

```
1 #File Header
2 #=====
3 # Author1 : Gaayadri Saravanan
4 # Author2 : Chew Ying Ying
5 # Date: 4/8/2023
6 # Course: BACS2093 Operating System
7 # Purpose: University Venue Management Menu
8
9 #!/bin/bash|
10
11 #Function to validate Patron ID format
12 function validate_patron_id()
13 {
14 #ID as per TARUMT format
15 pattern="^[0-9]{2}[A-Z]{3}[0-9]{5}$"
16 if [[ $1 =~ $pattern ]]; then
17 return 0 #valid ID
18 else
19 return 1 #invalid ID
20 fi
21 }
22
23 #Function to validate Full Name Format as per NRIC
24 function validate_full_name()
25 {
26 #Full name format
27 pattern="^[A-Za-z\s-]+$"
28 if [[ $1 =~ $pattern ]]; then
29 return 0 #valid name
30 else
31 return 1 #invalid name
32 fi
33 }
34
35 #Function to validate Contact Number format
36 function validate_contact_number()
37 {
```

Open ▾

*menuscrypt.sh
~/OSAsg

Save



```
36 function validate_contact_number()
37 {
38 #Contact Number Format
39 pattern="^[0-9]{3}-[0-9]{7}$"
40 if [[ $1 =~ $pattern ]]; then
41 return 0 #valid contact number
42 else
43 return 1 #invalid contact number
44 fi
45 }
46
47 #Function to validate Email format
48 function validate_email()
49 {
50 #TAR UMT Email format
51 pattern="^[a-zA-Z0-9._-]+@student\.tarc\.edu\.my$"
52 if [[ $1 =~ $pattern ]]; then
53 return 0 #valid email
54 else
55 return 1 #invalid email
56 fi
57 }
58
59 #Function to handle Registration of New Patron
60 function register_Patron()
61 {
62 echo "Option A - Register New Patron"
63
64 clear
65 echo "Patron Registration"
66 echo "======"
67
68 while true; do
69 read -p "Are you a student or staff? Press (S) for student or (T) for staff: " role
70 role=$(echo "$role" | tr '[:upper:]' '[:lower:]') #convert to lowercase
71
72 if [ "$role" == "s" ] || [ "$role" == "student" ]; then
```

```
68 while true; do
69 read -p "Are you a student or staff? Press (S) for student or (T) for staff: " role
70 role=$(echo "$role" | tr '[:upper:]' '[:lower:]') #convert to lowercase
71
72 if [ "$role" == "s" ] || [ "$role" == "student" ]; then
73 patron_type="Student"
74 break
75 elif [ "$role" == "t" ] || [ "$role" == "staff" ]; then
76 patron_type="Staff"
77 break
78 else
79 echo "Invalid input. Please enter either 'S' for student or 'T' for staff."
80 fi
81 done
82
83 #Prompt user to enter patron details
84 while true; do
85 read -p "Patron ID (As per TAR UMT format): " patron_id
86
87 #validate the Patron ID format
88 if validate_patron_id "$patron_id"; then
89 break
90 else
91 echo "Invalid Patron ID format. Please enter a valid Patron ID."
92 fi
93 done
94
95 #Prompt user to enter patron details
96 while true; do
97 read -p "Patron Full Name (As per NRIC): " full_name
98
99 #validate the full name format
100 if validate_full_name "$full_name"; then
101 break
102 else
103 echo "Invalid Full Name format. Please enter a valid Full Name."
104 fi
```

```
102 else
103 echo "Invalid Full Name format. Please enter a valid Full Name."
104 fi
105 done
106
107 #Prompt user to enter patron details
108 while true; do
109 read -p "Contact Number: " contact_number
110
111 #validate contact number format
112 if validate_contact_number "$contact_number"; then
113 break
114 else
115 echo "Invalid Contact Number format. Please enter a valid Contact Number (e.g., 019-7698312)."
116 fi
117 done
118
119 #Prompt user to enter patron details
120 while true; do
121 read -p "Email Address (As per TAR UMT format): " email
122
123 #validate email format
124 if validate_email "$email"; then
125 break
126 else
127 echo "Invalid TAR UMT email format. Please enter a valid email (e.g.,rosie-wm23@student.tarc.edu.my)."
128 fi
129 done
130
131
132 #save the patron details to the patron.txt file
133 echo "${patron_id}:${full_name}:${contact_number}:${email}:${patron_type}" >> patron.txt
134
135 read -p "Register Another Patron? (y)es or (q)uit: " choice
136
137 #keep asking for patron details until the user chooses to quit
138 while [ "$choice" == "y" ] || [ "$choice" == "Y" ]; do
```

