

c프로그래밍 및 실습
인터넷 사이트 로그인 및 타 사이트 로그인
정보로 회원가입
프로젝트 제안서

제출일자:

제출자명:이세준

제출자학번:233869

1. 프로젝트 목표

1) 배경 및 필요성

일반적인 로그인 시스템을 구현하고 사이트를 연계해 회원가입 및 로그인 할 수 있는 기능을 추가해 계속 새로 생겨나는 사이트마다 회원가입 해야 하는 불편함을 줄일 수 있도록 한다. 이 뿐만 아니라 학교사이트의 경우에는 주차 예약시스템 등등의 여러 편의성 기능을 추가해 제작한다.

2) 프로젝트 목표

다른 사이트의 로그인 정보를 활용해 다른 사이트에 회원가입 및 로그인을 할 수 있는 프로그램을 제작

3) 차별점

기존 프로그램의 경우 사이트마다 새로 회원가입을 해야 하는 불편함이 존재하고 너무나 많은 로그인 정보들과 회원가입때 생성하는 아이디/비번들의 조건(대문자, 글자 개수, 자, 특수문자)에 로그인에 실패하는 경우가 많음. 이러한 문제를 해결하기 위해 다른 사이트의 로그인 정보를 활용해 회원가입 및 로그인을 할 수 있는 기능을 제작한다

2. 기능 계획

1) 기능 1

- 회원가입 기능 (닉네임, 아이디 및 비밀번호 생성)

(1) 세부 기능 1 (문자열 조건 판별 기능)

- 아이디 비번 생성에 조건을 만들어 입력한 문자열이 조건에 일치하는지 판별

(2) 세부 기능 2 (문자열 중복 확인 기능)

- 입력한 문자열이 다른 유저의 정보와 중복되는지 확인

(3) 세부 기능 3 (타사이트 로그인 정보로 회원가입)

- 타사이트의 정보로 회원가입 기능 추가

2) 기능 2

- 로그인/아웃기능

(1) 세부 기능 1 (배열 내 정보 존재 확인 기능)

- 문자열 배열에 입력한 정보와 일치하는 정보가 있는지 확인하는 기능

(2) 세부 기능 2 (로그인 유지 기능)

- 로그인하였을 때 사이트를 종료해도 로그인을 유지할지 선택하는 기능

(3) 세부 기능 3 (타사이트 로그인 정보로 로그인 기능)

- 타사이트 정보로 로그인 기능 추가

3. 진척사항

1) 기능 구현

(1) 인터넷 사이트 출력

- main 함수에서 선언된 website_name 문자열 배열과 커서의 위치를 나타내는 y값을 입력받아 사이트 목록을 출력함.
- 직접 인터넷 사이트에 로그인을 할 수 없기 때문에 콘솔창에서 로그인이 되는 것으로 가정함.
- 반복문, 함수(문자열 배열을 매개변수로 사용)

```
void website_print(char website_name[][MAX_CHAR], int y) {
    for (int i = 0; i < MAX_WEBSITE; i++) {
        if (y == i) {
            printf("> ");
        } else {
            printf("  ");
        }
        printf("(%d) %s\n", i + 1, website_name[i]);
    }
}
```

(2) 키보드 입력(위/아래 및 엔터키)

- main 함수에서 선언된 커서의 위치값(cursor_position_y)과 목록의 최대값(max_scroll_num), 그리고 엔터키입력시 변경되는 select 변수를 전달받음
- 커서의 위치를 나타내는 변수를 포인터로 전달하고 함수가 실행되고 있을 때 위/아래 방향키를 입력받을 시 포인터로 넘겨받은 변수의 값을 증감 시키고 엔터키를 입력할 시 -1로 초기화된 select 변수를 변경해 목록을 선택하게 됨
- 포인터, 함수 사용

```
void keyboard_control(int* cursor_position_y, int max_scroll_num, int* select) {
    int key;
    key = _getch();
    if (key == 224) {
        key = _getch();
        switch (key) {
            case 72:
                if (*cursor_position_y > 0) {
                    *cursor_position_y -= 1;
                }
                break;
            case 80:
                if (*cursor_position_y < max_scroll_num - 1) {
                    *cursor_position_y += 1;
                }
                break;
        }
    } else if (key == 13) {
        *select = *cursor_position_y;
    }
}
```

