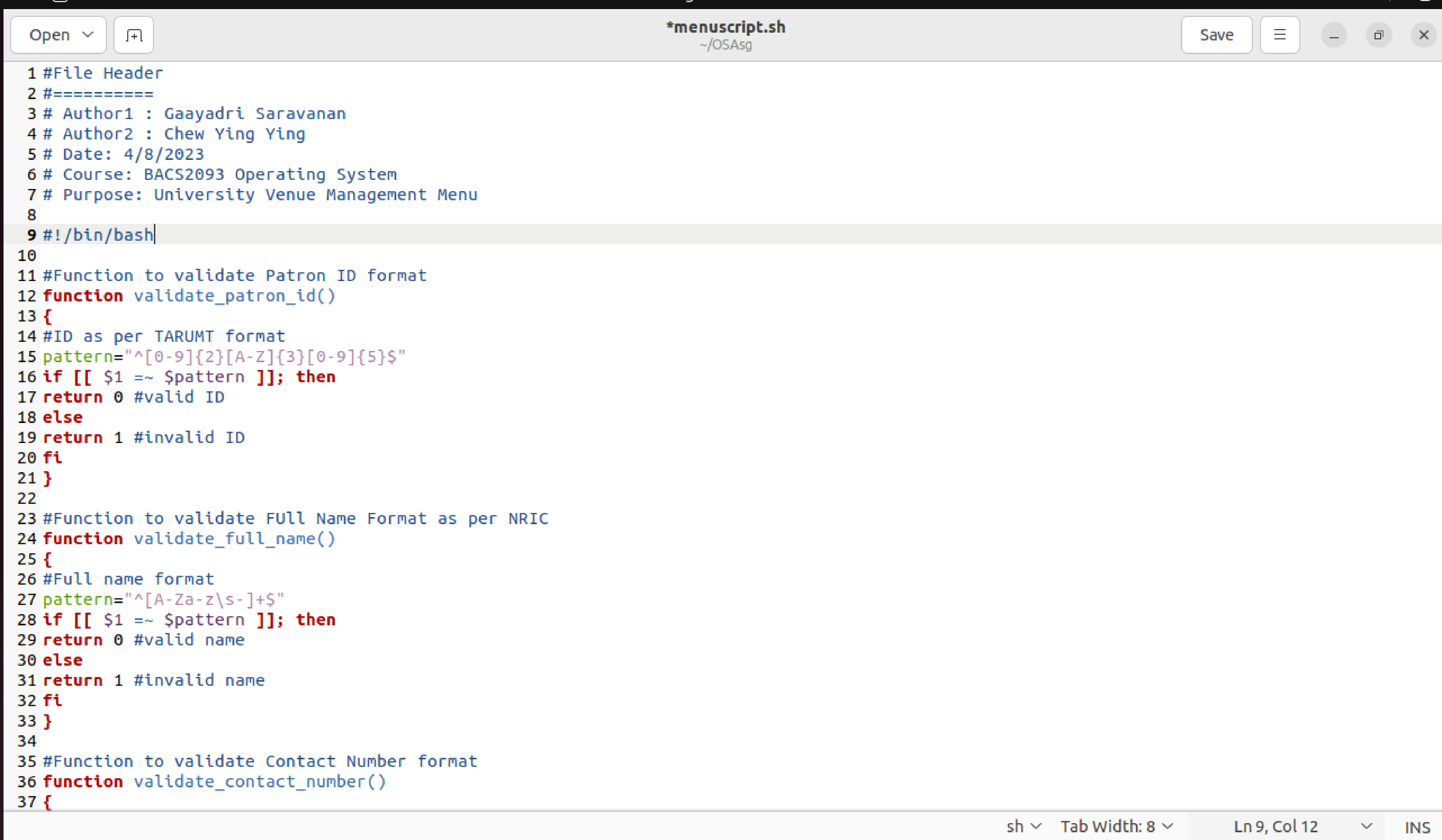
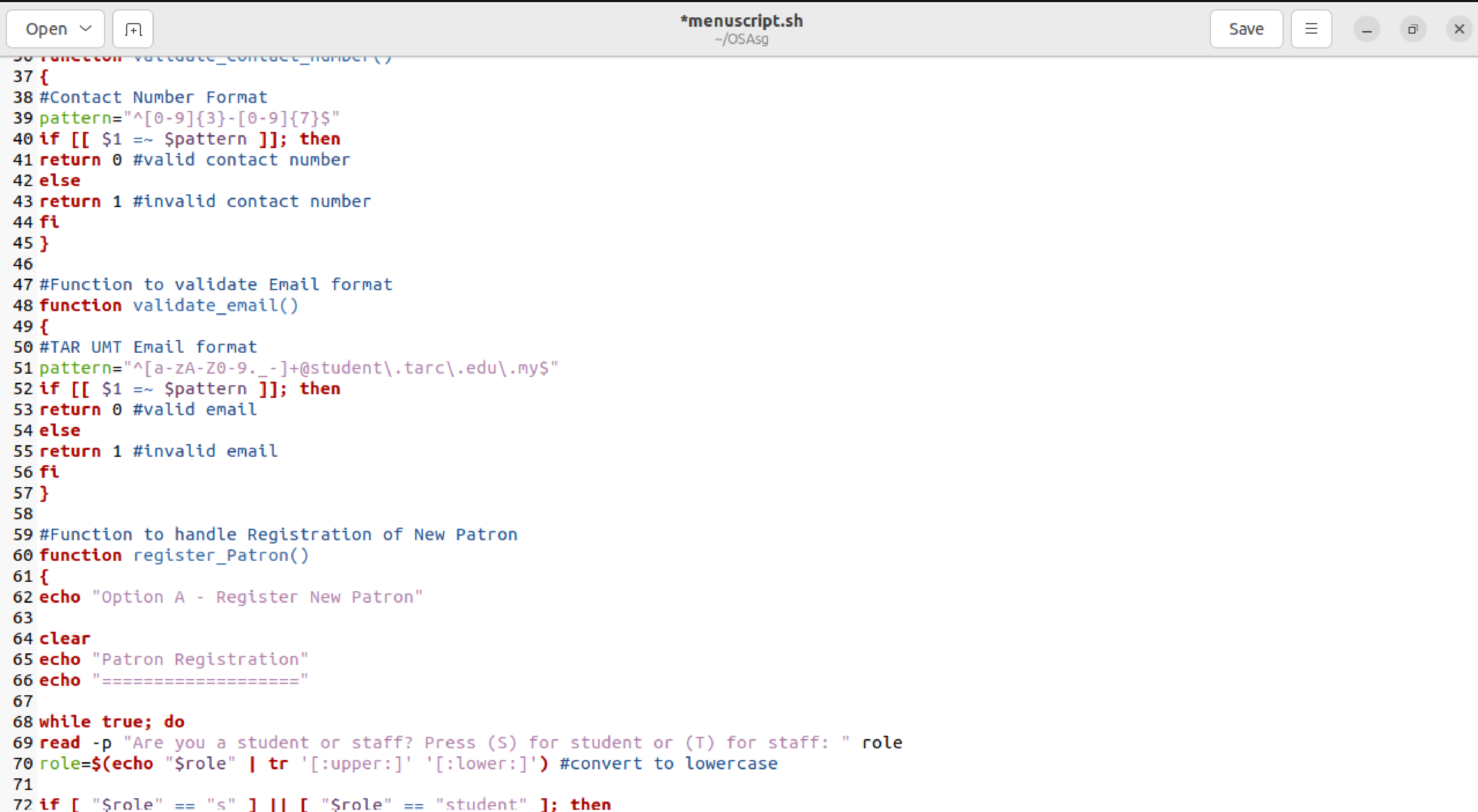
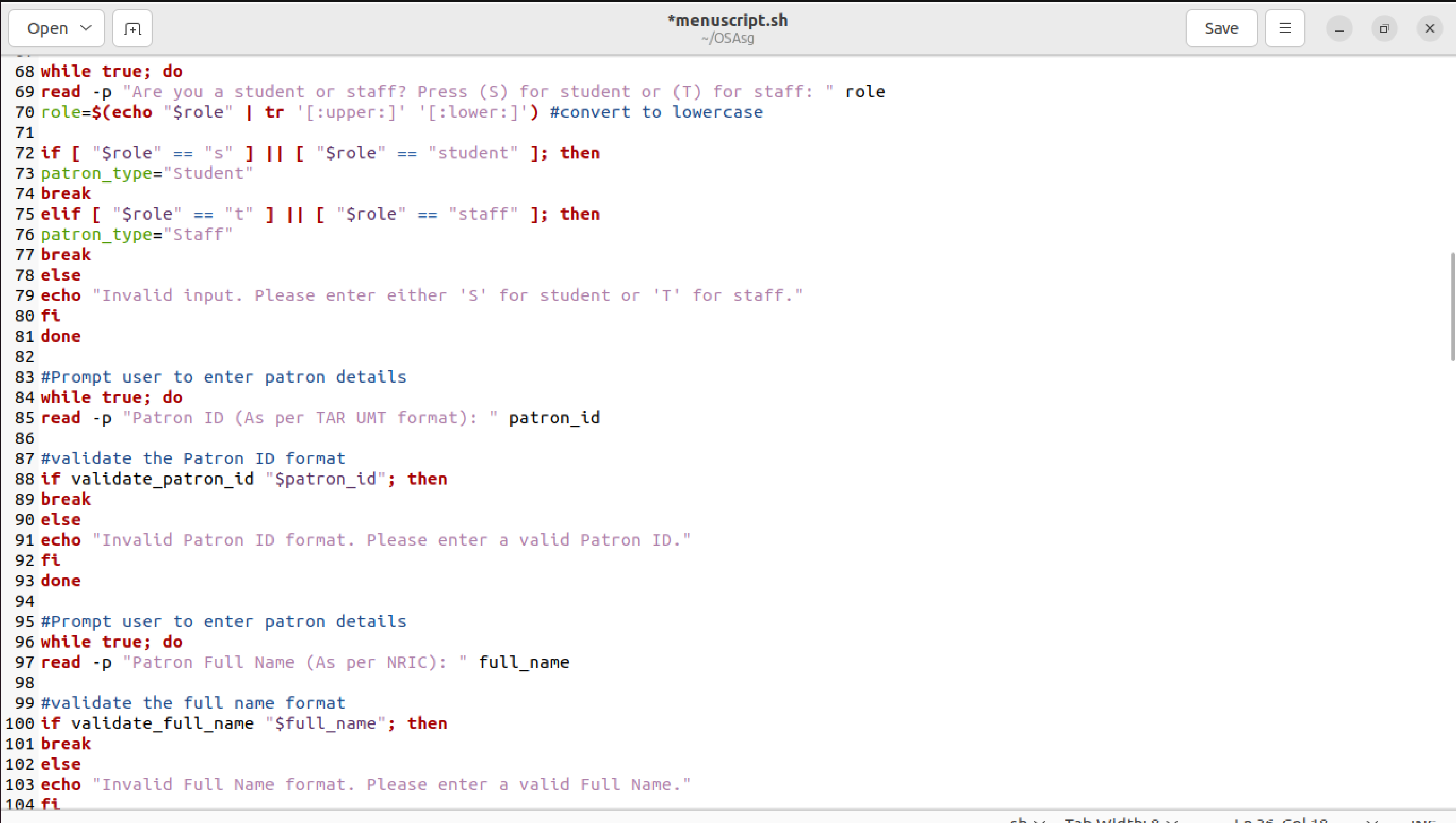


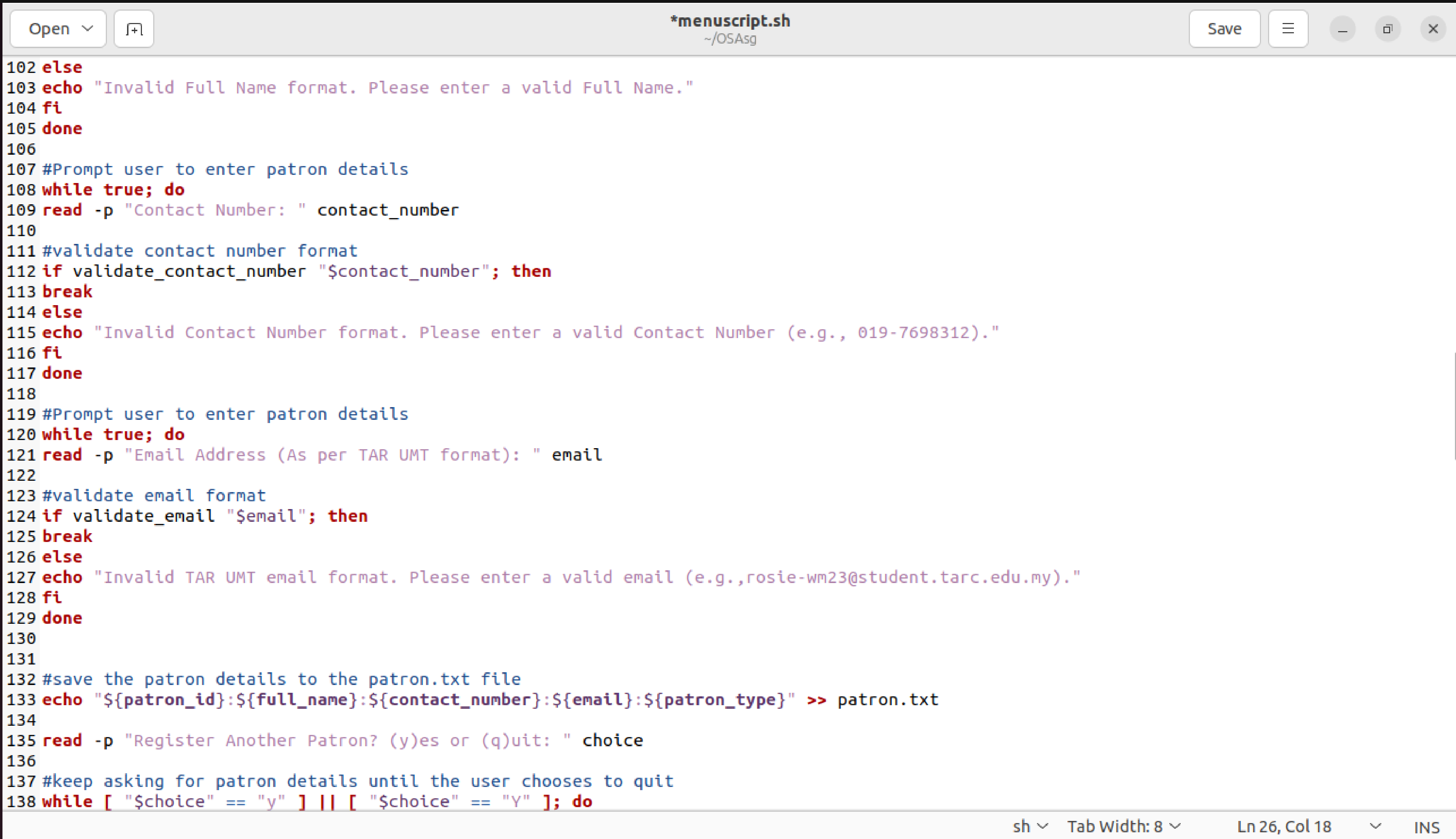
```
#File Header
#==========
# Author1 : Gaayadri Saravanan
# Author2 : Chew Ying Ying
# Date: 4/8/2023
# Course: BACS2093 Operating System
# Purpose: University Venue Management Menu

#!/bin/bash

#Function to validate Patron ID format
function validate_patron_id()
{
#ID as per TARUMT format
pattern="^[0-9]{2}[A-Z]{3}[0-9]{5}$"
if [[ $1 =~ $pattern ]]; then
return 0 #valid ID
else
return 1 #invalid ID
fi
}

#Function to validate FUll Name Format as per NRIC
function validate_full_name()
{
#Full name format
pattern="^[A-Za-z\s-]+$"
if [[ $1 =~ $pattern ]]; then
return 0 #valid name
else
return 1 #invalid name
fi
}

#Function to validate Contact Number format
function validate_contact_number()
