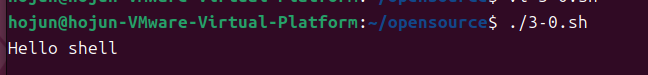
**오픈소스기초설계**

**[셸프로그래밍] 20211577 김호준**

**1. 3-0.sh**

* 출력 결과 스크린샷



* 코드

# 3-0.sh

#!/bin/bash

grep -qFx 'export MYENV="Hello Shell"' ~/.bashrc || echo 'export MYENV="Hello Shell"' >> ~/.bashrc

source ~/.bashrc

bash -c 'echo $MYENV'

sed -i '\#export MYENV="Hello Shell"#d' ~/.bashrc

unset MYENV

bash -c 'echo $MYENV'

**2. 3-1.sh**

* 출력 결과 스크린샷

텍스트, 스크린샷, 폰트이(가) 표시된 사진

AI 생성 콘텐츠는 정확하지 않을 수 있습니다.

* 코드
* #!/bin/bash
* read -p "두 개의 숫자: " num1 num2
* if [ -z "$num1" ] || [ -z "$num2" ]; then
* echo "error: 두 개의 숫자 입력."
* exit 1
* fi
* echo "입력: num1=$num1, num2=$num2"
* #덧셈
* sum=$(($num1 + $num2))
* echo "$num1 + $num2 = $sum"
* # 뺄셈
* diff=$(($num1 - $num2))
* echo "$num1 - $num2 = $diff"
* # 곱셈
* prod=$(($num1 \* $num2))
* echo "$num1 \* $num2 = $prod"
* if [ $num2 -eq 0 ]; then
* echo "$num1 / $num2 = 0, error:divide by zero"
* else
* quot=$(($num1 / $num2))
* echo "$num1 / $num2 = $quot"
* fi

**3. 3-2.sh**

* 출력 결과 스크린샷

텍스트, 스크린샷, 폰트이(가) 표시된 사진

AI 생성 콘텐츠는 정확하지 않을 수 있습니다.

* 코드
* #3-2.sh
* #!/bin/bash
* read -p "두 개 이상의 숫자 입력: " -a x\_array
* count=${#x\_array[@]} # 배열의 총 개수
* if [ $count -lt 2 ]; then
* echo "error: 두 개 이상의 숫자를 입."
* exit 1
* fi
* echo "y=1/2\*x^2"
* for x in "${x\_array[@]}"
* do
* y=$(echo "scale=4; 0.5 \* $x \* $x" | bc)
* echo "x = $x => y = $y"
* done

**4. 3-3.sh**

* 출력 결과 스크린샷

텍스트, 스크린샷이(가) 표시된 사진

AI 생성 콘텐츠는 정확하지 않을 수 있습니다.

* 코드
* #3-3.sh
* #!/bin/bash
* read -p "점수 입력 : " -a score\_array
* count=${#score\_array[@]}
* if [ $count -lt 2 ]; then
* echo "error: 두 개 이상의 점수를 입력."
* exit 1
* fi
* total\_sum=0
* echo "각각 등급"
* for score in "${score\_array[@]}"
* do
* if ! [[ "$score" =~ ^[0-9]+$ ]] || [ $score -lt 0 ] || [ $score -gt 100 ]; then
* echo "error: '$score' - 0<= score <=100"
* exit 1
* fi
* if [ $score -ge 90 ]; then
* grade="A"
* else
* grade="B"
* fi
* echo "점수: $score  =>  등급: $grade"
* total\_sum=$(($total\_sum + $score))
* done
* echo "평균 등급"
* avg\_score=$(($total\_sum / $count))
* echo "총점: $total\_sum"
* echo "과목 수: $count"
* echo "평균 점수: $avg\_score"
* if [ $avg\_score -ge 90 ]; then
* avg\_grade="A"
* else
* avg\_grade="B"
* fi
* echo "평균 등급: $avg\_grade"

**5. 3-4.sh**

* 출력 결과 스크린샷

텍스트, 스크린샷이(가) 표시된 사진

AI 생성 콘텐츠는 정확하지 않을 수 있습니다.

