**OSS Project1**

Student ID: 12214168

Name: YooHyun Kim

1. **line 1 - 11**

텍스트, 스크린샷, 폰트, 번호이(가) 표시된 사진

자동 생성된 설명

터미널에서 sh파일 실행 시 세 개의 파일을 입력 받고 있다. line 3에서 $#은 입력된 파일의 개수를 의미한다. 입력된 파일이 3개가 아니라면 세 개의 파일을 입력하도록 안내하는 메시지를 출력하고 프로그램을 종료한다.



세 개의 파일이 입력되지 않아서 에러 메시지가 출력되었다.

1. **line 12 - 65**

함수 f1()부터 f7()까지가 서술되어 있다. 뒤에 있는 코드를 먼저 보고 각 함수별로 살펴보겠다.

1. **line 66 – 70**

텍스트, 스크린샷, 폰트, 라인이(가) 표시된 사진

자동 생성된 설명

파일 세 개를 정상적으로 입력다면 출력되는 메시지다. 학번과 이름이 출력된다.

텍스트, 스크린샷, 폰트이(가) 표시된 사진

자동 생성된 설명

세 개의 파일을 입력 받았고 학번과 이름이 출력되었다.

1. **line 71 – 93**

텍스트, 스크린샷, 폰트, 문서이(가) 표시된 사진

자동 생성된 설명

1번부터 7번까지의 메뉴를 출력하고 read -p로 사용자의 메뉴 선택을 입력 받는다. case문을 통해 1번부터 7번까지 각각 함수 f1, … ,f7을 실행한다. 그리고 while문을 통해 사용자가 exit메뉴인 7번을 고르지 않는다면 위 코드가 반복되도록 하였다.

텍스트, 스크린샷, 폰트이(가) 표시된 사진

자동 생성된 설명

1번부터 7번까지 메뉴에 대한 설명이 출력됐고 사용자의 입력을 받고 있다.

1. **line 12 – 17**

텍스트, 폰트, 라인, 스크린샷이(가) 표시된 사진

자동 생성된 설명

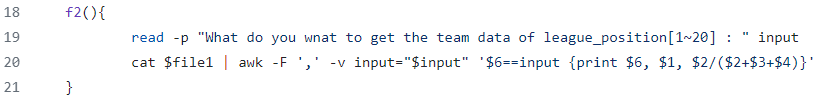
1번 메뉴를 선택했을 때 실행되는 함수다. 손흥민 선수의 데이터를 얻을 것인지 사용자에게 물어보고 y를 입력 받으면 line 15를 통해 손흥민 선수의 소속팀과 출전 경기 수, 골 수, 도움 수를 출력한다. file2인 players.csv에서 정보를 가져왔고 awk를 통해 구분자를 콤마(,)로 했다. 선수 이름이 있는 1열에서 손흥민 선수와 이름이 일치하는 행을 찾아 printf로 정보를 출력한다.

텍스트, 폰트, 스크린샷이(가) 표시된 사진

자동 생성된 설명

f1()의 실행 결과

1. **line 18 – 21**



2번 메뉴를 선택했을 때 실행되는 함수다. file1(teams.csv)으로부터 리그 순위에 대한 정보를 받아온다. 사용자가 1부터 20사이의 리그 순위를 입력하면 그 순위에 해당하는 팀과 그 팀의 승률을 출력한다. 승률은 ( wins / (wins+draws+losses) )로 계산했다. awk문을 보면 -F를 통해 구분자를 콤마로 했고 쉘 변수 $input을 사용하기 위해 -v 옵션을 사용했다. teams.csv의 6열($6)은 리그 순위이고 6열이 input과 같은 행에 대해 print문으로 순위($6), 팀명($1), 승률을 출력했다.



17위 브라이턴앤호브앨비언의 순위, 팀명, 승률이 출력됐다.

1. **line 22 – 28**

텍스트, 폰트, 스크린샷, 대수학이(가) 표시된 사진

자동 생성된 설명

3번 메뉴를 선택했을 때 실행되는 함수다. 최다 관중 수 경기 top3를 출력한다. file3(matches.csv)로부터 정보를 받아오고 **sort -r -t ‘,’ -n -k 2** 으로 구분자를 콤마로 하여 2열 기준 내림차순 정렬했다. 그리고 awk문을 통해 문제에서 요구한 형식으로 출력해준다. 관중 수 top3 경기를 출력하기 위해 공백까지 고려하여 마지막에 **head -n 9** 를 작성하였다.

텍스트, 스크린샷, 폰트이(가) 표시된 사진

자동 생성된 설명

top3 attendance 데이터가 출력됐다.

