[프로그래밍]20233110,장주원

ex3-0.sh

hello world를 출력.

#!/bin/sh

echo "hello world“



ex3-1.sh

숫자를 입력 받아 hello world를 여러 번 출력.

#!/bin/bash

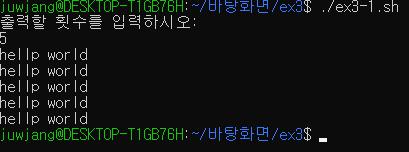
echo "출력할 횟수를 입력하시오:"

read count

for ((i = 0; i < count; i++)); do

echo "hellp world"

done



ex3-2.sh

두 숫자와 연산자(+ 또는 -)를 입력 받아 계산하기.

#!/bin/bash

IFS=',' read -p ": " num1 num2 operator

if [[ "$operator" == "+" ]]; then

result=$((num1 + num2))

operation="addition"

elif [[ "$operator" == "-" ]]; then

result=$((num1 - num2))

operation="subtraction"

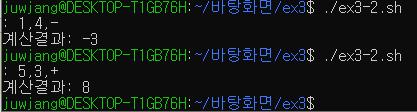
else

echo "+,-만 입력 가능"

exit 1

fi

echo "계산결과: $result“



ex3-3.sh

체질량 지수(BMI) 계산하여 비만 여부를 판단.

정상 체중:18.5 <=몸무게/(신장\*신장)<23

#!/bin/bash

read -p "몸무게(kg)를 입력하세요: " weight

read -p "신장(m)를 입력하세요: " height

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

if (( $(echo "$bmi >= 18.5 && $bmi < 23" | bc -l) )); then decision="정상체중입니다."

elif (( $(echo "$bmi < 18.5" | bc -l) )); then

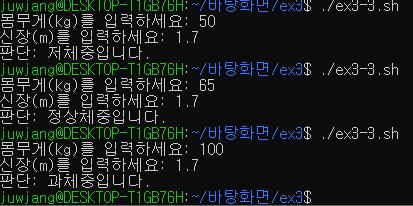
decision="저체중입니다."

else

decision="과체중입니다."

fi

echo "판단: $decision“



ex3-4.sh

case문을 활용한 예제 실습(p.439)

입력에 따라 분기하여 서로 다른 출력. 다양한 입력에 대한 방어 코드.

#!/bin/bash

while true; do

read -p "리눅스가 재미있나요? (yes/no): " answer

case "$answer" in

"yes"|"y"|"yesyesyes"|"Y")

echo "yes"

;;

"no"|"n"|"nonono"|"N"|"NO")

echo "no"

;;

\*)

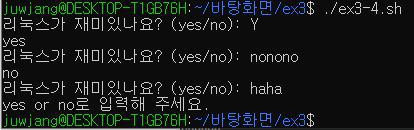
echo "yes or no로 입력해 주세요."

exit 0

;;

esac

done



ex3-5.sh

내부 함수를 만들어 리눅스 명령어(e.g.ls)를 실행.

입력 인자를 내부 함수 내의 리눅스 명령어 옵션으로 전달.

#!/bin/bash

run\_linux\_command() {

echo "프로그램을 시작합니다."

echo "함수 안으로 들어 왔음"

eval "$@"

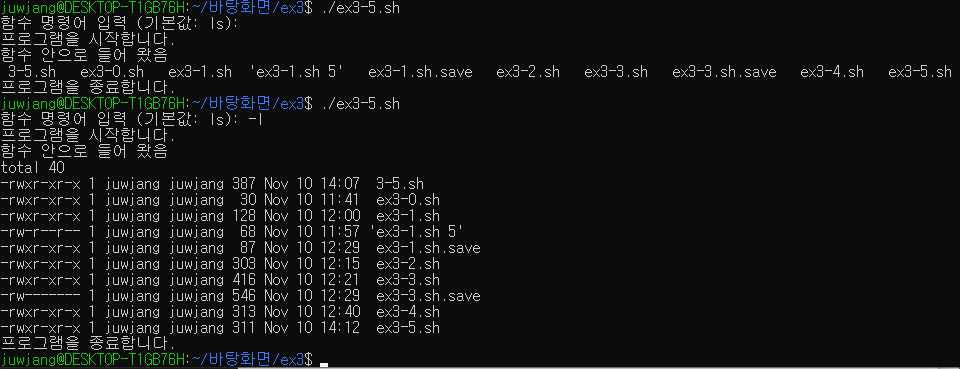
}

read -p "함수 명령어 입력 (기본값: ls): " user\_command

user\_command="ls $user\_command"

run\_linux\_command "$user\_command"

echo "프로그램을 종료합니다.“



ex3-6.sh

입력된 이름으로 폴더가 있는지 확인. 없다면 생성

5개의 파일을 만들고 압축. 새로운 폴더를 생성하여 압축 해제.

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

mkdir "$folder\_name"

echo "폴더 '$folder\_name'를 생성했습니다."

else

echo "폴더 '$folder\_name'가 이미 존재합니다."

fi

IFS=' ' read -r -a files <<< "$file\_names"

for file in "${files[@]}"; do

touch "$folder\_name/$file"

echo "$file" > "$folder\_name/$file"

done

tar -cvf "$folder\_name/files.tar" "$folder\_name"/\*

new\_folder\_name="extracted\_$folder\_name"

mkdir "$new\_folder\_name"

tar -xvf "$folder\_name/files.tar" -C "$new\_folder\_name"

echo "압축을 해제한 새로운 폴더: '$new\_folder\_name'“



ex3-7.sh

입력된 이름으로 폴더에서 5개 이상의 파일을 만듦,

파일 이름대로 하위 폴더들을 생성하고 각 폴더에 해당 파일을 링크.

