
The Frist Project

Console Board Java

2023.02.27- 2023.03.08

Kim Eun Ji

INDEX

1.기획

- CONCEPT
- 개발 목적

2.구현

- 기능 정의
 - 설계도
 - 소스 코드
 - 구현 화면
 - 프로젝트 후기
-

CONCEPT

‘자바 콘솔 게시판’

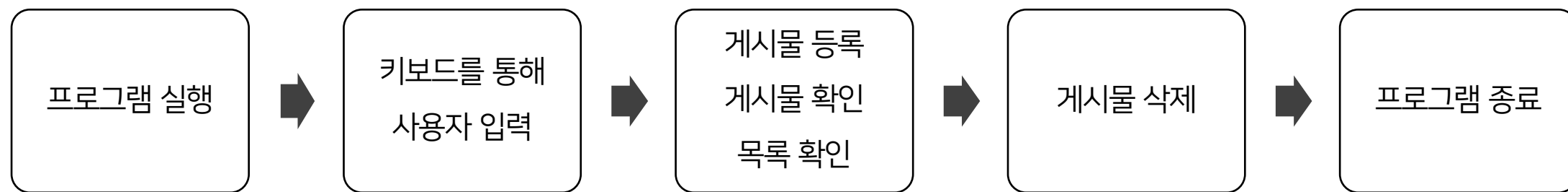
콘솔을 통해 사용자에게 데이터를 입력 받아
입력 받은 내용을 콘솔로 보여주는 게시판

```
=====
번호 제목 작성자   작성일
-----
1.목록  2.등록  3.내용  4.삭제  0.종료>1
=====
번호 제목 작성자   작성일
-----
1.목록  2.등록  3.내용  4.삭제  0.종료>2
제목>아아아
내용>안녕하세요
작성자>김말똥
{"msg":{"num":0,"title":"아아아","content":"안녕하세요","writer":"김말똥"},"type":"insert"}
글이 정상적으로 추가 되었습니다.
1.목록  2.등록  3.내용  4.삭제  0.종료>1
=====
번호 제목 작성자   작성일
-----
1   아아아   김말똥   Sep 25, 2022, 6:00:48 PM
1.목록  2.등록  3.내용  4.삭제  0.종료>3
=====
번호 제목 작성자   작성일
-----
1   아아아   김말똥   Sep 25, 2022, 6:00:48 PM
위의 게시판 번호중 하나를 넣어 주세요.>>1
=====
번호 제목 내용 작성자   작성일
-----
1   아아아   안녕하세요   김말똥   Sep 25, 2022, 6:00:48 PM
1.목록  2.등록  3.내용  4.삭제  0.종료>|
```

개발목적

- 간단한 명령어를 키보드로 입력하여 사용 할 수 있는 콘솔 게시판
- 게시글을 등록, 수정,삭제 할 수 있는 게시판
- 작성한 게시글 목록,내용 작성일자를 확인할 수 있는 게시판

기능 정의



설계도

사용자 입력[Scanner]

☐ 제목 [title]☐ 내용 [body]☐ 작성자 [username]☐ 작성 날짜/시간 [date]

콘솔 출력

☐ 목록
•게시글 번호 / 제목 / 작성자 / 날짜 시간☐ 상세 내용
•게시글 번호 / 제목 / 내용 / 작성자 / 날짜 시간☐ 기능 실행 메시지

기능

☐ 내용 입력☐ 명령어 실행☐ 등록☐ 조회☐ 삭제☐ 종료

소스 코드

- Class Article

데이터 입출력을 담당하는

인스턴스 생성

- toString 메서드 오버라이드

- 각각의 인스턴스의

Get() Set() 생성

```
public class Article {
    private int id;
    private String title;
    private String body;
    private String date;
    private String userName;

    Article(int id, String title, String body, String date, String userName) {

        this.id = id;
        this.title = title;
        this.body = body;
        this.date = date;
        this.userName = userName;
    }

    @Override
    public String toString() {
        return String.format("[No.%d] [제목:%s] [내용:%s] [작성자:%s] [작성일:%s]", id, title, body, userName, date);
    }

    public int getId() {
        return id;
    }

    public void setId(int id) {
        this.id = id;
    }

    public String getTitle() {
        return title;
    }
}
```

