

```
package sample;
```

```
public class Book {  
    String bookId  
    String title  
    String author ;  
    int state=1 ;    //0  1  2
```

```
    public Book(String bookId, String title, String author) {  
        super();  
        this.bookId = bookId  
        this.title = title  
        this.author = author  
    }
```

```
    public int getState() {  
        return state  
    }
```

```
    public void setState(int state) {  
        this.state = state  
    }
```

```
    public String getBookId() {  
        return bookId  
    }
```

```
    @Override  
    public String toString() {  
  
        return "Book [bookId=" + bookId + ", title=" + title + ", author=" + author + ", state=" + state + "]"  
    }  
}
```

```
package sample;
```

```
import java.text.SimpleDateFormat;
```

```
import java.util.ArrayList;
```

```
import java.util.Date;
```

```
import java.util.Scanner;
```

```
public class Lend {
```

```
    String lendId ;
```

```
    Date lendDate ;
```

```
    Date returnDate;
```

```
    Book book;
```

```
    boolean returnFlag;
```

```
    public Lend(String lendId, Date lendDate, Date returnDate, Book book) {
```

```
        super();
```

```
        this.lendId = lendId;
```

```
        this.lendDate = lendDate;
```

```
        this.returnDate = returnDate;
```

```
        this.book = book;
```

```
    }
```

```
    @Override
```

```
    public String toString() {
```

```
        SimpleDateFormat sf = new SimpleDateFormat("yyyy-MM-dd");
```

```
        String lendDateS = sf.format(this.lendDate.getTime());
```

```
        String returnDateS = sf.format(this.returnDate.getTime());
```

```
        //
```

```
        return "Lend [lendId=" + lendId + ", lendDate=" + lendDateS + ", returnDate=" + returnDateS + ", book="
```

```
+ book
```

```
        + ", returnFlag=" + returnFlag + "];
```

```
    }
```

```
    public void setReturnFlag(boolean returnFlag) {
```

```
        this.returnFlag = returnFlag;
```

```
    }
```

```
}
```

```
package sample;
```

```
import java.util.Date;
import java.text.SimpleDateFormat;
import java.util.ArrayList;
import java.util.Calendar;
import java.util.Scanner;
```

```
/*
 *
 * 1. 대여를 위한 책정보 생성
 * 2. 반복 (메뉴가 종료가 아닐 동안)
 *    - 메뉴선택
 *    - 대여
 *      - 대여번호, 도서번호입력
 *      - 도서가 대여가능상태인지 판단
 *      - 도서 대여가능
 *        :도서상태 대여중으로 변경
 *        :반납일자, 대여일자생성
 *        :대여리스트에 추가
 *      - 도서 대여 불가능
 *        : 도서불가능 메시지
 *    - 반납
 *      -대여번호 입력
 *      -반납상태변경
 *      -도서의 상태를 대여가능으로 변경
 *    - 조회
 *      - 대여리스트 전체 조회 *
 *
 */
```

```
public class BookProgram {
    public static void main(String[] args) {

        //책정보 생성
        ArrayList<Book> books = new ArrayList<>();
        Scanner sc = new Scanner(System.in);

        books.add( new Book("b001" ,"재미있는 책" , "강지훈"));
        books.add( new Book("b002" ,"신나는 책" , "김기엽"));
        books.add( new Book("b003" ,"재미있는 책" , "김수진"));
        books.add( new Book("b004" ,"재미있는 책" , "김연수"));
        books.add( new Book("b005" ,"재미있는 책" , "김예원"));
        books.add( new Book("b006" ,"재미있는 책" , "김종범"));

        // 대여시작 (메뉴제공)
        ArrayList<Lend> lendList = new ArrayList<>();

        loop1 :while( true) {

            System.out.println("1.대여   2. 조회   3.반납   4. 종료");
```

```

int menu = sc.nextInt();
switch( menu) {
case 1:
    System.out.println();
    System.out.println("대여코드입력");
    String lendId= sc.next();

    System.out.print("도서코드입력");
    String bookId = sc.next();

    boolean flag=false;
    // 책 대여가능여부 , 책상태 비교
    Book book=null;
    for(int i=0 ; i< books.size(); i++) {
        book = books.get(i);
        if(book.getBookId().equals(bookId) && book.getState() ==1 ) {
            book.setState(0); // 대여중
            flag=true;
            break; // 도서발견하면 반복문 나옴
        }
    }

    if(flag) {

        //대여일, 반납일 구하기
        Calendar cal = Calendar.getInstance();

        Date lendDate = cal.getTime() ;
        // 14일후 2주 후 , 일 기준
        cal.add(Calendar.DATE ,14);
        Date returnDate = cal.getTime() ;

        Lend lend = new Lend( lendId, lendDate, returnDate, book );
        lendList.add(lend);
        System.out.println( lend);
        System.out.println( "대여되었습니다");

    }else {
        System.out.println("대여불가능" );
    }

    break;
case 2:

    for(int i=0; i< lendList.size() ; i++) {
        System.out.println( lendList.get(i));
    }
    break;
case 3:
    System.out.println("반납할 대여번호를 입력하세요");

