

```
1 #! /bin/bash
2
3
4 name_array=('Constance_Castillo' 'Kerry_Goodwin' 'Dorothy_Carson'
... 'Craig_Williams'
5 'Daryl_Guzman' 'Sherman_Stewart' 'Marvin_Collier' 'Javier_Wilkerson'
... 'Lena_Olson'
6 'Claudia_George' 'Erik_Elliott' 'Traci_Peters' 'Jack_Burke' 'Jody_Turner'
... 'Kristy_Jenkins'
7 'Melissa_Griffin' 'Shelia_Ballard' 'Armando_Weaver' 'Elsie_Fitzgerald'
... 'Ben_Evans' 'Lucy_Baker'
8 'Kerry_Anderson' 'Kendra_Tran' 'Arnold_Wells' 'Anita_Aguilar'
... 'Earnest_Reeves' 'Irving_Stone'
9 'Alice_Moore' 'Leigh_Parsons' 'Mandy_Perez' 'Rolando_Paul' 'Delores_Pierce'
... 'Zachary_Webster'
10 'Eddie_Ward' 'Alvin_Soto' 'Ross_Welch' 'Tanya_Padilla' 'Rachel_Logan'
... 'Angelica_Richards'
11 'Shelley_Lucas' 'Alison_Porter' 'Lionel_Buchanan' 'Luis_Norman'
... 'Milton_Robinson' 'Ervin_Bryant'
12 'Tabitha_Reid' 'Randal_Graves' 'Calvin_Murphy' 'Blanca_Bell' 'Dean_Walters'
... 'Elias_Klein'
13 'Madeline_White' 'Marty_Lewis' 'Beatrice_Santiago' 'Willis_Tucker'
... 'Diane_Lloyd' 'Al_Harrison'
14 'Barbara_Lawson' 'Jamie_Page' 'Conrad_Reynolds' 'Darnell_Goodman'
... 'Derrick_Mckenzie'
15 'Erika_Miller' 'Tasha_Todd' 'Aaron_Nunez' 'Julio_Gomez' 'Tommie_Hunter'
... 'Darlene_Russell'
16 'Monica_Abbott' 'Cassandra_Vargas' 'Gail_Obrien' 'Doug_Morales' 'Ian_James'
... 'Jean_Moran'
17 'Carla_Ross' 'Marjorie_Hanson' 'Clark_Sullivan' 'Rick_Torres' 'Byron_Hardy'
... 'Ken_Chandler'
18 'Brendan_Carr' 'Richard_Francis' 'Tyler_Mitchell' 'Edwin_Stevens'
... 'Paul_Santos'
19 'Jesus_Griffith' 'Maggie_Maldonado' 'Isaac_Allen' 'Vanessa_Thompson'
... 'Jeremy_Barton'
20 'Joey_Butler' 'Randy_Holmes' 'Loretta_Pittman' 'Essie_Johnston'
... 'Felix_Weber' 'Gary_Hawkins'
21 'Vivian_Bowers' 'Dennis_Jefferson' 'Dale_Arnold' 'Joseph_Christensen'
... 'Billie_Norton'
22 'Darla_Pope' 'Tommie_Dixon' 'Toby_Beck' 'Jodi_Payne' 'Marjorie_Lowe'
... 'Fernando_Ballard'
23 'Jesse_Maldonado' 'Elsa_Burke' 'Jeanne_Vargas' 'Alton_Francis'
... 'Donald_Mitchell' 'Dianna_Perry'
24 'Kristi_Stephens' 'Virgil_Goodwin' 'Edmund_Newton' 'Luther_Huff'
... 'Hannah_Anderson' 'Emmett_Gill'
25 'Clayton_Wallace' 'Tracy_Mendez' 'Connie_Reeves' 'Jeanette_Hansen'
... 'Carole_Fox' 'Carmen_Fowler'
26 'Alex_Diaz' 'Rick_Waters' 'Willis_Warren' 'Krista_Ferguson' 'Debra_Russell'
... 'Ellis_Christensen'
27 'Freda_Johnston' 'Janis_Carpenter' 'Rosemary_Sherman' 'Earnest_Peters'
... 'Kelly_West'
```

```

28 'Jorge_Caldwell' 'Moses_Norris' 'Erica_Riley' 'Ray_Gordon' 'Abel_Poole'
... 'Cary_Boone'
29 'Grant_Gomez' 'Denise_Chapman' 'Vernon_Moran' 'Ben_Walker' 'Francis_Benson'
... 'Andrea_Sullivan'
30 'Wayne_Rice' 'Jamie_Mason' 'Jane_Figueroa' 'Pat_Wade' 'Rudy_Bates'
... 'Clyde_Harris' 'Andre_Mathis'
31 'Carlton_Oliver' 'Merle_Lee' 'Amber_Wright' 'Russell_Becker'
... 'Natalie_Wheeler' 'Maryann_Miller'
32 'Lucia_Byrd' 'Jenny_Zimmerman' 'Kari_Mccarthy' 'Jeannette_Cain' 'Ian_Walsh'
... 'Herman_Martin'
33 'Ginger_Farmer' 'Catherine_Williamson' 'Lorena_Henderson' 'Molly_Watkins'
... 'Sherman_Ford'
34 'Adam_Gross' 'Alfred_Padilla' 'Dwayne_Gibson' 'Shawn_Hall' 'Anthony_Rios'
... 'Kelly_Thomas'
35 'Allan_Owens' 'Duane_Malone' 'Chris_George' 'Dana_Holt' 'Muriel_Santiago'
... 'Shelley_Osborne'
36 'Clinton_Ross' 'Kelley_Parsons' 'Sophia_Lewis' 'Sylvia_Cooper'
... 'Regina_Aguilar'
37 'Sheila_Castillo' 'Sheri_Mcdonald' 'Lynn_Hodges' 'Patrick_Medina'
... 'Arlene_Tate' 'Minnie_Weber'
38 'Geneva_Pena' 'Byron_Collier' 'Veronica_Higgins' 'Leo_Roy' 'Nelson_Lopez')
39
40 function option_1
41 {
42     name_counter=0
43     echo
44     read -p "Enter the first name, or a partial start of the first name: "
... first_begins_with
45     echo
46     first_initial=$(echo ${first_begins_with:0:1} | tr [:lower:] [:upper:])
47     first_remaining=$(echo ${first_begins_with:1} | tr [:upper:] [:lower:])
48     first_begins_with="$first_initial$first_remaining"
49     for i in ${!name_array[*]};
50     do
51         first_name=$(echo ${name_array[$i]} | cut -d "_" -f 1)
52         if [[ $first_name = $first_begins_with* ]];then
53             echo ${name_array[$i]} | tr "_" " "
54             name_counter=1
55         fi
56     done
57     if [[ $name_counter -eq 0 ]];then
58         echo "No first name was found starting with $first_begins_with"
59     fi
60 }
61
62 function option_2
63 {
64     name_counter=0
65     echo
66     read -p "Enter the last name, or a partial start of the last name: "
... last_begins_with

