sample.csv`**

- Before moving on to the Procedure section, make sure you have gone through the ***Week 3 Live Session on Git and GitHub*** conducted on *April 21st, 2021*.
- You can find the link in the Live Sessions document.

Procedure

- Visit your **Private Repository of Assignment** that you must have received after performing the steps as mentioned in Introduction to GitHub Classroom.
- Copy the URL of your repository as shown in Figure 1.

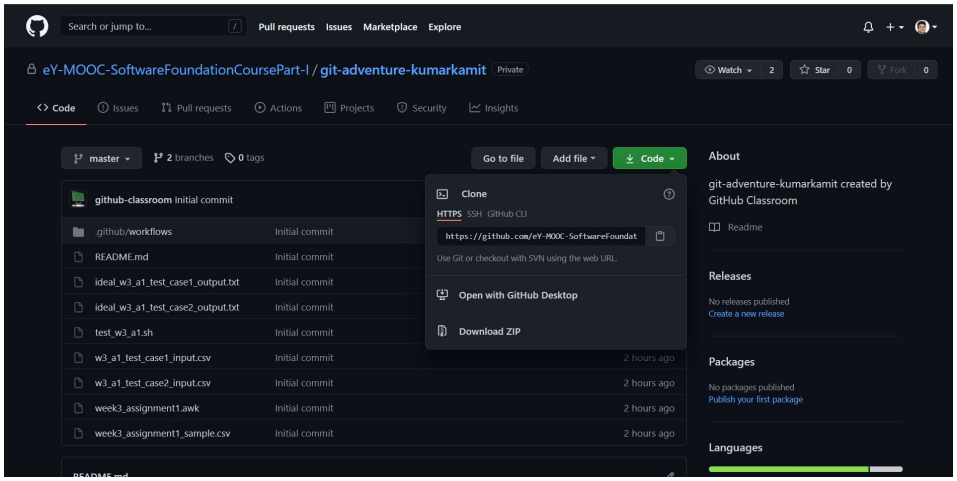


Figure 1: Copy Private Repository URL from the GitHub website.

- Open Terminal and navigate to the directory where you want to clone your repository.
- Now, clone the repository into your machine using the following command:

```
git clone https://github.com/eY-MOOC-SoftwareFoundationCoursePart-I/git-adventure-ku
```

- Navigate to the cloned repository and use the `ls` command to view the contents of the directory. The contents will be as shown in Figure 2.

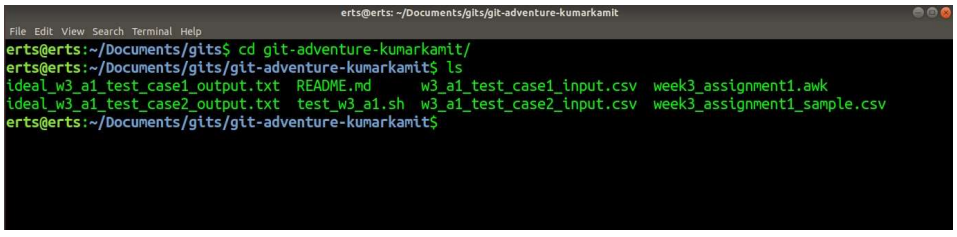


Figure 2: Contents of the 'Master' branch in the cloned repository.

- Use the following command to find the branches in your repository.

```
git branch -a
```

Figure 3: View the branches in the cloned repository.

- We will talk about the contents of `master` branch later. Let us first switch to the `feature` branch.
- Use the `checkout` command as mentioned below:

```
git checkout feature
```

- Use the `ls` command to view the contents of the directory. The contents will be as shown in Figure 4.

Figure 4: Contents of the **feature** branch after using **checkout** command.

