오픈소스 프로그래밍 과제 1 보고서

12181722 양진수

텍스트, 스크린샷, 디스플레이, 소프트웨어이(가) 표시된 사진

자동 생성된 설명

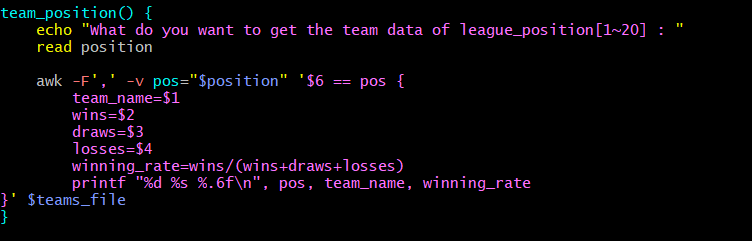
처음으로 $# 의 개수가 3개가 아니면, 제대로 실행이 되지 않도록 하고 사용법을 출력하도록 했다. 그 후 이 쉘 스크립트 내에서 사용될 teams.csv player.csv matches.csv 의 이름을 설정한다.

Show\_first 함수는 반복문 밖에서 출력하기 위한 함수이다

Menu 함수는 반복문안에서 7 이 입력되지 않았을 때 계속 출력될 메뉴를 표시하는 함수이다.

Hm7 함수는 과제의 첫번째 문제에 해당하는 함수이다. 사용자에게 데이터를 조회할지 묻고,

Y 를 입력하면 awk 문을 사용해 데이터를 출력한다. Player.csv 에서 첫번째 필드의 값이 손흥민 이라면, 그 행의 해당 정보 필드들을 알맞게 출력한다.

team\_position 함수는 사용자에게 몇등 팀의 데이터를 조회할지 묻고, 해당 등수를 가진 팀을 teams.csv 에서 조회해서 나타낸다. 승률을 데이터의 필드들을 가지고 계산하여 나타냈고, 소수점 아래 6글자까지 나타내도록 하였다.

텍스트, 스크린샷, 폰트이(가) 표시된 사진

자동 생성된 설명

Top3\_attendance 함수는 관중 수가 가장 많은 3개의 경기를 출력한다

필드 구분은 쉼표로 하고(-F’,’) 첫번째 행을 제외하고 처리했다(NR) sort 뒤의 옵션은 관중 수를 기준으로 데이터를 내림차순으로 정렬하는 것이다.

그리고 head -3 으로 위에서 3개의 행만 추출한다

과제 설명에 나온 예와 같은 형식으로 출력값을 가공하려 했으나, 숫자 외의 값이 지워지거나, 정렬이 올바르게 되지 않거나 하여 실패했다. 옵션을 제대로 다루지 못한 것으로 생각된다. 그렇기에 행 전체를 출력하는 방식으로 일단 작성하였다.

텍스트, 스크린샷, 소프트웨어이(가) 표시된 사진

자동 생성된 설명

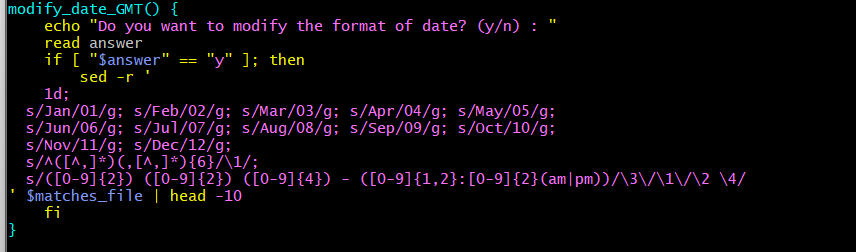
Position\_and\_scorer 함수는 앞의 함수들과 같이 사용자에게 데이터를 조회할지 물어보고 y가 입력으로 주어진다면 데이터 처리를 시작한다

먼저 sorted\_teams 라는 파일을 임시로 생성한다. 이는 문제에서 요구하는대로 순서 순위대로 정렬하여 정보를 나타내는데 어려움을 겪어 만들었다. 이 파일에서 순위 순서대로 정렬 한 후, 그 정보를 반복문을 돌며 출력할 것이다.

그 후 sorted teams에 순위대로 정렬한 팀 순위, 팀 이름을 가지고 반복문을 돈다. IFS 를 이용해 두 값을 분리하여 인식하고, 그 값을 먼저 echo 해준다.

그 후 awk 문을 이용해, 해당 팀 에서 가장 많은 득점을 한 선수를 찾고, 그 선수의 이름과 골 수 를 출력한다.

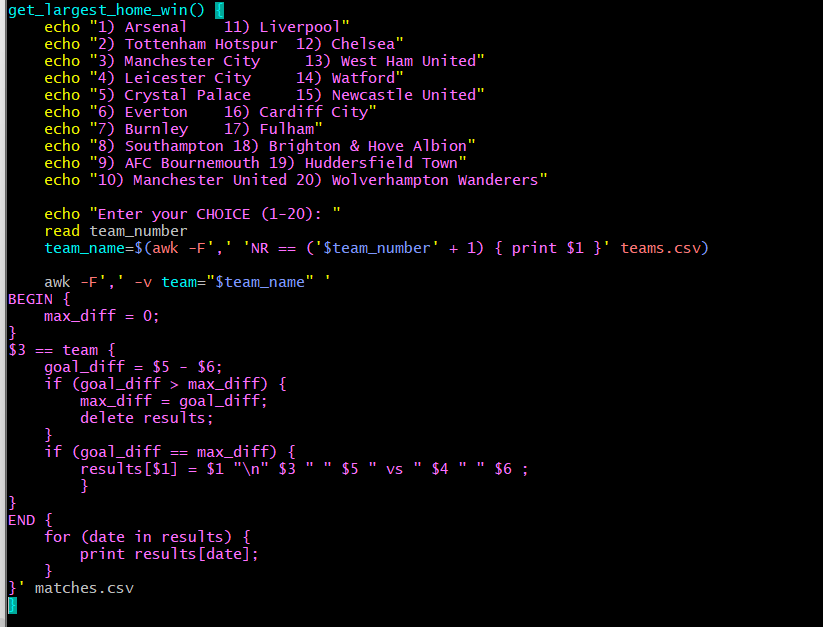
해당 과정이 끝나면, sorted\_teams 파일을 삭제한다.