{
```



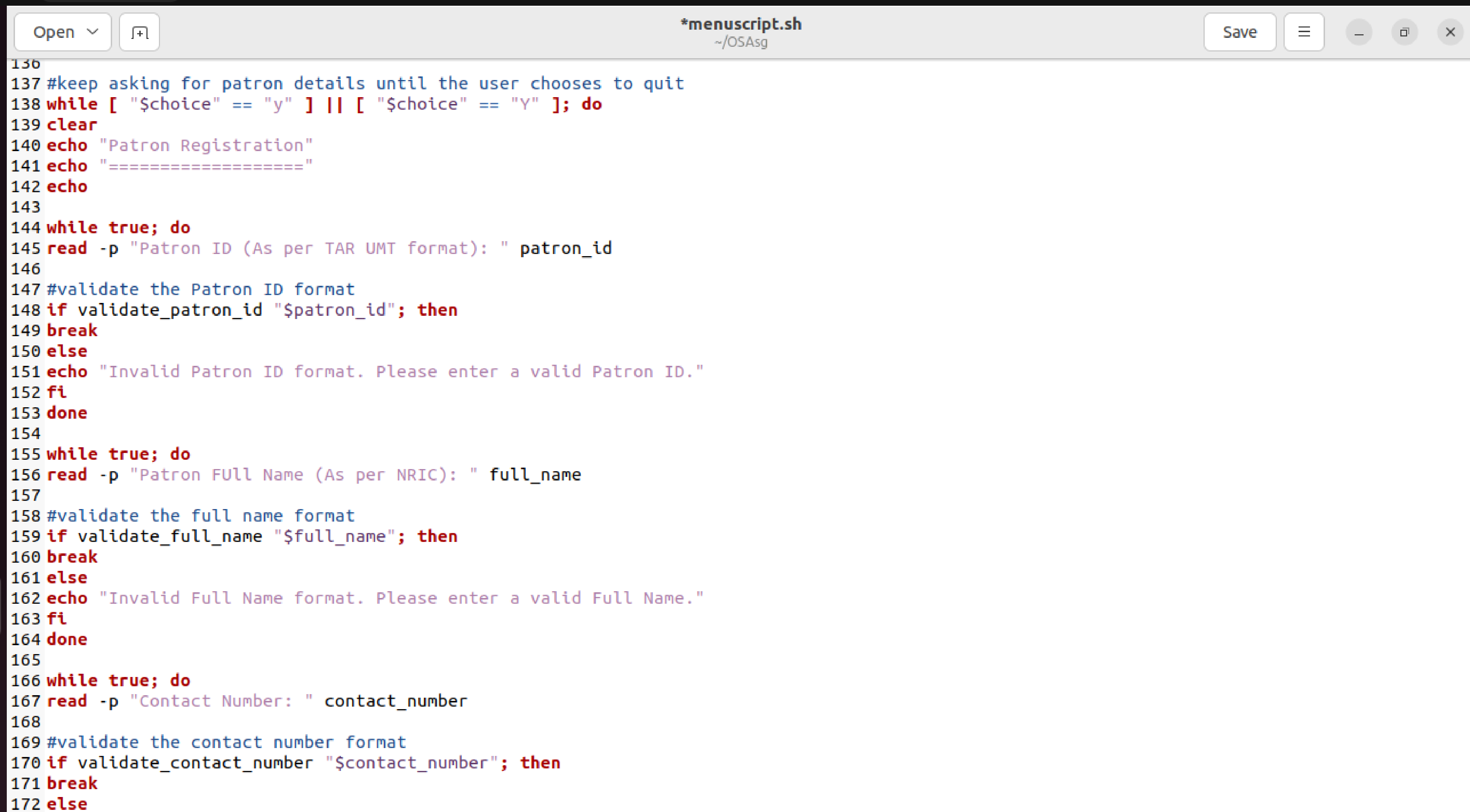
```
function validate_contact_number()
{
#Contact Number Format
pattern="^[0-9]{3}-[0-9]{7}$"
if [[ $1 =~ $pattern ]]; then
return 0 #valid contact number
else
return 1 #invalid contact number
fi
}

#Function to validate Email format
function validate_email()
{
#TAR UMT Email format
pattern="^[a-zA-Z0-9._-]+@student\.tarc\.edu\.my$"
if [[ $1 =~ $pattern ]]; then
return 0 #valid email
else
return 1 #invalid email
fi
}

#Function to handle Registration of New Patron
function register_Patron()
{
echo "Option A - Register New Patron"

clear
echo "Patron Registration"
echo "==================="

while true; do
read -p "Are you a student or staff? Press (S) for student or (T) for staff: " role
role=$(echo "$role" | tr '[:upper:]' '[:lower:]') #convert to lowercase

if [ "$role" == "s" ] || [ "$role" == "student" ]; then
