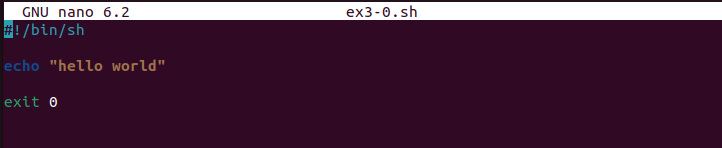
**오픈소스기초설계**

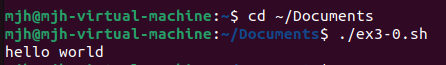
**과제 보고서**

Shell script 프로그래밍 실습

|  |  |
| --- | --- |
| **학과** | AI융합학부 |
| **학번** | 20243276 |
| **이름** | 문장혁 |
| **담당교수** | 김성흠 |
| **제출날짜** | 2024.11.08.금요일 |

**1. ex3-0** (코드,출력물)





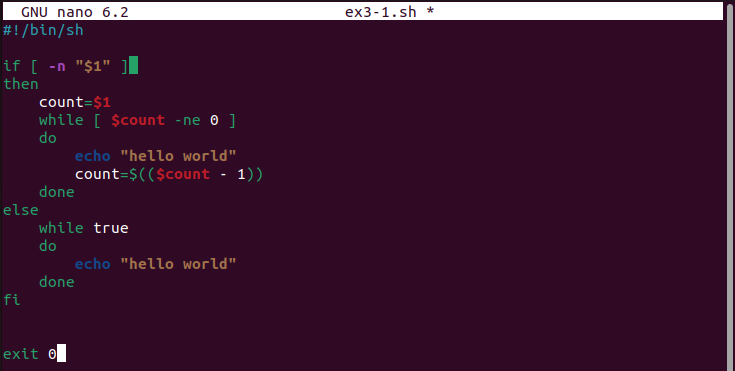
- 코드 설명

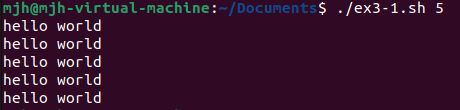
1) 스크립트할 셸을 지정하는 쉐뱅으로 시작하였다.

2) echo으로 hellow world 를 인자로 받아 그대로 출력하였다.

3) exit 0으로 스크립트를 종료하며 종료 상태로 0을 반환하였다.

**2. ex3-1**





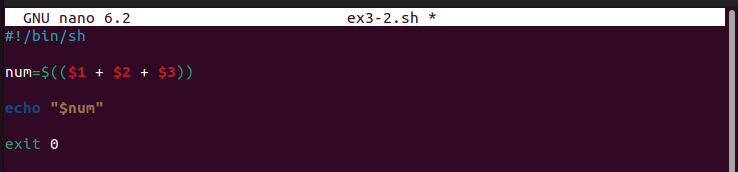
1) if 문을 통해 첫 번째 인자가 존재하는지 확인하였다. -n은 문자열이 비어있지 않으면 True로 반환한다. 만약 첫 번째 인자가 존재하면 해당 인자를 count 변수에 할당한다.

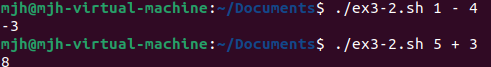
2) count 값을 기반으로 hello world가 출력될 횟수를 제한한다.

3) count가 0이 아닐 때 반복하고, 0이 되면 반복문을 종료한다.

4) 화면에 hello world를 출력한다.

**3. ex3-2**

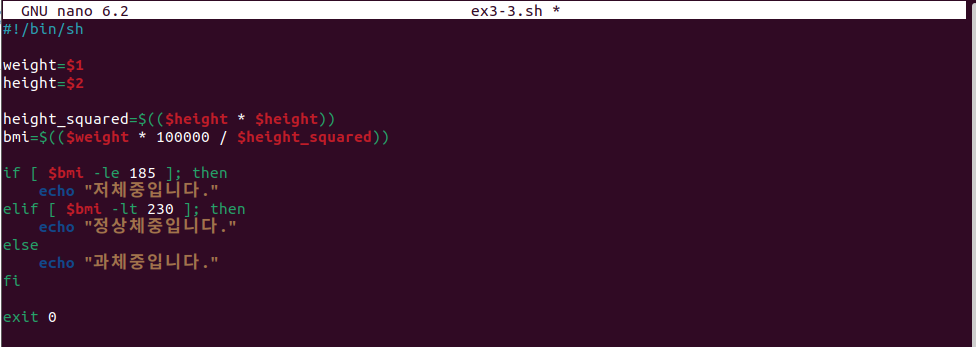


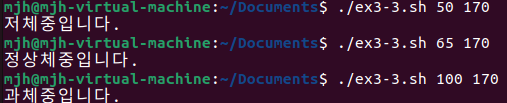


1) 세 개의 인자를 더해서 num 변수에 저장한다. ($(())는 산술 연산을 수행할 때 사용됨.)

