

오픈소스소프트웨어 10주차 실습과제

1. 내 Github Repository에 README.md 파일 생성하기

a. README.md 파일 소스

project1

오픈소스소프트웨어 프로젝트 1 (범죄자 관리 시스템)

프로젝트 사용법

1. 다음 프로젝트를 Local Repository에 clone하여 makefile을 이용하여 실행파일을 만드시오.

2. makefile 사용 예시

- * make main > 일반 실행 파일 생성

- * make main_debug > 디버그 모드 실행 파일 생성 (어떻게 실행 파일이 돌아가는지 자세시 알기 위해)

- * make clean > make를 통한 생성 파일 모두 삭제

3. ./main이나 생성된 실행파일 직접 접근하여 프로그램 실행

4. 프로그램 사용법에 대해서는 다음 설명을 읽으시오.

프로그램 기능

* Create

- * 범죄자 레코드 생성 (이름, 나이, 거주지역, 성별, 범죄분류)

* Read

- * all

- * 모드 레코드 정보 출력

- * name

- * 이름 별로 검색하기

- * age

- * 나이대 별로 검색하기

- * sex

- * 성 별로 검색하기

- * city

- * 거주지역 별로 검색하기

- * category

- * 범죄 분류 별로 검색하기

- * save file

- * 현재 레코드 정보에 따라 Criminal_List파일 저장

- * Quit

- * Read 메뉴 종료

* Update

- * 범죄자 레코드 수정 & 범죄 항목 추가

* Delete

- * 범죄자 레코드 삭제

* Read_file

- * Criminal_list 파일과 Criminal_Wicked 파일 읽어오기

* Read_Report

- * Criminal_Report 파일 읽어오기

* Save_Wicked

- * 현재 레코드 정보를 통하여 Criminal_Wicked 파일 수정하여 저장

* Optimize_record

- * 비어 있는 레코드들 앞으로 당겨주기

* Sort_Record

- * 이름 순서대로 레코드 정렬하기

* Create_user

- * 프로그램 사용자 추가 (아이디, 비밀번호)

* Quit

- * 프로그램 종료

b. 화면캡처

48 lines (48 sloc) | 1.68 KB

RawBlameHistory

project1

오픈소스소프트웨어 프로젝트 1 (범죄자 관리 시스템)

프로젝트 사용법

1. 다음 프로젝트를 Local Repository에 clone하여 makefile을 이용하여 실행파일을 만드시오.

2. makefile 사용 예시

- make main > 일반 실행 파일 생성
- make main_debug > 디버그 모드 실행 파일 생성 (어떻게 실행 파일이 돌아가는지 자세시 알기 위해)
- make clean > make를 통한 생성 파일 모두 삭제

