Ex3-0.sh

#!/bin/bash

echo "Hello, World!"

텍스트, 스크린샷, 폰트이(가) 표시된 사진

자동 생성된 설명

Ex3-1.sh

#!/bin/bash

# 사용자로부터 숫자 입력 받기

read -p "출력할 횟수를 입력하세요 (무한 루프를 위해 Enter 키만

누르세요): " num

if [ -z "$num" ]; then

# 무한 루프로 "Hello, World!" 출력

while true; do

echo "Hello, World!"

done

else

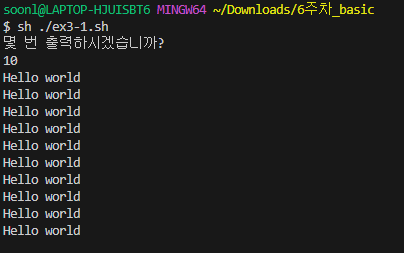
# 입력된 숫자만큼 "Hello, World!" 출력

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

echo "Hello, World!"

done

fi



Ex3-2.sh

#!/bin/bash

# 사용자로부터 두 숫자 입력 받기

read -p "첫 번째 숫자를 입력하세요: " num1

read -p "두 번째 숫자를 입력하세요: " num2

# 사용자로부터 연산자 입력 받기

read -p "덧셈 (+) 또는 뺄셈 (-) 연산을 선택하세요: " operator

# 입력된 연산자에 따라 계산 수행

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

result=$((num1 + num2))

operator\_text="덧셈"

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

result=$((num1 - num2))

operator\_text="뺄셈"

else

echo "유효하지 않은 연산자입니다."

exit 1

fi

# 결과 출력

echo "$num1 $operator\_text $num2 = $result"

텍스트, 폰트, 스크린샷, 디자인이(가) 표시된 사진

자동 생성된 설명

Ex3-3.sh

#!/bin/bash

# 인자로 몸무게(kg)와 키(cm)를 받는다

weight=$1

height\_cm=$2

# 키를 m 단위로 변환한다 (cm를 100으로 나눈다).

# expr는 정수 연산만 지원하므로 값 \* 10000 으로 조정

height\_m\_squared=`expr $height\_cm \\* $height\_cm / 10000`

# BMI 계산: (몸무게 \* 10000) / 키^2

# 여기서도 값 \* 10000 으로 조정하여 정수 연산 수행

BMI=`expr $weight \\* 10000 / $height\_m\_squared`

# BMI 값이 1850(18.5 \* 100) 이상 2300(23 \* 100) 미만인지 확인

if [ $BMI -ge 1850 ] && [ $BMI -lt 2300 ]; then

echo "적정 체중입니다."

else

echo "비만입니다."

Fi



Ex 3-4

#!/bin/bash

while true; do

# 사용자에게 질문

read -p "계속하시겠습니까? (yes / no) " answer

# 답변 확인

case $answer in

[Yy]\* ) exit;;

[Nn]\* ) exit;;

\* ) echo "yes or no로만 답변 주세요.";;

esac

done

텍스트, 스크린샷, 폰트이(가) 표시된 사진

자동 생성된 설명

Ex 3-5

#!/bin/bash

echo "프로그램을 시작합니다"

function\_a(){

echo "함수 안으로 들어 왔음."

ls $1

}

function\_a $1

echo "프로그램 종료합니다"

텍스트, 스크린샷, 폰트이(가) 표시된 사진

자동 생성된 설명

Ex 3-6

#!/bin/bash

# 사용자로부터 폴더 이름 입력 받기

read -p "폴더 이름을 입력하세요: " folder\_name

# 폴더가 존재하지 않으면 생성

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

mkdir "$folder\_name"

echo "폴더 '$folder\_name'를 생성했습니다."

fi

# 5개의 파일 생성 (예를 들어, 파일1.txt, 파일2.txt, ...)

for i in {1..5}; do

touch "$folder\_name/파일$i.txt"

echo "파일$i.txt를 생성했습니다."

done

# 파일들을 압축

tar -czvf "$folder\_name.tar.gz" "$folder\_name"echo "파일들을 압축했습니다: $folder\_name.tar.gz"

# 새로운 폴더를 생성하여 압축 해제

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

mkdir "$new\_folder"

tar -xzvf "$folder\_name.tar.gz" -C "$new\_folder"

echo "압축을 해제하여 새로운 폴더 '$new\_folder'를 생성했습니

다."

exit 0

텍스트, 스크린샷, 폰트이(가) 표시된 사진

자동 생성된 설명

Ex3-7

#!/bin/bash

if [ ! -d $1 ] ; then

mkdir $1

cd $1

i=0

while [ $i -lt 5 ]

do

mkdir file$i

touch file$i.txt

ln -s file$i.txt file$i/file$i.txt

i=$(($i + 1))

done

fi

exit 0

텍스트, 스크린샷, 소프트웨어, 멀티미디어 소프트웨어이(가) 표시된 사진

자동 생성된 설명

EX 3-8

#!/bin/bash

# 입력값 검사

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

echo "사용 방법: $0 [이름] [전화번호]"

exit 1

fi

# 변수 할당

name=$1

phone=$2

# DB.txt 파일에 이름과 전화번호 추가

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

# DB.txt 파일에 구분자 추가

echo "---my friends---" >> DB.txt

텍스트, 스크린샷, 소프트웨어, 멀티미디어 소프트웨어이(가) 표시된 사진

자동 생성된 설명

3-9

#!/bin/bash

# 입력값 검사

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

echo "사용 방법: $0 [검색할 이름의 일부]"

exit 1

fi

# 변수 할당

search\_string=$1

# DB.txt 파일에서 입력받은 문자열로 검색하여 일치하는 라인 출력

grep "$search\_string" DB.txt

텍스트, 스크린샷, 폰트이(가) 표시된 사진

자동 생성된 설명