

# CS 108 - Bash Project

Akshay Karthik

April 28, 2024

## **Abstract**

A brief overview of my bash project

# Contents

<b>1</b>	<b>Objective</b>	<b>3</b>
<b>2</b>	<b>Introduction to Bash</b>	<b>3</b>
<b>3</b>	<b>Applications</b>	<b>3</b>
<b>4</b>	<b>Basic Idea of Code</b>	<b>3</b>
4.1	Combine . . . . .	3
4.2	Upload . . . . .	4
4.3	Total . . . . .	4
4.4	git init . . . . .	4
4.5	git commit . . . . .	4
4.6	Update . . . . .	4
<b>5</b>	<b>Customisations</b>	<b>5</b>
<b>6</b>	<b>Conclusion</b>	<b>5</b>

# 1 Objective

The bash file `submission.sh` implement a Bash Grader using Bash and Git. Another files `plots.py`, `bargraph.py` has some added coustomisations.

## 2 Introduction to Bash

- **Bash:** Bash is a command language interpreter for Unix and Linux operating systems.
- **Bash Scripting:** Bash scripting involves writing sequences of commands for the Bash shell to execute.

## 3 Applications

Some of the applications of the *Bash* are described below

1. **System Administration:** Bash scripts are extensively used for automating system administration tasks such as backups, log rotation, user management, software installations, and system monitoring. They help streamline routine tasks and maintain system stability and security.
2. **File and Data Manipulation:** Bash scripts are adept at processing files and data. They can perform tasks such as parsing log files, extracting specific information, transforming data formats, and generating reports. This is particularly useful in data processing pipelines and data analysis tasks.
3. **Backup and Disaster Recovery:** Bash scripts are used to create automated backup solutions for critical data and system configurations. They can schedule regular backups, manage backup storage, and verify backup integrity. In the event of data loss or system failure, these scripts facilitate swift recovery processes.
4. **Task Automation:** Beyond system administration, bash scripts are handy for automating various tasks in everyday computing, such as file organization, batch image processing, text manipulation, and email notifications. They help users save time and effort by eliminating repetitive manual tasks
5. **Custom Utilities:** Beyond system administration, bash scripts are handy for automating various tasks in everyday computing, such as file organization, batch image processing, text manipulation, and email notifications. They help users save time and effort by eliminating repetitive manual tasks

## 4 Basic Idea of Code

Suppose you are provided with a large number of such files. Each such file has data for a student for a particular exam. The format of the data is as follows: Roll Number,Name,Marks. Every .csv file will have a heading and then rows of data for various students. For running your bash script, you need to type: “bash submission.sh command any other extra arguments(if needed)”. Here are some

### 4.1 Combine

When we run the script with combine argument, it will create a file named `main.csv`. In the `main.csv` the header is Roll Number,Name,List of exams. Basically combine should do combining all the .csv files which are present in our current directory. And if a student is absent for some particular exam then his name will not present in that corresponding exam. Combine argument will do like it will represent as “a” if he is absent in particular exam.

## 4.2 Upload

There should be a provision of uploading/importing new csv files. So, on running the command “bash submission.sh upload /Desktop/project.csv, your bash script should copy the corresponding file into the script’s directory. Now, whenever you run the “combine” command next, your `main.csv` file will be updated accordingly.

When we run the script with upload argument ,the script will copy the provided path of .csv file into script’s directory.

## 4.3 Total

Total command will simply create a total column in `main.csv` without disturbing the previous header.And total for each row is updated and appended under the total column for each student.In `main.csv` if a student is absent it is shown as "a".On calculating the total for each row our script will consider "a" as 0 marks.

If we try to run this toal argument multiple times it will throw an error message like "Error: 'Total' header already exists in main.csv. Script cannot be executed."

## 4.4 git init

This command on running will initialize the current repository.It will ask for a path of folder to create .git folder in the current repository. If there exist already a .git file then it will show a message like "Reinitialized existing Git repository".

The git init command initializes a new Git repository in the current directory or in a specified directory. When executed, it creates a hidden directory named .git, which serves as the repository’s metadata store. This directory contains essential files and subdirectories that Git uses to manage the repository’s history, configuration settings, and object database. By running git init, developers can start tracking changes to their project files, commit snapshots of the project at different stages, and leverage Git’s version control features such as branching, merging, and collaboration. Essentially, git init marks the beginning of version control for a project, allowing developers to systematically manage their codebase and track changes over time.

## 4.5 git commit

This command on running will create a subfolder in .git directory having name as it’s commit id.This sub folder will store the all the committed changes at that particular instant , producing a timestamp coressponding to present time.

Git commit is a pivotal operation in version control systems, enabling developers to record changes to their codebase systematically. Before committing, developers stage changes using the git add command,but here we directly add all the files present in the current directory, allowing granular control over which modifications to include. When a developer commits changes with git commit, Git records the staged changes as a new commit object in the repository’s history, complete with metadata such as the author’s information and a descriptive commit message. Each commit represents a snapshot of the project’s state at a specific point in time and is immutable, ensuring the integrity and traceability of the project’s history.

## 4.6 Update

It is used to update the marks of student.When we run this script it will ask for an input roll number,name,exam name .If all the three matches then the code will proceed otherwise it will throw an error message like "Error: Roll number and name do not match".If matches then it will ask for to enter the updated marks ,finally the marks are updated in the corresponding exam file.

## 5 Customisations

The files `plot1.py` and `plot2.py` will provide the graphs.

1. The `plot1.py` will provide the graphs for each exam having names on x-axis and marks on y-axis , and it will provide statistics like mean,median,standard deviation with their values on top left side of each subplot,horizontal lines are drawn at mean,median,standard deviation with different colour to notice them.
2. The `plot2.py` will provide a bar graph having total marks on y-axis and roll numbers on x-axis and will indicate the mean,median,standard deviation on graphs with different colours.

## 6 Conclusion

Hence