

Programming Assignment #2

Mini-SNS

Due Date: 2020년 4월 24일 금요일 23:59

담당조교: 장원종 (wonjong@postech.ac.kr)

1. Objective

본 과제에서는 상용되고 있는 Social Network Service의 축소판인 Mini-SNS를 구현해보고, 이를 통해 Class 개념과 Design 방법을 익힌다. SNS의 기본 기능인 회원 가입 및 탈퇴, 회원 검색 및 친구 추가, 그리고 게시글과 댓글 작성 기능 등을 구현해본다.

2. 과제 설명

2.1 개요

시스템 상에서 사용되는 텍스트 기반 메뉴 형태는 다음과 같다.

```
=====
Menu - Main
1. Sign up
2. Sign in
3. Load command
4. Program exit
=====
```

Figure 1. 텍스트 기반 메뉴 형태 (메인 메뉴 예시)

본 시스템의 사용자는 표시된 메뉴 중 하나의 메뉴를 숫자 입력을 통해 선택하여 선택된 기능을 수행하게 된다. 시스템 상에는 크게 회원 가입 기능, 친구 관리 기능, 그리고 피드 관리 기능이 있다.

1) 회원 관리 메뉴 (메인 메뉴)

프로그램을 시작하면 제일 처음 표시되는 회원 관리 메뉴에는 1.회원 가입, 2.로그인, 3.명령어 불러오기, 4.프로그램 종료가 있다.

회원 가입은 ID / password / name / birthday 의 정보를 받아서 진행되며, ID의 경우 DB 상에 중복되는 ID가 있는 지 체크하여 중복되는 경우 회원 가입이 불가능하여야 한다. 로그인에 입력되는 ID와 password를 시스템에 저장된 정보와 일치하는 지 확인하는 verification 과정을 거쳐서 일치하는 경우에만 다음 메뉴로 이동하게 한다. 명령어 불러오는 채점을 위한 기능으로 command.txt를 불러와 해당 명령 순서대로 작동하게 한다. 프로그램 종료는 수동으로 프로그램을 종료할 수 있게 하는 메뉴이다.

2) 회원 메뉴

로그인에 성공하고 나면 상단에 회원 정보가 표시되며, 아래에는 SNS와 관련된 기능들이 나타난다. 관련 기능에는 크게 1.친구 관리와 2.피드 관리가 있다. 그 외에도 3.로그아웃 기능과 4.계정 삭제 기능이 있다. 로그아웃을 하게 되면 회원 정보는 프로그램 상에 남은 채 1)의 회원 관리 메뉴로 돌아간다. 계정 삭제를 하게 되면 회원 정보와 작성한 게시글이 프로그램 상에서 삭제되며 마찬가지로 1)의 회원 관리 메뉴로 돌아가게 된다.

```
=====
Menu - My page
@saero2 - Park saeroy 2000.12.17
1. Friends
2. Feed
3. Sign out
4. Delete my account
=====
```

Figure 2. 회원 메뉴 예시

3) 친구 메뉴

친구 메뉴를 선택하는 경우 아래와 같은 네 가지 기능을 제공한다. 크게 친구를 추가하고 삭제하는 기능과 나의 친구 목록을 조회하는 기능으로 이루어져 있다.

```
=====
Menu - Friends
1. Add friends
2. Delete friends
3. My friends
4. Previous menu
=====
```

Figure 3. 친구 메뉴 예시

4) 피드 메뉴

피드 메뉴를 선택하는 경우 아래와 같은 네 가지 기능을 제공한다. 크게 1.모든 피드를 확인하는 기능과 2.게시글 작성 기능 그리고 3.나의 게시글을 모아서 보는 기능이 있다.

```
=====
Menu - Feed
1. All Feed
2. Post
3. My posting
4. Previous menu
=====
```