- An AWK script by the name `week3_assignment1.awk` will be present.
- You need to edit this file in order to solve the ***2nd part of the assignment***.
- The contents of `week3_assignment1.awk` will be as follows:

```
#!/bin/awk -f

BEGIN{
        FS=",";
        know_less_than_eq_two_before = 0;
        know_great_than_eq_five_after = 0;
}

{
        if(NR > 0)
        {
                if($1 <= 1)
                        know_less_than_eq_two_before = 1;

                if($2 >= 8)
                        know_great_than_eq_five_after = 1;
        }
}

END{
        percen_incr = ( ( know_great_than_eq_five_after - know_less_than_eq_two_befo
        printf "\nPercentage increase in knowledge gained from level 2 or less befor
}
```

- After you have edited the file, you can check your output by running the following command:

```
bash test_w3_a1_feature.sh week3_assignment1.awk
```

- If all the test cases are passed, you will find the following statement printed in your terminal:

```
Congrats! All test cases PASSED.
```

- If the test case(s) is/are not passed, you will find the following statement printed in your terminal-

```
Test case number 1: FAIL
```

- After you are satisfied with your output and want to push your code to the repo, follow the below steps:

  1. Run,

     ```
     git status
     ```

     to check the status of your file. You will observe that state is **modified**.

  2. Now run,

     ```
     git add .
     ```

     to stage all the modified files in the directory.

  3. Run

     ```
     git status
     ```

     again, to verify whether the files are staged.

  4. Run

     ```
     git commit -m "type message for your commit here"
     ```

     to commit all the files which are staged.

  5. Run

     ```
     git status
     ```

     to verify that your working tree is clean.

  6. Lastly, run

     ```
     git push origin feature
     ```

     Make sure that you enter your **GitHub username and password**.

- Refer Figure 5 to check the output of above commands.

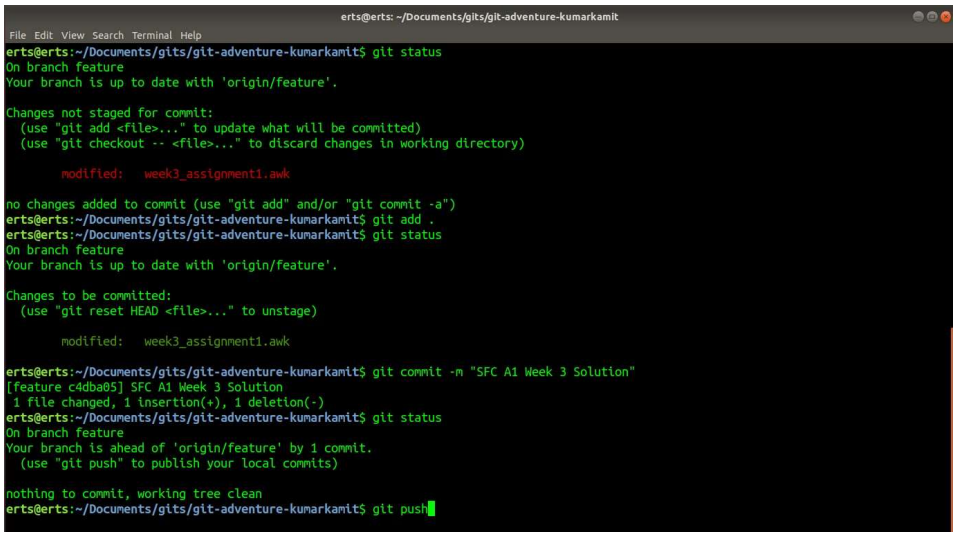


Figure 5: Output of commands required to push a file on remote repository.

- If your code **did not pass all the test cases**, you will observe a :x: next to the commit as shown in Figure 6.

