

파이썬 크롤러를 이용한 나만의 교보문고 만들기

기본설정 방법

1. Mysql 설치

이미 설치되어 있으면 바로 2번으로 넘어갑니다.

2. Bitnami 설치

Download now!

Please sign in to Bitnami to download **WAMP**. It is completely free and gives you access to updates, community support forums, our once-a-month newsletter and more.

Sign In with Google

Sign In with Facebook

Sign In with Github

Sign In with Microsoft

Are you looking for Yahoo Sign In? [Click here](#)

or [Sign in with an existing Bitnami account](#)

Don't have a Bitnami account? [Create one](#)

[No thanks, just take me to the download](#)

Home > Applications > Infrastructure > WAMP

WAMP 7.3.16 4.5 (24 ratings) php.net

Updated 7 days ago Version 7.3.16

DEPLOYMENT OFFERING

On my computer

Win / Mac / Linux

BITNAMI WAMP STACK INSTALLERS

Bitnami native installers automate the setup of a Bitnami application stack on Windows, Mac OS and Linux. Each installer includes all of the software necessary to run out of the box (the stack). The process is simple: just download, click next-next-next and you are done!

Try, test and work with the application in your local environment

Our installers are self contained. They don't interfere with other software on your system

ADDITIONAL RESOURCES

- Documentation
- Application credentials
- Readme
- Changelog
- Support

DOWNLOAD INSTALLERS FOR BITNAMI WAMP STACK

Download for Windows 64-bit

7.3.16-2 Checksum

Other versions

Version	Arch	Size	Checksum
WAMP 7.2.29-2	64-bit	275 MB	sha256
WAMP 7.4.41 (Development)	64-bit	275 MB	sha256

Download for Windows 64-bit

Download for Windows 64-bit

<https://bitnami.com/stack/wamp/installer> 로 접속해서 비트나미 WAMP를 설치합니다.

설치는 계속 next 누르고 비밀번호는 000000으로 설정합니다.

3. PHP파일 준비

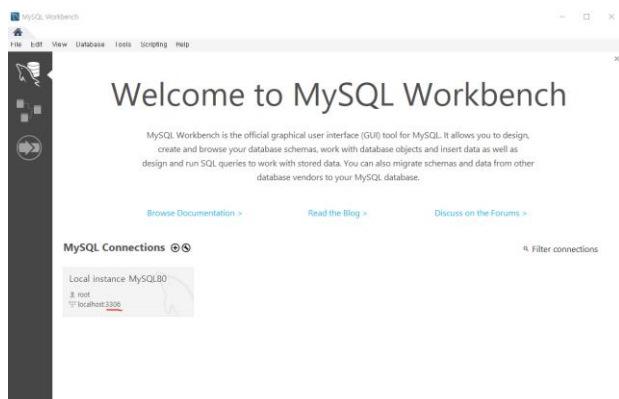
Bookstore_php.zip 파일을 압축을 풀어서 해당 폴더를 잘라낸뒤 C:\Bitnami\wampstack-7.3.16-0\apache2\htdocs 에 붙여 넣습니다.

그리고 php 파일들인 order_list, product, product-detail, sign_up, purchase, admin, deliver 의

```
$pdo = new PDO('mysql:host=localhost;dbname=bookstore;  
charset=utf8', 'root', '4885');
```

부분을 찾아 마지막 4885부분을 본인의 mysql 비밀번호로 바꿔줍니다.

4. Mysql 데이터 Import

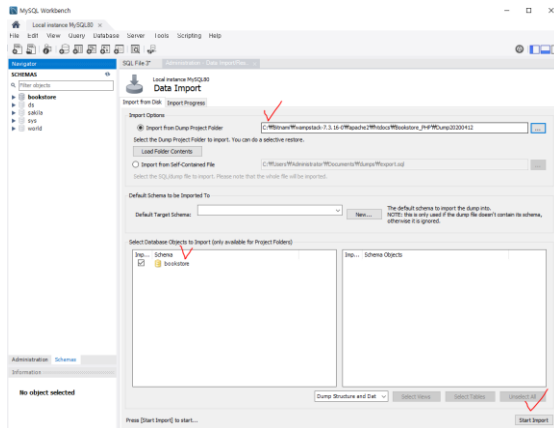


mysql workbench를 실행시켜 **포트번호가 3306인 계정**으로 로그인합니다. 사진처럼 **Local instance MYSQL**이라고 써 있으면 기존에 있던 mysql에 등록한 비밀번호를, **WAMP**라고 써 있으면 000000을 입력해서 로그인합니다.