(3)웹사이트 접속 기능

- main 함수에서 select 변수가 변경될 시 작동하는 함수
- log_in : 유저의 각 사이트 로그인 상태 및 로그인 유지 선택 상태를 나타내는 배열

website_name : 각 사이트 이름을 나타내는 배열

select : 유저가 엔터키를 사용해 선택한 사이트 번호

DB : 각 사이트에 회원가입한 유저의 데이터(이름, 아이디, 비밀번호)

user_Count : 각 사이트에 회원가입한 유저 수

- 목록에서 선택한 사이트를 출력하는 함수로 각 사이트가 포함하고 있는 목록을 배열로 가지고 있으며 또한 각 사이트가 포함하고 있는 목록이 다른 사이트라면 그 사이트로 이동이 가능함

- 배열, 다차원 배열, 조건문

(함수가 너무 길어 다 못 넣었습니다)

```
int website_connect(int log_in[][3], char website_name[][MAX_CHAR], int select,
    char DB[MAX_WEBSITE][MAX_USER][4][MAX_CHAR],
    int user_Count[]) {
    int choice = -1, scroll = 0;
    char website_collection[MAX_WEBSITE][MAX_WEBSITE_LIST][100] = {
        // index=0 은 그 배열에 포함된 빈 칸이 아닌 배열 갯수
        {"7", "메일", "NAVER", "JNU PORTAL", "GITHUB", "YOUTUBE", "PROGRAMMERS",
        "내 정보/로그인"},
        {"7", "네이버메일", "GOOGLE", "JNU PORTAL", "GITHUB", "YOUTUBE",
        "PROGRAMMERS", "내 정보/로그인"},
        {"3", "내학사행정", "e클래스", "내 정보/로그인", "", "", ""},
        {"2", "Repositories", "내 정보/로그인", "", "", "", ""},
        {"3", "쇼츠", "뮤직", "내 정보/로그인", "", "", ""},
        {"2", "코딩테스트", "내 정보/로그인", "", "", "", ""}};
    while (1) {
        system("cls");
        printf("===접속된 사이트 : %s===\n", website_name[select]);
        for (int i = 0; i < atoi(website_collection[select][0]) + 1; i++) {
            if (scroll == i) {
                printf("> ");
            } else {
                printf(" ");
            }
            if (i == atoi(website_collection[select][0])) {
                printf("(%d) 나가기\n", i + 1);
                break;
            } else if (i == atoi(website_collection[select][0]) - 1) {
                if (log_in[select][0] == 1) {
                    printf("(%d) 내 정보\n", i + 1);
                } else {
                    printf("(%d) 로그인\n", i + 1);
                }
            } else {
                printf("(%d) %s\n", i + 1, website_collection[select][i + 1]);
            }
        }
    }
}
```

문제가 검색되지 않음

(4)회원가입 기능

- log_in : 유저의 각 사이트 로그인 상태 및 로그인 유지 선택 상태를 나타내는 배열
- website_name : 각 사이트 이름을 나타내는 배열
- select : 유저가 엔터키를 사용해 선택한 사이트 번호
- DB : 각 사이트에 회원가입한 유저의 데이터(이름, 아이디, 비밀번호)
- user_Count : 각 사이트에 회원가입한 유저 수
- 사용자가 회원가입때 입력한 정보를 DB에 저장하는 함수
입력한 정보들을 제한에 따라 다시 입력하도록 함
- 문자열 배열, 함수 사용

```
int sign_up(int web_number, char DB[][MAX_USER][4][MAX_CHAR], int user_Count[],
            int log_in[][3], char website_name[][MAX_CHAR]) {
    char check_id[MAX_CHAR], check_pw[MAX_CHAR], check_name[MAX_CHAR];
    int checker=-1;
    while (1) {
        checker = -1;
        printf("=====%s=====\n", website_name[web_number]);
        printf("=====SIGNUP===== (회원가입 취소:stop)\n");
        printf("ID=");
        scanf_s("%s", check_id, (int)sizeof(check_id));
        if (!strcmp(check_id, "stop")) {
            return 0;
        }
        printf("=====\n");
        if (strlen(check_id) >= MAX_CHAR) {
            system("cls");
            printf("문자 최대 제한 초과(MAX=%d)", MAX_CHAR);
            continue;
        }
        for (int i = 0; i < user_Count[web_number]; i++) {
            if (!strcmp(DB[web_number][i][1], check_id)) {
                system("cls");
                printf("중복된 아이디입니다\n.");
                checker = 1;
                break;
            }
        }
        if (checker == 1)
            continue;
        else
            break;
        system("cls");
    }
    char ch = getchar();
    while (1) {
```

