Your 20% Linux Shell Scripting Assessment 2025

Contents

ASSESSMENT SPECIFICATION

The goal of this project is to, using Linux bash shell scripting, create a system that automates the process of tracking and managing records in a system of choice.

Your system of choice is totally your own preference, where some, of many, possibilities might be:

- Conference attendance catalogue

- Vehicles catalogue

- Plants catalogue

- Endangered species catalogue

- Fuel prices catalogue

- Music Events catalogue etc.

The system will allow the user, at a minimum, to:

- **add** in full details of new records, with *minimal* input being name, phone number & email address

- be able to **search** records within the system and **output** in a user friendly manner,

- be able to **remove** records from the system,

- provide a user-friendly interface for managing the data and

- be able to exit at any stage.

This should all be created with user friendliness and error handling in mind.

Your 20% Linux Shell Scripting Assessment 2025

Features of the system

1. **Data Input to Add a New Record:**
   - Read information from the user and store new records appropriately
2. **Search data:**
   - Allow high level searches with filtering
3. **Remove data:**
   - Allow specific record(s) to be deleted.
4. **User Interaction:**
   - Create a user-friendly menu-based interface for the end user.
   - Options in the menu should include facilities such as:

     add,

     remove,

     view overall data,

     view specific data,

     return to main menu and to

     exit.

5. **Error Handling:**
   - Implement appropriate error handling such as (but not limited to):
     1. Check the expected user inputs i.e. is the mobile number digits only?
     2. Allow the user to exit from within any menu
     3. Ensure there are no infinite loops

Your 20% Linux Shell Scripting Assessment 2025

5%   **Lab Scripting Homework**   End of 1st Reading Week

5%   **NetAcad Exams Chapters 1-11 incl**   End of 2nd Reading Week

25%   **CA Linux Shell Scripting**   End of 2nd Reading Week

SUBMISSION REQUIREMENTS **- 25% Linux Shell Scripting**

You are required to take the specification above and create a suitable system for the end user

You are required to upload to Moodle:

1) **Your code**

You may wish to compress all your code into one directory

Ensure you've *not included* any hardcoded links to scripts on your own machine (or I won't be able to access)

2) A **link to your video demonstration**

You are required to demonstrate the execution of your system, where:

> this video is to be a MAX. of 10 minutes long
> it is not necessary that you appear in this video, this is optional but I do need your voiceover to describe your coding approach

HALF AND HALF: When recording your <10 minute video demonstration, please:

- have the **code open** on one side of the screen and
- **demo** the execution of this code on the other side of the screen (where you will describe the inner workings and your approach to the scripting i.e. "I used an exit code here to ensure that…", "I choose to use awk here to…")

Your Linux Shell Scripting Assessment 2024 Grading Rubric

Shell Programming Sample Assignment Grading Rubric			
NAME			**COMMENTS**
Workings (30)	Menu	3	
	Add new unique record	8	
	Search for record(s)	8	
	Remove a single record	6	
	Exit (an optional action at all times)	5	
Robustness (35)	Unexpected data: error handling of something other than menu options	2	
	Can a user return to the main menu easily?	4	
	Error handling of a user adding blank field(s)	2	
	Is user input correct format i.e. is phone numbers only, email in correct format etc.?	7	
	Can user find duplicate records (maybe the same name) & return appropriate clear message to user?	5	
	Does the user have options to (i) delete multiple records at once and (ii) delete one specific record from list?	10	
	Can a user verify info is correct before storing/deleting etc?	5	
Usability (20)	General appearance & layout of UI	6	
	Interface easy to understand and navigate with no internal looping	7	
	Are there any UI extras i.e. colour schemes used, welcome note etc.	3	
	Is case insensitive implemented?	2	
	Can a user view contents if required i.e. before searching etc.?	2	
Comments, Code Style & video demo (15)	Good code comments	3	
	Indentation & spacing	4	
	Video: logical demo, audible, good oral presentation	8	
Total		*100*	

-

```bash
#!/bin/bash

students_records="students"
reports_directory="reports"

clear
while true; do
    echo "Student Progress"
    echo "1. View student details"
    echo "2. Report on any students needing intervention"
    echo "3. Exit"
    read -p "Select an option: " choice

    case $choice in
        1)
            read -p "Enter student ID: " student_id
            if [ -f "$students_records/$student_id.txt" ]; then
                cat "$students_ records/$student_id.txt"
            else
                echo "Student not found." # what should happen for the end user next?
            fi
            ;;
        2)
            # Logic to read user input & generate a report on students averaging below a threshold
            ;;
        3)
            echo "Exiting..."
            exit 0
            ;;
        *)
            echo "Invalid option. Please choose a valid menu option"
            ;;
    esac
done
```

Figure 1: Sample draft of code

Student ID: 20102010
Name: Caroline Cahill
Exam Results: 45,36,37,34

Figure 2 Sample Student file

Student Progress Reporting System
1. View student details
2. Report on any students needing intervention
3. Exit
Choose one of the options above: 1
Enter Student: 20102010   #should a user be able to view all students to pick a field to filter by?
Here are the details for Student ID: 20102010
Student ID: 20102010
Name: Caroline Cahill
Exam Results: 45,56,67,54

Student Progress Reporting System
1. View student details
2. Report on any students needing intervention
3. Exit
Choose one of the options above: 2
You wish to check if a students average is below a particular threshold.
Please enter the threshold percentage: 40%
REPORT: The following students are currently averaging below the threshold of **40%**
Caroline Cahill   Average: 38%

Student Progress Reporting System
1. View student details
2. Report on any students needing intervention
3. Exit
Choose one of the options above: 3
Are you sure you wish to exit the system [Y/N]: y
Exiting…

Figure 3: Sample Report Generation