이 함수는 날짜 형식을 matches.csv 에서 가져와 과제의 요구사항대로 바꿔서 출력한다. 먼저 사용자의 입력에 y 가 들어온다면 함수의 그 다음 부분을 실행한다.

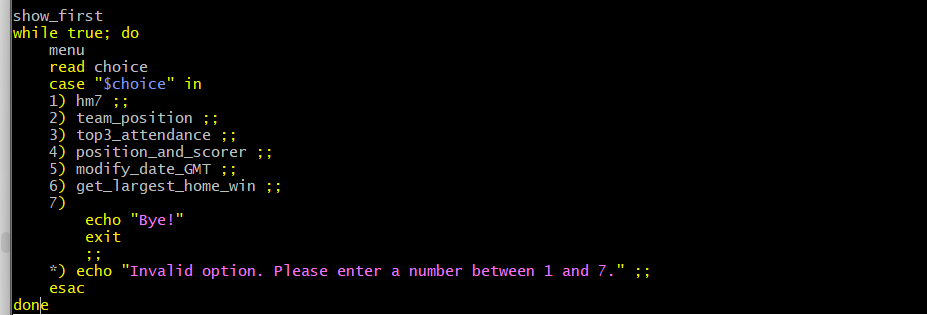
Sed 명령을 통해 작성하였으며. 1d 를 통해 첫번째 행을 삭제하고, 영문으로 적힌 월(month)에 관한 정보를 숫자로 바꾼다. 그 후 정규표현식을 이용해 각 행의 첫번째 필드만 추출 한 뒤, 형식을 변경해 dd mm yyyy – time 식으로 표현되어있던 문자를 yyyy/mm/dd time 모양으로 변경한다

그 후 head -10 옵션으로 10개만 추출해 출력한다.



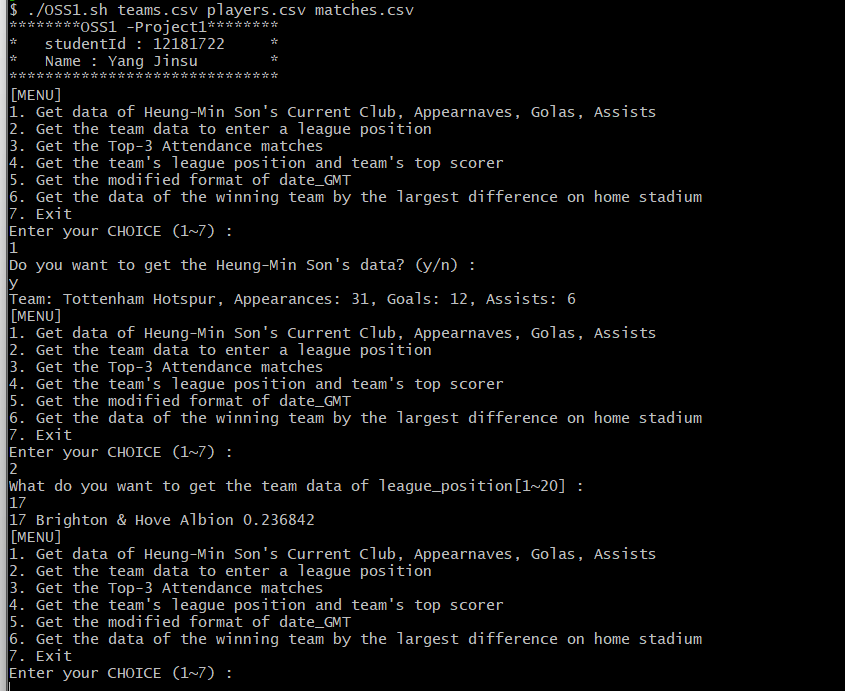
이 함수는 홈에서 가장 큰 점수차로 승리한 경기를 찾아 보여준다. 사용자에게 어떤 팀을 선택할지 묻고, 해당하는 팀을 teams.csv 에서 찾아 team\_name 에 저장한다.

그 후 awk 문 안에서 max\_diff 를 초기화 하고, matches 의 세번째 필드, 즉 홈 팀이 선택한 팀인 row를 찾는다. 그리고 거기서 원정팀과의 점수 차이를 goal\_diff 에 저장하고, 원래의 max\_diff 보다 더 큰 값이라면 max\_diff 를 바꿔준다. 그리고 max\_diff 와 goal\_diff 가 같다면 results라는 배열에 해당 경기정보를 추가해준다. 이렇게 matches.csv 의 파일을 다 돌고 나면 result 에 남아있는 값이 홈에서 최대 점수차로 승리한 경기일것이다. 따라서 result 안의 element 들을 출력한다.



위 함수들을 가지고 위와 같이 반복문을 돌며 실행한다.

실행예시



텍스트, 스크린샷, 폰트이(가) 표시된 사진

자동 생성된 설명