```



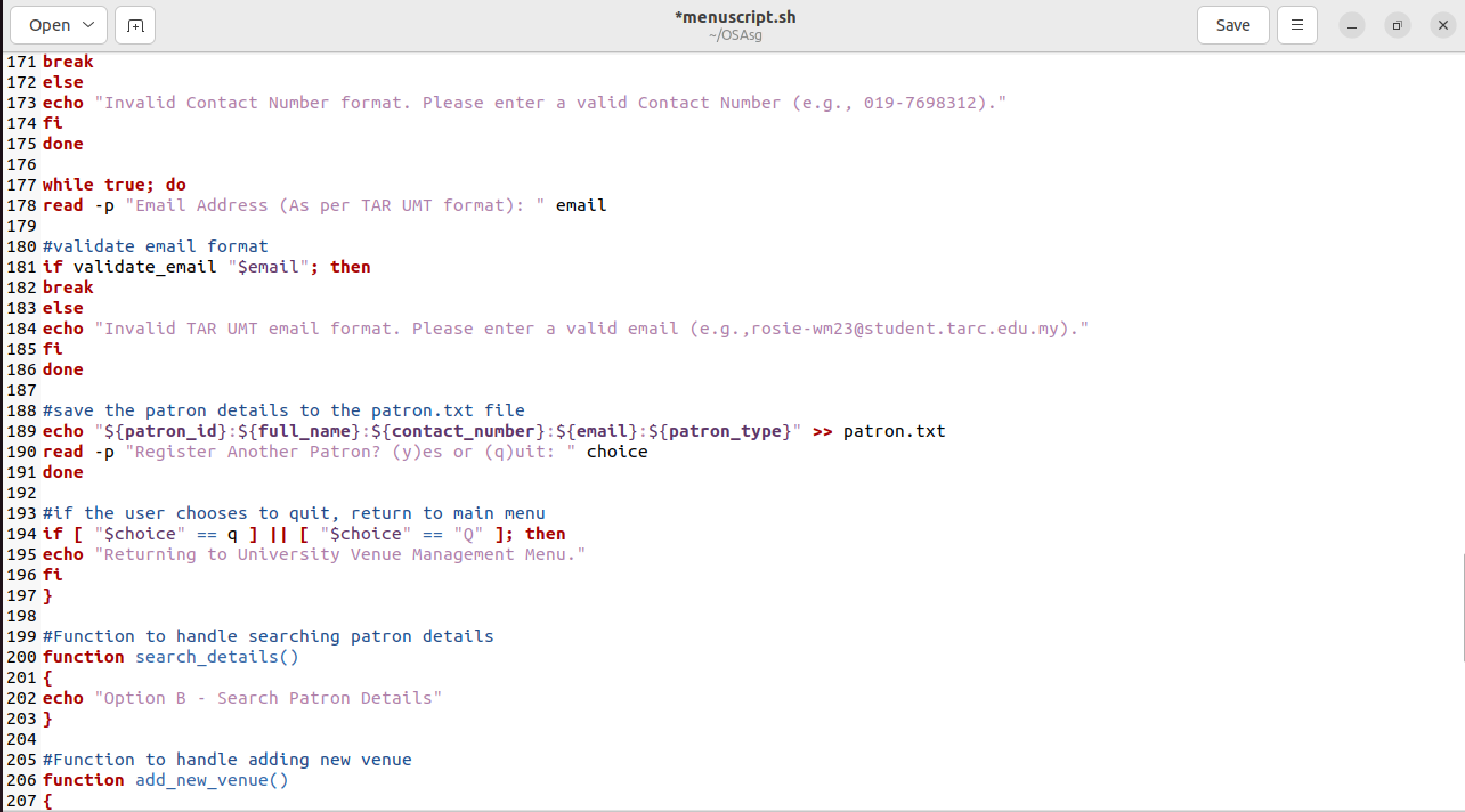
```
while true; do
read -p "Are you a student or staff? Press (S) for student or (T) for staff: " role
role=$(echo "$role" | tr '[:upper:]' '[:lower:]') #convert to lowercase

if [ "$role" == "s" ] || [ "$role" == "student" ]; then
patron_type="Student"
break
elif [ "$role" == "t" ] || [ "$role" == "staff" ]; then
patron_type="Staff"
break
else
echo "Invalid input. Please enter either 'S' for student or 'T' for staff."
fi
done

#Prompt user to enter patron details
while true; do
read -p "Patron ID (As per TAR UMT format): " patron_id

#validate the Patron ID format
if validate_patron_id "$patron_id"; then
break
else
echo "Invalid Patron ID format. Please enter a valid Patron ID."
fi
done

#Prompt user to enter patron details
while true; do
read -p "Patron Full Name (As per NRIC): " full_name

