



실전 프로젝트1





7

TodoListApp

Update #2

공지

- 출석확인 방법
 - 화요일 (대면 / 줌)
 - 금요일 (줌), LMS 인 경우 HD LMS에서 자동출석체크
 - Hisnet 온라인 출석
 - 수강과목 선택 > 강의정보 > 강의출석현황 메뉴에서 확인
- 5주차
 - 9/28(화) - 기존과 동일하게 대면/줌 병행 수업 진행
 - 10월 1일(금) - HDLMS 동영상 시청 + 실시간 온라인 실습으로 수업 진행함
 - **LMS 5주차 수업 동영상을 아래 실습 시간 시작 전까지 시청 완료해야 함! (자동 출석 체크)**
 - 1분반(김광) - 오전 10시부터 줌으로 실시간 온라인 실습 진행
 - 2분반(장소연) - 오전 10시부터 줌으로 실시간 온라인 실습 진행
 - 3분반(장소연) - 오후 5시부터 줌으로 실시간 온라인 실습 진행

TodoListApp Update#1 review

- 내용 입력 기능 개선
 - 항목 추가할 때, 내용에 단어만 입력되는 오류 개선하여 문장이 입력되도록 개선
- TodoUtil.java
 - updateItem()
 - createItem()

```
public static void createItem(TodoList l) {  
  
    String title, desc;  
    Scanner sc = new Scanner(System.in);  
  
    System.out.print("[항목 추가]\n"  
        + "제목 > ");  
  
    title = sc.next();  
    if (l.isDuplicate(title)) {  
        System.out.println("제목이 중복됩니다!");  
        return;  
    }  
    sc.nextLine();  
    System.out.print("내용 > ");  
    desc = sc.nextLine().trim();  
  
    TodoItem t = new TodoItem(title, desc);  
    l.addItem(t);  
    System.out.println("추가되었습니다.");  
}
```

TodoListApp Update#1 review

- 항목 내 날짜 정보를 문자열로 다루도록 수정
 - 파일에 저장하고 불러오는 기능의 편의성을 위해 문자열로 보관하는 것이 좋음!
- TodoItem.java
 - String current_date;
 - 생성자 변경 SimpleDateFormat 사용

```
public class TodoItem {  
    private String title;  
    private String desc;  
    private String current_date;
```

```
public TodoItem(String title, String desc){  
    this.title=title;  
    this.desc=desc;  
    SimpleDateFormat f = new SimpleDateFormat("yyyy/MM/dd kk:mm:ss");  
    this.current_date = f.format(new Date());  
}
```

TodoListApp Update#1 review

- Class SimpleDateFormat

Constructors

Constructor and Description

`SimpleDateFormat()`

Constructs a SimpleDateFormat using the default pattern and date format

`SimpleDateFormat(String pattern)`

Constructs a SimpleDateFormat using the given pattern and the default locale

`SimpleDateFormat(String pattern, DateFormatSymbols formatSymbols)`

Constructs a SimpleDateFormat using the given pattern and date format symbols

`SimpleDateFormat(String pattern, Locale locale)`

Constructs a SimpleDateFormat using the given pattern and the default locale

Letter	Date or Time Component
G	Era designator
y	Year
Y	Week year
M	Month in year
w	Week in year
W	Week in month
D	Day in year
d	Day in month
F	Day of week in month
E	Day name in week
u	Day number of week (1 = Monday, ..., 7 = Sunday)
a	Am/pm marker
H	Hour in day (0-23)
k	Hour in day (1-24)
K	Hour in am/pm (0-11)
h	Hour in am/pm (1-12)

<https://docs.oracle.com/javase/7/docs/api/java/text/SimpleDateFormat.html>

TodoListApp Update#1 review

- 데이터파일 (todolist.txt) 저장 기능 구현
 - 프로그램 종료 시 저장
 - TodoUtil.java
 - saveList()
 - FileWriter 사용

```
public static void saveList(TodoList l, String filename) {  
    try {  
        Writer w = new FileWriter(filename);  
        for (TodoItem item : l.getList()) {  
            w.write(item.toSaveString());  
        }  
        w.close();  
        System.out.println("모든 데이터가 저장되었습니다.");  
    } catch (IOException e) {  
        // TODO Auto-generated catch block  
        e.printStackTrace();  
    }  
}
```

TodoItem.java

```
public String toSaveString() {  
    return title + "##" + desc + "##" + current_date + "\n";  
}
```


