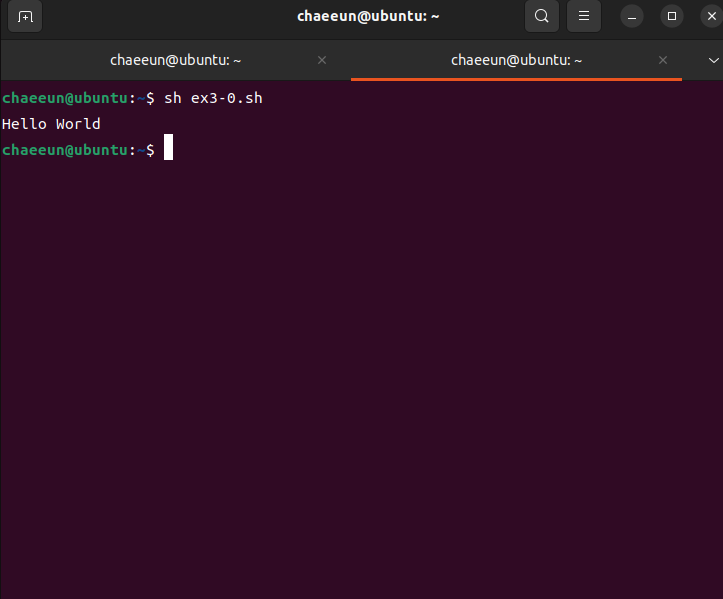
[프로그래밍] 20233093, 박채은

Assignment #3: shell script 프로그래밍

**ex3-0.sh**

▪hello world를 출력



소스코드

**ex3-0.sh**

#!/bin/sh

echo "Hello World"

exit 0

**ex3-1.sh**

#!/bin/sh

echo "인사 반복 횟수 입력"

read answer

for i in $(seq 1 $answer); do

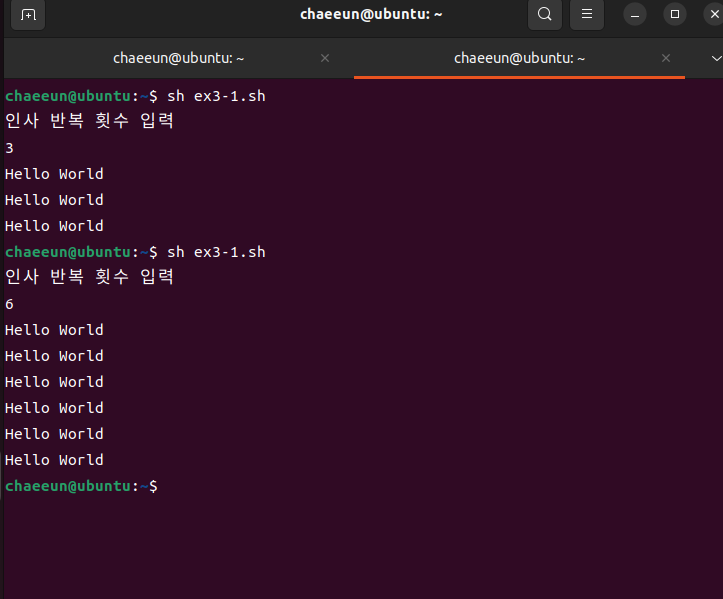
echo "Hello World"

done

exit 0

**ex3-1.sh**

▪숫자를 입력 받아 hello world를 여러 번 출력

****

**ex3-2.sh**

소스코드

**ex3-2.sh**

#!/bin/sh

echo "두 숫자와 연산 기호 입력"

read expression

echo "계산 결과: $(echo "$expression" | bc)"

