NATIONAL UNIVERSITY OF MODERN LANGUAGES

ISLAMABAD

Operating System (LAB PROJECT)

Submitted to
Dr. Messam Raza

Group Members
Adeel Naeem
(BSAI-146)

Junaid Asif

(BSAI-144)

Syed Qasim Ali

(BSAI-161)

Submission Date: December 16, 2024

Title: Class Maintenance System

Objective

The objective of the Class Maintenance System is to create a simple and effective tool for managing student records and attendance using Ubuntu commands. The system is implemented with Bash scripting and leverages standard Linux utilities to perform various operations like adding, updating, deleting, and viewing records.

Initial Setup

- **Objective**: Set up the system directory and create necessary files for storing student records.

- **Commands Used**:

  - mkdir class_maintenance: Create a directory named class_maintenance to store all related files.

  - cd class_maintenance: Navigate to the created directory.

Creating a File for Student Records

- **Objective**: Create a file to store student information.

- **Commands Used**:

  - cat > student.txt: Create a file named student.txt and allow the user to input student data. Input is in a structured format (e.g., ID,Name,Program).

```
root@DESKTOP-677C6JO:~# mkdir class_maintenance
root@DESKTOP-677C6JO:~# cd class_maintanence
-bash: cd: class_maintanence: No such file or directory
root@DESKTOP-677C6JO:~# cd class_maintenance
root@DESKTOP-677C6JO:~/class_maintenance# cat >student.txt
146,adeel naeem,BSAI
144,junaid asif,BSAI
161,syyed qasim ali,BSAI
root@DESKTOP-677C6JO:~/class_maintenance# cat >attendence.txt
4-12-24,146,present
4-12-24,144,present
4-12-24,161,absent
root@DESKTOP-677C6JO:~/class_maintenance# cat attendence.txt
4-12-24,146,present
4-12-24,144,present
4-12-24,161,absent
```

Features

The system is divided into the following modules:

1. Student Management

- Add student details (ID, Name, Class).

- Update existing student details.

- Delete student records.

- View all student records.

2. Attendance Management

- Add attendance for students.

- Update attendance records.

- Delete attendance records.

- View all attendance records.

Commands Used

File Management

- **[-f filename]**: Check if a file exists.

   o Ensures that required files (student.txt, attendance.txt) are available.

- **cat**: Display the contents of a file.

   o Used to display records to users.

Input Handling

- **read -p**: Prompt the user for input.

   o Collects student details, attendance details, or user-selected menu options.

Decision Making

- **if-else**: Handles conditional logic.

   o Determines if a record exists or if user input is valid.

- **case**: Implements a menu-driven interface.

   o Directs the user to specific operations based on their selection.

Search and Validation

- **grep -q**: Searches for patterns in files.

   o Validates the existence of a record before updating or deleting.

Data Manipulation

- **Appending Data (>>)**:
  - o Adds new records to the end of the file.
  - o Example: echo "$id,$name,$class" >> student.txt.
- **sed**: Stream editor for text manipulation.
  - o Update records: Replace specific lines in the file.
    - ▪ Example: sed -i "/^$id,/c\\$id,$name,$class" student.txt.
  - o Delete records: Remove specific lines from the file.
    - ▪ Example: sed -i "/^$id,/d" student.txt.

Feedback and Output

- **echo**: Prints success or error messages.
  - o Example: echo "Student added successfully" confirms a successful operation.