```

```

67     echo
68     last_initial=$(echo ${last_begins_with:0:1} | tr [:lower:] [:upper:])
69     last_remaining=$(echo ${last_begins_with:1} | tr [:upper:] [:lower:])
70     last_begins_with="$last_initial$last_remaining"
71     for i in ${!name_array[*]};
72     do
73         last_name=$(echo ${name_array[$i]} | cut -d "_" -f 2)
74         if [[ $last_name = $last_begins_with* ]];then
75             echo ${name_array[$i]} | tr "_" " "
76             name_counter=1
77         fi
78     done
79     if [[ $name_counter -eq 0 ]];then
80         echo "No last name was found starting with $last_begins_with"
81     fi
82 }
83
84
85 function option_3
86 {
87     while true;
88     do
89         read -p "Enter the new first name: " new_first_name
90         if [[ $new_first_name == *" "* ]];then
91             echo "Just the first name, please.  No spaces."
92             continue
93         fi
94         read -p "Enter the new last name: " new_last_name
95         first_initial=$(echo ${new_first_name:0:1} | tr [:lower:] [:upper:])
96         first_remaining=$(echo ${new_first_name:1} | tr [:upper:] [:lower:])
97         new_first_name="$first_initial$first_remaining"
98
99         last_initial=$(echo ${new_last_name:0:1} | tr [:lower:] [:upper:])
100        last_remaining=$(echo ${new_last_name:1} | tr [:upper:] [:lower:])
101        new_last_name="$last_initial$last_remaining"
102        new_name="$new_first_name $new_last_name"
103        new_name=$(echo $new_name | tr " " "_")
104        name_array+=("$new_name")
105        echo
106        echo "$new_name has been added" | tr "_" " "
107        break
108    done
109 }
110
111 function option_4
112 {
113     while true;
114     do
115
116         delete_counter=0
117         echo

```

```

118     echo "Delete a name by entering the full name ie: John Smith"
119 #   read -p "Enter Q to return to the main menu, 1 to view names, or enter
... the name: " delete_name
120     read -p "Enter the full name (Q to return to the main menu, 1 to search
... first names): " delete_name
121     if [[ $delete_name == "Q" || $delete_name == "q" ]]; then
122         break
123     elif [[ $delete_name == "1" ]];then
124         option_1
125         continue
126     fi
127     delete_name=$(echo $delete_name | tr " " "_")
128     delete_name=$(echo $delete_name | tr [:upper:] [:lower:])
129     first_name=$(echo $delete_name | cut -d "_" -f 1)
130     last_name=$(echo $delete_name | cut -d "_" -f 2)
131
132     first_initial=$(echo ${first_name:0:1} | tr [:lower:] [:upper:] )
133     first_remaining=$(echo ${first_name:1})
134     first_name="$first_initial$first_remaining"
135
136     last_initial=$(echo ${last_name:0:1} | tr [:lower:] [:upper:] )
137     last_remaining=$(echo ${last_name:1})
138     last_name="$last_initial$last_remaining"
139
140     delete_name="$first_name $last_name"
141     delete_name=$(echo $delete_name | tr " " "_")
142     for i in ${!name_array[*]};
143     do
144         if [[ $delete_name = ${name_array[$i]} ]];then
145             echo
146             name_array=(${name_array[*]:0:$i} ${name_array[*]:$((i+1))})
147             echo "$delete_name has been deleted from the array" | tr "_" " "
148             delete_counter=1
149         fi
150     done
151
152     if [[ $delete_counter -ne 1 ]]; then
153         echo "$delete_name was not found in the array." | tr "_" " "
154     fi
155
156 done
157
158 }
159
160
161
162
163
164 while true;
165 do
166     echo "

```

```
167 Please select from the following options:
168
169     1. List all names starting with one or more letters of the first name
170     2. List all names starting with one or more letters of the last name
171     3. Add a name
172     4. Delete a name
173     5. Exit
174 "
175
176 read -p "Option #: " user_menu_choice
177
178 if [[ $user_menu_choice -eq 1 ]];then
179     option_1
180 elif [[ $user_menu_choice -eq 2 ]];then
181     option_2
182 elif [[ $user_menu_choice -eq 3 ]];then
183     option_3
184 elif [[ $user_menu_choice -eq 4 ]];then
185     option_4
186 elif [[ $user_menu_choice -eq 5 ]];then
187     break
188 else
189     echo "That is not a valid option. Please try again."
190     continue
191 fi
192
193
194
195 done
196
```