

# Shell Scripting Exercises (1)

by Sanka Dharmarathna

---

***Exercise\_1 - Write a shell script that prints “Shell Scripting is Fun!” on the screen***

---

```
#!/bin/bash

echo "Shell Scripting is Fun!"
```

---

***Exercise\_2 - Modify the shell script from exercise 1 to include a variable. The variable will hold the contents of the message “Shell Scripting is Fun!”***

---

```
#!/bin/bash

NAME="Shell Scripting is Fun!"
echo $NAME
```

---

***Exercise\_3 - Store the output of the command “hostname” in a variable. Display “This script is running on \_.” where “\_” is the output of the “hostname” command.***

---

```
#!/bin/bash

HOSTNAME=$(hostname)
echo "This script is running on $HOSTNAME"
```

***Exercise\_4 - Write a shell script to check to see if the file “file\_path” exists. If it does exist, display “file\_path passwords are enabled.” Next, check to see if you can write to the file. If you can, display “You have permissions to edit “file\_path.””If you cannot, display “You do NOT have permissions to edit “file\_path””***

```
#!/bin/bash

FILE="/home/Assignment"

if [ -e "$FILE" ]
then
    echo "$FILE passwords are enabled"
fi

if [ -x "$FILE" ]
then
    echo "You have permission to execute $FILE"

else
    echo "You do Not have permissions to execute $FILE"
fi
```

***Exercise\_5 - Write a shell script that displays “man”, “bear”, “pig”, “dog”, “cat”, and “sheep” on the screen with each appearing on a separate line. Try to do this in as few lines as possible.***

```
#!/bin/bash

ANIMALS="man bear pig dog cat sheep"

for ANIMAL in $ANIMALS
do
    echo $ANIMAL
done
```

---

***Exercise\_6 - write a shell script that prompts the user for a name of a file or directory and reports if it is a regular file, a directory, or another type of file. Also perform an ls command against the file or directory with the long listing option.***

---

```
#!/bin/bash

echo "Enter the file path"
read FILE

if [ -f "$FILE" ]
then
    echo "$FILE is a regular file"

elif [ -d "$FILE" ]
then

    echo "$FILE is another type of file"
fi

ls -l $FILE
```

---

***Exercise\_7 - Modify the previous script to that it accepts the file or directory name as an argument instead of prompting the user to enter it.***

---

```
#!/bin/bash

FILE=$1

if [ -f "$FILE" ]
then
    echo "$FILE is a regular file"

elif [ -d "$FILE" ]
then
    echo "$FILE is a directory"

else
    echo "$FILE is another type of file"
fi

ls -l $FILE
```

---

***Exercise\_8 - Write a shell script that displays, “This script will exit with 0 exit status.” Be sure that the script does indeed exit with a 0 exit status.***

---

```
#!/bin/bash

echo "This script will exit with 0 exit status."
exit 0
```