텍스트, 스크린샷, 디자인이(가) 표시된 사진

AI 생성 콘텐츠는 정확하지 않을 수 있습니다.

텍스트, 폰트, 스크린샷, 디자인이(가) 표시된 사진

AI 생성 콘텐츠는 정확하지 않을 수 있습니다.

* 코드
* #3-4.sh
* #!/bin/bash
* scores=()
* calculate\_average() {
* count=${#scores[@]}
* if [ $count -eq 0 ]; then
* echo "error: 입력된 점수가 없음."
* return 1 # 실패 반환
* fi
* total\_sum=0
* for s in "${scores[@]}"; do
* total\_sum=$(($total\_sum + $s))
* done
* avg\_score=$(($total\_sum / $count))
* return 0
* }
* while true
* do
* # --- 1. 메뉴 출력 ---
* echo "==================="
* echo "1) 과목 성적 추가"
* echo "2) 입력된 모든 점수 보기"
* echo "3) 평균 점수 확인"
* echo "4) 평균 등급 (A/B) 변환"
* echo "5) 종료"
* echo "==================="
* read -p "메뉴를 선택하세요 (1-5): " choice
* echo ""
* case $choice in
* 1)
* echo "--- 1) 과목 성적 추가 ---"
* read -p "추가할 점수를 입력하세요 (0-100): " new\_score
* if ! [[ "$new\_score" =~ ^[0-9]+$ ]] || [ $new\_score -lt 0 ] || [ $new\_score -gt 100 ]; then
* echo "오류: '$new\_score' - 점수는 0에서 100 사이의 정수여야 합니다."
* else
* scores+=($new\_score)
* echo "$new\_score 점이 성공적으로 추가되었습니다."
* fi
* ;;
* 2)
* echo "--- 2) 입력된 모든 점수 보기 ---"
* count=${#scores[@]}
* if [ $count -eq 0 ]; then
* echo "입력된 점수가 없습니다."
* else
* # ${scores[@]} : 배열의 모든 요소를 출력
* echo "총 ${count}개 점수: ${scores[@]}"
* fi
* ;;
* 3)
* echo "--- 3) 평균 점수 확인 ---"
* # 헬퍼 함수 호출, 'if'로 성공(0) 여부 확인
* if calculate\_average; then
* echo "총점: $total\_sum"
* echo "과목 수: $count"
* echo "평균 점수 (정수): $avg\_score"
* fi
* ;;
* 4)
* echo "--- 4) 평균 등급 (A/B) 변환 ---"
* if calculate\_average; then
* if [ $avg\_score -ge 90 ]; then
* avg\_grade="A"
* else
* avg\_grade="B"
* fi
* echo "평균 점수 $avg\_score 는 등급으로 '$avg\_grade' 입니다."
* fi
* ;;
* 5)
* echo "프로그램을 종료합니다."
* break
* ;;
* \*)
* echo "잘못된 입력입니다. 1에서 5 사이의 숫자를 입력하세요."
* ;;
* esac
* echo ""
* done

**6. 3-5.sh**

* 출력 결과 스크린샷

텍스트, 스크린샷, 폰트이(가) 표시된 사진

AI 생성 콘텐츠는 정확하지 않을 수 있습니다.

* 코드
* #3-5.sh
* #!/bin/bash
* run\_ls\_with\_options() {
* local command\_to\_run="ls $@"
* echo "--- eval로 실행될 명령어 문자열 ---"
* echo "$command\_to\_run"
* eval $command\_to\_run
* }
* echo "스크립트에 전달된 인자: $@"
* echo ""
* run\_ls\_with\_options "$@"

**7. 3-6.sh**

* 출력 결과 스크린샷

텍스트, 스크린샷, 폰트이(가) 표시된 사진

AI 생성 콘텐츠는 정확하지 않을 수 있습니다.