2) num 변수에 저장된 값을 출력한다.

**4. ex3-3**





1) weight와 height를 변수에 저장한다.

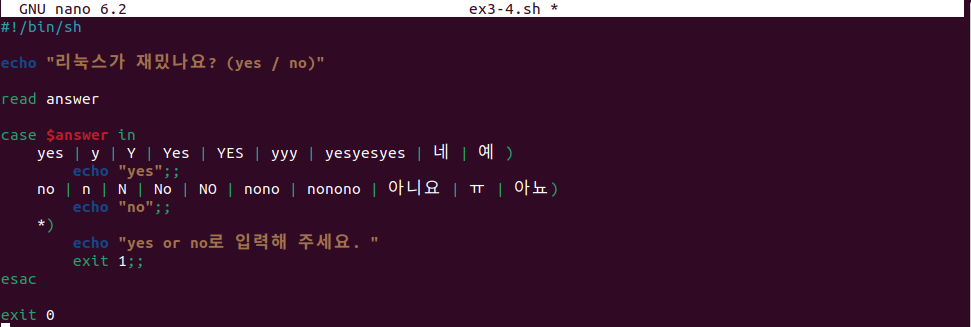
2) height\_squared에 키의 제곱을 계산하여 저장한다. 나중에 bmi를 계산할 때 사용한다.

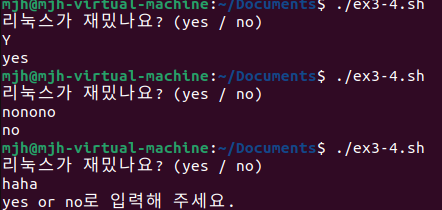
3) bmi 계산 공식인 체중/을 사용한다. (여기서 키는 cm로 받기 때문에 10000을 곱해주어

으로 변환해준다.

4) if 조건문을 사용하여 bmi 값이 185 이하인 경우 저체중, 230 미만인 경우 정상체중, 위의 조건을 모두 만족하지 않으면 과체중으로 출력한다.

**5. ex3-4**





1) 사용자에게 “리눅스가 재밌나요?” 라는 질문을 출력한다

2) read 명령어를 통해 사용자의 입력을 받고 answer에 저장한다.

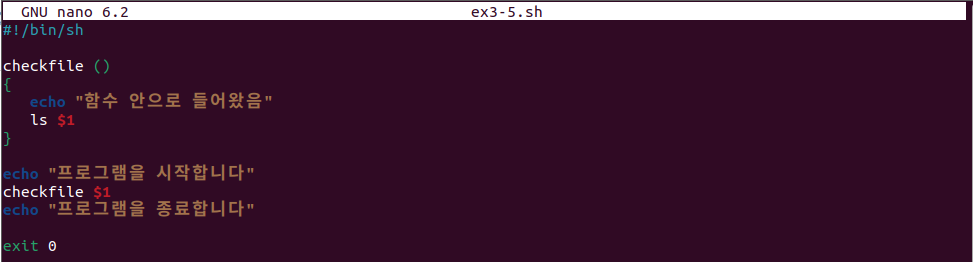
3) case문을 사용하여 answer 변수에 저장된 값에 따라 다른 동작을 실행한다.

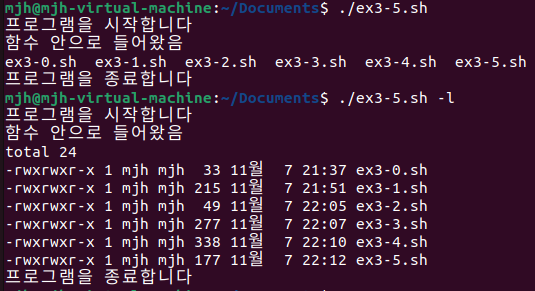
- 첫 번째로 yes, y, Y … 등등 다른 변형들을 입력한 경우 yes를 출력한다.