기존에 만들어 둔 bookstore DB가 **있으면** drop database bookstore; 를 실행시킨 뒤 create database bookstore; use bookstore;를 실행시키고

기존에 만들어 둔 bookstore DB가 **없으면** create database bookstore; use bookstore; 를 실행시킵니다.

그 뒤에 상단의 **Server**탭의 **data import**를 실행한 뒤



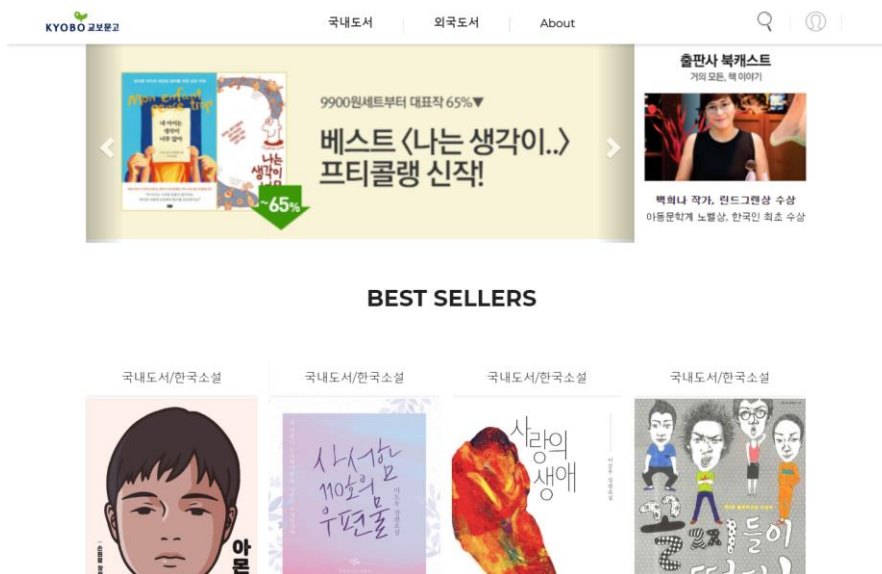
import from dump folder에서 **Bookstore_PHP**아래의 **Dump** 폴더를 선택한 뒤에 좌하단에 bookstore가 생겼는지 확인한 뒤 우하단의 start import를 클릭 후 성공적으로 import 되었는지 확인합니다.

book 테이블에는 각 책들의 데이터들이 들어가 있고, mem 테이블에는 회원들의 정보, orders에는 주문내역이 저장되어 있습니다.

5. 설정 완료 확인

브라우저를 켜고 주소에 http://localhost/BookStore_php/index.php 를 입력합니다.

