ex3-0.sh

#!/bin/bash

echo "hello world"

exit 0

텍스트, 폰트, 스크린샷이(가) 표시된 사진

자동 생성된 설명

ex3-1.sh

#!/bin/sh

if [ -n "$1" ]

then

    c=$1

    while [ $c -ne 0 ]

    do

        echo "hello world"

        c=$(($c - 1))

    done

else

    while [ 1 ]

    do

        echo "Hello World"

    done

fi

exit 0

텍스트, 스크린샷, 폰트이(가) 표시된 사진

자동 생성된 설명

명령어 뒤 숫자를 입력 안한경우

텍스트, 스크린샷, 폰트, 디자인이(가) 표시된 사진

자동 생성된 설명

ex3-2.sh

#!/bin/bash

# 두 숫자와 연산자를 입력받습니다.

read -p "Enter the first number: " num1

read -p "Enter the second number: " num2

read -p "Enter an operator (+ or -): " operator

# 계산을 수행하고 결과를 출력합니다.

case $operator in

    +)

        echo "Result: $(($num1 + $num2))"

        ;;

    -)

        echo "Result: $(($num1 - $num2))"

        ;;

    \*)

        echo "Error: Invalid operator!"

        ;;

esac

텍스트, 폰트, 스크린샷이(가) 표시된 사진

자동 생성된 설명

ex3-3.sh

#!/bin/bash

# 사용자로부터 몸무게(kg)와 신장(m)을 입력받습니다.

read -p "몸무게(kg)를 입력하세요: " weight

read -p "신장(m)을 입력하세요: " height

# BMI를 계산합니다.

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

# BMI에 따라 체중 상태를 판단하고 결과를 출력합니다.

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

    echo "당신은 정상 체중입니다."

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

    echo "당신은 과체중입니다."

else

    echo "당신은 저체중입니다."

fi

텍스트, 폰트, 타이포그래피, 디자인이(가) 표시된 사진

자동 생성된 설명

ex3-4.sh

#!/bin/bash

while true; do

    # 질문을 던집니다.

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

    # 입력값에 따라 분기하여 처리합니다.

    case $answer in

        [Yy]|[Yy][Ee][Ss])

            echo "리눅스를 즐기고 있어서 다행입니다!"

            break

            ;;

        [Nn]|[Nn][Oo])

            echo "아쉽네요. 더 재미있는 경험이 되도록 해봅시다!"

            break

            ;;

        \*)

            echo "잘못된 입력입니다. 'yes' 또는 'no'로 답변해주세요."

            ;;

    esac

done

텍스트, 폰트, 스크린샷, 타이포그래피이(가) 표시된 사진

자동 생성된 설명

ex3-5.sh

#!/bin/bash

# 내부 함수 정의

run\_command() {

  # 함수에 전달된 첫 번째 인자를 ls 명령어의 옵션으로 사용합니다.

  ls $1

}

# 스크립트를 사용하여 함수를 호출하고 인자를 전달합니다.

# 예시: 현재 디렉토리의 파일들을 자세히(list) 보여주는 옵션을 사용

run\_command "-l"

# 사용자로부터 입력받은 옵션으로 ls 명령어를 실행하는 예시

read -p "Enter the ls command options: " options

run\_command "$options"

텍스트, 스크린샷, 폰트이(가) 표시된 사진

자동 생성된 설명

ex3-6.sh

#!/bin/bash

# 폴더 이름을 변수로 설정합니다.

폴더\_이름="files"

# 해당 이름의 폴더가 없으면 생성합니다.

if [ ! -d "$폴더\_이름" ]; then

  mkdir "$폴더\_이름"

fi

# 해당 폴더로 이동합니다.

cd "$폴더\_이름"

# 5개의 파일을 생성합니다.

touch file0.txt file1.txt file2.txt file3.txt file4.txt

# 생성한 파일들을 files.tar로 압축합니다.

tar -cvf files.tar \*.txt

# 압축 해제할 새로운 폴더를 생성합니다.

새\_폴더\_이름="new\_files"

mkdir "$새\_폴더\_이름"

# 압축 파일을 새 폴더에 해제합니다.

tar -xvf files.tar -C "$새\_폴더\_이름"

텍스트, 스크린샷, 폰트이(가) 표시된 사진

자동 생성된 설명

ex3-7.sh

#!/bin/sh

# 첫 번째 인자로 받은 폴더 이름이 존재하는지 확인합니다.

if [ -d "$1" ]

then

   # 폴더가 이미 존재하면 아무 것도 하지 않습니다. ':'는 빈 연산입니다.

   :

else

   # 폴더가 존재하지 않으면 새로운 폴더를 만듭니다.

   mkdir $1

   # 새로 만든 폴더로 이동합니다.

   cd $1

   # 5개의 파일을 만들기 위한 카운터 변수를 초기화합니다.

   i=0

   # i가 5보다 작을 때까지 반복합니다.

   while [ $i -lt 5 ]

   do

       # file0, file1, ...과 같은 이름으로 폴더를 생성합니다.

       mkdir file$i

       # file0.txt, file1.txt, ...과 같은 이름으로 파일을 생성합니다.

       touch "file$i.txt"

       # 각 파일에 대한 심볼릭 링크를 해당 폴더 내에 생성합니다.

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

       # 카운터를 1 증가시킵니다.

       i=$((i + 1))

   done

fi

exit 0

텍스트, 스크린샷, 폰트이(가) 표시된 사진

자동 생성된 설명

ex3-8.sh

#!/bin/sh

# 데이터베이스 파일명을 변수에 저장합니다.

database="DB.txt"

# 데이터베이스 파일이 존재하지 않으면 새로 생성합니다.

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

    touch "$database"

fi

# 첫 번째 인자(이름)를 변수에 저장합니다.

name="$1"

# 두 번째 인자(정보)를 변수에 저장합니다.

info="$2"

# 이름과 정보를 데이터베이스 파일에 추가합니다.

echo "$name: $info" >> "$database"

# 사용자에게 정보 추가 완료 메시지를 출력합니다.

echo "정보가 DB.txt 파일에 추가되었습니다."

exit 0

텍스트, 스크린샷, 폰트이(가) 표시된 사진

자동 생성된 설명

ex3-9.sh

#!/bin/sh

# 데이터베이스 파일명을 변수에 저장합니다.

database="DB.txt"

# 데이터베이스 파일 존재 여부를 확인합니다.

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

    # 데이터베이스 파일이 없으면 경고 메시지를 출력하고 스크립트를 종료합니다.

    echo "DB 파일이 없습니다."

    exit 1

fi

# 검색할 이름을 첫 번째 인자로부터 가져옵니다.

search\_name="$1"

# 데이터베이스 파일에서 이름에 해당하는 정보를 검색합니다.

info=$(grep "^$search\_name:" "$database")

# 검색된 정보가 있으면 그 정보를 출력합니다.

if [ -n "$info" ]; then

    echo "$info"

else

    # 정보가 없으면 사용자에게 검색 결과가 없음을 알립니다.

    echo "검색 결과가 없습니다."

fi

exit 0

텍스트, 폰트, 스크린샷, 라인이(가) 표시된 사진

자동 생성된 설명