

```
1  #! /bin/sh
2
3  #This was my test array
4  #declare -a myArray=(44 seven 7 120 monkey cat heck 3.1 -1382 banana)
5
6  clear
7  read -p "Press enter to move from task to task"
8  clear
9
10 RandoInsert () {
11 myArrayLength=${#myArray[*]}
12 indexPosition=$(( ($RANDOM%myArrayLength) ))
13 myArray=(${myArray[*]:0:indexPosition} "$1" ${myArray[*]:indexPosition})
14 }
15
16 #Task 1 Fill the array with user input
17 echo -e "Task 1\n"
18 echo "Create an array filled with 10 items"
19 echo -e "Be sure to get a mix of floating point numbers, integers and
... text\n"
20 inputCounter=0
21 while [ $inputCounter -lt 10 ];
22 do
23     read -p "Please enter a number or word for index position $inputCounter:
... " userInput
24     #echo
25     userInput=$( echo $userInput | tr " " "_" )
26     myArray+=($userInput)
27     inputCounter=$((inputCounter+1))
28 done
29 read
30 clear
31
32 #Task 2 Print the length of the array
33 echo -e "Task 2\n"
34 myArrayLength=${#myArray[*]}
35 echo "This array has $myArrayLength items"
36 read
37 clear
38
39 #Task 3 Print out the array
40 echo -e "Task 3\n"
41 echo -e "This is the array:\n"
42 echo ${myArray[*]}
43 read
44
45
46 #Task 4 Swap the first item with the last item in the array and print it out
47 echo -e "Task 4\n"
48 firstThing=${myArray[0]}
49 myArray[0]=${myArray[-1]}
```

```

50 myArray[-1]=$firstThing
51 echo -e "This is the array after swapping the first and last items:\n"
52 echo ${myArray[*]}
53 read
54 #clear
55
56 #Task 5 Print the first 3 and last 3 items in the array
57 echo -e "Task 5\n"
58 echo -e "These are the first three and last three items in the array:\n"
59 echo ${myArray[*]:0:3} ${myArray[*]:(-3)}
60 read
61 clear
62
63 #Task 6 Loop through and print out the individual items in the array
64 echo -e "Task 6\n"
65 echo -e "These are the elements of the array\n"
66 for i in ${myArray[*]};
67 do
68     echo $i
69 done
70 read
71 clear
72
73 #Task 7 Check to see if cat is in my array and let the user know
74 echo -e "Task 7\n"
75 catFlag=0
76 for i in ${!myArray[*]};
77 do
78     if [[ ${myArray[$i]} == "cat" ]]; then
79         catFlag=1
80         break
81     fi
82 done
83
84 if [[ $catFlag -eq 1 ]];then
85     echo "There is a cat in my array"
86 else
87     echo "There is no cat in my array"
88 fi
89 read
90 clear
91
92 #Task 8 Get the name of a Marvel character from the user and pass it to the
... random insert function
93 echo -e "Task 8\n"
94 read -p "Please insert the name of a Marvel character: " userHero
95 userHero=$( echo $userHero | tr " " "_")
96 RandoInsert $userHero
97 echo
98 read
99 clear

```

```

100
101 #Task 9 Print out the index position of the hero, then the array
102 echo -e "Task 9\n"
103 for i in ${!myArray[*]};
104 do
105     if [[ ${myArray[$i]} == $userHero ]];then
106         echo "$userHero is at position $i in the array"
107     fi
108 done
109 echo
110 read
111 clear
112
113 #Task 10 Show the final state of the array
114 echo -e "Task 10\n"
115 echo "The array is now: ${myArray[*]}"
116 read
117 clear
118
119 #Task 11 Copy integers to new array, sort, print
120 echo -e "Task 11\n"
121 for i in ${!myArray[*]};
122 do
123     if [[ ${myArray[$i]} =~ ^-?[0-9]+$ ]];then
124         intArray+=(${myArray[$i]})
125     fi
126 done
127
128 intArray=($(for i in ${intArray[*]}; do echo $i; done | sort -n ))
129 echo "The integers in the original array, sorted, are: ${intArray[*]}"
130 read
131 clear
132
133 #Task 12 Create and unpack an array of arrays
134 echo -e "Task 12\n"
135 echo -e "These are the elements of the array of arrays\n"
136 subarray1=(1 2 3)
137 subarray2=(a b c)
138 packing_array=('subarray1[@]' 'subarray2[@]')
139 my_array=('packing_array[@]')
140
141 for i in "${my_array[@]}"
142 do
143     for j in "${!i}"
144     do
145         for k in "${!j}"
146         do
147             echo "$k"
148         done
149     done
150 done

```

151
152
153
154
155

`echo`

`read -p "Press enter to end the script"`