Figure 4. 피드 메뉴 예시

2.2 메뉴 기능 설명

- 해당 메뉴가 선택될 시, 현재 선택된 메뉴 명을 출력하고(ex. : 1. Sign up), 다음 예시와 같이 **4.2 로그파일 형식**에 명시된 순서대로 각각의 정보 입력 요청 문장을 출력하고, 정보를 입력 받아 필요한 기능을 수행한다.

```
=====
Menu - Main
1. Sign up
2. Sign in
3. Load command
4. Program exit
=====
Select Menu:1
Selected Menu: 1. Sign up
ID: saero2
Name: Park saeroy
Birthday: 2000/12/17
Password: itaewonclass
Registration Done!
```

Figure 5. 메뉴 선택 예시

- 메뉴 기능을 제대로 수행했을 경우, 화면상에 성공하였다는 메시지를 출력하며, 실패 시 실패하였다는 메시지를 출력하도록 한다.
- 또한 선택된 메뉴 번호 및 메뉴 선택 후 입력되는 정보는 command.txt 파일에 기록한다.

- 구현의 편의상 입력 받는 정보의 길이는 아래의 “**각 메뉴 기능 수행과 관련한 제한사항**”에 명시된 최대치보다 항상 작은 값을 입력 받는 것으로 생각한다. 즉, 회원 아이디 길이, 등록하는 회원의 수 등은 매 입력 시 마다 검사할 필요가 없다.

각 메뉴 기능 수행과 관련한 제한 사항
0) 숫자와 영문만 입력으로 받는 것으로 가정한다. 1) 저장 가능한 최대 회원 수: 50 2) 저장 가능한 최대 게시글 수: 100 3) 각 게시글 별 최대 길이: 500 단어 4) 각 댓글 별 최대 길이: 100 단어 5) ID, Password, Name 최대 길이: 32 글자 6) 커맨드 로그 파일 이름: "command.txt"

각각의 메뉴는 다음과 같은 기능을 수행한다.

각 메뉴별 수행 기능
<p>Menu - Main</p> <p>1. Sign up</p> 1) 입력 정보: - ID, name, birthday, password 2) 회원가입을 통해 나의 계정을 생성하는 기능 3) Valid case: - 입력된 ID 값이 기존에 있던 ID 와 일치하는 경우 적절한 안내 메시지 출력 4) Failure case: - 입력된 ID 값이 기존에 있던 ID 와 일치함에도 불구하고 중복되어 계정이 생성되는 경우
<p>2. Sign in</p> 1) 입력 정보: - ID, password 2) 가입된 ID 와 password 를 통해 로그인하는 기능 3) Valid case: - 입력된 ID 와 password 가 sign up 된 ID 와 password 와 일치하는 경우 - 일치하지 않는다면 적절한 안내 메세지 출력 4) Failure case - 가입되지 않은 아이디임에도 불구하고 로그인이 되는 경우.

3. Load command

1) 입력 정보:

- 없음.

2) 채점용 command.txt 파일을 불러와 프로그램을 자동으로 실행시키는 기능

3) Valid case:

- 실행파일과 같은 경로 상에 command.txt 파일이 존재하는 경우
- 같은 경로 상에 command.txt 가 존재하지 않는 경우 적절한 안내 메시지 출력

4) Failure case:

- command.txt 를 불러오는 데에 실패하는 경우
- 같은 경로 상에 command.txt 가 존재하지 않는데 안내 메시지를 출력하지 않은 경우

4. Program exit

1) 입력 정보:

- 없음

2) 기능

- 프로그램을 종료하는 경우

3) Valid case:

- 정상적으로 종료되는 경우

4) Failure case:

- 정상적으로 종료되지 않는 경우

Menu - My page

1. Friends

1) 입력 정보:

- 없음.

2) 기능

- Menu - friends 로 이동

2. Feed

1) 입력 정보:

- 없음.

2) 기능

- Menu - Feed 로 이동

3. Sign out

1) 입력 정보:

- 없음

2) 기능

- 로그아웃, Menu - Main 으로 이동

4. Delete my account

1) 입력 정보:

- 없음

2) 기능

- 계정 삭제, Menu - Main 으로 이동

3) Valid case:

- 해당 계정이 계정 목록에서 제대로 삭제되어 더 이상 다른 계정에서 친구 추가가 되지 않으며 친구 목록에 나타나지 않음.

