Shell scirpt 프로그래밍 실습

20243265 김수현

**Ex3-0.sh**

-Hello world를 출력

**실행코드:**

!/bin/bash

echo "Hello, World!"

**스크린샷**

**텍스트, 스크린샷, 폰트, 소프트웨어이(가) 표시된 사진

자동 생성된 설명**

**Ex3-1.sh**

- 숫자를 입력 받아 hello world를 여러 번 출력.

**실행코드:**

#!/bin/bash

echo "몇 번 'Hello, World!'를 출력할지 숫자를 입력하세요:"

read num

for ((i=1; i<=num; i++))

do

echo "Hello, World!"

done

**스크린샷**

**텍스트, 스크린샷, 폰트, 소프트웨어이(가) 표시된 사진

자동 생성된 설명**

----------------------------------------------------------------------------------------------------

**Ex3-2.sh**

- 두 숫자와 연산자(+ 또는-)를 입력 받아 계산하기.

**실행코드:**

#!/bin/bash

read input

# 입력 받은 값을 공백을 기준으로 분리 (첫 번째 숫자, 연산자, 두 번째 숫자)

num1=$(echo $input | cut -d ' ' -f 1)

operator=$(echo $input | cut -d ' ' -f 2)

num2=$(echo $input | cut -d ' ' -f 3)

# 연산자에 따라 계산 수행

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

result=$((num1 + num2))

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

result=$((num1 - num2))

else

echo "잘못된 연산자입니다. + 또는 - 만 사용하세요."

exit 1

**스크린샷**

**텍스트, 스크린샷, 폰트이(가) 표시된 사진

자동 생성된 설명**

**Ex3-3.sh**

- 체질량지수(BMI) 계산하여 비만 여부를 판단.

**실행코드:**

#!/bin/bash

read height\_cm weight\_kg

# 신장을 미터 단위로 변환

height\_m=$(echo "scale=2; $height\_cm / 100" | bc)

# BMI 계산

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

# 비만 여부 판단

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

echo "BMI: $bmi - 저체중"

elif (( $(echo "$bmi >= 18.5 && $bmi < 23" | bc -l) )); then

echo "BMI: $bmi - 정상체중"

else

echo "BMI: $bmi - 비만"

fi

**스크린샷**

**텍스트, 스크린샷, 폰트이(가) 표시된 사진

자동 생성된 설명**

**Ex3-4.sh**

- case문을 활용한 예제 실습

-입력에 따라 분기하여 서로 다른 출력. 다양한 입력에 대한 방어코드.

**실행코드:**

!/bin/bash

echo "리눅스가 재미있나요? (yes / no)"

read answer

case $answer in

yes | y | Y | Yes | YES)

echo "좋아요! 리눅스를 즐기고 계시군요.";;

no | n | N | No | NO)

echo "아쉽네요. 더 재미있는 점을 찾아보세요!";;

\*)

echo "yes 또는 no로 입력해 주세요.";;

esac

**스크린샷**

**텍스트, 스크린샷, 폰트, 소프트웨어이(가) 표시된 사진

자동 생성된 설명**

**Ex3-5.sh**

- 내부 함수를 만들어 리눅스 명령어 실행.

-입력 인자를 내부 함수 내의 리눅스 명령어 옵션으로 전달

**실행코드:**

!/bin/bash

# 프로그램 시작 메시지

echo "프로그램을 시작합니다."

# 내부 함수 정의

list\_files() {

# 전달받은 인자를 ls 명령어에 전달하여 실행

ls "$@"

}

# 함수 호출 및 인자 전달

list\_files "$@"

# 프로그램 종료 메시지

echo "프로그램을 종료합니다."

**스크린샷**

**텍스트, 스크린샷, 폰트, 소프트웨어이(가) 표시된 사진

자동 생성된 설명**

**Ex3-6.sh**

- 입력된 이름으로 폴더가 있는지 확인. 없다면 생성.