exit 0

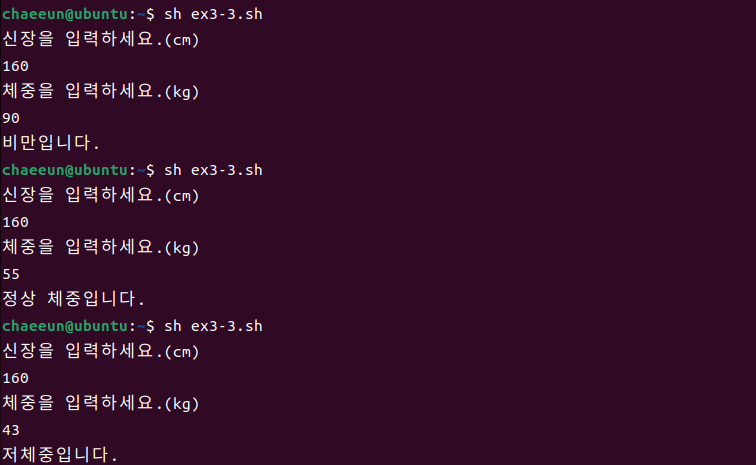
▪두 숫자와 연산자(+ 또는-)를 입력 받아 계산하기



**ex3-3.sh**

▪체질량 지수(BMI) 계산하여 비만 여부를 판단.

▪정상체중: 18.5 <= 몸무게/ (신장\* 신장) < 23



#!/bin/sh

echo "신장을 입력하세요.(cm)"

read height

echo "체중을 입력하세요.(kg)"

read weight

ratio=23.00

ratio2=18.50

bmi=$(echo "scale=2; $weight \* 10000 / $height / $height" | bc)

if [ "$(echo "$bmi > $ratio" | bc)" -ne 0 ]; then

echo "비만입니다."

elif [ "$(echo "$bmi > $ratio2" | bc)" -ne 0 ]; then

echo "정상 체중입니다."

else

echo "저체중입니다."

fi

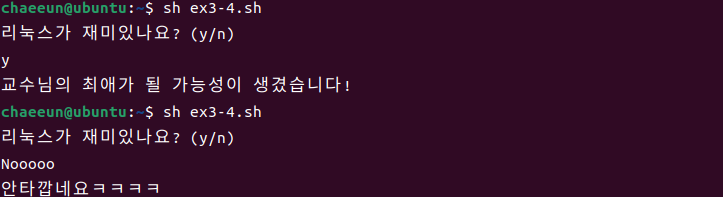
exit 0

**ex3-4.sh**

▪case문을 활용한 예제실습(p.439)

▪입력에 따라 분기하여 서로 다른 출력.

다양한 입력에 대한 방어코드.



#!/bin/sh

echo "리눅스가 재미있나요? (y/n)"

read answer

case $answer in

y|Y|yes|Yes|YES)

echo "교수님의 최애가 될 가능성이 생겼습니다!";;

[nN]\*)

echo "안타깝네요ㅋㅋㅋㅋ";;

\*)

echo "잘못된 입력입니다."

exit 1;;

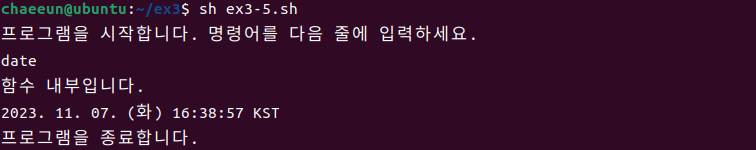
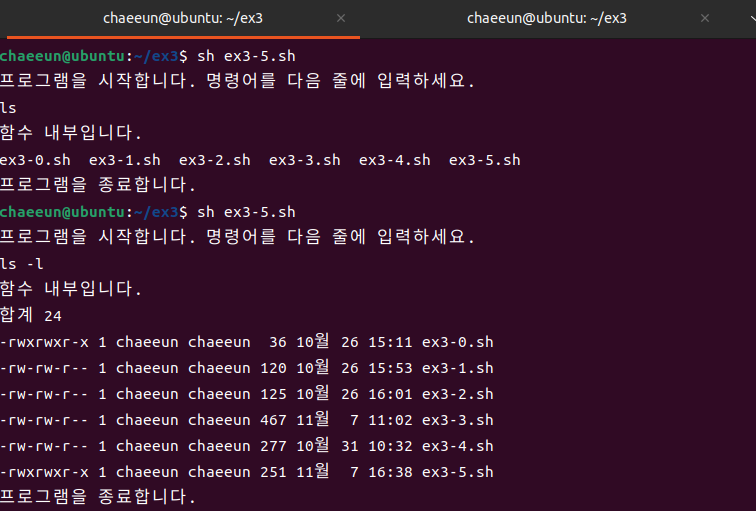
esac

