

ASSIGNMENT 1 - Running Linux Commands

Part - 1

```
1000 ls
1001 pwd
1002 mkdir linux-a
1003 cd linux-a
1004 touch file1.txt
1005 cat > file1.txt
1006 cat
1007 cat file1.txt
1008 cat file1.txt
1009 cp file1.txt file2.txt
1010 cat file2.txt
1011 rename s/file1.txt/renamed.txt/' files
1012 head file2.txt
1013 tail file2.txt
1014 tac file2.txt
1015 more file2.txt
1016 less file2.txt
1017 su temp_user
1018 id
1019 useradd temp
1020 passwd temp
1021 groupadd group1
1022 cat file2.txt | tac
1023 cat file2.txt | tac | cat \n
1024 cut -d- -f2 file2.txt
1025 cat file2.txt | grep h
1026 comm file1.txt file2.txt
1027 echo class7 | sed 's/class/jtp/vnclass7
1028 who
1029 whoami
1030 cal
1031 date
1032 cat file1.txt | tee new.txt | cat
1033 cat file1.txt | tr 'hello' 'HELLO'
1034 sort file1.txt |uniq
1035 wc file2.txt
1036 od -b file2.txt
1037 gzip file1.txt file2.txt
1038 ls
1039 gunzip file1.txt file2.txt
1040 find -name *.txt
1041 find -name *.txt
1042 locate file1.txt
1043 sleep 2
1044 rm file1.txt
1045 rm file2.txt
1046 rmdir linux-a
kshama@Kshamas-MacBook-Air linux-a %
```

Directory commands

1. pwd-display the location of the current working directory
2. mkdir - create new directory
3. rmdir -delete directory
4. ls - display list of contents
5. cd - change directory

File commands

6. touch - create file
7. cat > filename - write into file
8. cat - display contents of file
9. cp - copy file contents
10. mv - move file

User commands

11. su - administrative access to another use
12. id - display user ID
13. useradd - add or remove user
14. passwd - create or change password
15. groupadd - create a user group

Filter commands

16. comm - compare 2 files
17. wc - count words
18. cut - select specific column
19. gzip - compress file
20. gunzip - decompress file

Utility commands

21. cal - current months calendar
22. date - display date, time

Part - 2

1. Swap two numbers without using third variable

Code:

```
UW PICO 5.09 File: swap.sh

#!/bin/sh
echo "Enter first number:"
read a
echo "Enter second number:"
read b
b=$((a+b))
a=$((b-a))
b=$((b-a))
echo "The swapped numbers are:"
echo $a
echo $b
```

Output:

```
kshama@Kshamas-MacBook-Air assignment2 % nano swap.sh
kshama@Kshamas-MacBook-Air assignment2 % ./swap.sh
zsh: permission denied: ./swap.sh
kshama@Kshamas-MacBook-Air assignment2 % chmod +x
usage:  chmod [-fhv] [-R [-H | -L | -P]] [-a | +a | =a [i]#[ n]] mode|entry file ...
        chmod [-fhv] [-R [-H | -L | -P]] [-E | -C | -N | -i | -I] file ...
kshama@Kshamas-MacBook-Air assignment2 % chmod +x swap.sh
kshama@Kshamas-MacBook-Air assignment2 % ./swap.sh
Enter first number:
10
Enter second number:
30
The swapped numbers are:
30
10
kshama@Kshamas-MacBook-Air assignment2 %
```

2. Accept one integer argument and print its multiplication table

Code:

```
UW PICO 5.09 File: multi.sh

#!/bin/bash
echo "Enter a number:"
read n
echo "The multiplication table of $n is:"
for((i=1;i<=10;i++))
do
    echo -n "$n * $i = `expr $n \* $i`"
    echo ""
done
```

Output:

```
kshama@Kshamas-MacBook-Air assignment2 % chmod +x multi.sh
kshama@Kshamas-MacBook-Air assignment2 % ./multi.sh
Enter a number:
10
The multiplication table of 10 is:
10 * 1 = 10
10 * 2 = 20
10 * 3 = 30
10 * 4 = 40
10 * 5 = 50
10 * 6 = 60
10 * 7 = 70
10 * 8 = 80
10 * 9 = 90
10 * 10 = 100
kshama@Kshamas-MacBook-Air assignment2 %
```

ASSIGNMENT 2 - Running a simple C program on linux

Instructions given had a straight forward approach and followed the same.

Created a C file and printed a text.