(5) 로그인 기능

- log_in : 유저의 각 사이트 로그인 상태 및 로그인 유지 선택 상태를 나타내는 배열

website_name : 각 사이트 이름을 나타내는 배열

select : 유저가 엔터키를 사용해 선택한 사이트 번호

DB : 각 사이트에 회원가입한 유저의 데이터(이름, 아이디, 비밀번호)

user_Count : 각 사이트에 회원가입한 유저 수

- 사용자가 입력한 정보가 DB 배열에 존재하는지 확인하는 함수
- 문자열, 배열, 함수 사용

```
int LOGIN(int web_number, char DB[][MAX_USER][4][MAX_CHAR], int user_Count[],
          int log_in[][3], char website_name[][MAX_CHAR]) {
    char login_id[MAX_CHAR], login_pw[MAX_CHAR];
    char ch, response[MAX_CHAR];
    int check = -1;
    while (1) {
        system("cls");
        printf("=====%s=====\n", website_name[web_number]);
        printf("=====LOGIN=====\n");
        printf("ID=");
        scanf_s("%s", login_id, (int)sizeof(login_id));
        printf("=====\n");
        if (strlen(login_id) >= MAX_CHAR) {
            printf("문자 최대 제한 초과(MAX=14)");
        } else {
            break;
        }
    }
    ch = getchar();
    while (1) {
        system("cls");
        printf("=====%s=====\n", website_name[web_number]);
        printf("=====LOGIN=====\n");
        printf("PW=");
        scanf_s("%s", login_pw, (int)sizeof(login_pw));
        printf("=====\n");
        if (strlen(login_pw) >= MAX_CHAR) {
            printf("문자 최대 제한 초과(MAX=14)");
        } else {
            break;
        }
    }
    system("cls");
    for (int i = 0; i < *user_Count; i++) {
        if ((strcmp(DB[web_number][i][1], login_id)) == 0) {
            if ((strcmp(DB[web_number][i][2], login_pw)) == 0) {
                log_in[web_number][0] = 1;
                log_in[web_number][2] = i;
                check = 1;
                printf("=====\n");
                printf("===상시 로그인(예/아니요)====\n");
                scanf_s("%s", response, (int)sizeof(response));
                if (strcmp(response, "예") == 0) {
                    log_in[web_number][1] = 1;
                }
            }
        }
    }
}
```

2) 테스트 결과

(1) 인터넷 사이트 출력 기능

- 인터넷 사이트 목록을 출력하는 기능(숫자 0은 키보드 입력 횟수를 표시해둔 것)

```
0=====접속할 사이트 선택=====
> (1) GOOGLE
    (2) NAVER
    (3) JNU PORTAL
    (4) GITHUB
    (5) YOUTUBE
    (6) PROGRAMMERS
```

(2) 키보드 입력

- 키보드 입력을 이용해 목록에서 이동하는것

```
4=====접속할 사이트 선택=====
    (1) GOOGLE
    (2) NAVER
> (3) JNU PORTAL
    (4) GITHUB
    (5) YOUTUBE
    (6) PROGRAMMERS
```

(3) 웹사이트 접속

- 사용자가 선택한 사이트에 접속하는 기능


```
====접속된 사이트 : GOOGLE====  
> (1) 메일  
    (2) NAVER  
    (3) JNU PORTAL  
    (4) GITHUB  
    (5) YOUTUBE  
    (6) PROGRAMMERS  
    (7) 로그인  
    (8) 나가기  
-1 0
```