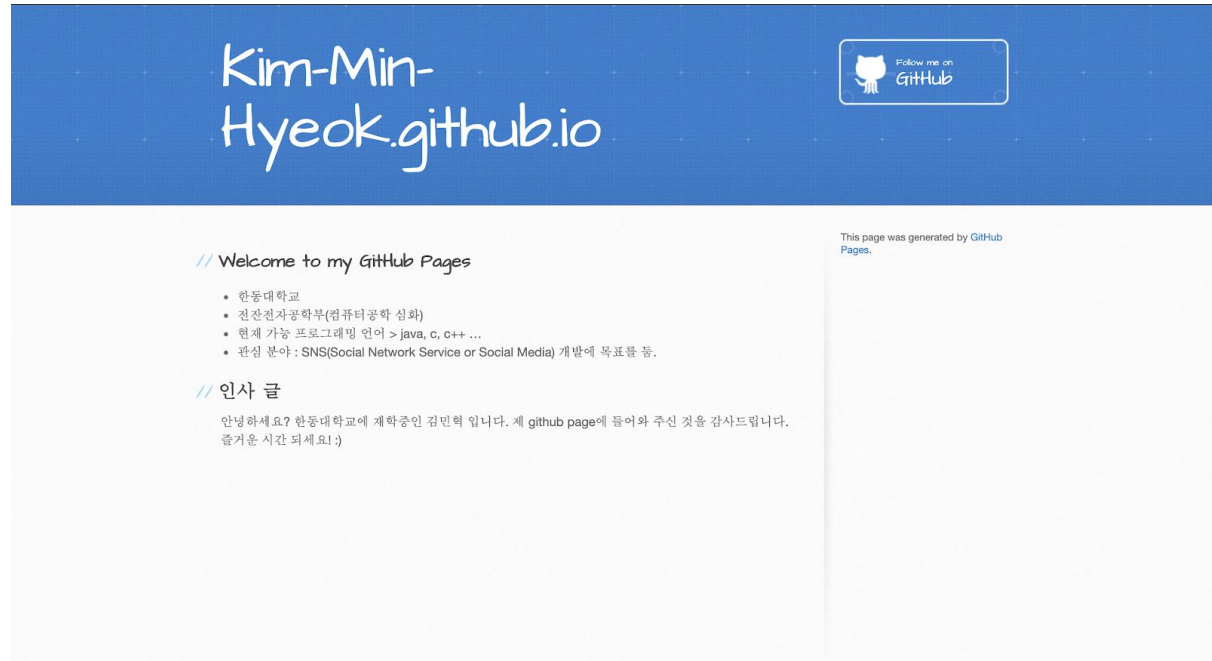
3. ./main이나 생성된 실행파일 직접 접근하여 프로그램 실행

4. 프로그램 사용법에 대해서는 다음 설명을 읽으시오.

프로그램 기능

- Create
 - 범죄자 레코드 생성 (이름, 나이, 거주지역, 성별, 범죄분류)
- Read
 - all
 - 모드 레코드 정보 출력
 - name
 - 이름 별로 검색하기
 - age
 - 나이대 별로 검색하기
 - sex
 - 성 별로 검색하기
 - city
 - 거주지역 별로 검색하기
 - category
 - 범죄 분류 별로 검색하기
 - save file
 - 현재 레코드 정보에 따라 Criminal_List파일 저장
 - Quit
 - Read 메뉴 종료
- Update
 - 범죄자 레코드 수정 & 범죄 항목 추가
- Delete
 - 범죄자 레코드 삭제
- Read_file
 - Criminal_list 파일과 Criminal_Wicked 파일 읽어오기
- Read_Report
 - Criminal_Report 파일 읽어오기
- Save_Wicked
 - 현재 레코드 정보를 통하여 Criminal_Wicked 파일 수정하여 저장
- Optimize_record
 - 비어 있는 레코드들 앞으로 당겨주기
- Sort_Record
 - 이름 순서대로 레코드 정렬하기
- Create_user
 - 프로그램 사용자 추가 (아이디, 비밀번호)
- Quit
 - 프로그램 종료

2. 내 Github 계정에 github page 생성하기
 - a. github page 주소 : Kim-Min-Hyeok.github.io
 - b. 화면 캡처



3. Clone & PR 을 보냈던 팀원의 repository에 대하여 팀원이 수정하거나 추가한 내용을 내게 어떻게 반영해야 하는가?

> 먼저 현재 remote되고 있는 원격 repository는 origin이라는 별칭의 나의 원격 repository이다. 그러나 팀원의 원격 repository의 변경 내용을 내 local repository에 업데이트 하려면 팀원의 원격 repository를 remote하여 pull하여야 한다.

- a. 팀원의 repository를 remote 하기

```
[s21900102@peace:~/oss2020/Pulltest$ git remote add team https://github.com/KimKangHoon4201/OSS_2020_Project
```

다음과 같이 팀원의 repository를 team이라는 별칭으로 add 하였다.

```
[s21900102@peace:~/oss2020/Pulltest$ git remote
origin
team
```

현재 remote 되고 있는 원격 repository 별칭

- b. pull 하기

```
[s21900102@peace:~/oss2020/Pulltest$ git pull team master
remote: Enumerating objects: 12, done.
remote: Counting objects: 100% (12/12), done.
remote: Compressing objects: 100% (8/8), done.
remote: Total 12 (delta 4), reused 12 (delta 4), pack-reused 0
Unpacking objects: 100% (12/12), done.
From https://github.com/KimKangHoon4201/OSS_2020_Project
* branch          master      -> FETCH_HEAD
* [new branch]     master      -> team/master
```

다음과 같이 team의 별칭을 가진 팀원의 원격 repository에서 나의 local repository로 pull 하였다. (fetch + merge)

```
[s21900102@peace:~/oss2020/Pulltest$ ls  
hospital2.c hospital.c hospital.h main.c Makefile patient_info.txt
```

다음과 같이 비어있던 디렉토리에 pull이 된 것을 알 수 있다.

c. remote 제거

```
[s21900102@peace:~/oss2020/Pulltest$ git remote remove team  
[s21900102@peace:~/oss2020/Pulltest$ git remote  
origin
```

다음과 같이 git remote remove를 통하여 team의 별칭을 가진 원격 repository를 삭제하여 주었다.

- d. git push origin master을 이용하여 나의 원격 repository에 최신 버전을 push한다.
- e. 팀원의 원격 repository로 나의 원격 repository의 내용을 Pull Request한다.