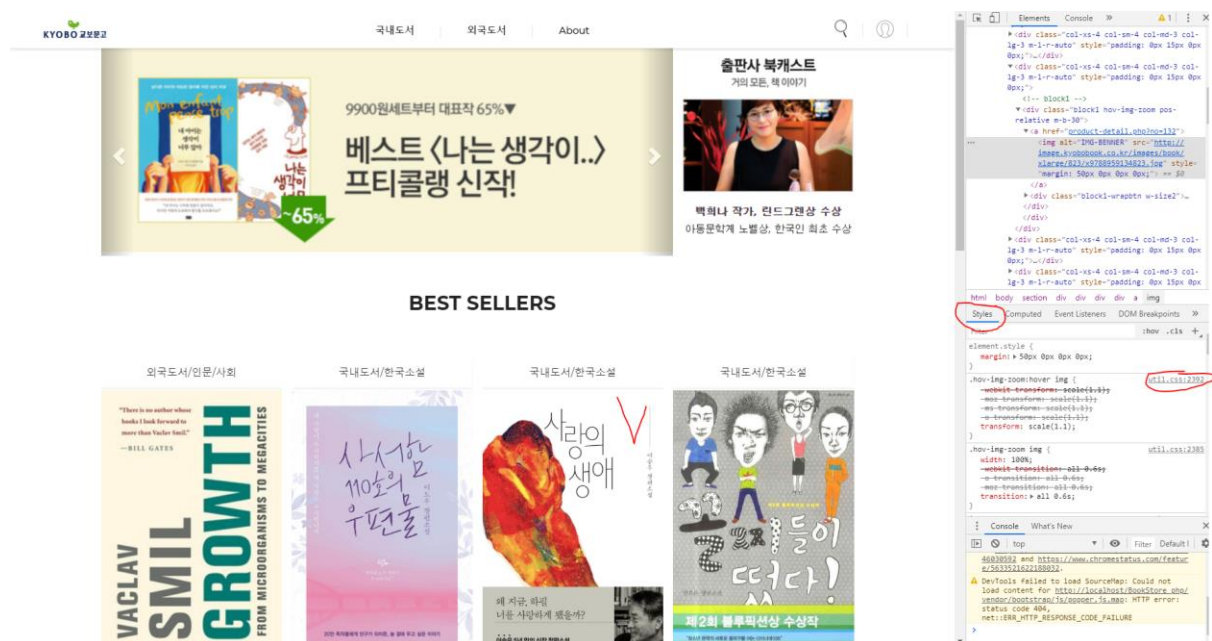
다음과 같은 창이 나오면 모든 설정을 성공적으로 완료하신 겁니다.



간단한 팁

이 프로젝트는 CSS 파일이 하나만 있지도 않고 php 파일에 적혀있는 것도 아니라서 **해당 웹 요소에 어떤 CSS가 적용되었는지 찾기가 굉장히 불편한데** 이는 크롬의 **개발자 모드**를 사용하면 비교적 쉽게 찾을 수 있습니다.

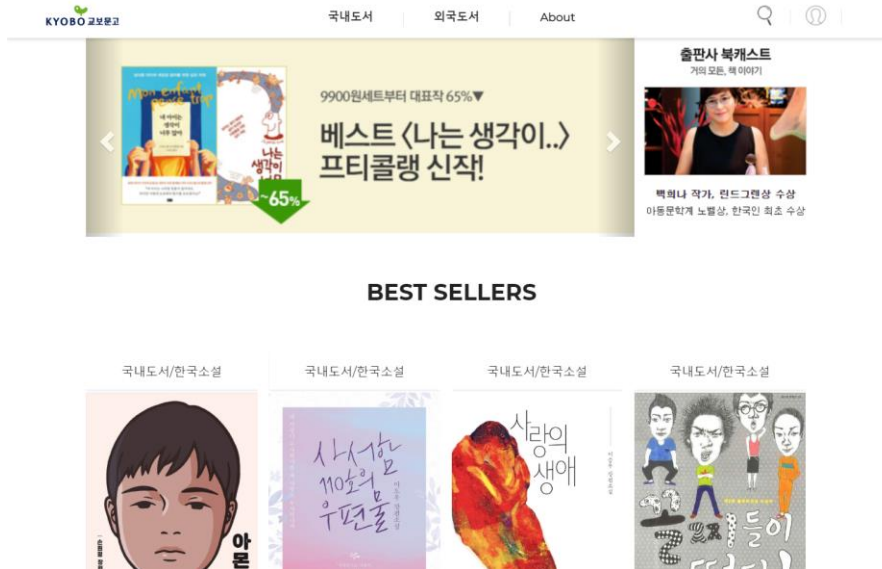
ctrl + shift + c를 누르거나 **F12**를 누르면 **개발자 모드**가 켜집니다



여기에서 예를 들어 붉은 색으로 V 표시한 이미지에 적용된 CSS들을 알고 싶다면 우측 개발자 모드 창의 **중단에 Style이 클릭되었는지 확인하면 하단에 적용된 CSS들의 종류가 전부** 나오고, **해당 CSS가 어느 파일인지** 또한 나옵니다. 따라서 해당 요소의 CSS를 수정하고 싶으면 모든 CSS 파일은 프로젝트 폴더의 CSS에 있기 때문에 CSS 폴더에서 필요한 CSS파일을 찾아서 수정하면 됩니다.