#!/bin/bash

echo "폴더의 이름을 입력하세요:"

read folder\_name

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

mkdir "$folder\_name"

else

echo "폴더 '$folder\_name'가 이미 존재합니다."

fi

cd "$folder\_name"

for i in {1..5}; do

echo "내용을 채워 넣은 파일 $i" > "file$i.txt"

done

for file in \*.txt; do

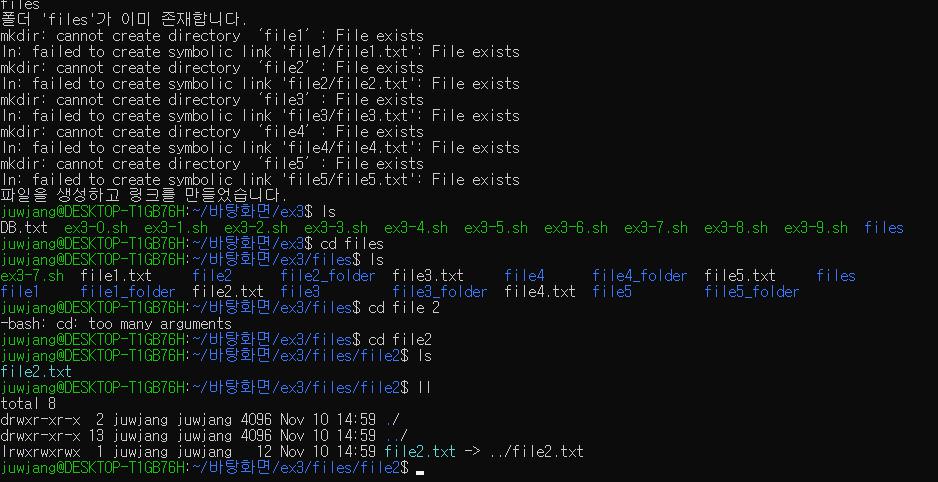
folder="${file%.\*}"

mkdir "$folder"

ln -s "../$file" "$folder/$file"

done

echo "파일을 생성하고 링크를 만들었습니다."



ex3-8.sh

팀원의 이름과 생일 또는 전화번호를 DB.txt.dp 기록

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

#!/bin/bash

database\_file="DB.txt"

echo "팀원의 이름과 생일 또는 전화번호를 기록합니다."

if [ -e "$database\_file" ]; then

read -p "DB.txt 파일이 이미 존재합니다. 기존 내용에 추가하시겠습니까? (y/n): " append\_choice

if [ "$append\_choice" != "y" ]; then

echo "기존 파일을 삭제하고 처음부터 작성합니다."

rm "$database\_file"

fi

fi

while true; do

read -p "팀원의 이름을 입력하세요 (끝내려면 'exit' 입력): " member\_name

if [ "$member\_name" == "exit" ]; then

echo "프로그램을 종료합니다."

break

fi

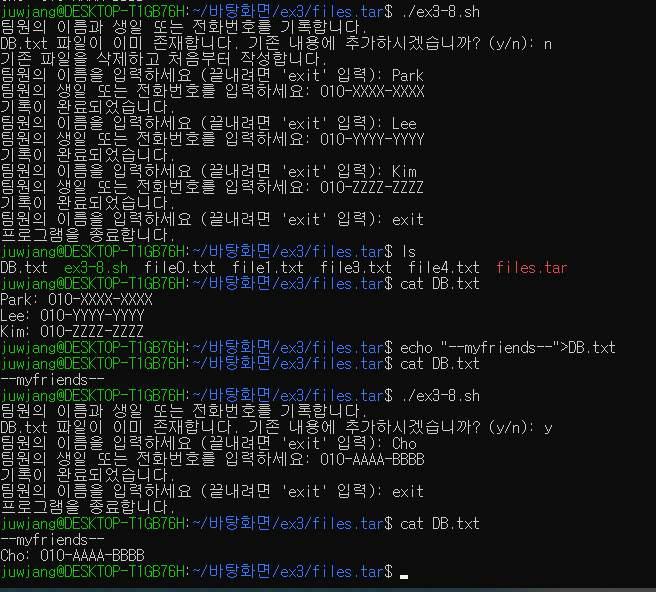
# 생일 또는 전화번호 입력 받기

read -p "팀원의 생일 또는 전화번호를 입력하세요: " member\_info

echo "$member\_name: $member\_info" >> "$database\_file"

echo "기록이 완료되었습니다."

done



ex3-9.sh

이름으로 검색. DB.txt.에서 검색된 팀원의 정보를 확인.

#!/bin/bash

database\_file="DB.txt"

if [ ! -e "$database\_file" ]; then

echo "DB.txt 파일이 존재하지 않습니다. 프로그램을 종료합니다."

exit 1

fi

while true; do

read -p "검색할 팀원의 이름 또는 전화번호 일부를 입력하세요 (끝내려면 'exit' 입력): " search\_input

if [ "$search\_input" == "exit" ]; then

echo "프로그램을 종료합니다."

break

fi

# 이름 또는 전화번호의 일부로 검색

search\_result=$(grep -i "$search\_input" "$database\_file")

if [ -z "$search\_result" ]; then

echo "입력한 정보와 일치하는 팀원이 없습니다."

else

echo "검색 결과: $search\_result"

fi

done