```

```
String idLend = sc.next();
```

```
//정상인경우
```

```
//대여정보 출력하기
```

```
for( int i=0 ; i< lendList.size() ; i++) {
```

```
    Lend lend = lendList.get(i);
```

```
    if( lend.lendId.equals( idLend)) {
```

```
        lend.setReturnFlag(true);
```

```
        lend.book.setState(1);
```

```
        System.out.println("반납되었습니다");
```

```
        System.out.println("반납정보" + lend);
```

```
        break;
```

```
    }
```

```
}
```

```
break;
```

```
case 4:
```

```
    System.out.println(" 종료");
```

```
    break loop1;
```

```
default:
```

```
}
```

```
}
```

```
}
```

```
}
```

```

package sample;

import java.util.Date;

import java.util.ArrayList;
import java.util.Calendar;
import java.util.Scanner;

public class BookProgram2 {
    ArrayList<Book> books ;
    ArrayList<Lend> lendList;
    Scanner sc = new Scanner(System.in);

    BookProgram2(){
        books = new ArrayList<>();
        lendList = new ArrayList<>();
        loadBookInfo();
    }

    private void loadBookInfo() {
        books.add( new Book("b001" ,"재미있는 책" , "강지훈"));
        books.add( new Book("b002" ,"신나는 책" , "김기엽"));
        books.add( new Book("b003" ,"재미있는 책" , "김수진"));
        books.add( new Book("b004" ,"재미있는 책" , "김연수"));
        books.add( new Book("b005" ,"재미있는 책" , "김예원"));
        books.add( new Book("b006" ,"재미있는 책" , "김종범"));
    }

    private void run() {
        loop1 :while( true) {

            System.out.println("1.대여  2. 조회  3.반납  4. 종료");
            int menu = sc.nextInt();
            switch( menu) {
                case 1:
                    lend();
                    break;
                case 2:
                    searchAll();
                    break;
                case 3:
                    returnBook();
                    break;
                case 4:
                    System.out.println(" 종료");
                    break loop1;
                default:

```

```

        }
    }

}

private void searchAll() {
    for(int i=0; i< lendList.size() ; i++) {
        System.out.println( lendList.get(i));
    }
}

private void lend() {
    System.out.println();
    System.out.println("대여코드입력");
    String lendId= sc.next();

    System.out.print("도서코드입력");
    String bookId = sc.next();

    boolean flag=false;
    // 책 대여가능여부 , 책상태 비교
    Book book=null;
    for(int i=0 ; i< books.size(); i++) {
        book = books.get(i);
        if(book.getBookId().equals(bookId) && book.getState() ==1 ) {
            book.setState(0); // 대여불가능 상태로 변경
            flag=true;
            break;
        }
    }

    if(flag) {
        Date lendDate = getLendDate();
        Date returnDate = getReturnDate();
        Lend lend = new Lend( lendId, lendDate, returnDate, book );
        lendList.add(lend);
        System.out.println( lend);
        System.out.println( "대여되었습니다");

    }else {
        System.out.println("대여불가능" );
    }
}

```

```

private void returnBook() {
    System.out.println("반납할 대여번호를 입력하세요");
    String idLend = sc.next();
    //정상인 경우

    //대여정보 출력하기
    for( int i=0 ; i< lendList.size() ; i++) {
        Lend lend = lendList.get(i);
        if( lend.lendId.equals( idLend)) {
            lend.setReturnFlag(true);
            lend.book.setState(1);
            System.out.println("반납되었습니다");
            System.out.println("반납정보" + lend);
            break;
        }
    }
}

```

```

private Date getReturnDate( ) {
    Calendar cal = Calendar.getInstance();
    // 14일후 , 일 기준 + 14
    cal.add(Calendar.DATE, 14);
    return cal.getTime() ;
}

```

```

private Date getLendDate() {
    Calendar cal = Calendar.getInstance();
    //현재일을 정해진 포맷으로
    return cal.getTime() ;
}

```

```

public static void main(String[] args) {
    new BookProgram2().run();

    // 두 줄로 풀어서 작성할 수 있다
    // BookProgram2 program = new BookProgram2();
    // program.run();
}

```

```

}

```