코드 설명

1. index.php



우리 사이트의 메인 페이지로, 제일 상단의 메뉴바 부분은 **header.php**, 제일 하단의 footer부분은 **footer.php** 파일로 되어있으며 body부분은 상단의 **이벤트 배너 캐러셀**과 **각 장르별 베스트셀러**들을 보여줍니다.

겹치는 내용이기엔 최상단에 있는 header.php를 설명하고 footer.php를 설명한 뒤에 다시 index.php를 설명하겠습니다.

2. header.php

header.php는 우리 사이트의 **상단 메뉴바를 나타내는 코드**인데 각 페이지마다 같은 메뉴바를 사용하기엔 이렇게 **별도로 php파일로 만든 뒤 코드를 include**해줘야 중복으로 코딩할 필요가 없습니다.

1~6번줄은 로그인 여부 확인을 위해 **세션 데이터**를 받아오기 위해 **session_start()** 함수를 실행하고, 혹시나 모를 에러 발생시 에러내역을 보기 위해 3,5번줄을 작성했습니다.

7~13번줄은 상단바의 **검색기능을 동작 시키기 위한 자바스크립트 함수**입니다. 검색버튼을 클릭할 시 onclick으로 실행되는 함수이고 값을 받아와 제품 정보를 보여주는

product.php에 get방식으로 검색어와 함께 보냅니다

16번줄부터 102번줄까지는 **PC크기의 화면에서의 상단바에 대한 코드**입니다.

19번줄은 로고 이미지입니다. 클릭하면 메인화면으로 가게 a태그를 씌워두고 이미지는 images 폴더 아래에 있습니다.

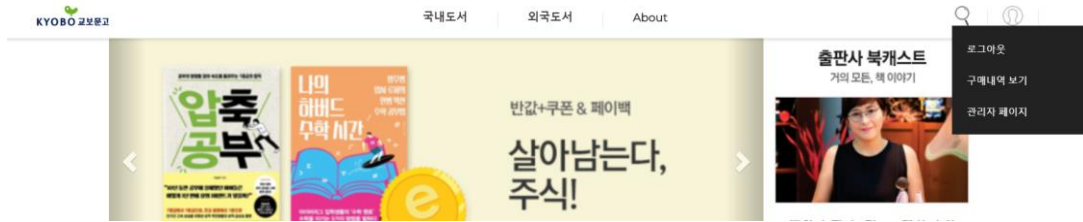


23~45번줄 **main_menu** 클래스 **ul**태그는 **상단바의 메뉴들**을 나타내는 리스트이고, 그 하단의 **sub_menu** 클래스 **ul**태그의 리스트들은 **main_menu**에 hover하면 drop-down되게 css를 작성해 두었으며 각각의 리스트를 클릭하면 해당 카테고리의 도서들을 보여주는 **product.php**로 이동하게 하였습니다.

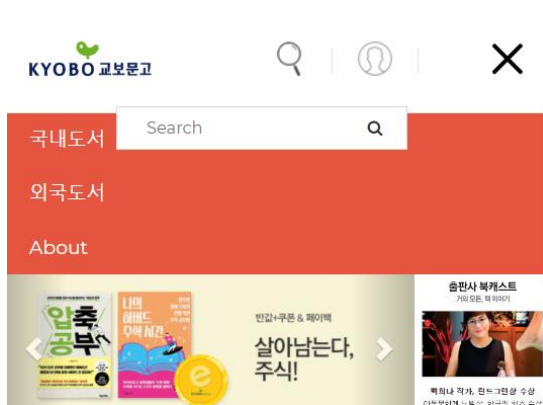
49~100번줄은 상단바의 우측에 있는 검색 아이콘과 옆의 로그인 메뉴에 대한 코드입니다.

51~ 65번줄은 **검색버튼**에 관한 코드이고 52번 줄의 **img**가 **검색버튼**이고 이를 누를 경우 평소에는 **hidden** 상태여서 안보이던 53~64번줄이 나타나게 합니다. 나타난 창은 input창과 옆의 작은 돋보기모양 검색 버튼이고 input창에 검색하려는 책제목을 입력하고 작은 돋보기 버튼을 누르면 위에서 설명한 **search함수**가 작동되어 검색창으로 이동합니다.