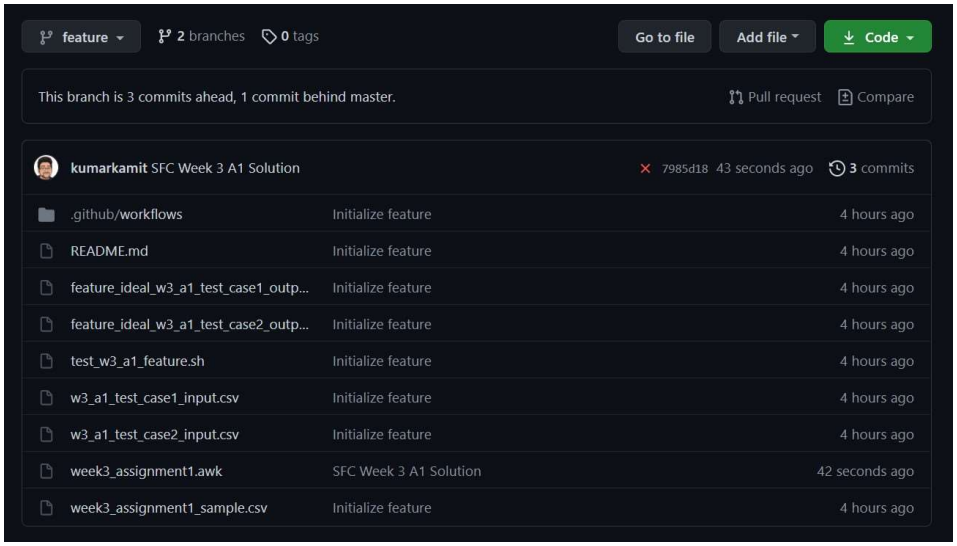


Figure 6: Unsuccessful attempt to clear all the test cases.

- If your code passed all the test cases, you will observe a :heavy_check_mark: next to the commit as shown in Figure 7.

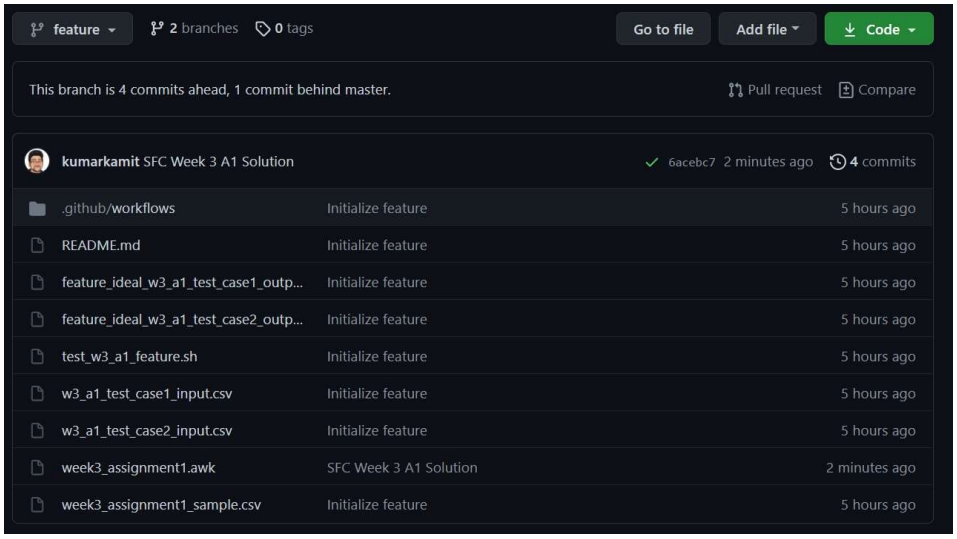


Figure 7: Successful attempt to clear all the test cases.

- You can review the result of all your commits by visiting the **Actions** tab in your Assignment repository as shown in Figure 8.

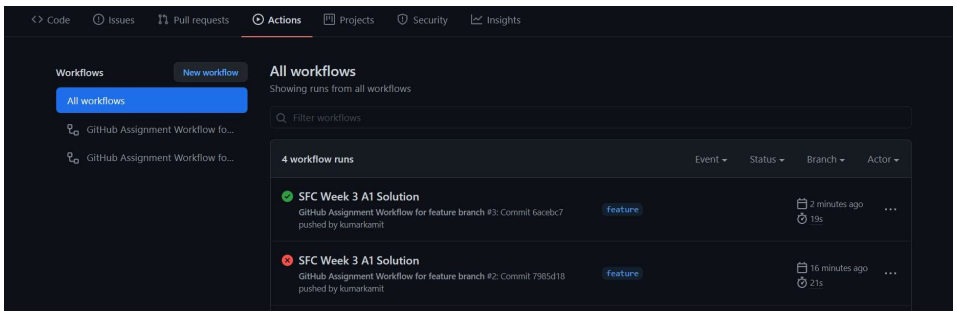


Figure 8: Reviewing the result of each commit on GitHub.