Viewing Records

- **cat**: Displays all records stored in files.
  - o Example: cat student.txt shows all student information.

```bash
#!/bin/bash
[ ! -f student.txt ] && cat student.txt
[ ! -f attendence.txt ] && cat attendence.txt
echo  "class system ment "
echo   "1.Add student"
echo   "2.update student"
echo   "3.Delete student"
echo   "4.view student"
echo   "5.Add attendence"
echo   '6.update attendence '
echo   "7.delete attendence"
echo   "8.View attendence"
read -p "choose an option :" option
case $option in
1)
read -p "Enter ID: " id
read -p "Enter Name: "  name
read -p "Enter class : " class

 echo $id,$name,$class>>student.txt
 echo "Student added successfully"
 ;;
2)
read -p "Enter ID of a student to update :" id
if grep -q "^$id," student.txt;
then
read -p "Enter new name: " name
read -p "Enter new Class: " class
 sed -i "/^$id,/c\\$id,$name,$class " student.txt
 echo "student added successfully"
else
 echo "student ith ID $id not found"
fi
;;
3)
 read -p  "enter ID of a student to delete : " id
if grep -q "^$id," student.txt;
then
 sed -i "/^$id,/d" student.txt
 echo "student deleted successfully"
```

```bash
3)
 read -p  "enter ID of a student to delete : " id
if grep -q "^$id," student.txt;
then
 sed -i "/^$id,/d" student.txt
 echo "student deleted successfully"
else
 echo "student with id $id is not found"
fi
;;
4)

 echo "student Record"
 cat student.txt
;;
5)
read -p  "Enter date : " date
read -p "Enter  Student ID : " id
read -p "Enter status(present/absent) : " status
 echo "$date,$id,$status " >> attendence.txt
 echo "attendence added successfully" ;;
6)
read -p "Enter date and student ID to update attendence : " search
if grep -q "^$search," attendence.txt;
then
read -p "enter new status: " status
 sed -i "/^$search,/c\\$search,$status" attendence.txt
 echo "attendence updated successfully "
else
 echo "attendence record not found "
fi
;;
7)
read -p "Enter date and student ID to delete Record : " search
if
 grep -q "^$search," attendence.txt ;
then
 sed -i "/^$search,/d" attendence.txt
 echo "attendence deleted Sucessfully"
else
```

```
^G Help        ^O Write Out    ^W Where Is     ^K Cut
^X Exit        ^R Read File    ^\ Replace      ^U Paste
```

```
  echo "attendence deleted Sucessfully"
else
  echo "record not found"
fi
;;
8)

  echo  "attendence record "
  cat attendence.txt
;;
*)
  echo "Invalid option"
  ;;
esac
```

Implementation Flow

1. **Startup Check**

   o The script verifies if the required files (student.txt and attendance.txt) exist. If they don't, it proceeds without errors.

2. **Interactive Menu**

   o Presents users with options for managing student and attendance data.

   o Users select operations by entering a corresponding number.

3. **Perform Operations**

   o Based on user input, the script executes:

      ▪ Add, update, or delete operations.

      ▪ View existing records.

4. **Error Handling**

   o Ensures operations are performed only if valid records exist.

   o Displays appropriate error messages for invalid actions.

5. **Data Persistence**

   o All changes are saved in text files for future reference.


```
root@DESKTOP-677C6JO:~/class_maintenance# nano class_system.sh
root@DESKTOP-677C6JO:~/class_maintenance# ./class_system.sh
class system ment
1.Add student
2.update student
3.Delete student
4.view student
5.Add attendence
6.update attendence
7.delete attendence
8.View attendence
choose an option :4
student Record
146,Adeel Naeem,OS
144,Junaid Asif,DM
161,Syed Qasim Ali,DM
root@DESKTOP-677C6JO:~/class_maintenance#
```

Advantages

1. **Simplicity**:

   o Uses basic Bash commands and text files, eliminating the need for complex database systems.

2. **Portability**:

   o Can run on any Linux/Ubuntu system with Bash installed.

3. **Modular Design**:

   o Easy to extend or modify for additional functionalities.

4. **Low Resource Usage**:

   o Does not require heavy computational resources or external tools.

Example Workflow

1. **Start the Script**:

   o The user runs the script on a terminal.

2. **Select an Operation**:

   o Example: Add a new student by choosing "1" from the menu.

3. **Input Data**:

   o Enter the student ID, name, and class when prompted.

4. **Verify Operation**:

   o View the updated records using the "View student" option.

5. **Manage Attendance**:

   o Add, update, or delete attendance records as required.

```
root@DESKTOP-677C6JO:~/class_maintenance# nano class_system.sh
root@DESKTOP-677C6JO:~/class_maintenance# ./class_system.sh
class system ment
1.Add student
2.update student
3.Delete student
4.view student
5.Add attendence
6.update attendence
7.delete attendence
8.View attendence
choose an option :4
student Record
146,Adeel Naeem,OS
144,Junaid Asif,DM
161,Syed Qasim Ali,DM
root@DESKTOP-677C6JO:~/class_maintenance#
```

Conclusion

The Class Maintenance System is a simple yet powerful solution for managing student and attendance records. It leverages the flexibility of Bash scripting and standard Ubuntu commands to provide an efficient and user-friendly interface. By focusing on core Linux utilities, the system ensures ease of use, portability, and extensibility.