- 두 번째로 no, n, N … 등등 다른 변형들을 입력한 경우 no를 출력한다.

- 세 번째로 위의 조건에 맞지 않는 경우, “yes or no로 입력해주세요”가 출력되고 exit1으로 스크립트를 비정상 종료한다.

**6. ex3-5**





1) 인자로 전달된 파일, 디렉터리의 정보를 확인하는 기능을 수행하는 checkfile 함수를 정의한다.

2) “함수 안으로 들어왔음”을 출력한다.

3) 함수의 첫번째 인자로 전달된 값을 ls 명령어로 출력한다. (ls : 파일이나 디렉터리 정보 보여줌.)

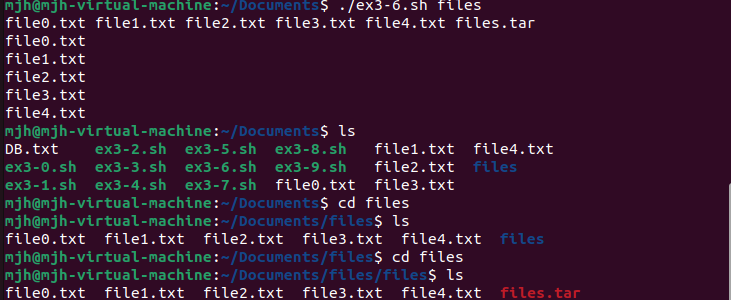
4) “프로그램을 시작합니다.”를 출력한다.

5) checkfile 함수를 호출하며 스크립트를 시작할 때 저장된 첫 번째 인자를 함수에 전달한다.

6) “프로그램을 종료합니다”를 출력하고 exit 0을 이용하여 정상적으로 스크립트를 종료한다.

**7. ex3-6**

****



1) 아까와 같이 if 문을 사용해 첫 번째 인자로 전달된 디렉토리가 존재하는지 확인한다. (-d는 해당 경로가 디렉토리인지 확인하는 역할)

2) then : 디렉터리가 이미 존재하면 정상적으로 종료한다.

3) else : 디렉터리가 존재하지 않는 경우 아래의 작업을 수행한다.

4) $1 이름으로 디렉터리를 생성한 후 cd를 이용하여 디렉터리로 이동한다.

5) i=0으로 변수를 초기화하고 while문을 이용하여 i를 1씩 증가시키며 5번 반복문을 실행한다.

6) done을 사용하여 while문을 종료하고 file0.txt부터 file4.txt까지의 파일을 $1.tar라는 이름의 tar 파일로 압축한다.

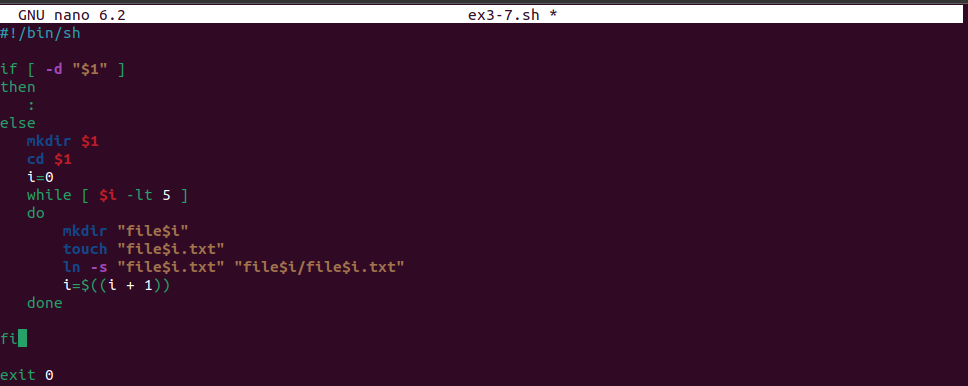
7) 생성된 tar 파일을 $1 디렉터리로 이동시킨 후, $1.tar 파일을 해제하여 디렉터리 안에 파일을 복원한다. (> /dev/null은 출력 결과를 화면에 표시하지 않고 버린다.)