66번줄은 아이콘 사이의 | 모양을 나타냅니다.



67~98번줄은 로그인 메뉴에 대한 php코드인데 session데이터가 없으면(93~98번줄) 로그인 상태가 아니기에 누르면 로그인 화면으로 이동되는 메뉴를 붙여넣고, session데이터가 있으면(80~91번줄) 로그인 된 상태이기에 마우스를 hover하면 로그아웃과 구매내역을 볼 수 있는 메뉴를 붙여넣고, 마지막으로, id가 admin, 즉, 관리자이면(68~80번줄) 관리자 메뉴를 붙여넣습니다. 관리자 페이지에서는 유저들이 구매한 내역들에 대해 배송처리(배송완료를 눌러 DB에서 사라지게 할 수 있습니다)



107~156번줄은 모바일에서 본 상단바의 모습을 나타내는 코드입니다.

115~127번줄이 검색부분 코드이며 PC버전에서 설명한 것과 동일합니다.

130~134번줄은 로그인 메뉴 관련이며 로그인 상태에 따라 로그인/로그아웃 기능을 실행하게 만들어주는 코드입니다.

144~159번줄은 우측의 햄버거버튼을 누르면 나오는 메뉴에 관한 코드이며 PC버전의 1차 메뉴리스트만 나오게 구현해줬습니다.

header.php 설명완료

3. footer.php



footer는 각 화면의 최 하단에 사이트에 대한 정보 및 간단한 사이트맵을 보여주는 부분입니다.

특별히 복잡할 내용은 없고 각 도서 장르별로 **product.php**로 옮겨가게 a태그를 씌워 두었습니다.

footer.php 설명완료

4. index.php

다시 index로 돌아와서 설명을 하겠습니다.

50번째줄에서 앞에서 설명한 **header.php**를 **index.php**에 불러오는 php코드입니다.

52번부터 104번까지가 실질적인 index.php의 코드입니다.

55~71번 줄이 메인페이지의 **캐러셀**(사진이 바뀌는 배너) 관련 코드입니다.

55번줄은 **캐러셀의 크기**를 결정하는데 class를 보시면 **col-xs-9** 이런 식으로 가운데만 다르고 col과 9는 같게 써 있는 것들이 있는데 이 클래스들은 **화면 사이즈에 따라 해당 태그의 사이즈를 반응형으로 바꿔주는 클래스**이며 화면을 **12등분**한 사이즈에서 얼마나 차

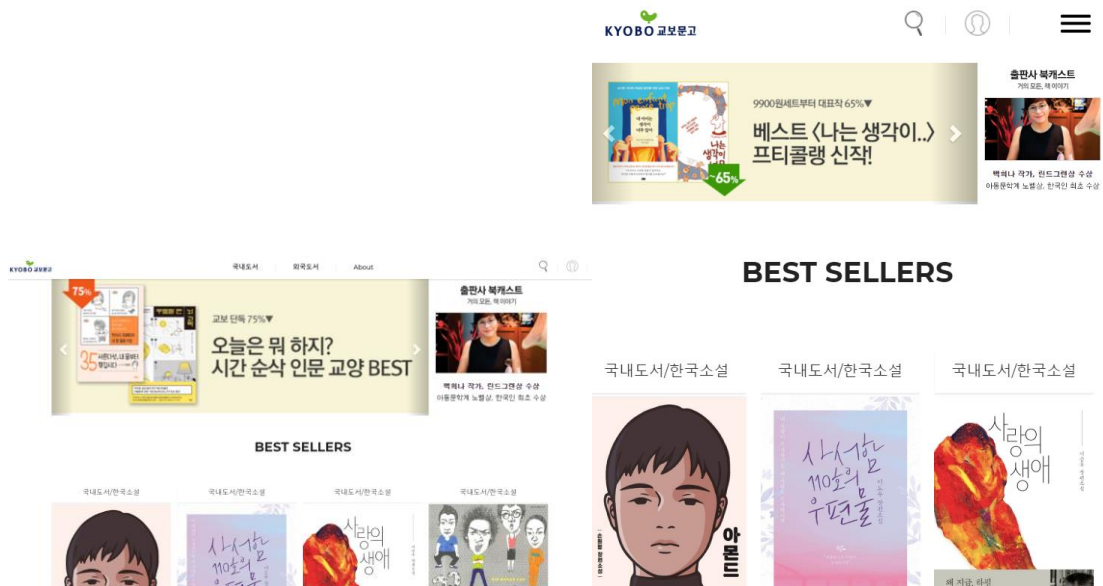
지할자를 나타냅니다. 즉, 9이기에 화면 크기가 바뀌어도 화면의 3/4를 계속 유지합니다.