exit 0

**ex3-5.sh**

▪내부함수를 만들어 리눅스 명령어(e.g. ls)를 실행.

▪입력인자를 내부함수 내의 리눅스 명령어 옵션으로 전달



#!/bin/sh

function(){

read userCommand

echo "함수 내부입니다."

eval "$userCommand"

}

echo "프로그램을 시작합니다. 명령어를 다음 줄에 입력하세요."

function "$userCommand"

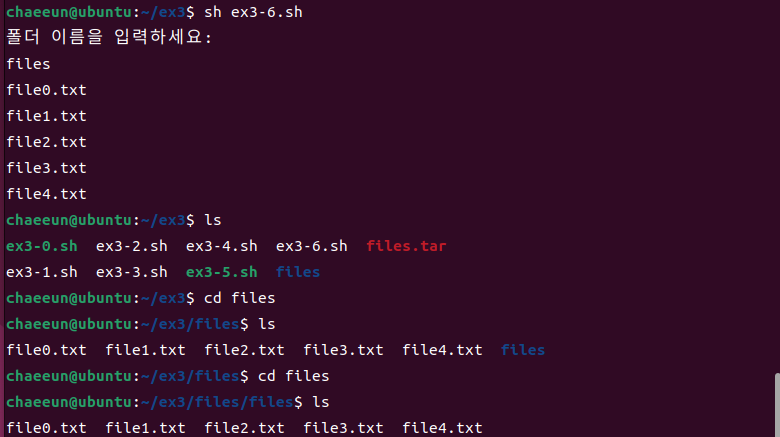
echo "프로그램을 종료합니다."

exit 0

**ex3-6.sh**

▪입력된 이름으로 폴더가 있는지 확인. 없다면 생성.

▪5개의 파일을 만들고 압축. 새로운 폴더를 생성하여 압축 해제

#!/bin/sh

folder\_func(){

echo "폴더 이름을 입력하세요:"

read folder\_name

if [ ! -d "$folder\_name" ]; then

mkdir "$folder\_name"

fi

for i in 0 1 2 3 4 ; do

touch "$folder\_name/file$i.txt"

echo "file$i.txt"

done

tar -cf "$folder\_name.tar" "$folder\_name"

tar -xf "$folder\_name.tar" -C "$folder\_name"

}

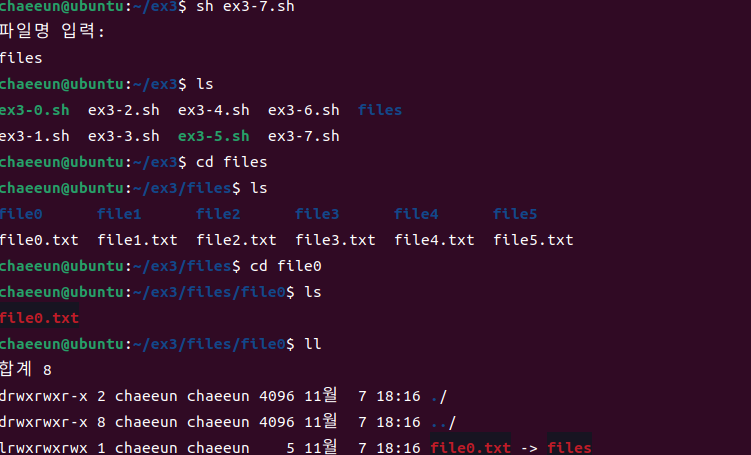
folder\_func

exit 0

**ex3-7.sh**

▪입력된 이름으로 폴더에서 5개 이상의 파일을 만듦.