- Once you have passed all test cases on `feature` branch, then move to the `master` branch of the repository. Use the `checkout` command as mentioned below:

```
git checkout master
```

- Use the `ls` command to view the contents of the directory. The contents will be as shown in Figure 9.

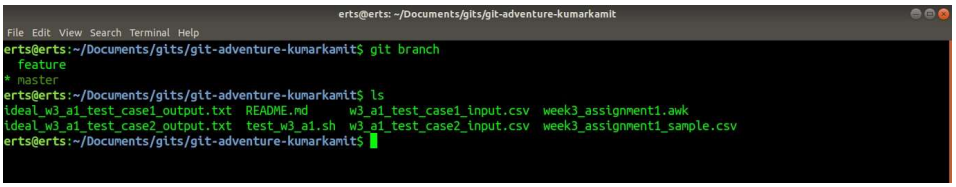


Figure 9: Contents of the **master** branch after using **checkout** command.

- You will notice that another `week3_assignment1.awk` is present here. The contents of this file will be as follows:

```
#!/bin/awk -f

BEGIN{
        printf "Average value of knowledge gained (on a scale of 1 to 10) before and

        FS=",";
        sum_scale_know_before = 0;
        sum_scale_know_after = 0;
}

{
        if(NR > 0)
        {
                sum_scale_know_before = sum_scale_know_before + 1;
                sum_scale_know_after = sum_scale_know_after + 2;
        }
}

END{
        printf "Before: %.3f\n", sum_scale_know_before/NR;
        printf "After: %.3f\n",  sum_scale_know_after/NR;
}
```

- Before we edit the contents of this file, we need to **merge ** ***this file with the contents of the same file present*** in the `feature` branch.
- To merge the contents of both the files, use the following command:

```
git merge feature --allow-unrelated-histories
```

- You will observe, that some conflicts arise when you try to merge as shown in Figure 10. To learn more about this, make sure you have thoroughly gone through the **Week 3 Live Session on Git and GitHub**.

Figure 10: Conflicts shown in terminal when trying to merge.

- After you have resolved these conflicts, you can test your code with the following command:

```
bash test_w3_a1.sh week3_assignment1.awk
```

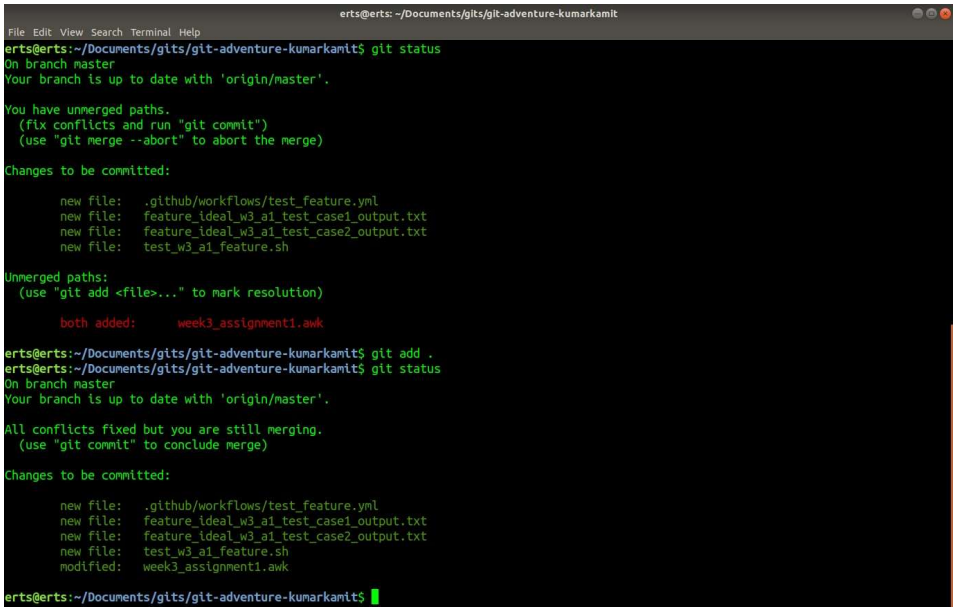
- If your code **passed all the test cases**, you will get the output as shown below:

```
Running tests...

====================================
Evaluating for Test case: 1

Pass => Program executed correctly.

Pass => Output is correct!
Test case number 1: PASS
====================================
====================================
Evaluating for Test case: 2

Pass => Program executed correctly.

Pass => Output is correct!
Test case number 2: PASS
====================================

Congrats! All test cases PASSED.
```