57번줄의 **div**로 감싸져있는 58~66번줄의 **div**들이 움직이는 배너들을 나타냅니다. 우리 사이트는 이벤트 페이지는 특별히 없기에 배너를 누르면 제품들을 볼 수 있는 **product.php**로 보내기로 해놓았고 각 **div**의 **a**태그에 감싸진 **img**가 각각의 배너 이미지입니다.

69~70번줄은 캐러셀의 사진을 이전, 다음으로 넘기는 버튼에 관한 코드입니다.

72번줄은 캐러셀 옆에있는 이미지 관련 코드입니다. 캐러셀이 3/4를 차지하기에 이 이미지는 1/4를 차지하게 하기위해 **col-size-3**으로 클래스를 지정해줍니다.

75번줄에서는 구역이 나뉘를 보여주는 **h2**태그를 삽입합니다. 75번줄 위아래로 있는 73,74,76번줄 **div**는 **h2**위아래를 띄우기 위해(**style height** 사용) 사용했습니다.



index.php의 PC버전 화면과 모바일 화면

78~102번 코드는 베스트셀러들을 보여주기 위한 코드입니다.

79~84번코드는 **mysql과 연동을 위한 코드**입니다. 79~81번줄 코드로 **DB에 연결**합니다.

81번코드로 **DB에서 데이터를 가져올 sql**을 작성합니다. 가져올 데이터가 가장 많이 팔린 책 12권이기에 각 **베스트셀러의 썸네일, 1차 카테고리, 3차카테고리, 책번호를 가져오기 위한 sql**입니다.

83,84번줄 코드로 **sql을 실제로 실행**시킵니다.

86번줄의 **while문으로 select문의 결과**를 뽑아냅니다. **fetch로 결과**를 한 줄씩 뽑아냅니다.

87번줄은 각 책들의 정보를 감쌀 최상위 **div**인데 class가 기존과 다르게 sm이하 사이즈와 md이상의 사이즈가 다른데, **PC버전에서는 4개씩, 모바일에서는 3개씩** 나오게 하기 위함입니다.

90번은 **도서를 클릭했을 때 해당 도서의 상세정보**로 넘어가게하기 위해 **product-detail.php**로 이동시키는 코드입니다.

92번코드는 **이미지태그에 도서의 썸네일 주소**를 넣는 코드입니다.

94~99번 코드는 **베스트셀러 각각의 도서 위의 카테고리부분**을 나오게 하는 코드입니다.

a태그에 상세정보로 가는 url을 넣어주고, 내용으로 카테고리들을 /로 구분해서 넣습니다.

index.php 설명완료

5. about.php

about.php는 상단바에 있는 about을 클릭하면 접근할 수 있는 페이지로, 교보문고의 교보문고란 페이지를 참고했으며, **텍스트만 있는 페이지라 설명은 생략**하겠습니다.

6. login.html

The image displays two versions of a login form. The left version is a simplified form with two input fields: the first contains the text 'admin' and the second contains four dots '....'. Below these are two green buttons: '체크하기' (Check) and '로그인 하기' (Login). At the bottom, there is a link: '아직 계정이 없으신가요? 계정 만들기' (Don't you have an account yet? Create account). The right version is a more detailed form with five input fields: 'admin', a password field with four dots, 'Email', '주소' (Address), and '핸드폰번호' (Phone number). Below these are two green buttons: '체크하기' (Check) and '계정 만들기' (Create account). At the bottom, there is a link: '아직 계정이 없으신가요? 로그인 하기' (Don't you have an account yet? Login).

login.html은 로그인 및 회원가입을 하는 페이지입니다. php문이 필요 없기에 html 파일입니다. '계정 만들기'와 '로그인 하기'로 서로 토글이 가능하며 로그인, 회원가입 모두 처음에는 로그인하기/회원가입 버튼이 비활성화되어 있어 사용불가능하고 '체크하기' 버튼을 눌러서 정상적으로 입력을 했는지 확인해야 아래의 로그인하기/회원가입 버튼이 활성화됩니다.