#validate the full name format
if validate_full_name "$full_name"; then
break
else
echo "Invalid Full Name format. Please enter a valid Full Name."
fi
```



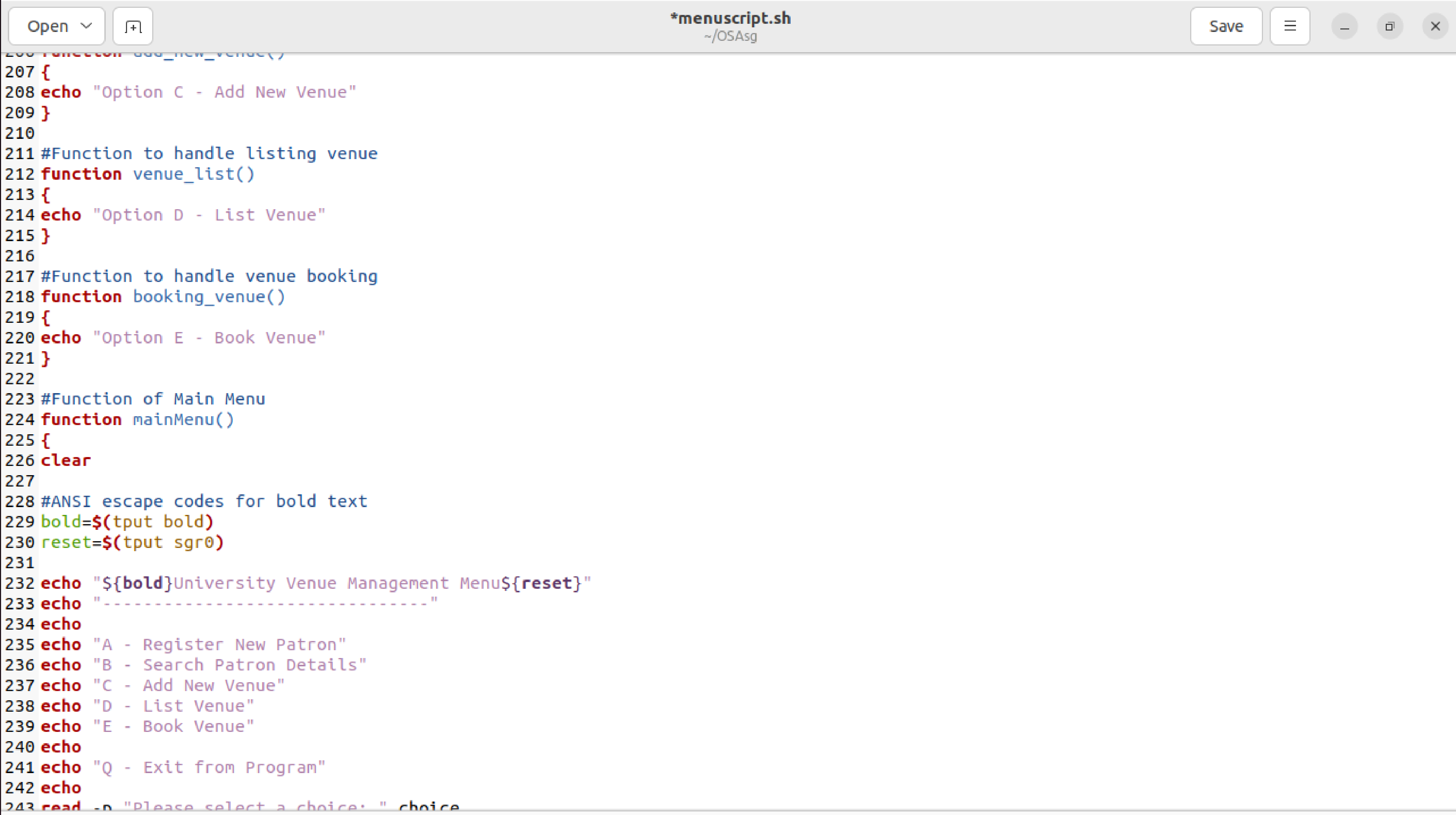
```
else
echo "Invalid Full Name format. Please enter a valid Full Name."
fi
done

#Prompt user to enter patron details
while true; do
read -p "Contact Number: " contact_number

#validate contact number format
if validate_contact_number "$contact_number"; then
break
else
echo "Invalid Contact Number format. Please enter a valid Contact Number (e.g., 019-7698312)."
fi
done

#Prompt user to enter patron details
while true; do
read -p "Email Address (As per TAR UMT format): " email

#validate email format
if validate_email "$email"; then
break
else
echo "Invalid TAR UMT email format. Please enter a valid email (e.g.,rosie-wm23@student.tarc.edu.my)."
fi
