

우선 전체를 while true 로 감싼후에 read -p 로 Enter your choice 를 출력하고 사용자 입력을 받아 choice 에 저장한다.

choice 가 1 이라면 Please enter the 'movie id' (1~1682):를 출력하고 사용자 입력을 받아 movieid 에 저장한다. 이후 awk 명령어에서 -v 옵션을 통해 movieid 를 id 에 저장하고 -F 옵션을 통해 구분자를 |로 한다. u.item 을 한 줄씩 |로 구분하여 그 첫번째 인자가 입력받은 movie id 즉 id와 같으면 \$0 를 print 하여 그 줄을 모두 출력한다.

choice 가 2 라면 Do you want to get the data of 'action' genre movies from 'u.item'? (y/n)를 출력하고 사용자 입력을 받아 이를 check 에 저장한다. check 가 y 이면 앞에서와 마찬가지로 -F 옵션을 통해 u.item 을 |로 구분자로 데이터를 불러오고 |로 구분된 데이터 중 7 번째 숫자가 action 이므로 \$7 이 1 이면 action 영화이다 이때 movie id 와 title 을 출력해야 하므로 \$1 과 \$2 를 출력하고 10 개만 출력하기 위해 count 가 10 이 되면 exit 한다.

choice 가 3 이라면 Please enter the 'movie id' (1~1682):를 출력하고 사용자 입력을 받아 이를 movieid 에 저장한다. awk -v 를 통해 movieid 를 id 에 저장하고 u.data 를 불러온다. 이때 u.data 의 2 번째가 데이터가 movie id 이므로 이를 id 와 비교하여 같으면 total 에 더하고 count 를 증가시킨다. 데이터를 모두 확인하면 END 블록에서 total 을 count 로 나눈 결과를 5 자리로 출력한다.

choice 가 4 라면 Do you want to delete the 'IMDb URL' from 'u.item'? (y/n)를 출력하고 사용자 입력을 받아 이를 check 에 저장한다. check 가 y 이면 u.item 에서 불러온 데이터를 sed 명령어를 통해 |로 시작하고 |로 끝나며 http 로 시작하는 문자열의 그룹을 찾아서 빈문자열로 치환하고 그 중 10 개만 출력한다.

choice 가 5 라면 Do you want to get the data about users from 'u.user'? (y/n)를 출력하고 사용자 입력을 받아 이를 check 에 저장한다. check 가 y 이면 u.user 에서 불러온 데이터를 sed -e 옵션을 통해 F 는 female 로 M 은 male 로 치환한다. 이후 awk -F 를 통해 |로 구분자로 불러온 데이터에서 각각 user id age gender occupation 을 출력한다 이때 10 개만 출력하기 위해 count 가 10 이 되면 exit 한다.

choice 가 6 이라면 Do you want to Modify the format of 'release data' in 'u.item'? (y/n)를 출력하고 사용자 입력을 check 에 저장한다. check 가 y 이면 awk -F 를 이용해 u.item 의 데이터를 |로 구분자로 불러온다. 이때 BEGIN 블록에서 OFS=FS 를 통해 출력시 구분자도 |가 되도록 해줬다. 이후 split 을 통해 3 번째 항목인 날짜를 -로 구분자로 a 에 배열 형식으로 저장한다. 이제 if 문을 통해 month 를 나타내는 2 번째 항목을 원하는 형식으로 변환하여 m 에 저장한다..

이후 원래 데이터가 있던 3 번째 항목에 a[3](년도) m(월) a[1](날짜) 순으로 대입하고 if 문을 통해 1673 부터 1682 번째 라인까지만 출력한다.

choice 가 7 이라면 Please enter the 'user id' (1~943): 를 출력하고 사용자 입력값을 userid 에 받아온다. 이후 awk -v 를 통해 userid 를 id 로 받아오고 u.data 파일에서 첫번째 항목이

id와 일치하면 2 번째 항목인 movieid 를 숫자 순서대로 정렬하고 이를 movieids 에 저장한다. 이후 sed 명령어에서 반복문을 통해 movieids 를 한줄로 만든 후 %n 문자를 모두 |로 치환하여 출력한다. 이제 movieids 의 값을 배열로 변환하고 1 번째부터 10 번째 원소를 하나씩 id 에 할당한 후 awk -v movieid u.item 의 첫번째 항목과 일치하면 movieid 와 title 을 |로 구분하여 출력한다. 이때 이미 movieids 에서 정렬했으므로 정렬하는 과정은 필요없다.

choice 가 8 이라면 Do you want to get the average 'rating' of movies rated by users with 'age' between 20 and 29 and 'occupation' as 'programmer'?(y/n)를 출력하고 사용자 입력값을 check 가 y 라면 awk -F 로 u.user 파일에서 직업이 programmer 고 연령이 20 에서 29 사이인 user 의 user id 를 id 변수에 저장한다. BEGIN 블록으로 id 에 있는 값들을 ids 라는 배열에 공백으로 분리하여 저장한다. 이후 배열의 id 가 u.data 의 id 와 일치하면 sum 배열에 movie id 를 인덱스로 하여 ratings 를 더하고 같은 방식으로 count 도 1 더한다. 모든 데이터에 대하여 연산이 끝나면 소수점 5 자리까지 출력한다 이때 0 을 없애기 위해 .6g 를 사용하였다. 이후 이것들을 오름차순으로 정렬한다.