9~22줄 코드는 회원가입을 위한 칸들을 전부 채웠는지 확인하는 자바스크립트 함수입니다. 각 폼들의 값을 ID로 가져와서 전부 값을 채웠을 경우에만 버튼 비활성화를 풀고, 아닌 경우 빈칸없이 모든 칸을 입력하라고 합니다.

24~34줄 코드는 같은 맥락의 로그인 체크용 함수입니다.

html부분은 특별한건 없고 계정만들기/로그인하기 버튼이 disabled로 설정되어 있으며 각각의 form이 login.php와 sign_up.php로 post방식으로 전달되게 되어있습니다.

login.html 설명완료

7. sign_up.php

회원가입을 진행하는 php입니다. login.html에서 post로 전달받은 정보를 통해 회원가입을 진행시킵니다.

5~10번줄 코드로 DB와 연결하고 12번 코드로 sql을 설정합니다.

`$_POST['id']`는 POST방식으로 전달받은 데이터 중 id 데이터의 값을 가져오고, `isset` 함수는 해당 데이터가 존재하는지 확인합니다.

12번 줄의 `:id` 같은 식으로 입력된 것들은 15~19번줄 `bindValue` 함수를 통해서 값을 대체해줍니다.

DB에 쿼리를 날리는데 문제가 없었으면 34~35번줄 자바스크립트가 실행됩니다.

각각 회원가입 성공 alert와 login화면으로 보내는 자바스크립트입니다.

sign_up.php 설명완료

8. login.php

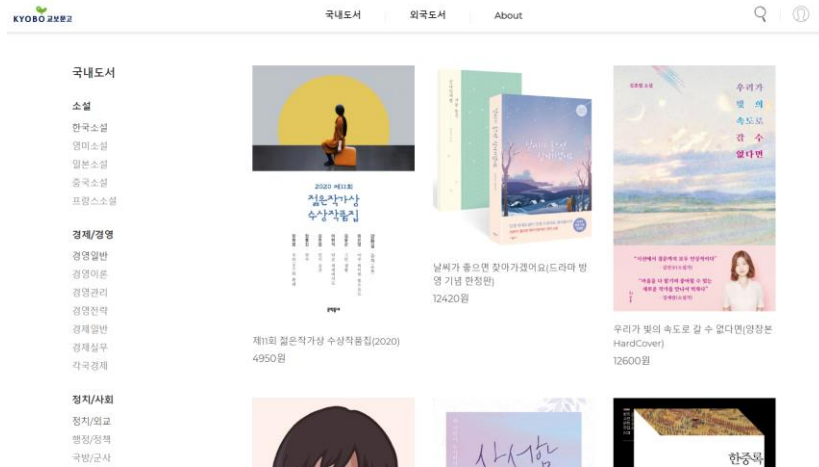
`login.php`도 `sign_up.php`와 비슷한 맥락입니다. `post`로 전달받은 ID, PWD를 DB에서 확인 후 존재하면 로그인, 없으면 로그인 페이지로 돌려보내는 `php`입니다.

12~14번줄 코드로 세션데이터를 확인해서 이미 로그인된 유저인지 확인하고 alert창을 띄웁니다.

15~43번줄 코드는 ID,PWD를 DB에서 조회해서 해당 유저가 존재할 경우 로그인, 없을 경우 해당 유저가 없다고 alert창을 띄우는 코드입니다.

login.php 설명완료

9. product.php



product.php는 상단바나 footer의 국내도서, 외국도서 등의 메뉴를 눌렀을 시 해당 카테고리
고리의 책들을 보여주는 페이지입니다. **get** 방식으로 데이터를 받아야하기 때문에 **url**뒤
에 카테고리나 **search** 변수가 붙어있습니다.

51~318번줄 코드는 좌측에 있는 카테고리들을 위한 바입니다. 특별할 것 없이 각 카테고리
고리들을 리스트로 나열한 것입니다.