▪파일 이름대로 하위 폴더들을 생성하고 각 폴더에 해당 파일을 링크



#!/bin/sh

link\_func(){

echo "파일명 입력:"

read file\_name

if [ ! -d "$file\_name" ]; then

mkdir "$file\_name"

fi

for i in 0 1 2 3 4 5; do

touch "$file\_name/file$i.txt"

subfolder\_name="$file\_name/file$i"

mkdir "$subfolder\_name"

ln -s "$file\_name" "$subfolder\_name/file$i.txt"

done

}

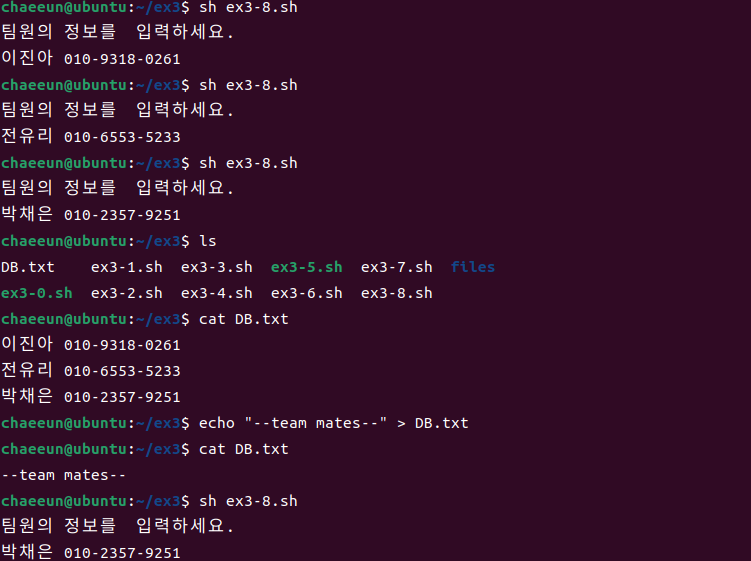
link\_func

exit 0

**ex3-8.sh**

▪팀원의 이름과 생일 또는 전화번호를 DB.txt에 기록

▪이름과 정보를 DB.txt에 계속 추가할 수 있음. 또는 처음부터 다시 작성



#!/bin/sh

db\_file="DB.txt"

if [ ! -f "$db\_file" ]; then

touch "$db\_file"

fi

echo "팀원의 정보를 입력하세요."

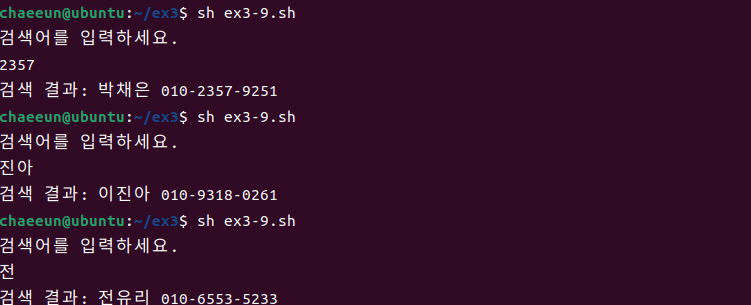
read name

echo "$name" >> "$db\_file"

exit 0

**ex3-9.sh**

▪이름으로 검색. DB.txt에서 검색된 팀원의 정보를 확인.



#!/bin/sh

echo "검색어를 입력하세요."

read search

result=$(grep "$search" "DB.txt")

if [ -n "$result" ]; then

echo "검색 결과: $result"

else

echo "검색 결과를 찾을 수 없습니다."

fi

exit 0