-5개의 파일을 만들고 압축. 새로운 폴더를 생성하여 압축 해제.

**실행코드:**

#!/bin/bash

# 폴더 이름 입력 받기

read -p "폴더 이름을 입력하세요: " folder\_name

# 폴더가 존재하는지 확인하고 없으면 생성

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

mkdir "$folder\_name"

echo "$folder\_name 폴더를 생성했습니다."

else

echo "$folder\_name 폴더가 이미 존재합니다."

fi

# 지정된 폴더로 이동

cd "$folder\_name"

# 5개의 텍스트 파일 생성

for i in {0..4}; do

touch "file$i.txt"

done

echo "5개의 파일을 생성했습니다."

# 생성된 파일들을 tar로 압축

tar -cvf files.tar file\*.txt

echo "파일들을 files.tar로 압축했습니다."

# 새로운 폴더 생성 후 압축 해제

mkdir extracted\_files

tar -xvf files.tar -C extracted\_files

echo "압축을 extracted\_files 폴더에 해제했습니다."

**스크린샷**

**텍스트, 스크린샷, 폰트이(가) 표시된 사진

자동 생성된 설명**

**Ex3-7.sh**

- 입력된 이름으로 폴더에서 5개이상의 파일을 만듦.

- 파일 이름대로 하위 폴더들을 생성하고 각 폴더에 해당파일을 링크.

**실행코드:**

!/bin/bash

# 폴더 이름 입력 받기

read -p "폴더 이름을 입력하세요: " folder\_name

# 폴더가 존재하는지 확인하고 없으면 생성

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

mkdir "$folder\_name"

echo "$folder\_name 폴더를 생성했습니다."

else

echo "$folder\_name 폴더가 이미 존재합니다."

fi

# 지정된 폴더로 이동

cd "$folder\_name"

# 5개의 텍스트 파일 생성

for i in {0..4}; do

touch "file$i.txt"

done

echo "5개의 파일을 생성했습니다."

# 각 파일에 대해 하위 폴더 생성 및 심볼릭 링크 생성

for i in {0..4}; do

subfolder="file$i"

mkdir "$subfolder" # 하위 폴더 생성

ln -s "../file$i.txt" "$subfolder/" # 파일에 대한 심볼릭 링크 생성

done

echo "각 파일에 대해 하위 폴더를 만들고 링크를 생성했습니다."

**스크린샷**

**텍스트, 전자제품, 스크린샷, 폰트이(가) 표시된 사진

자동 생성된 설명**

**Ex3-8.sh**

- 팀원의 이름과 생일 또는 전화번호를 DB.txt에 기록

- 이름과 정보를 DB.txt에 계속 추가할 수 있음. 또는 처음부터 다시작성.

**실행코드:**

#!/bin/bash

# 사용법 안내

if [ "$#" -ne 2 ]; then

echo "사용법: $0 이름 전화번호"

exit 1

fi

# 인자 저장

name=$1

phone=$2

# 정보 추가

echo "$name $phone" >> DB.txt

echo "$name님의 정보가 DB.txt에 추가되었습니다."

**스크린샷**

**텍스트, 스크린샷, 소프트웨어, 폰트이(가) 표시된 사진

자동 생성된 설명**

**Ex3-9.sh**

- 이름으로 검색. DB.txt에서검색된 팀원의 정보를 확인.

**실행코드:**

!/bin/bash

# 검색할 이름을 입력받음

if [ "$#" -ne 1 ]; then

echo "사용법: $0 이름"

exit 1

fi

name=$1

# DB.txt에서 이름 검색

grep -i "$name" DB.txt

# 검색 결과가 없을 경우 메시지 출력

if [ $? -ne 0 ]; then

echo "$name에 해당하는 정보를 찾을 수 없습니다."

Fi

**스크린샷**

**텍스트, 스크린샷, 폰트이(가) 표시된 사진

자동 생성된 설명**