Code:

```
#include <stdio.h>
int main(){
    printf("Kshama 12345\n");
    return 0;
}
```

Output:

```
kshama@Kshamas-MacBook-Air lab0 % nano hello.c
kshama@Kshamas-MacBook-Air lab0 % gcc hello.c -o hello.o
kshama@Kshamas-MacBook-Air lab0 % ls
hello.c hello.o
kshama@Kshamas-MacBook-Air lab0 % ./hello.o
Kshama 12345
kshama@Kshamas-MacBook-Air lab0 %
```

ASSIGNMENT 3 - Shell scripts - Research on Shell scripts and solve these questions.

1. Script to Print All .txt and .c Files

Code:

```
#!/opt -s nullglob
# /bin/bash
txt_files=$(ls *.txt)
c_files=$(ls *.c)
if [[ ${#txt_files[@]} -gt 0 || ${#c_files[@]} -gt 0 ]]; then
    ls *.txt *.c 2>/dev/null
else
    echo "Creating dummy files"
    touch file1.txt file2.txt file1.c file2.c
    echo "created:"
    ls *.txt *.c
fi
```

Output:

```
kshama@Kshamas-MacBook-Air assignment3 % nano file.sh
kshama@Kshamas-MacBook-Air assignment3 % chmod +x file.sh
kshama@Kshamas-MacBook-Air assignment3 % ./file.sh
add.sh file.sh
kshama@Kshamas-MacBook-Air assignment3 %
```

2. Script to Add Two Numbers

Code:

```
UW PICO 5.09 File: add.sh
#!/bin/sh
echo "Enter first number:"
read a
echo "Enter second number:"
read b
c=$((a+b))
echo "The sum of 2 numbers is:"
echo $c
```

Output:

```
kshama@Kshamas-MacBook-Air assignment3 % nano add.sh
kshama@Kshamas-MacBook-Air assignment3 % chmod +x add.sh
kshama@Kshamas-MacBook-Air assignment3 % ./add.sh
Enter first number:
6
Enter second number:
7
The sum of 2 numbers is:
13
kshama@Kshamas-MacBook-Air assignment3 %
```

3. Installing Docker Using a Shell Scripts

Code:

```
UW PICO 5.09
#!/bin/bash
echo "Checking if Homebrew is installed..."
if ! command -v brew &> /dev/null; then
    echo "Homebrew is not installed. Installing Homebrew..."
    /bin/bash -c "$(curl -fsSL https://raw.githubusercontent.com/Homebrew/install/HEAD/install.sh)"
else
    echo "Homebrew is already installed."
fi
echo "Updating Homebrew..."
brew update
echo "Installing Docker..."
brew install --cask docker
echo "Starting Docker..."
open /Applications/Docker.app
echo "Waiting for Docker to start..."
sleep 10
echo "Checking Docker version..."
docker --version
echo "Running a test container (hello-world)..."
docker run hello-world
```

Output:

```
kshama@Kshamas-MacBook-Air assignment3 % % ./docker.sh
Checking if Homebrew is installed...
Homebrew is already installed.
Updating Homebrew...
=> Updating Homebrew...
Updated 2 taps (homebrew/core and homebrew/cask).
=> New Formulae
redocly-cli          sd13                umka-lang
=> New Casks
dockfix              linearmouse@beta    valhalla-freq-echo
flashspace           muteme              valhalla-space-modulator
Installing Docker...
=> Downloading https://raw.githubusercontent.com/Homebrew/homebrew-cask/20167b42eeae59
##### 100.0%
=> Downloading https://desktop.docker.com/mac/main/arm64/179585/Docker.dmg
##### 100.0%
=> Installing Cask docker
=> Moving App 'Docker.app' to '/Applications/Docker.app'
=> Linking Binary 'docker-compose.zsh-completion' to '/opt/homebrew/share/zsh/site-fun
=> Linking Binary 'docker-fish-completion' to '/opt/homebrew/share/fish/vendor
=> Linking Binary 'docker' to '/usr/local/bin/docker'
Password:
=> Linking Binary 'docker-credential-desktop' to '/usr/local/bin/docker-credential-ds
=> Linking Binary 'docker-credential-gcr-login' to '/usr/local/bin/docker-credential-e
=> Linking Binary 'docker-credential-osxkeychain' to '/usr/local/bin/docker-credential
=> Linking Binary 'hub-tool' to '/usr/local/bin/hub-tool'
=> Linking Binary 'docker-compose' to '/usr/local/cli-plugins/docker-compose'
=> Linking Binary 'docker.bash-completion' to '/opt/homebrew/etc/bash_completion.d/doc
=> Linking Binary 'docker.zsh-completion' to '/opt/homebrew/share/zsh/site-functions/_
=> Linking Binary 'docker.fish-completion' to '/opt/homebrew/share/fish/vendor_comple
=> Linking Binary 'kubectl' to '/usr/local/bin/kubectl.docker'
=> Linking Binary 'docker-compose.bash-completion' to '/opt/homebrew/etc/bash_completi
🍺 docker was successfully installed!
Starting Docker...
Waiting for Docker to start...
Checking Docker version...
Docker version 27.4.0, build bde2b89
Running a test container (hello-world)...
docker: Cannot connect to the Docker daemon at unix:///var/run/docker.sock. Is the docker daemon running?.
See 'docker run --help'.
kshama@Kshamas-MacBook-Air assignment3 % █
```