Open ▾



*manuscript.sh

~/OSAsg

Save



```
136
137 #keep asking for patron details until the user chooses to quit
138 while [ "$choice" == "y" ] || [ "$choice" == "Y" ]; do
139 clear
140 echo "Patron Registration"
141 echo "======"
142 echo
143
144 while true; do
145 read -p "Patron ID (As per TAR UMT format): " patron_id
146
147 #validate the Patron ID format
148 if validate_patron_id "$patron_id"; then
149 break
150 else
151 echo "Invalid Patron ID format. Please enter a valid Patron ID."
152 fi
153 done
154
155 while true; do
156 read -p "Patron Full Name (As per NRIC): " full_name
157
158 #validate the full name format
159 if validate_full_name "$full_name"; then
160 break
161 else
162 echo "Invalid Full Name format. Please enter a valid Full Name."
163 fi
164 done
165
166 while true; do
167 read -p "Contact Number: " contact_number
168
169 #validate the contact number format
170 if validate_contact_number "$contact_number"; then
171 break
172 else
```

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

174 fi

175 done

176

177 while true; do

178 read -p "Email Address (As per TAR UMT format): " email

179

180 #validate email format

181 if validate_email "\$email"; then

182 break

183 else

184 echo "Invalid TAR UMT email format. Please enter a valid email (e.g.,rosie-wm23@student.tarc.edu.my)."

185 fi

186 done

187

188 #save the patron details to the patron.txt file

189 echo "\${patron_id}:\${full_name}:\${contact_number}:\${email}:\${patron_type}" >> patron.txt

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

191 done

192

193 #if the user chooses to quit, return to main menu

194 if ["\$choice" == q] || ["\$choice" == "Q"]; then

195 echo "Returning to University Venue Management Menu."

196 fi

197 }

198

199 #Function to handle searching patron details

200 function search_details()

201 {

202 echo "Option B - Search Patron Details"

203 }

204

205 #Function to handle adding new venue

206 function add_new_venue()

207 {

Open ▾

J+l

*menuscript.sh
~/OSAsg

Save

≡

—

□

✕

```
206 function add_new_venue()
207 {
208 echo "Option C - Add New Venue"
209 }
210
211 #Function to handle listing venue
212 function venue_list()
213 {
214 echo "Option D - List Venue"
215 }
216
217 #Function to handle venue booking
218 function booking_venue()
219 {
220 echo "Option E - Book Venue"
221 }
222
223 #Function of Main Menu
224 function mainMenu()
225 {
226 clear
227
228 #ANSI escape codes for bold text
229 bold=$(tput bold)
230 reset=$(tput sgr0)
231
232 echo "${bold}University Venue Management Menu${reset}"
233 echo "-----"
234 echo
235 echo "A - Register New Patron"
236 echo "B - Search Patron Details"
237 echo "C - Add New Venue"
238 echo "D - List Venue"
239 echo "E - Book Venue"
240 echo
241 echo "Q - Exit from Program"
242 echo
243 read -n 1 "Please select a choice: " choice
```

```
232 echo "${bold}University Venue Management Menu${reset}"
233 echo "-----"
234 echo
235 echo "A - Register New Patron"
236 echo "B - Search Patron Details"
237 echo "C - Add New Venue"
238 echo "D - List Venue"
239 echo "E - Book Venue"
240 echo
241 echo "Q - Exit from Program"
242 echo
243 read -p "Please select a choice: " choice
244
245 case "$choice" in
246 A|a) register_Patron ;;
247 B|b) search_details ;;
248 C|c) add_new_venue ;;
249 D|d) venue_list ;;
250 E|e) booking_venue ;;
251 Q|q) echo "Exiting..."; exit 0 ;;
252 *) echo "Invalid choice. Please try again." ;;
253 esac
254 }
255
256 #Loop to keep the menu running until the user chooses to exit
257 while true; do
```