- If your code **did not pass all the test cases**, you may get the output as shown below:

```
Running tests...

====================================
Evaluating for Test case: 1

Pass => Program executed correctly.

Fail => Expected output is:
Average value of knowledge gained (on a scale of 1 to 10) before and after the task
Before: 2.673
After: 5.925

Percentage increase in knowledge gained from level 2 or less before to level 5 or mo

But got the output:
Average value of knowledge gained (on a scale of 1 to 10) before and after the task
Before: 2.669
After: 5.918

Percentage increase in knowledge gained from level 2 or less before to level 5 or mo

Test case number 1: FAIL
```

- After you are satisfied with your output, follow the steps mentioned above to push your code to GitHub. You can refer Figure 11 for these steps as well.



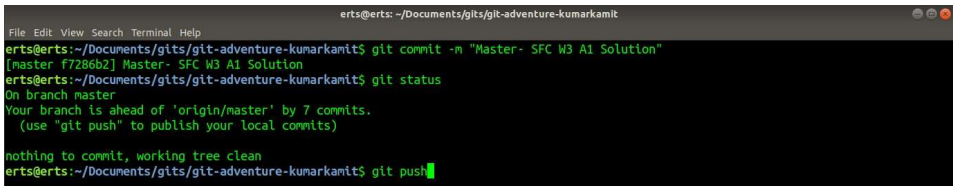


Figure 11 (a) and (b): Git commands required to push code on 'master' branch.

- Visit the **Actions** tab in your Assignment repository to verify the test results of your code.

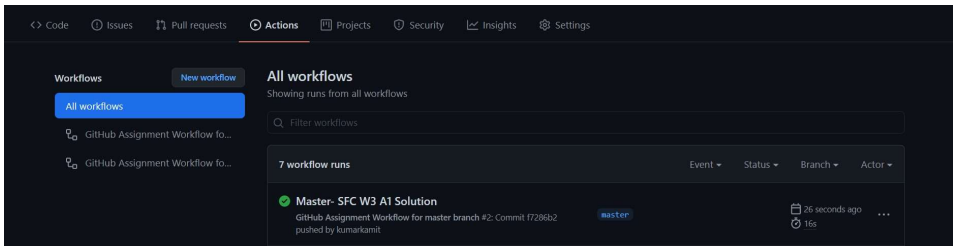


Figure 12: Reviewing the final result of the assignment.

Voila! You have now successfully completed this Assignment.

The deadline of this Assignment is `26th Apr 2021, 1700 Hrs.`

The total marks (10) will be updated after the deadline on or before `27th Apr 2021`, provided the test cases on both `master` and `feature` branches are `PASSED`.

References

- Nano Editor
  - How to use Nano Text Editor
  - Nano Editor Official Docs

- Vim Editor
  - Interactive Vim Tutorial
- AWK
  - Advanced Bash Scripting Guide
  - AWK: Introduction and Tutorial
  - Very Useful Command Line Utilities
  - SED AWK Examples by Unix School

All The Best!