ToDoListApp Update#1 review

- 데이터파일 (todolist.txt)
읽기 기능 구현
 - 프로그램 시작 시 읽기
 - TodoUtil.java
 - loadList()
 - BufferedReader,
FileReader,
StringTokenizer
사용

todolist.txt

```
1 과제##운영체제, 알고리즘 과제하기##2021/09/10 10:00:01
2 회식##동아리 동기들과 밥먹기##2021/09/12 12:10:21
3 운동##1킬로미터 빠르게 걷기##2021/09/13 20:30:31
4 세미나##랩실 세미나 준비하기##2021/09/14 09:20:45
```

5|

```
public static void loadList(ToDoList l, String filename) {
    try {
        BufferedReader br = new BufferedReader(new FileReader(filename));
        String line;
        int count = 0;
        while((line = br.readLine()) != null) {
            StringTokenizer st = new StringTokenizer(line, "##");
            String title = st.nextToken();
            String description = st.nextToken();
            String current_date = st.nextToken();
            TodoItem item = new TodoItem(title, description);
            item.setCurrent_date(current_date);
            l.addItem(item);
            count++;
        }
        br.close();
        System.out.println(count+"개의 항목을 읽었습니다.");
    } catch (FileNotFoundException e) {
        // TODO Auto-generated catch block
        System.out.println(filename+" 파일이 없습니다.");
        //e.printStackTrace();
    } catch (IOException e) {
        // TODO Auto-generated catch block
        e.printStackTrace();
    }
}
```


4주차 실습 - TodoListApp Update #2

1. 항목 목록에서 개수를 표현할 것!

- TodoUtil 클래스 내 listAll() 변경

2. TodoItem 필드 추가 및 기능에 반영

- 카테고리 필드 - String category; / 마감일자 필드 - String due_date;
- 추가, 삭제, 수정, 목록출력, 파일읽기 등의 기능에 변경 사항 반영

3. 기능 변경

- 목록 출력 시 일련번호를 함께 출력하도록 하여, 특정 항목을 선택할 때에 이 번호를 사용할 것.
- [항목 수정], [항목 삭제] 에서 제목으로 찾는 대신, 번호를 입력하도록 변경하여 기능 수행

```
Command > ls
```

[전체 목록, 총 4개]

1. [과제] 3주차과제하기 - 운영체제, 알고리즘 과제하기 - 2021/09/21 - 2021/09/10 10:00:01
2. [기타] 동아리회식 - 동아리 동기들과 밥먹기 - 2021/09/29 - 2021/09/12 12:10:21
3. [취미] 운동 - 1킬로미터 빠르게 걷기 - 2021/09/15 - 2021/09/13 20:30:31
4. [과제] 세미나발표 - 랩실 세미나 준비하기 - 2021/09/27 - 2021/09/14 09:20:45

Command > **ls**

[전체 목록, 총 4개]

1. [과제] 3주차과제하기 - 운영체제, 알고리즘 과제하기 - 2021/09/21 - 2021/09/10 10:00:01
2. [기타] 동아리회식 - 동아리 동기들과 밥먹기 - 2021/09/29 - 2021/09/12 12:10:21
3. [취미] 운동 - 1킬로미터 빠르게 걷기 - 2021/09/15 - 2021/09/13 20:30:31
4. [과제] 세미나발표 - 랩실 세미나 준비하기 - 2021/09/27 - 2021/09/14 09:20:45

Command > **del**

[항목 삭제]

삭제할 항목의 번호를 입력하시오 > **3**

3. [취미] 운동 - 1킬로미터 빠르게 걷기 - 2021/09/15 - 2021/09/13 20:30:31

위 항목을 삭제하시겠습니까? (y/n) > **y**

삭제되었습니다.

Command > **ls**

[전체 목록, 총 3개]

1. [과제] 3주차과제하기 - 운영체제, 알고리즘 과제하기 - 2021/09/21 - 2021/09/10 10:00:01
2. [기타] 동아리회식 - 동아리 동기들과 밥먹기 - 2021/09/29 - 2021/09/12 12:10:21
3. [과제] 세미나발표 - 랩실 세미나 준비하기 - 2021/09/27 - 2021/09/14 09:20:45

Command > **edit**

[항목 수정]

수정할 항목의 번호를 입력하시오 > **3**

3. [취미] 운동 - 1킬로미터 빠르게 걷기 - 2021/09/15 - 2021/09/13 20:30:31

새 제목 > **운동하기**

새 카테고리 > **취미**

새 내용 > **평봉필드 2바퀴 돌기**

새 마감일자 > **2021/10/10**

수정되었습니다.