4) Failure case:

- 해당 계정이 제대로 삭제되지 않아 해당 계정의 정보가 다른 기능들에서 노출되는 경우

Menu - Friends

1. Add friends

1) 입력 정보:

- ID

2) 기능

- ID 를 통해 해당 계정을 나의 친구 목록에 추가하는 기능

3) Valid case:

- 해당 ID 의 계정이 존재할 경우

- 이미 추가된 계정일 경우 추가하지 않고 적절한 안내 메시지를 출력

4) Failure case:

- 해당 ID 의 계정이 존재하지 않을 경우

2. Delete friends

1) 입력 정보:

- ID

2) 기능

- 내 친구 목록에서 입력된 ID 의 친구를 삭제

3) Valid case:

- 해당 ID 의 계정이 존재할 경우
- 존재하지 않을 경우 적절한 안내 메시지 출력

4) Failure case:

- 잘못된 계정이 삭제되는 경우

3. My friends

1) 입력 정보:

- 없음.

2) 기능

- 내 계정에 추가된 모든 친구 리스트를 출력하는 기능

3) Valid case:

- 내 계정에 추가된 모든 친구 리스트를 알파벳 순서대로 오름차순(숫자는 문자보다 앞) 출력
- 친구가 하나도 없을 경우 적절한 안내 메시지를 출력

4) Failure case:

- 내 계정에 추가된 모든 친구들을 출력하지 않을 경우

4. Previous menu

1) 입력 정보:

- 없음

2) 기능

- Menu- Member 로 돌아감.

Menu - Feed

1. All feed

1) 입력 정보:

- 게시물 번호

2) 기능

- 나의 친구들이 작성한 게시물을 시간순으로 출력하는 기능
- 게시물의 번호를 선택하여 추가 기능을 수행할 수 있도록 한다.
- 게시물의 번호는 0 부터 1 씩 증가하며 최근 게시물일 수록 낮은 번호를 받는다.
- -1 을 입력 시 Menu - Feed 로 돌아온다.

3) Valid case:

- 정상적으로 나의 친구로 등록된 계정의 게시물만 시간순으로 출력해주는 경우

- 최근에 작성한 게시물일 수록 번호가 0 에 가까우며 위에 출력하도록 한다.
- 게시물 번호 입력 시 해당 게시물의 추가 작업으로 이동한다.
- -1 입력 시 이전 메뉴로 돌아간다.

4) Failure case:

- 친구가 아닌 계정의 게시물이 보이는 경우
- 시간순으로 출력하지 않은 경우
- 게시물 번호를 입력했을 때 적절한 작업이 수행되지 않는 경우
- -1 을 입력했을 때 이전 메뉴로 돌아가지지 않는 경우

2. Post

1) 입력 정보

- 게시물 내용

2) 기능

- 게시글을 작성하는 기능
- 게시물 내용이 없으면 이전 메뉴로 돌아간다.

3) Valid case

- 적절하게 게시물이 작성이 되어 나의 친구 계정에서의 feed 에 보이고 My posting 에서 보이는 경우.
- 아무것도 입력받지 않으면 이전 메뉴로 돌아가는 경우.

4) Failure case

- 나의 친구 계정의 feed 에서 보이지 않거나 My posting 에서 보이지 않는 경우
- 게시물 내용이 없는데 게시물이 등록되는 경우

3. My posting

1) 입력 정보

- 게시물 번호

2) 기능

- 내가 작성한 게시글을 시간순으로 보여주는 기능

3) Valid case

- 1. All feed 와 동일하게 게시물 번호를 0 부터 시간순으로 부여해주고 최신 게시물일 수록 상단에 노출되는 경우
- 1. All feed 와 동일하게 게시물 번호를 입력할 경우 추가 작업을 수행하는 창으로 넘어간다.
- -1 을 입력했을 때 정상적으로 이전 메뉴로 돌아가는 경우
- 아무 게시물이 없다면 적절한 안내 메시지를 출력하는 경우

4) Failure case

- 게시물 번호가 시간순으로 출력되지 않는 경우
- 최신 게시물일수록 상단에 노출되지 않는 경우
- 게시물 번호를 입력하였을 때 적절하게 다음 메뉴로 넘어가지 않는 경우
- -1 을 입력했을 때 정상적으로 이전 메뉴로 돌아가지지 않는 경우

4. Previous menu

1) 입력 정보

- 없음

2) 기능

- Menu - Member 로 돌아감.