소스 코드

자바 콘솔 게시판의 코드를 출력하는 메인

- Scanner 생성
- Class Board 생성
- setScanner 생성
- Run() Board 소스코드 실행
- sc.close() 종료

```
public class Main {  
    public static void main(String[] args) {  
        Scanner sc = new Scanner(System.in);  
        Board board = new Board();  
        board.setScanner(sc);  
        board.run();  
        sc.close();  
    }  
}
```

소스 코드

- 사용자로부터 명령어 입력 받을 수 있도록 스캐너 연결
- ArrayList <>생성
- 입력 데이터를 받은 setScanner ()

```
class Board {  
    Scanner sc;  
  
    int userId = 0; // 게시글 갯수 증가  
    List<Article> articles = new ArrayList<>();  
  
    void setScanner(Scanner sc) {  
        this.sc = sc;  
    }  
}
```

소스 코드

- run() 프로그램 시작
- 0번 종료 명령어 처리
- 1번 게시물 목록 출력

```
void run() {
    System.out.println("=====게시판=====");
    System.out.println("=====");
    System.out.println("[번호]   [제목]   [작성자]   [작성일]");
    System.out.println("-----");
    System.out.println("1.목록 2.등록 3.내용 4.삭제 0.종료");

    while (true) {

        System.out.printf("번호를 입력해주세요. > ");

        String cmd = sc.nextLine().trim();

        if (cmd.equals("0")) {
            System.out.println("=====프로그램 종료=====\\n");
            break;
        } else if (cmd.equals("1")) {
            System.out.println("=====게시물 목록=====");
            System.out.println("-----");
            System.out.println("No. / 제목 ");
            System.out.println("-----");

            for (int i = articles.size() - 1; i >= 0; i--) {
                Article article = articles.get(i);
                System.out.printf("%d / %s / %s / %s \\n", article.getId(), article.getTitle(),
                    article.getUserName(), article.getDate());
            }
        }
    }
}
```

소스 코드

- 2번 입력 받은 내용 등록
- 등록한 날짜/시간 데이터
- Article 객체로 생성된 게시물 저장

```
} else if (cmd.equals("2")) { // 입력

    CmdAdd();

private void CmdAdd() {

    System.out.println("=====게시물 등록=====");
    System.out.printf("제목 : ");
    String title = sc.nextLine();
    System.out.printf("내용 : ");
    String body = sc.nextLine();
    System.out.printf("작성자 : ");
    String userName = sc.nextLine();

    SimpleDateFormat dt = new SimpleDateFormat("yy/MM/dd [HH:mm:ss]");
    String date = dt.format(new Date());

    int id = userId + 1;
    userId = id;

    Article article = new Article(id, title, body, date, userName); // 데이터 저장 객체 생성

    articles.add(article); // 생성된 게시물 목록 추가

    System.out.println("생성된 게시물 : " + article);

    System.out.printf("%d번 게시물이 등록 되었습니다.\n", article.getId());
}
```

소스 코드

- 3번 입력 받은 게시글 내용 확인
- 게시글 존재 여부 확인
- get() 각각의 데이터 출력

```
} else if (cmd.equals("3")) { // 내용확인
    System.out.print("내용을 확인할 게시글 번호를 입력해주세요. \n명령 > ");
    String num = sc.nextLine();

    if (articles.isEmpty()) {
        System.out.println("게시물이 존재하지 않습니다.");
        continue;
    }
    boolean b = false;
    for (Article article : articles) {
        try {
            if (article.getId() == Integer.parseInt(num)) {
                b = true;
                System.out.println("=====게시물 상세보기=====");
                System.out.printf("No.: %s\n", article.getId());
                System.out.printf("제목: %s\n", article.getTitle());
                System.out.printf("내용: %s\n", article.getBody());
                System.out.printf("작성일: %s\n", article.getUserName());
                System.out.printf("작성자: %s\n", article.getDate());
                break;
            }
        } catch (Exception e) {
            System.out.println("잘못된 입력값입니다.");
        }
    }
    if (!b) {
        System.out.println("게시물이 없습니다.");
    }
}
```