-접속된 사이트 목록에 다른 사이트가 존재하고 그 사이트를 선택하면 그 사이트로 이동

```
====접속된 사이트 : NAVER====  
> (1) 네이버 메일  
    (2) GOOGLE  
    (3) JNU PORTAL  
    (4) GITHUB  
    (5) YOUTUBE  
    (6) PROGRAMMERS  
    (7) 로그인  
    (8) 나가기  
-1 1
```

현재 이미지 상태: GOOGLE ->NAVER 이동

(4) 회원가입 기능

- 사용자가 회원가입하기 위해 정보를 입력하는 기능 (각 사이트 로그인 -> 회원가입)

```
====접속된 사이트 : GOOGLE====
(1) 메일
(2) NAVER
(3) JNU PORTAL
(4) GITHUB
(5) YOUTUBE
(6) PROGRAMMERS
> (7) 로그인
(8) 나가기
-1 0
```

```
(1) 로그인
> (2) 회원가입
(3) 나가기
```

```
=====GOOGLE=====
=====SIGNUP===== (회원가입 취소:stop)
=ID=
```

```
====접속된 사이트 : GOOGLE====
(1) 메일
(2) NAVER
(3) JNU PORTAL
(4) GITHUB
(5) YOUTUBE
(6) PROGRAMMERS
> (7) 로그인
(8) 나가기
abcde abcde abcde-1 0
```

(목록 아래의 알파벳은 테스트를 위해 사용자 이름,아이디,비번을 표시해둔 것)

(5) 로그인 기능

- 사용자가 로그인하기 위해 정보를 입력하는 기능 (각 사이트 로그인 -> 로그인)

```
> (1) 로그인
    (2) 회원가입
    (3) 나가기
```

```
=====GOOGLE=====
=====LOGIN=====
=ID=
```

(회원가입에서 입력한 데이터 입력)

```
=====
====상시 로그인(예/아니요)=====
```

```
====접속된 사이트 : GOOGLE====
(1) 메일
(2) NAVER
(3) JNU PORTAL
(4) GITHUB
(5) YOUTUBE
(6) PROGRAMMERS
> (7) 내 정보
    (8) 나가기
```

상시 로그인에서 "예"를 입력 시 사이트에서 나가도 로그인 상태 저장

```
====접속된 사이트 : GOOGLE====
(1) 메일
(2) NAVER
(3) JNU PORTAL
(4) GITHUB
(5) YOUTUBE
(6) PROGRAMMERS
(7) 내 정보
> (8) 나가기
```

```
13=====접속할 사이트 선택=====
> (1) GOOGLE
    (2) NAVER
    (3) JNU PORTAL
    (4) GITHUB
    (5) YOUTUBE
    (6) PROGRAMMERS
```

```
=====접속된 사이트 : GOOGLE=====
> (1) 메일
    (2) NAVER
    (3) JNU PORTAL
    (4) GITHUB
    (5) YOUTUBE
    (6) PROGRAMMERS
    (7) 내 정보
    (8) 나가기
```

4. 계획 대비 변경 사항

1) 키보드 입력 방식 변경

- 사용자가 목록에 표시된 사이트 중 하나를 입력해 접속하는 방식
- 사용자가 키보드 방향키를 이용해 사이트를 선택하는 방식으로 변경
- 기존 방식으로 하면 사용자가 입력해야 하는 정보가 너무 많아져 방향키 입력 방식으로 변경

2) 기능 없는 사이트에 기능 추가

- 배경에 작성되어 있는 학교 사이트와 주차 예약 시스템도 구현한다
- 기능이 없는 각 사이트에 어울리는 기능을 추가한다
- 현재 기능이 없는 메일, 네이버 메일 내학사행정 등등 여러 목록에 어울리는 기능을 추가할 계획입니다.

5. 프로젝트 일정

업무		11/3	11/10	11/17
제안서 작성		완료			
기능1	세부기능1	완료	완료		
	세부기능2	완료	완료		
	세부기능3		-----	진행중	-----→
기능2	세부기능1	완료	완료		
	세부기능2	완료	완료		
	세부기능3		-----	진행중	-----→