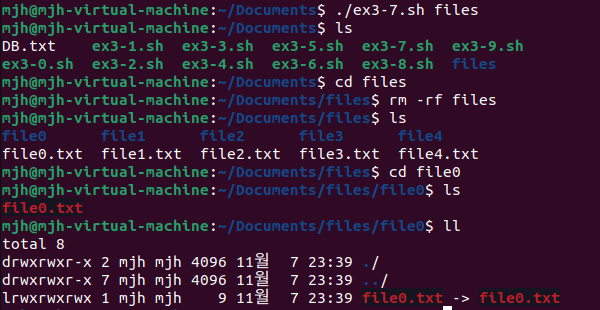
8) echo를 사용하여 현재 디렉터리에 있는 파일 목록을 출력하고 cd를 통해 스크립트가 시작된 위치로 돌아간다.

9) $1 디렉터리 안의 .txt 파일의 이름만 출력하고, basename 명령어를 통해 파일의 경로를 제외하고 파일 이름만 출력한다. (xargs -n 1을 사용하여 한 번에 하나씩 basename에 전달함.)

10) fi 로 if 문을 종료한다.

**8. ex3-7**





1) if 문을 이용하여 첫 번째 인자로 전달된 디렉토리가 존재하는지 확인한다.

2) then : 디렉토리가 이미 존재할 경우 바로 종료한다.

3) else : 디렉토리가 존재하지 않을 경우 아래 작업을 실행한다

4) mkdir 을 이용해 첫 번째 인자로 받은 이름의 디렉토리를 생성한다.

5) cd를 이용해 생성된 디렉토리로 이동한다.

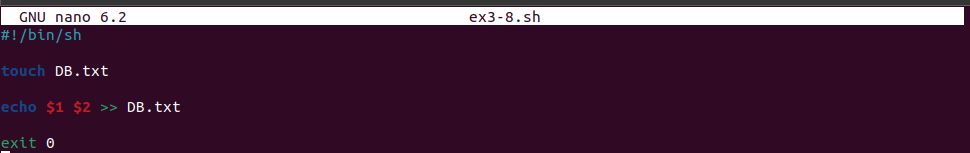
6) 변수 i=0으로 초기화하고 i가 5보다 작을 동안 반복문을 실행한다.

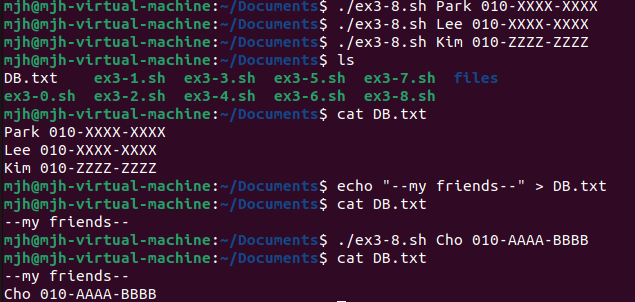
7) file0, file1 … 와 같은 이름의 하위 디렉토리를 순서대로 생성한다.

8) 심볼릭 링크 (ln -s)를 생성한다. 그리고 files$i.txt 파일에 대한 심볼릭 링크를 file$i 디렉터리 내부에 file$i.txt라는 이름으로 만든다.

9) 카운터 변수 i를 1씩 증가시키고 while문을 종료한다. fi를 이용하여 if문을 종료한다.

**9. ex3-8**

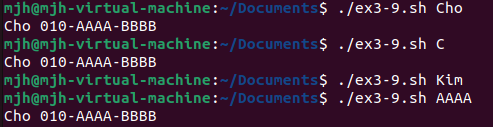


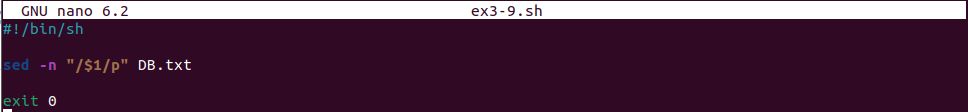


1) DB.txt 파일을 생성한다.

2) echo를 사용하여 첫 번째 인자와 두 번째 인자를 DB.txt 파일에 추가한다. (>> 연산자는 기존 파일의 내용을 유지하면서 새 내용을 끝에 추가한다.)

**10. ex3-9**





1) sed 명령어를 이용하여 DB.txt 파일에서 첫 번째 인자에 해당하는 문자열이 포함된 줄을 출력한다. (-n 옵션은 sed의 기본 출력 기능을 비활성화한다, “/$1/p 부분은 $1에 해당하는 문자열을 검색하고 그 줄을 출력한다.”)