1. **line 29 – 40**

텍스트, 스크린샷, 폰트, 대수학이(가) 표시된 사진

자동 생성된 설명

4번 메뉴를 선택했을 때 실행되는 함수다. 사용자에게 y를 입력받으면 각 팀의 순위와 팀별 최다 득점 선수 1명을 출력한다. line 33 – 38의 for문을 i=1일 때부터 20일 때까지 반복한다. line 34의 awk문을 보면 -F로 구분자를 콤마로 하였고 변수 i를 awk문에서 사용하기 위해 -v를 사용했다. file1(teams.csv)의 6열(순위)가 col과 같다면 순위($6)와 팀명($1)을 출력한다. line 35는 쉘 변수 $team을 line 36에서 사용하기 위해 지정해줬다. $team에는 순위($i)에 해당되는 팀명이 file1(teams.csv)으로부터 입력된다. line2에서는 file2(players.csv)에서 해당 팀의 선수의 이름과 골 수를 불러온다. 여기서 {print $1, $7}이 아니라 {print $7, $1, $7}으로 해준 이유는 골 수로 내림차순 정렬을 하기 위해서이다. 선수마다 이름에 있는 공백 수가 달라 {print $1, $7}으로 한다면 awk문까지 실행했을 때 골 수가 몇 번째 열에 오는지 알 수 없다. 그래서 일단 {print $7, $1, $7}으로 제일 앞에 골 수를 출력하고 **sort -r -t ‘,’ -n -k 1** 으로 골 수 기준 내림차순 정렬했다. 그 다음에 **head -n 1**을 통해 해당 팀의 최다 득점자만 출력되게 하였다. 그리고 **awk '{$1=""; sub(/^ \*/,"");print}'**를 통해 제일 앞의 열(골 수)을 지워줬다.

텍스트, 스크린샷, 폰트이(가) 표시된 사진

자동 생성된 설명

1위부터 20위까지의 팀명과 최다 득점 선수가 출력된다. 사진에는 1위부터 6위까지만 첨부하였다.

1. **line 41 – 50**

텍스트, 스크린샷, 폰트, 대수학이(가) 표시된 사진

자동 생성된 설명

5번 메뉴를 선택했을 때 실행되는 함수다. 사용자로부터 y를 입력 받으면 시간의 포맷을 바꿔 출력한다. line44에서 NR!=1을 통해 1행은 제외해줬다. line 45에서 ‘s/’를 통해 영어로 출력된 월을 숫자로 변형시켰다. 그리고 line 46에서 월, 일, 년을 년, 월, 일의 순서로 바꿔줬다. line 47에서는 ‘ – ‘를 공백으로 바꿔줬다. 마지막으로 **head -n 10**을 통해 위에서부터 10줄만 출력했다.

텍스트, 스크린샷, 폰트이(가) 표시된 사진

자동 생성된 설명

match.csv의 상위 10개 매치의 시간의 포맷이 바뀌어 출력됐다.

1. **line 51 – 60**

텍스트, 스크린샷, 폰트이(가) 표시된 사진

자동 생성된 설명

6번 메뉴를 선택하면 실행되는 함수다. line 52 – 55의 코드는 20개의 팀을 file1(teams.csv)로부터 받아와 출력한다. line 56에서 read -p를 통해 사용자의 팀 입력을 받는다. line 57에서 awk문에 쓰기 위한 쉘 변수 $team을 생성했다. $team은 file1에서 팀명을 받아 저장한다. line 58의 maxi 변수에는 사용자가 고른 팀의 홈경기 중 가장 크게 승리한 경기의 점수차를 file3(matches.csv)에서 받아와 저장한다. -v를 통해 쉘 변수 $team을 사용한 것을 확인할 수 있다. (($5-$6))은 점수 차이다. $5는 홈팀의 점수, $6은 원정팀의 점수다. 그리고 파이프라인 뒤의 awk문에서 최대값을 찾아내 출력하는 것을 확인할 수 있다. line 59에서는 $team과 $maxi를 이용해 해당 팀의 홈에서의 가장 큰 승리 경기를 출력해준다.

텍스트, 스크린샷, 메뉴, 폰트이(가) 표시된 사진

자동 생성된 설명

크리스탈 팰리스의 홈에서 가장 큰 점수차(2점)의 승리 경기가 출력됐다.

1. **line 61 – 65**

텍스트, 스크린샷, 폰트, 도표이(가) 표시된 사진

자동 생성된 설명

7번 메뉴를 선택했을 때 실행되는 함수로 “Bye!”를 출력하며 프로그램을 종료시킨다.



f7()의 실행 결과