CS 108 - Bash Grader

Siddhant Mulkikar

April 26, 2024

Contents

1   Objective

The objective of this CS108 Project was to create a csv file manager and interpreter with a command line interface.

2   Introduction

Bash Grader is a command line interface for instructors to grade students and manage their marks. It is an incredibly useful tool with its many functionalities and features.

3   List of Functionalities

3.1   Basic functionalities

- Upload a marklist in the form of a csv file.

- Total the marks of each student over all exams

- Combine the marks of all exams into a single csv file

- Update the marks of a student in a particular exam

4 Usage

4.1 Upload

Run the following command to upload a csv file in your current directory.

```
bash submission upload <filename>
```

4.2 Total

Run the following command to total the marks of each student over all exams into a total column in `main.csv`.

```
bash submission total
```

4.3 Combine

The following command combines all the csv files in the current directory into `main.csv`.

```
bash submission combine
```

4.4 Update

Run the following command to update the marks of a particular student

```
bash submission update
```

5 Working

5.1 Upload

```
siddhant@siddhant:~/Desktop/bash-grader-test$ ls
bash_scripts   main.csv  quiz1.csv  quiz2.csv  submission.sh   test
siddhant@siddhant:~/Desktop/bash-grader-test$ bash submission.sh upload test/mid
sem.csv
siddhant@siddhant:~/Desktop/bash-grader-test$ ls
bash_scripts   main.csv  midsem.csv  quiz1.csv  quiz2.csv  submission.sh  test
siddhant@siddhant:~/Desktop/bash-grader-test$
```

In the above example, the file `quiz1.csv` is uploaded i.e. copied into the current directory from the filepath given as argument by the user. The script checks if the file exists and if it is a csv file. If it is, only then is the file copied into the current directory.
Script files : upload.sh

5.2 Total

```
siddhant@siddhant:~/Desktop/bash-grader-test$ cat main.csv
Roll_Number,Name,quiz1,midsem
22B1003,Saksham Rathi,6,16
22B009,Mayank Kumar,7,07
23B0069,Sunny,4,14
23B0108,Ramesh,a,24siddhant@siddhant:~/Desktop/bash-grader-test$ bash submission
.sh total
siddhant@siddhant:~/Desktop/bash-grader-test$ cat main.csv
Roll_Number,Name,quiz1,midsem,Total
22B1003,Saksham Rathi,6,16,22
22B009,Mayank Kumar,7,07,14
23B0069,Sunny,4,14,18
23B0108,Ramesh,a,24,24
siddhant@siddhant:~/Desktop/bash-grader-test$
```

In the above example, the script totals the marks of each student over all exams and adds a new column `Total` to the `main.csv` file. If the marks column has `a` as its entry, the script skips that row while totaling.
Script files : total.awk

5.3 Combine

```
main.csv > data
1   Roll_Number,Name,midsem,quiz1
2   23B0901,Kaushal Vijayvergiya,a,11.38
3   23B0902,Arya Suwalka,4.91,20
4   23B0903,Abhi Jain,6.4,9.28
5   23B0904,Prakhar Jain,7.39,a
6   23B0905,Sagnik Nandi,7.23,8.92
7   23B0906,Velagala Deeraj Sathvik Reddy,6.44,14.46
8   23B0907,Hari Shankar Karthik,a,13.3
9   23B0908,Raghav Goyal,8.37,7.53
10  23B0909,Varri Shyam Sri Vardhan,6.05,a
11  23B0910,Sonal Kumari,4.41,9.7
```

```
main.csv > data
1   Roll_Number,Name,endsem,midsem,quiz1,quiz2
2   23B0901,Kaushal Vijayvergiya,a,a,11.38,14.47
3   23B0902,Arya Suwalka,4.57,4.91,20,a
4   23B0903,Abhi Jain,0.91,6.4,9.28,9.7
5   23B0904,Prakhar Jain,8.72,7.39,a,14.03
6   23B0905,Sagnik Nandi,6.31,7.23,8.92,11.74
7   23B0906,Velagala Deeraj Sathvik Reddy,5.63,6.44,14.46,7.88
8   23B0907,Hari Shankar Karthik,6.4,a,13.3,10.26
9   23B0908,Raghav Goyal,6.08,8.37,7.53,4.31
10  23B0909,Varri Shyam Sri Vardhan,8.15,6.05,a,11.72
11  23B0910,Sonal Kumari,4.78,4.41,9.7,a
```

| (a) Before combine | (b) After combine |

In the above example, the script combines all the csv files in the current directory into a single `main.csv` file. The script checks if the file is a csv file and if it is not the `main.csv` file.
`main.csv` is constructed from scratch in the following steps:

1. First, an array of all unique roll numbers and names is created.

2. Then, based on the exam files, a header is created with the roll numbers and names.

3. A mesh of `a`'s is created with the dimensions of the header.

4. Then the script iterates over all exam files and fills in the marks of each student in the mesh, whenever they are found. This ensures that anyone who is not present in a particular exam file is marked as `a`(absent) in `main.csv` for that particular exam.

Script files : combine.sh combine.awk

5.4 Update

(a) quiz1.csv before update

(b) quiz2.csv after update

(a) main.csv before update

(b) main.csv after update

In the above example, the script updates the marks of **23B0956** in `quiz1.csv` and `main.csv`.
The script first checks if the student is present in the `main.csv` file and if the exam file exists. If the student is present in the `main.csv` file, the script updates the marks of the student in `main.csv` by calling `update_main.awk` and the exam file by calling `update_exam.awk`.
It also asks the user if they want to update the marks of the student in some other exam and repeats the same process. If the user does not want to update the marks of the student in any other exam, the script checks if `main.csv` had been totaled before. If it was, then the script calls `total.awk` to correct the total, as the marks in `main.csv` have been updated.
Script files : update.sh update_exam.awk update_main.awk total.awk