소스 코드

- 4번 게시물 삭제
- getId() 삭제할 글 번호 확인
- articles.remove
해당 번호의 게시물 삭제

```
} else if (cmd.equals("4")) {  
    // 삭제  
    System.out.println("삭제할 글 번호");  
    String s = sc.nextLine();  
    for (Article a : articles) {  
        try {  
            if (a.getId() == Integer.parseInt(s)) {  
                articles.remove(a);  
                System.out.println(a.getId() + " 번 글 " + a.getTitle() + " 이 정상적으로 삭제되었습니다.");  
                break;  
            }  
        } catch (Exception e) {  
            System.out.println("잘못된 입력값");  
        }  
    }  
} else {  
    System.out.printf("입력된 명령어 : %s\n", cmd);  
}  
}
```

+

구현 화면



+



-2번 명령어를 입력하여 게시물 등록



=====게시판=====

=====

[번호] [제목] [작성자] [작성일]

1. 목록 2. 등록 3. 내용 4. 삭제 0. 종료

번호를 입력해주세요. > 2

=====게시물 등록=====

제목 : 안녕하세요.

내용 : 안녕하세요 자바 첫번째 프로젝트를 완성했습니다.

작성자 : 김은지

생성된 게시물 : [No.1] [제목:안녕하세요.] [내용:안녕하세요 자바 첫번째 프로젝트를 완성했습니다.] [작성자:김은지] [작성일:23/03/08 [19:09:12]]

1번 게시물이 등록 되었습니다.

번호를 입력해주세요. > 2

=====게시물 등록=====

제목 : java

내용 : 자바랑 빨리 친해지고 싶네요.

작성자 : 김은지

생성된 게시물 : [No.2] [제목:java] [내용:자바랑 빨리 친해지고 싶네요.] [작성자:김은지] [작성일:23/03/08 [19:09:50]]

2번 게시물이 등록 되었습니다.