2.3 실행 예시

시스템은 텍스트 기반 UI에 의해 작동되며, 유저는 해당 메뉴 번호를 입력하여 메뉴를 선택할 수 있다. 실행 예시에서는 2.2 에서 자세히 다루지 않은 게시글 관련 기능을 위주로 설명한다.

```
=====
Menu - Main
1. Sign up
2. Sign in
3. Load command
4. Program exit
=====
Select Menu:1
=====
ID: saero2
Name: Park saeroy
Birthday: 2000/12/17
Password: itaewonclass
Registration Done!
=====
Menu - Main
1. Sign up
2. Sign in
3. Load command
4. Program exit
=====
```

Select Menu:2

=====

ID: saero3

Password: gangnamstyle

Unregistered account!

=====

Menu - Main

1. Sign up

2. Sign in

3. Load command

4. Program exit

=====

Select Menu:2

=====

ID: saero2

Password: itaewonclass

=====

Menu - My page

@saero2 - Park saeroy 2000.12.17

1. Friends

2. Feed

3. Sign out

4. Delete my account

=====

Select Menu: 2

=====

Menu - Feed

1. All Feed

2. Post

3. My posting

4. Previous menu

=====

Select Menu: 1

=====

0. @seojun (Like: 2)

- Hello world

1. @taehee (Like: 4)

- Have a nice day

```
2. @minjeong (Like: 10)
- I wanna go home
3. @pytorch (Like: 25)
- I am better than tensorflow
4. @tensorflow (Like: 12)
- Deep learning is amazing
=====
Select number: 4
=====
@tensorflow
- Deep learning is amazing
Reply:
@pytorch - I think so.
@minjeong - I wanna learn it
=====
Do you like it? (y/n): y
Write Comment:
=====
0. @seojun (Like: 2)
- Hello world
1. @taehee (Like: 4)
- Have a nice day
2. @minjeong (Like: 10)
- I wanna go home
3. @pytorch (Like: 25)
- I am better than tensorflow
4. @tensorflow (Like: 13)
- Deep learning is amazing
=====
Select number: 4
=====
@tensorflow (Like: 13)
- Deep learning is amazing
Reply:
@pytorch - I think so.
@minjeong - I wanna learn it
=====
Write Comment: Where can I learn it
=====
```

```
0. @seojun (Like: 2)
- Hello world
1. @taehee (Like: 4)
- Have a nice day
2. @minjeong (Like: 10)
- I wanna go home
3. @pytorch (Like: 25)
- I am better than tensorflow
4. @tensorflow (Like: 13)
- Deep learning is amazing
=====
Select number: -1
=====
Menu - Feed
1. All feed
2. Post
3. My posting
4. Previous menu
=====
Select Menu: 2
=====
Post:
=====
Menu - Feed
1. All feed
2. Post
3. My posting
4. Previous menu
=====
Select Menu: 2
=====
Post: Programming is so ez
=====
Menu - Feed
1. All feed
2. Post
3. My posting
4. Previous menu
=====
```

Select Menu: 3

=====

0. @saero2 (Like: 0)

- Programming is so ez

1. @saero2 (Like: 23)

- Programming is so funny

=====

Select number: 1

=====

@saero2

- Programming is so funny

Reply:

@pytorch - I agree it

@tensorflow - train a deep neural network using me

=====

Write Comment:

=====

0. @saero2 (Like: 0)

- Programming is so ez

1. @saero2 (Like: 23)

