

**코드**



3-0

#!/bin/sh: 스크립트를 sh 셸에서 실행하도록 지정합니다.

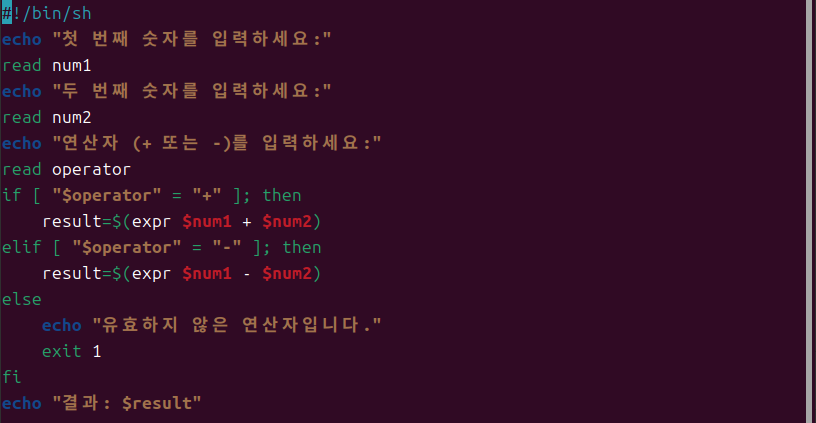
echo "Hello World": "Hello World"라는 문구를 출력합니다.



3-1  
echo "반복할 횟수를 입력하세요:": 사용자에게 반복할 횟수를 입력하도록 요청합니다.

read count: 입력된 횟수를 count 변수에 저장합니다.

for i in $(seq 1 $count); do ... done: seq 1 $count를 통해 1부터 count까지의 숫자를 생성하고, 그 횟수만큼 Hello World를 출력합니다.



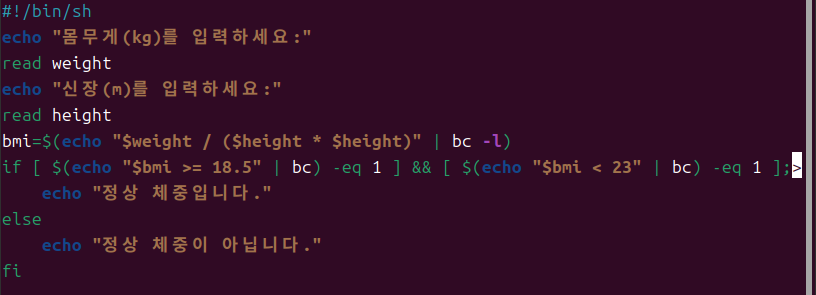
3-2

echo/read 명령어로 첫 번째 숫자, 두 번째 숫자, 연산자를 입력받습니다.

if [ "$operator" = "+" ]; then ... elif [ "$operator" = "-" ]; then ... else ... fi: 입력된 연산자가 + 또는 -인지 확인하여 해당 연산을 수행합니다.

expr: 산술 연산을 수행합니다.

echo "결과: $result": 계산된 결과를 출력합니다.



3-3

read weight 및 read height: 몸무게와 신장을 입력받습니다.

bmi=$(echo "$weight / ($height \* $height)" | bc -l): BMI를 계산합니다. bc -l은 정확한 소수 계산을 위해 사용됩니다.

if ... fi: BMI 값을 기준으로 18.5 이상, 23 미만이면 정상 체중으로 판단하여 출력합니다.



3-4

read action: 사용자가 선택한 옵션을 action 변수에 저장합니다.

case ... esac: action의 값에 따라 다른 메시지를 출력합니다. 입력 값이 start, stop, status 중 하나가 아니면 기본 메시지를 출력합니다.