번호를 입력해주세요. > 2

=====게시물 등록=====

제목 : 언제쯤

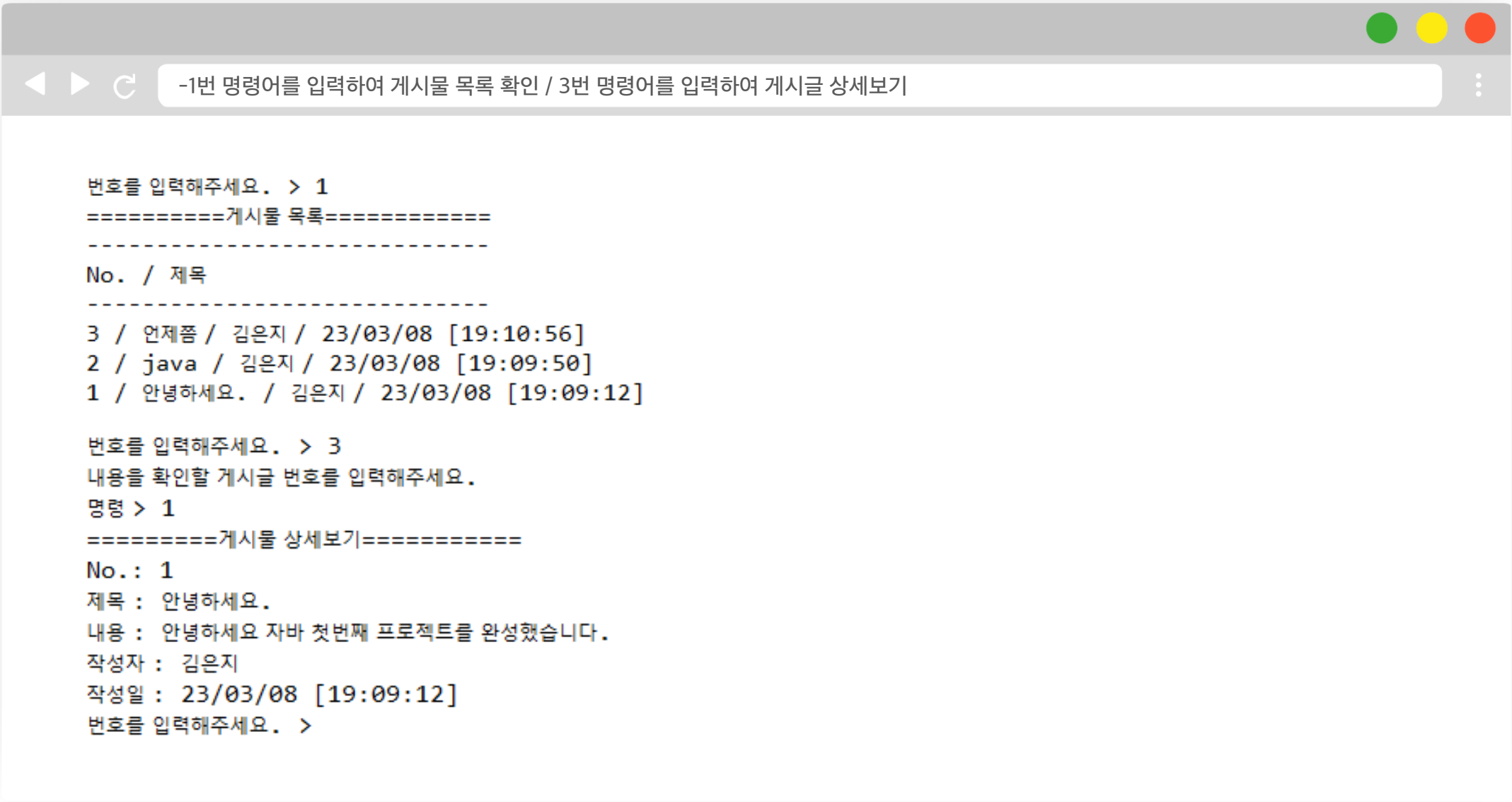
내용 : 언제쯤 자바랑 친해질 수 있을까요.. 모두 화이팅합니다.

작성자 : 김은지

생성된 게시물 : [No.3] [제목:언제쯤] [내용:언제쯤 자바랑 친해질 수 있을까요.. 모두 화이팅합니다.] [작성자:김은지] [작성일:23/03/08 [19:10:56]]

3번 게시물이 등록 되었습니다.

번호를 입력해주세요. >



=====게시물 목록=====

3 / 언제쯤 / 김은지 / 23/03/08 [19:10:56]
2 / java / 김은지 / 23/03/08 [19:09:50]
1 / 안녕하세요. / 김은지 / 23/03/08 [19:09:12]

삭제할 글 번호 작성자 작성일

번호를 입력해주세요. > 1

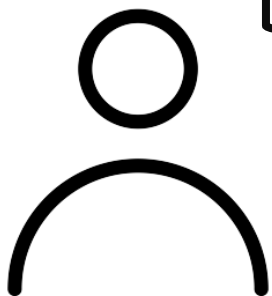
=====게시물 목록=====

3 / 언제쯤 / 김은지 / 23/03/08 [19:10:56]
1 / 안녕하세요. / 김은지 / 23/03/08 [19:09:12]
번호를 입력해주세요. >

‘ 첫 프로젝트를 마치며 ’

구현해야 할 기능들을 리스트화 시켜 그동안 수업 시간에 배웠던 문법을 사용하여 하나하나 기능을 구현하는 과정에서 프로그램 하나를 동작 시키는 전반적인 구현 방법에 대해 배워가는 프로젝트였습니다.

이번 프로젝트를 하면서 사용자에게 데이터를 받고 출력하는 개념에 대해 많은 공부가 되었고, 객체지향 언어인 자바를 사용하여 프로그래밍 하였지만 클래스들은 나누고 함수를 받아오는 각각의 기능별로 클래스를 나누고 함수를 사용하는 객체지향 개념이 많이 부족하다고 느꼈습니다.



Hello my name is Kim Eun Ji