done


#save the patron details to the patron.txt file
echo "${patron_id}:${full_name}:${contact_number}:${email}:${patron_type}" >> patron.txt

read -p "Register Another Patron? (y)es or (q)uit: " choice

#keep asking for patron details until the user chooses to quit
while [ "$choice" == "y" ] || [ "$choice" == "Y" ]; do
```



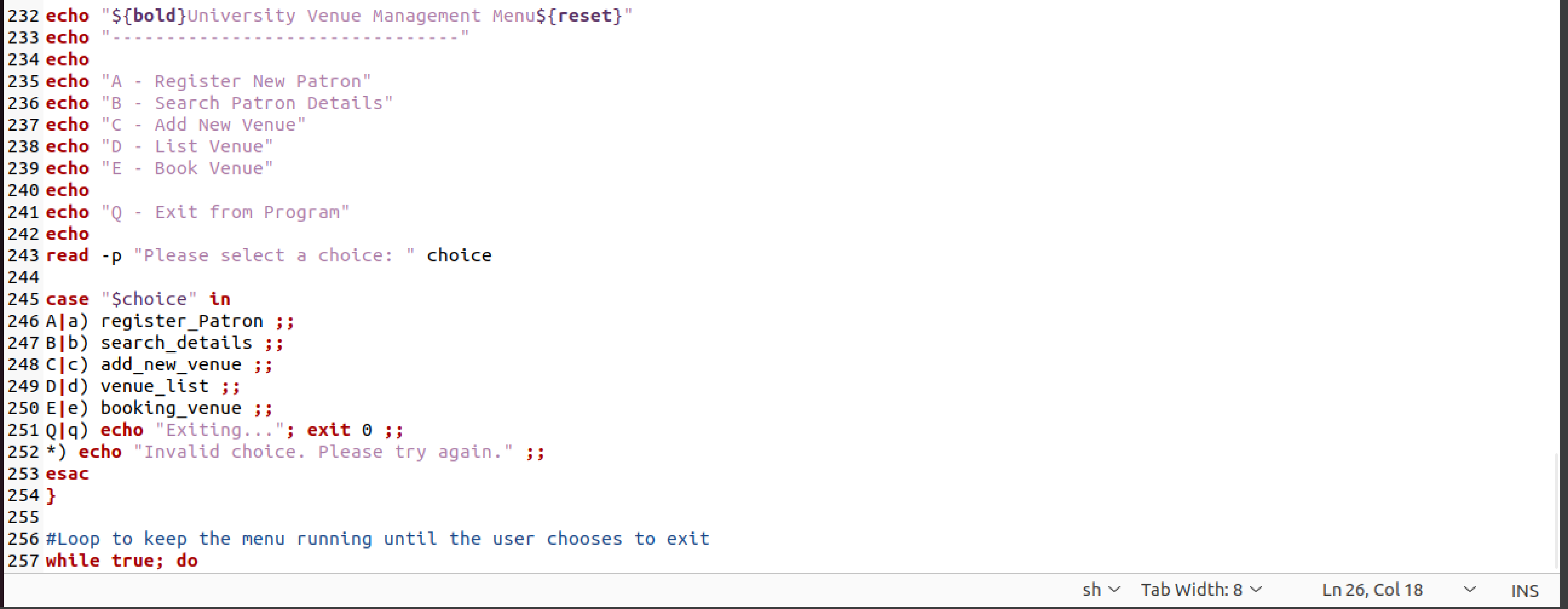
```
#keep asking for patron details until the user chooses to quit
while [ "$choice" == "y" ] || [ "$choice" == "Y" ]; do
clear
echo "Patron Registration"
echo "==================="
echo

while true; do
read -p "Patron ID (As per TAR UMT format): " patron_id

#validate the Patron ID format
if validate_patron_id "$patron_id"; then
break
else
echo "Invalid Patron ID format. Please enter a valid Patron ID."
fi
done

while true; do
read -p "Patron FUll Name (As per NRIC): " full_name

