

## 1. 구현된 함수의 기능 명세

set\_table 함수 : strcpy(문자열 복사)함수를 이용해 구조체player, club, league의 column, data 파트에 각각 문자열을 넣어주고 구조체에 해당하는 파일을 쓰기모드(w)로 오픈해 strcpy로 구조체의 column, data파트에 넣어주었던 문자열을 불러와 txt로 저장.

get\_all\_element : set\_table에서 저장한 txt파일에서 fgets를 이용해 문자열을 가져와 구조체에 삽입해줌.

print\_all\_table : set\_table과 get\_all\_element 함수를 거쳐 구조체에 저장된 데이터(문자열)을 전부 출력해주는 함수.

find\_element\_by\_key : 1차적으로 table\_Player, table\_Club, table\_League 중 하나의 테이블을 선택해 해당 테이블의 내용을 출력하고 그 테이블에서 사용자가 찾으려는 key값을 입력하여 해당 데이터를 출력하는 함수.

## 2. 코드설명

```
void set_table(Player* table_Player, Club* table_Club, League* table_League) {
    int fp_in_num = 0;
    FILE* fp1 = NULL, * fp2 = NULL, * fp3 = NULL;

    strcpy(table_Player->column, "key,p_name,club_key\n");
    ""

    strcpy(table_Player->data[0], "1,손흥민,1\n");
    ""

    strcpy(table_Club->data[0], "1,토트넘,1\n");
    ""

    strcpy(table_League->data[0], "1,1,2,1:2\n");
    ""

    fp1 = fopen("Player.txt", "w");
    ""

    fputs(table_Player->column, fp1);
```

```

    ""

    for (fp_in_num = 0; fp_in_num < 3; fp_in_num++)
        fputs(table_Player->data[fp_in_num], fp1);

    ""

    fclose(fp1);

    ""

```

set\_table 함수의 경우 player.txt , club.txt , league.txt 파일을 생성하기위해 파일포인터 fp1, fp2, fp3를 생성해주고, 우선 strcpy함수를 이용해 각 테이블에 column파트와 data파트에 문자열을 복사해주고 fp1~fp3를 각각 player, club, league.txt "w" 쓰기모드로 열어주고 strcpy로 복사해준 문자열을 fputs를 이용해 파일에 문자열을 써주고 fclose를 이용해 파일을 닫아줌.

---

```

void get_all_element(Player* table_Player, Club* table_Club, League* table_League)
{
    int fp_in_num = 0;
    FILE* fp1 = NULL, * fp2 = NULL, * fp3 = NULL;
    char str[100];

    fp1 = fopen("Player.txt", "r");
    ""

    fgets(str, 100, fp1);
    ""

    for (fp_in_num = 0; fp_in_num < 3; fp_in_num++)
        fgets(table_Player->data[fp_in_num], 100, fp1);
    ""

    fclose(fp1);
    ""

```

get\_all\_element 함수의 경우 set\_table함수에서 fputs로 구조체에 문자열을 삽입해 주었던 것을 fgets로 문자열을 가져와 주는 역할을 수행.

---

```

void print_all_table(Player* table_Player, Club* table_Club, League* table_League) {
    int p, c, l;
    printf("-----Player_table-----\n");
    for (p = 0; table_Player != 0; p++)
    {
        if (p == 0)
            printf("%s", table_Player->column);
        else {
            if (table_Player->data[p - 1][0] == 0)
                break;
            printf("%s", table_Player->data[p - 1]);
        }
    }
    ""

```

print\_all\_table 함수의 경우 for문과 if-else문을 이용해 get\_all\_element 함수로 불러온 구조체 데

이터를 단순 출력해 주는 역할을 수행.

---

```
void find_element_by_key(Player* table_Player, Club* table_Club, League* table_League)
{
    int p, c, l;
    char str1[100] = { 0 };
    int num = 0;
    printf("어떤테이블을 사용하시겠습니까? (table_Player, table_Club, table_League) : ");
    gets(str1);
    printf("-----테이블 구성내용-----\n");

    {
        if (strcmp(str1, "table_Player") == 0)
        {
            for (p = 0; table_Player != 0; p++)
            {
                if (p == 0)
                    printf("%s", table_Player->column);
                else {
                    if (table_Player->data[p - 1][0] == 0)
                        break;
                    printf("%s", table_Player->data[p - 1]);
                }
            }
        }

        ....

        printf("\n연관된 자료를 찾을 key를 입력하십시오(0 = 종료):");
        scanf_s("%d", &num);
        {
            if (num == 0)
            {
                return 0;
            }

            else if (strcmp(str1, "table_Player") == 0 && num == 1)
            {
                printf("테이블 정보 : %s", table_Player->column);
                printf("선택된 데이터 : %s", table_Player->data[0]);
                printf("키 값으로 찾은 클럽정보 : %s", table_Club->data[0]);
            }
        }

        ....
    }
}
```

find\_element\_by\_key 함수의 경우 print\_all\_table에서 출력한 데이터중 1차적으로 player, club, league 테이블중 원하는 테이블을 입력받아 strcmp함수(string.h 헤더파일에 선언)로 문자열비교를 통해 if문을 거쳐 해당하는 테이블만 출력한 후, 2차적으로 그 테이블에서 사용자가 찾기를 희망하는 연관된 자료의 key값을 입력받아(0입력시 종료) 그 데이터를 출력함.

### 3. 결과화면에 대한 분석

<pre>어떤테이블을 사용하시겠습니까? (table_Player, table_Club, table_League) : table_Player -----테이블 구성내용----- key,p_name,club_key 1,손흥민,1 2,박지성,2 3,박주영,2</pre>	
Table_Player 입력	
<pre>연관된 자료를 찾을 key를 입력하십시오(0 = 종료):3 테이블 정보 : key,p_name,club_key 선택된 데이터 : 3,박주영,2 키 값으로 찾은 클럽정보 : 2,맨유,1</pre>	
Table_Player 중 3번key입력 -> 박주영에 대한 club key값 정보 출력	

<pre>어떤테이블을 사용하시겠습니까? (table_Player, table_Club, table_League) : table_Club -----테이블 구성내용----- key,c_name,League_key 1,토트넘,1 2,맨유,1 3,아스날,2 4,맨시티,2</pre>	
Table_club 입력	
<pre>연관된 자료를 찾을 key를 입력하십시오(0 = 종료) : 4 테이블 정보 : key,c_name,League_key 선택된 데이터 : 4,맨시티,2 키 값으로 찾은 경기정보 : 2,3,4,2:3</pre>	
Table_Club 중 4번key입력 -> 맨시티에 관한 league key값 정보 출력	

<pre>어떤테이블을 사용하시겠습니까? (table_Player, table_Club, table_League) : table_League -----테이블 구성내용----- key,club_key1,club_key2,score 1,1,2,1:2 2,3,4,2:3</pre>	
Table_league 입력	
<pre>연관된 자료를 찾을 key를 입력하십시오(0 = 종료) : 2 테이블 정보 : key,club_key1,club_key2,score 선택된 데이터 : 2,3,4,2:3 키 값으로 찾은 클럽정보 : 3,아스날,2 키 값으로 찾은 클럽정보 : 4,맨시티,2</pre>	
Table_league 중 2번key입력 -> league key값이 2인 아스날과 맨시티 클럽정보 출력	