Installed docker using homebrew, opened on applications and ran it.

4. Download and Install MySQL Database

Code:

```
UW PICO 5.09

#!/bin/bash
echo "Checking if Homebrew is installed..."
if ! command -v brew &> /dev/null; then
    echo "Home brew is not installed. Installing Homebrew..."
    /bin/bash -c "$(curl -fsSL https://raw.githubusercontent.com/Homebrew/install/HEAD/install.sh)"
else
    echo "Homebrew is already installed."
fi
echo "Updating Homebrew..."
brew update
echo "Installing MySQL..."
brew install mysql
echo "Starting MySQL service..."
brew services start mysql
echo "Checking MySQL version..."
mysql --version
echo "MySQL installation and setup completed!"
```

Output:

```
-- Pouring mysql--2.0.0_arm64_bottle.tar.gz
--> /usr/homebrew/Cellar/mysql/9.2.0/bin/mysqld --initialize-insecure --user=kshama --basedir=/opt/homebrew/Cellar/mysql/9.2.0 --datadir=/opt/homebrew/var/mysql --tmpdir=/tmp
--> Casks
Upgrading from MySQL 8.4 to MySQL 9.2 requires running MySQL 8.4 first:
- brew services stop mysql
- brew install mysql@8.4
- brew services start mysql@8.4
- brew services stop mysql@8.4
- brew services start mysql

We've installed your MySQL database without a root password. To secure it run:
  mysql_secure_installation

MySQL is configured to only allow connections from localhost by default
To connect run:
  mysql -u root

To start mysql now and restart at login:
  brew services start mysql
Or, if you don't want/need a background service you can just run:
  /opt/homebrew/opt/mysql/bin/mysqld_safe --datadir=/opt/homebrew/var/mysql

Summary
🍺 /opt/homebrew/Cellar/mysql/9.2.0: 529 files, 294 MB
🔧 Running brew cleanup mysql...
Disable this behaviour by setting HOMEBREW_NO_INSTALL_CLEANUP.
Hide these hints with HOMEBREW_NO_ENV_HINTS (see 'man brew').

--> Casks
Upgrading from MySQL 8.4 to MySQL 9.2 requires running MySQL 8.4 first:
- brew services stop mysql
- brew install mysql@8.4
- brew services start mysql@8.4
- brew services stop mysql@8.4
- brew services start mysql

We've installed your MySQL database without a root password. To secure it run:
  mysql_secure_installation

MySQL is configured to only allow connections from localhost by default
To connect run:
  mysql -u root

To start mysql now and restart at login:
  brew services start mysql
Or, if you don't want/need a background service you can just run:
  /opt/homebrew/opt/mysql/bin/mysqld_safe --datadir=/opt/homebrew/var/mysql

Starting MySQL service...
--> Tapping Homebrew services
Cloning into '/opt/homebrew/Library/Taps/homebrew/homebrew-services'...
remote: Enumerating objects: 3967, done.
remote: Compressing objects: 188M (152/152), done.
remote: Total 3967 (delta 389), reused 351 (delta 329), pack-reused 3486 (from 3)
Receiving objects: 188M (3967/3967), 1.15 MiB | 713.80 KiB/s, done.
Resolving deltas: 188M (1527/1527), done.
Tapped 2 commands (54 files, 1.0MB).
Successfully started 'mysql' (label: homebrew.mysql.mysql)

Checking MySQL version...
mysql Ver 9.2.0 for macOS17 on arm64 (Homebrew)
MySQL installation and setup completed!
kshama@Kshamas-MacBook-Air assignment3 % █
```

Installed Mysql using Homebrew, started sql server and verified installation.