도서 관리 프로그램 보고서

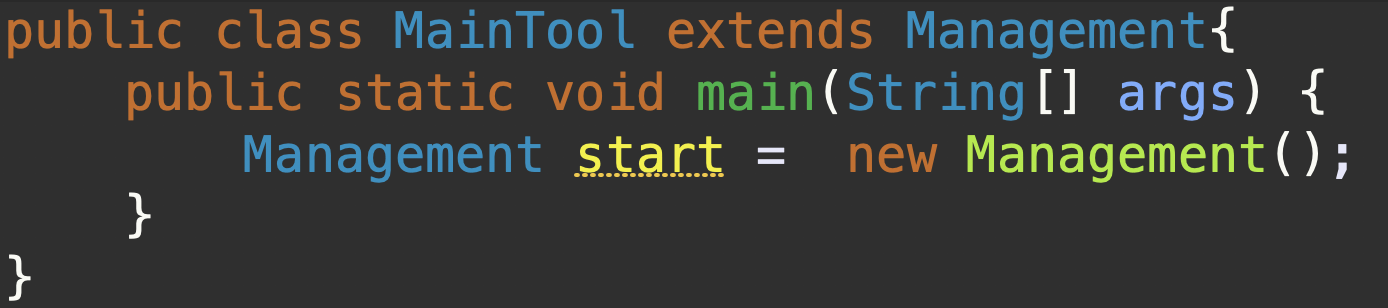
1305강민석

1. 클래스 리스트

* MainTool
* Management
* BookInfo
* BookManager
* UserInfo
* UserManager
* BorrowManager

1. 클래스 코드 설명

**<MainTool>**



**main 메소드**: Management 인스턴스(객체) 생성을 하여 Management 클래스의 생성자를 호출.

**<Management>**

스크린샷이(가) 표시된 사진

자동 생성된 설명

**Management클래스 생성자**:

스위치문에서 selected변수에 입력된 기능 값(0~6)에 따라 각각의 기능이 있는 클래스의 객체에 접근하여 메소드를 실행.

**<BookInfo>**

텍스트이(가) 표시된 사진

자동 생성된 설명

**BookInfo클래스 생성자**:

- 책이름과 분류 번호를 입력 받아 저장하고, 나머지는 기본값으로 초기화 해준다.

**<BookManager>**

**스크린샷, 앉아있는이(가) 표시된 사진

자동 생성된 설명**

**insertBook메소드**:

스위치문에서 책의 분류 번호를 입력 받아 각 분류 번호에 맞는 bookList 인덱스번호 위치에 최대 30권씩 저장 될 수 있도록 저장. (0~29: 예술, 30~59: 문학, 60~89: 과학, 90~120: 역사)

**new BookInfo(inputName, inputGroup)**:

inputName, inputGroup변수에 저장된 책 이름과 분류 번호를 BookInfo생성자에 넘겨줌.

스크린샷이(가) 표시된 사진

자동 생성된 설명

**current메소드**:

bookList배열에 저장된 책을 for문을 사용해 분류번호별로 나누어 검색.

**각각의 for문**:

for문을 분류번호마다 30번씩 돌려 bookList배열에 저장돼있는 각 책마다의 isBorrow변수(책 대출 여부)를 검사하여 대출한 책(borrowCnt)을 증가시키거나 남아있는 책(remaining)의 카운트를 증가시킴.

스크린샷이(가) 표시된 사진

자동 생성된 설명

**Static public BookInfo findBook(String inputName)**:

for문을 사용하여 bookList에 있는 책의 name(책 이름)을 매개변수인 inputName과 비교하여 같다면 bookList안에 있는 그 책을 반환하여 준다.

**Public BookInfo findBook(int userNum)**:

for문을 사용하여 bookList에 있는 책의 userNum(회원번호)을 매개변수인 userNum과 비교하여 같다면 foundBookList배열에 저장하여 준다. 그리고 foundBookList배열이 비어있지 않다면 자신을 반환하여 준다.

**void findBook()**:

foundBook에 findBook(String inputName)메소드를 호출하여 bookName과 같은 책을 반환 받는다.

foundBook에 있는 책의 isBorrow변수(책 대출 여부)를 검사하여 대출중인지 대출가능한지를 출력 해준다.

**<UserInfo>**

스크린샷이(가) 표시된 사진

자동 생성된 설명

**UserInfo클래스 생성자**:

이름, 주소, 전화번호, 회원번호를 받아 초기화 해준다.

(private변수인 address와 phoneNum은 setter를 통하여 초기화)

**<UserManager>**

스크린샷이(가) 표시된 사진

자동 생성된 설명

BookManager를 상속받음.

**UserManager클래스 생성자**: 10명의 도서 회원을 추가함.

**UserInfo findUser(int userNum)**:

for each문을 사용하여 users배열에 있는 각 유저의 userNum과 매개변수 userNum과 비교하여 같다면 해당 회원을 반환.

**void findUser()**:

foundUser에 findUser(int userNum)메소드를 호출하여 회원번호를 통하여 회원을 반환 받아 저장한다. foundUser에 값이 있다면 foundBook에 findBook(int userNum)메소드를 호출하여 해당 회원번호의 책을 반환받아 저장한다. foundBook을 통하여 대출한 책을 출력한다.

**<BorrowManager>**

스크린샷이(가) 표시된 사진

자동 생성된 설명

**borrowBook메소드**:

foundUser에 findUser(int userNum)메소드를 호출하여 회원을 반환받아 저장함.

foundUser의 borrowCount변수(대출 권수)가 3미만이라면 foundBook에 findBook(String bookName)메소드를 호출하여 반환값을 저장함.

대출기능-> {foundBook의 isBorrow값이 true가 아니라면 foundBook의 userNum에 회원이름을 넣어주고, isBorrow를 true값을 넣고, returnDate에 반납일을 넣어준다.}

foundUser의 borrowCount변수(대출 권수)를 1증가 시킨다. 대출 완료를 출력 한다.

스크린샷이(가) 표시된 사진

자동 생성된 설명

**returnBook메소드**: (책 반납은 시험삼아 구현해보았습니다.)

foundUser에 findUser(int userNum)메소드를 호출하여 회원을 반환받아 저장함.

foundUser의 borrowCount변수(대출 권수)가 0초과라면 foundBook에 findBook(String bookName)메소드를 호출하여 반환값을 저장함.

반납기능-> {foundBook에 값이 있다면, foundBook의 userNum(회원번호)을 0으로 초기화 해주고, isBorrow에 false값을 넣고, returnDate에 빈 문자열을 넣어준다. foundUser의 borrowCount변수(대출 권수)를 1증가 시킨다.}

foundUser의 borrowCount변수(대출 권수)를 1감소 시킨다. 반납완료를 출력시킨다.

**date메소드**: 현재날짜를 기준으로 7일 이후의 날짜를 반환한다.