* 코드

Python 코드

* #!/usr/bin/env python3
* import sys
* print("--- [Python] 실행 파일 시작 ---")
* arguments = sys.argv[1:]
* print(f"[Python] 수신된 인자 리스트: {arguments}")
* print(f"[Python] 수신된 인자 개수: {len(arguments)}")
* print("--- [Python] 실행 파일 종료 ---")

shell 코드

#3-6.sh

#!/bin/bash

read -p "전달할 2개 이상의 인자를 입력: " -a input\_array

count=${#input\_array[@]}

if [ $count -lt 2 ]; then

    echo "error: 2개 이상의 인자가 필요."

    exit 1

fi

echo "[셸] Python(3-6.py)를 호출"

echo "[셸] Python에 전달할 인자: ${input\_array[@]}"

python3 3-6.py "${input\_array[@]}"

echo "--- [셸] 3-6.sh 종료 ---"

**8. 3-7.sh**

* 출력 결과 스크린샷

텍스트, 스크린샷, 폰트이(가) 표시된 사진

AI 생성 콘텐츠는 정확하지 않을 수 있습니다.

텍스트, 폰트, 스크린샷, 번호이(가) 표시된 사진

AI 생성 콘텐츠는 정확하지 않을 수 있습니다.

텍스트, 스크린샷, 폰트이(가) 표시된 사진

AI 생성 콘텐츠는 정확하지 않을 수 있습니다.

텍스트, 스크린샷, 폰트이(가) 표시된 사진

AI 생성 콘텐츠는 정확하지 않을 수 있습니다.

* 코드
* #3-7.sh
* #!/bin/bash
* while true
* do
* echo "============================="
* echo "Linux 시스템 상태 확인"
* echo "============================="
* echo "1) 사용자 정보 (w)"
* echo "2) GPU / CPU 사용률 (nvidia-smi / top)"
* echo "3) 메모리 사용량 (free)"
* echo "4) 디스크 사용량 (df)"
* echo "5) 종료"
* echo "============================="
* read -p "메뉴를 선택하세요 (1-5): " choice
* echo ""
* case $choice in
* 1)
* echo "--- 1) 사용자 정보 ---"
* echo "현재 시스템에 접속된 사용자 정보를 표시합니다."
* w
* ;;
* 2)
* echo "--- 2) GPU / CPU 사용률 ---"
* if command -v nvidia-smi &> /dev/null
* then
* echo "[NVIDIA GPU 사용률]"
* nvidia-smi
* else
* echo "[CPU 사용률 (1회 실행)]"
* top -b -n 1 | grep "Cpu(s)"
* fi
* ;;
* 3)
* echo "--- 3) 메모리 사용량 ---"
* echo "시스템의 전체, 사용 중, 여유 메모리 (MB/GB 단위)"
* free -h
* ;;
* 4)
* echo "--- 4) 디스크 사용량 ---"
* echo "마운트된 파일 시스템별 디스크 사용량 (MB/GB 단위)"
* df -h
* ;;
* 5)
* echo "프로그램을 종료합니다."
* break
* ;;
* \*)
* echo "잘못된 입력입니다. 1에서 5 사이의 숫자를 입력하세요."
* ;;
* esac
* echo ""
* done

**9. 3-8.sh**

* 출력 결과 스크린샷

텍스트, 스크린샷이(가) 표시된 사진

AI 생성 콘텐츠는 정확하지 않을 수 있습니다.

텍스트, 스크린샷, 번호이(가) 표시된 사진

AI 생성 콘텐츠는 정확하지 않을 수 있습니다.

DB 파일 내에 파일

텍스트, 스크린샷, 폰트, 번호이(가) 표시된 사진

AI 생성 콘텐츠는 정확하지 않을 수 있습니다.

Train에 링크 된 파일

텍스트, 스크린샷, 폰트, 디자인이(가) 표시된 사진

AI 생성 콘텐츠는 정확하지 않을 수 있습니다.