- Programming is so funny

=====

Select number: -1

=====

Menu - Feed

1. All feed

2. Post

3. My posting

4. Previous menu

=====

Select Menu: 4

=====

Menu - My page

@saero2 - Park saeroy 2000.12.17

1. Friends

2. Feed

3. Sign out

4. Delete my account

=====

```

Select Menu: 4
=====
Menu - Main
1. Sign up
2. Sign in
3. Load command
4. Program exit
=====
Select menu: 4

```

1) 해당 메뉴가 선택되면(예시: 1. Register as Member), 각 메뉴 별 필요한 정보(4.2 로그파일 형식 참조)를 입력 받도록 한다. 메뉴 디자인은 임의로 변경이 가능하나, **반드시 메뉴 번호, 순서, 번호별 기능은 기존과 같게 하도록 한다.**

2) Feed와 Post와 관련한 추가 설명

- 위의 예시에서는 "Programming is so funny" 게시물을 직접 작성하지는 않았지만 적절한 출력 예시를 보여주기 위해 미리 작성되어 있다고 가정한다.
- 게시물 및 댓글 작성 시에 아무것도 입력하지 않은 채 Enter키를 누른다면, 아무 일도 일어나지 않고 이전 메뉴로 이동해야 한다.
- 좋아요를 누르면 바로 반영이 되어야 하며(위의 예시에서 Like: 12 -> Like: 13), n 을 선택하더라도 좋아요가 차감되지는 않는다.
- 한번 좋아요를 누른 게시물에는 Do you like it? 메시지가 출력되지 않아야 한다.
- 항상 아무것도 출력할 것이 없는 경우에는 적절한 안내 메시지가 안내되어야 한다.

3. Class Design

5) 클래스 모듈화

본 과제의 주요 목표인 클래스 설계를 달성하기 위해, 클래스 설계는 가능한 분리된 기능을 수행하도록 설계한다. 예를 들어, 하나의 클래스에 모든 기능(회원 정보를 담는 기능, 회원 리스트 관리 등) 수행하도록 하는 경우는 지양하도록 하며 이는 채점에 반영될 예정이다.

6) Class Design 예시

- Member Class

회원에 대한 정보를 담는 클래스

- Member Collection Class:

회원 리스트를 관리하는 클래스

- 리스트 구현 시, 배열을 사용하여 구현하며, 최대 저장 가능한 회원의 수는 한정되어 있음.

Member	MemberCollection
멤버 변수 id name birthday ...	멤버 변수 // 멤버 리스트 배열 member_list[MAX_NUM_MEMBER]; ...
멤버 함수 get_login_id(); set_password(); ...	멤버 함수 get_member(); ...

*** 유의사항 ***

자신이 설계한 클래스를 위와 같이 보기 쉽게 표로 작성하여 보고서에 반드시 첨부하도록 합니다.

4. Load command

본 과제에서는 채점의 공정성을 위해 사전에 작성된 test case들을 이용하여 채점할 예정이며, 이를 위해 본 기능은 반드시 구현되어야 한다. 앞서 언급한 것처럼 모든 메뉴의 순서 및 기능은 안내된 것과 동일해야 한다. 그렇지 않을 경우, 채점에 불이익을 받을 수 있다.

7) Command file 형식

각 명령들은 개행문자로 인해 구분되어야 한다. **2.3 실행 예시**에 사용된 예제를 command.txt 로 나타내면 다음과 같다.

```
1 1
2 saero2
3 Park saeroy
4 2000/12/17
5 itaewonclass
6 2
7 saero3
8 gangnamstyle
9 2
10 saero2
11 itaewonclass
12 2
13 1
14 4
15 y
16
17 4
18 Where can I learn it
19 -1
20 2
21
22 2
23 Programming is so ez
24 3
25 1
26
27 -1
28 4
29 4
30 4
```

Figure 6. command.txt 예시

5. 과제 제출

- 1) 제출 파일은 **assn2.zip, assn2_report.pdf** 형태로 **두가지 파일을 따로 업로드**할 것
- 2) 실행 파일 파일을 제외한 나머지 **makefile, 소스 및 헤더 파일**만을 제출하고, 해당 압축파일을 풀었을 때 **make** 커맨드만으로 컴파일이 가능하도록 제출할 것