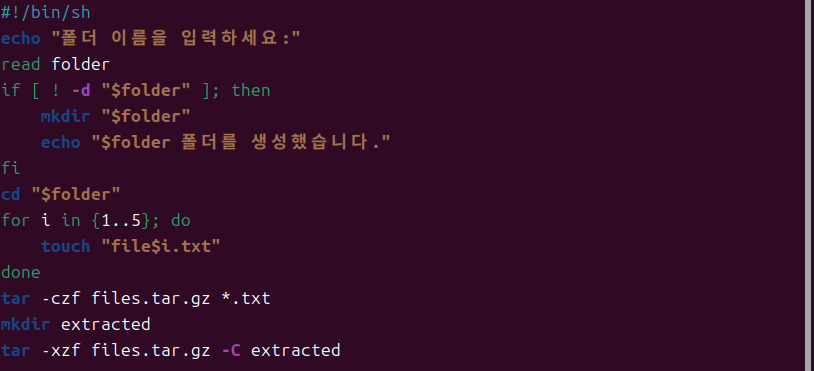


3-5

list\_files(): 지정한 디렉토리의 파일 목록을 출력하는 함수입니다.

read dir: 디렉토리 이름을 입력받아 dir 변수에 저장합니다.

list\_files "$dir": 입력받은 디렉토리를 list\_files 함수에 전달하여 해당 디렉토리의 파일 목록을 출력합니다.



3-6

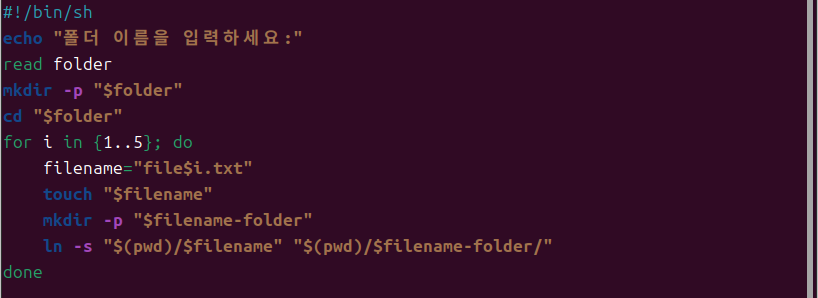
read folder: 폴더 이름을 입력받아 저장합니다.

mkdir "$folder": 폴더가 없으면 생성합니다.

for ... touch "file$i.txt": 5개의 빈 텍스트 파일을 생성합니다.

tar -czf files.tar.gz \*.txt: 파일들을 files.tar.gz로 압축합니다.

mkdir extracted와 tar -xzf ... -C extracted: 압축 파일을 extracted 폴더에 해제합니다.



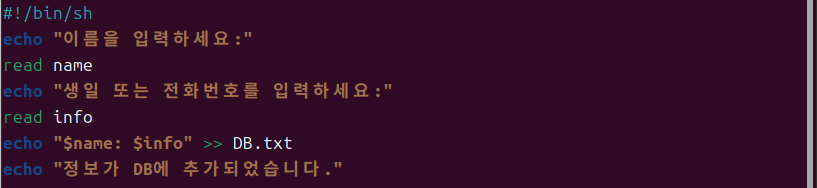
3-7

mkdir -p "$folder": 지정된 폴더가 없으면 생성합니다.

for ... touch "$filename": file1.txt에서 file5.txt까지 생성합니다.

mkdir -p "$filename-folder": 각 파일에 대응하는 하위 폴더를 생성합니다.

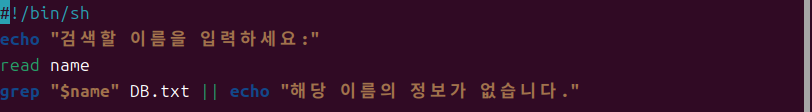
ln -s ...: 파일의 심볼릭 링크를 하위 폴더에 만듭니다.



3-8

read name 및 read info: 이름과 정보를 입력받아 저장합니다.

echo "$name: $info" >> DB.txt: DB.txt 파일에 입력받은 정보가 추가됩니다.



3-9

read name: 검색할 이름을 입력받아 name 변수에 저장합니다.

grep "$name" DB.txt || ...: DB.txt에서 이름을 검색하여 찾으면 결과를 출력하고, 없으면 메시지를 출력합니다.