* 코드
* #3-8.sh
* #!/bin/bash
* echo "--- 1. 'DB' 폴더 확인 및 생성 ---"
* mkdir -p "DB"
* echo "'DB' 폴더가 준비되었습니다."
* echo ""
* echo "DB 폴더에 5개 파일 생성 중 (file1.txt ~ file5.txt)..."
* for i in {1..5}
* do
* echo "이것은 file${i}의 내용입니다." > "DB/file${i}.txt"
* done
* echo "DB 폴더 내에서 파일 압축 중 (db\_files.tar.gz)..."
* tar -czvf "DB/db\_files.tar.gz" -C "DB" file1.txt file2.txt file3.txt file4.txt file5.txt
* echo "압축 완료."
* echo ""
* echo "--- 3. 'train' 폴더 생성 및 5개 파일 링크 ---"
* mkdir -p "train"
* CURRENT\_DIR=$(pwd)
* echo "'train' 폴더에 5개 원본 파일의 심볼릭 링크 생성..."
* for i in {1..5}
* do
* local\_target="$CURRENT\_DIR/DB/file${i}.txt"
* local\_link="train/link\_file${i}.txt"
* ln -sf "$local\_target" "$local\_link"
* done
* echo "[DB 폴더 내용]"
* ls -l DB
* echo ""
* echo "[train 폴더 내용]"
* ls -l train
* echo ""
* echo "[링크된 파일 내용 확인 (예: link\_file1.txt)]"
* cat "train/link\_file1.txt"

**10. 3-9.sh**

* 출력 결과 스크린샷

텍스트, 스크린샷이(가) 표시된 사진

AI 생성 콘텐츠는 정확하지 않을 수 있습니다.

텍스트, 스크린샷, 폰트이(가) 표시된 사진

AI 생성 콘텐츠는 정확하지 않을 수 있습니다.

* 코드
* #3-9.sh
* #!/bin/bash
* DB\_FILE="DB.txt"
* touch "$DB\_FILE"
* while true
* do
* echo "==================="
* echo "1) 팀원 정보 추가"
* echo "2) 팀원과 한 일 기록"
* echo "3) 팀원 검색 (이름)"
* echo "4) 수행 내용 검색 (날짜/키워드)"
* echo "5) 종료"
* echo "==================="
* read -p "메뉴를 선택하세요 (1-5): " choice
* echo ""
* case $choice in
* 1)
* echo "--- 1) 팀원 정보 추가 ---"
* read -p "추가할 팀원 이름: " name
* read -p "생일 또는 전화번호: " info
* echo "MEMBER | $name | $info" >> "$DB\_FILE"
* echo "[$name] 님의 정보를 $DB\_FILE 에 저장했습니다."
* ;;
* 2)
* echo "--- 2) 팀원과 한 일 기록 ---"
* default\_date=$(date +%Y-%m-%d)
* read -p "날짜 (기본값: $default\_date): " log\_date
* if [ -z "$log\_date" ]; then
* log\_date=$default\_date
* fi
* read -p "수행한 일 내용: " activity
* echo "LOG | $log\_date | $activity" >> "$DB\_FILE"
* echo "[$log\_date] 활동을 $DB\_FILE 에 저장했습니다."
* ;;
* 3)
* echo "--- 3) 팀원 검색 (이름) ---"
* read -p "검색할 팀원 이름: " search\_name
* echo ""
* echo "[$search\_name] (으)로 검색된 팀원 정보:"
* result=$(grep "^MEMBER" "$DB\_FILE" | grep -i "$search\_name")
* if [ -z "$result" ]; then
* echo "일치하는 팀원 정보가 없습니다."
* else
* echo "$result"
* fi
* ;;
* 4)
* echo "--- 4) 수행 내용 검색 (날짜/키워드) ---"
* read -p "검색할 날짜(YYYY-MM-DD) 또는 키워드: " search\_keyword
* echo ""
* echo "[$search\_keyword] (으)로 검색된 수행 내용:"
* result=$(grep "^LOG" "$DB\_FILE" | grep -i "$search\_keyword")
* if [ -z "$result" ]; then
* echo "일치하는 수행 내용이 없습니다."
* else
* echo "$result"
* fi
* ;;
* 5)
* echo "프로그램을 종료합니다."
* break
* ;;
* \*)
* echo "잘못된 입력입니다. 1에서 5 사이의 숫자를 입력하세요."
* ;;
* esac
* done