#validate the full name format
if validate_full_name "$full_name"; then
break
else
echo "Invalid Full Name format. Please enter a valid Full Name."
fi
done

while true; do
read -p "Contact Number: " contact_number

#validate the contact number format
if validate_contact_number "$contact_number"; then
break
else
```



```
break
else
echo "Invalid Contact Number format. Please enter a valid Contact Number (e.g., 019-7698312)."
fi
done

while true; do
read -p "Email Address (As per TAR UMT format): " email

#validate email format
if validate_email "$email"; then
break
else
echo "Invalid TAR UMT email format. Please enter a valid email (e.g.,rosie-wm23@student.tarc.edu.my)."
fi
done

#save the patron details to the patron.txt file
echo "${patron_id}:${full_name}:${contact_number}:${email}:${patron_type}" >> patron.txt
read -p "Register Another Patron? (y)es or (q)uit: " choice
done

#if the user chooses to quit, return to main menu
if [ "$choice" == q ] || [ "$choice" == "Q" ]; then
echo "Returning to University Venue Management Menu."
fi
}

#Function to handle searching patron details
function search_details()
{
echo "Option B - Search Patron Details"
}

#Function to handle adding new venue
function add_new_venue()
{
```



```
function add_new_venue()
{
echo "Option C - Add New Venue"
}

#Function to handle listing venue
function venue_list()
{
echo "Option D - List Venue"
}

#Function to handle venue booking
function booking_venue()
{
echo "Option E - Book Venue"
}

#Function of Main Menu
function mainMenu()
{
clear

#ANSI escape codes for bold text
bold=$(tput bold)
reset=$(tput sgr0)

echo "${bold}University Venue Management Menu${reset}"
echo "---------------------------------"
echo
echo "A - Register New Patron"
echo "B - Search Patron Details"
echo "C - Add New Venue"
echo "D - List Venue"
echo "E - Book Venue"
echo
echo "Q - Exit from Program"
echo
read -p "Please select a choice: " choice
```



```
echo "${bold}University Venue Management Menu${reset}"
echo "--------------------------------"
echo
echo "A - Register New Patron"
echo "B - Search Patron Details"
echo "C - Add New Venue"
echo "D - List Venue"
echo "E - Book Venue"
echo
echo "Q - Exit from Program"
echo
read -p "Please select a choice: " choice

case "$choice" in
A|a) register_Patron ;;
B|b) search_details ;;
C|c) add_new_venue ;;
D|d) venue_list ;;
E|e) booking_venue ;;
Q|q) echo "Exiting..."; exit 0 ;;
*) echo "Invalid choice. Please try again." ;;
esac
}

#Loop to keep the menu running until the user chooses to exit
while true; do
```

sh   Tab Width: 8   Ln 26, Col 18   INS