**[오픈소스SW개론 과제1 보고서 – 12201838임정섭]**

[과제 스크립트 작성 방법]

저는 개발환경의 편리성을 위하여 WSL2와 Ubuntu를 설치하고, VisualStudioCode를 Ubuntu환경에서 실행하여 스크립트를 개발하였습니다.

[스크립트 구현 내용]

shellAssignment.sh라는 하나의 파일을 가지고 과제를 구현했습니다.

과제는 세 개의 파일을 인자로 받기에, 터미널에서 아래와 같은 명령어를 통해 스크립트 파일을 실행했습니다.



과제에 대한 스크립트 구현 내용은 아래와 같습니다.

우선 세 가지 파일을 위치기반인자를 통해 입력을 받고, 기본 출력양식(학번 이름 및 구현번호)를 출력했습니다.

번호가 부여된 과제들에 대한 구현 내용은 다음과 같습니다.

1] 사용자로부터 read -p 명령어를 통해 movieId를 입력받고, u.item파일 내용 중 awk를 통하여 행을 선택하는 출력을 했습니다.

텍스트, 스크린샷, 소프트웨어, 폰트이(가) 표시된 사진

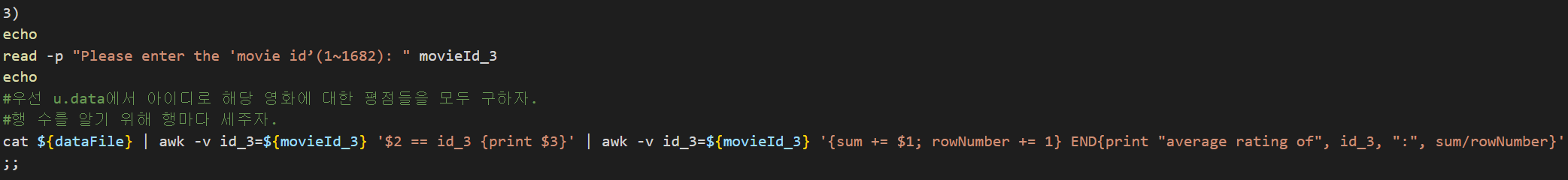
자동 생성된 설명

2] 7번째 필드가 1이면 Action 장르이기에 awk의 필터에 해당 내용을 작성했습니다. sort와 head 명령어를 통해 정렬 후 상위 10개의 행만 뽑아 출력했습니다.

텍스트, 스크린샷, 멀티미디어 소프트웨어, 폰트이(가) 표시된 사진

자동 생성된 설명

3] 받은 movieId를 기반으로 u.data에서 awk를 통해 해당 행의 평점부분만 뽑아낸 후, 평점의 합을 행수의 합으로 나누어 평균을 구했습니다. Awk에 END패턴을 부여하면 마지막 행까지 처리 후 END패턴에 해당하는 동작을 수행하기 때문에, 평균을 구할 수 있습니다.



4] 요구 사항인 10라인만 출력과 ‘IMDB Url’필드를 제거한 출력은, NR<=10이라는 패턴과 ‘IMDB Url’에 해당하는 5번째 필드만 제외한 출력 포맷을 지정해 하나의 awk명령어를 통해 처리할 수 있었습니다.

스크린샷, 텍스트, 라인이(가) 표시된 사진

자동 생성된 설명

5] 우선 u.user의 M과 F로 표시된 성별을 male과 female이라는 문자열로 바꿔줘야 하기에, sed명령어와 정규표현식을 통해 해결했습니다. 정확한 문자열 캐치를 위해 \b를 사용하여 word boundary를 지정했습니다. 이후엔 ()를 사용하여 u.user의 각 필드들을 그룹으로 묶고, \1 \2 방식으로 지정한 그룹의 순서를 재배치했습니다.

스크린샷, 텍스트이(가) 표시된 사진

자동 생성된 설명

6] 우선 last 10 lines를 지정하기 위해 tail 명령어를 사용했습니다. 이후 5번과 비슷하게, 모든 영문으로 된 Month들을 sed명령어로 전부 해당하는 숫자형 Month로 치환 후 그룹개념을 사용하여 필드들을 원하는 순서로 배치했습니다. 이 과제를 해결하면서 sed의 -E옵션에 대해 학습했고, 일반 정규표현식보다 더 사용이 용이한 확장정규표현식을 사용했습니다.

스크린샷, 텍스트이(가) 표시된 사진

자동 생성된 설명

7] 출력 상 윗쪽에 해당하는 부분은, u.data에서 입력으로 받은 userId를 패턴에 지정해 사용자가 평가한 영화의 id들만 출력할 수 있었습니다. 다만 마지막 영화id출력 뒤에는 구분자 ‘|’가 붙지 않기에, sed명령어를 통해 처리해줬습니다. 끝을 가리키는 $문자를 사용했습니다.  
출력 상 아랫쪽에 해당하는 부분을 해결한 내용을 설명하겠습니다.  
우선 윗 부분의 출력내용 중 상위 10개의 영화id만 행 별로 담아주는 별도의 파일(targetMovIdAndMovName\_12201838)을 리다이렉션(>)으로 생성했습니다. 이 파일을 while문의 입력으로 주어, 파일의 행을(즉, 영화 id를) while문의 조건변수(movieId\_7)로 삼아 u.item파일에 대한 awk의 패턴으로 사용하여 검색할 수 있었습니다.  
 텍스트, 스크린샷이(가) 표시된 사진

자동 생성된 설명

8] 우선 u.user에서 조건에 해당하는 userId만 awk를 통해 뽑아내어, targetUser\_12201838에 저장했습니다. 이 파일을 while문의 입력으로 주고, while문에서는 해당하는 userId를 바탕으로 u.data에서 userId, movieId, score을 뽑아내어 targetData\_12201838에 매번 append해주어 파일을 만들었습니다. append개념이기에, while문 진입 전 매번 해당 파일을 truncate해주어야 원하는 데이터만을 뽑아낼 수 있었습니다. 평균은 3번처럼 awk를 통해 평균을 구한 뒤 오름차순으로 정렬해 출력했습니다.

텍스트, 스크린샷, 소프트웨어, 폰트이(가) 표시된 사진

자동 생성된 설명

9]

텍스트, 폰트, 스크린샷, 타이포그래피이(가) 표시된 사진

자동 생성된 설명

[GitHub add, commit, push]

텍스트, 전자제품, 스크린샷, 소프트웨어이(가) 표시된 사진

자동 생성된 설명

마찬가지로 VisualStudioCode의 WSL-Ubuntu 터미널을 사용하여 깃허브에 관련 부속파일 및 쉘코드를 업로드했습니다.