Command > **ls**

[전체 목록, 총 4개]

1. [과제] 3주차과제하기 - 운영체제, 알고리즘 과제하기 - 2021/09/21 - 2021/09/10 10:00:01
2. [기타] 동아리회식 - 동아리 동기들과 밥먹기 - 2021/09/29 - 2021/09/12 12:10:21
3. [과제] 세미나발표 - 랩실 세미나 준비하기 - 2021/09/27 - 2021/09/14 09:20:45
4. [취미] 운동하기 - 평봉필드 2바퀴 돌기 - 2021/10/10 - 2021/09/22 21:09:00

4주차 실습 - TodoListApp Update #2

4. 검색 기능 추가

- 명령어 형식 : **find <키워드>**
- 현재 목록 내의 제목과 내용에서 <키워드> 를 포함하고 있는 모든 항목을 출력한다.
- help 명령의 메뉴에 추가, 명령어에 대한 기능 추가

```
Command > ls
```

```
[전체 목록, 총 4개]
```

1. [과제] 3주차과제하기 - 운영체제, 알고리즘 과제하기 - 2021/09/21 - 2021/09/10 10:00:01
2. [기타] 동아리회식 - 동아리 동기들과 밥먹기 - 2021/09/29 - 2021/09/12 12:10:21
3. [취미] 운동 - 1킬로미터 빠르게 걷기 - 2021/09/15 - 2021/09/13 20:30:31
4. [과제] 세미나발표 - 랩실 세미나 준비하기 - 2021/09/27 - 2021/09/14 09:20:45

```
Command > find 3주
```

1. [과제] 3주차과제하기 - 운영체제, 알고리즘 과제하기 - 2021/09/21 - 2021/09/10 10:00:01
- 총 1개의 항목을 찾았습니다.

```
Command > find 하기
```

1. [과제] 3주차과제하기 - 운영체제, 알고리즘 과제하기 - 2021/09/21 - 2021/09/10 10:00:01
 4. [과제] 세미나발표 - 랩실 세미나 준비하기 - 2021/09/27 - 2021/09/14 09:20:45
- 총 2개의 항목을 찾았습니다.

4주차 실습 - TodoListApp Update #2 (선택과제)

5. 최신순 정렬 기능(선택기능)

- 명령어 형식, `ls_date_desc` (`ls_date` 의 역순으로 정렬)
- 현재 목록을 날짜 역순으로 출력한다.
- help 명령의 메뉴에 추가, 명령어에 대한 기능 추가

6. 카테고리 검색 기능(선택기능)

- 명령어 형식, `find_cate` <키워드>
- 현재 목록의 카테고리에 검색 <키워드>를 포함하고 있는 모든 항목을 출력한다.
- help 명령의 메뉴에 추가, 명령어에 대한 기능 추가

Command > `find_cate` 과제

```
1. [과제] 3주차과제하기 - 운영체제, 알고리즘 과제하기 - 2021/09/21 - 2021/09/10 10:00:01
4. [과제] 세미나발표 - 랩실 세미나 준비하기 - 2021/09/27 - 2021/09/14 09:20:45
총 2개의 항목을 찾았습니다.
```

4주차 실습 - TodoListApp Update #2 (선택과제)

7. 카테고리 목록 출력 기능(선택기능)

- 명령어 형식, `ls_cate`
- 현재 등록되어 있는 카테고리를 중복되지 않도록 출력한다.
- help 명령의 메뉴에 추가, 명령어에 대한 기능 추가

```
Command > ls
```

```
[전체 목록, 총 4개]
```

1. [과제] 3주차과제하기 - 운영체제, 알고리즘 과제하기 - 2021/09/21 - 2021/09/10 10:00:01
2. [기타] 동아리회식 - 동아리 동기들과 밥먹기 - 2021/09/29 - 2021/09/12 12:10:21
3. [취미] 운동 - 1킬로미터 빠르게 걷기 - 2021/09/15 - 2021/09/13 20:30:31
4. [과제] 세미나발표 - 랩실 세미나 준비하기 - 2021/09/27 - 2021/09/14 09:20:45

```
Command > ls_cate
```

```
과제 / 기타 / 취미
```

```
총 3개의 카테고리가 등록되어 있습니다.
```

4주차 실습과제 제출

- Due date - 2021년 9월 26일 (일) 밤 11시 59분
- 제출방법 - LMS 과제로 PDF 파일 제출
- PDF 파일 내 포함 내용
 - 파일명 형식 : 이름(학번)_4주차.pdf
 - 4주차 실습 수행을 통해 업데이트된 프로젝트 실행 결과
 - 프로젝트 최종 소스가 Commit & Push 된 자신의 Github URI 기입
 - 각각의 미션 수행을 확인할 수 있는 실행 화면 캡처 이미지 첨부 & 이미지 별로 간단한 설명 기입
 - 항목을 10개 이상 등록하여 저장된 todolist.txt 파일 내용 캡처 이미지 첨부
 - 4주차 실습을 수행하면서 느낀 점

과제 평가 부여 기준 (만점)

- 필수 미션만 완료 - 80%
- 선택 미션까지 완료 - 100%