328번 줄은 **product.php**는 검색으로 들어왔을 경우 **get**방식으로 **search**를 넘겨주기에
해당 변수가 있으면 검색 결과를 도출해주기 위해 확인하는 코드입니다.

330번 줄의 **sql**은 검색의 결과가 너무 많을 경우 로드하는데 조금 오래걸려서 최대 100
개까지만 로드하게 해놓았습니다.

334번의 **while**문은 검색된 레코드 마다 도서를 보여주는 **html**을 출력하는 코드인데 336
번줄을 보시면 **PC**화면에서는 4사이즈(줄 당 3개), **핸드폰**에서는 12사이즈(줄 당 1개)로
보이게 설정을 해 두었으며, 썸네일, 책제목을 누르면 상세정보로 이동하게 **a**태그를 설정
하였습니다.

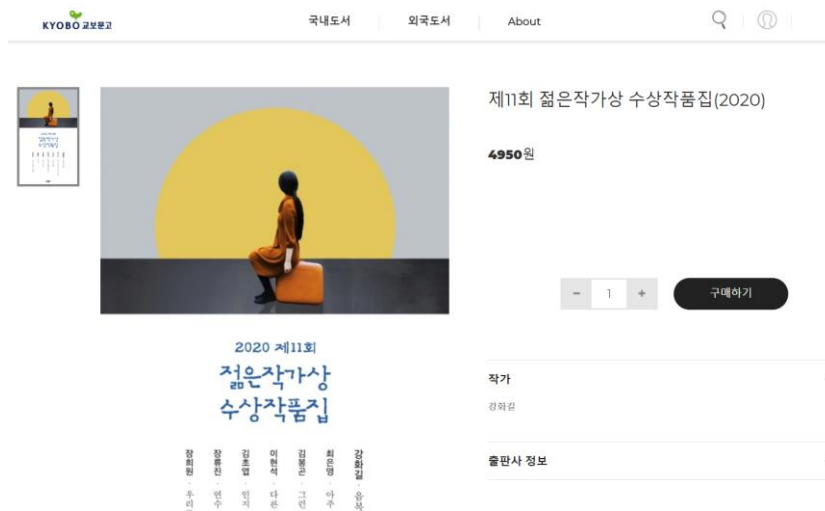
356번줄 **elseif**문은 아무런 **get**변수없이 주소에 **product.php**만 입력했을 시 오류를 피하
기위해 **get** 변수가 없어도 default로 국내도서/소설/한국소설 의 데이터를 보여주는 코

드립니다.

385번줄의 else문은 검색으로 들어온 것도 아니고 **get변수가 정상적으로 있는 경우**이기
에 각각의 카테고리에 맞는 도서 정보들을 출력해주게 하는 코드입니다.

product.php 설명완료

10.product-detail.php



product-detail.php는 특정 도서의 상세정보를 조회하고 해당 도서를 구매할 수 있는 페이지로 get방식의 변수 no(해당 책의 번호)를 사용합니다.

38~54번의 자바스크립트는 구매하기 버튼을 눌렀을 시 동작하는 함수 buy()에 대한 정의인데 로그인 된 상태이면 구매를 진행하는 purchase.php로 id, 가격, 수량, 책제목 변수들을 함께 보내고, 로그인 된 상태가 아니면 로그인을 하라는 알림창과 로그인창으로 리다이렉트합니다.

56~65번 코드는 DB와 연동해서 no번 책의 데이터를 가져옵니다.

74번 코드가 좌측의 작은 도서 썸네일, 76번 코드가 오른쪽의 큰 이미지이기에 각각에 thumb 데이터를 출력해줍니다.

85번은 **title**을, 90번은 **price**를 출력하는 코드이며 97~99번과 103~105번코드는 수량을 +/- 하는 버튼들의 코드이고 111~113번 코드가 구매하기 버튼에 대한 코드이며, 클릭 시 앞에서 정의한 자바스크립트 함수 **buy()**를 실행합니다.

121~133번 코드는 구매하기 버튼 아래의 작가정보 보기 관련 코드이며 135~146번 코드가 출판사 정보 코드입니다.

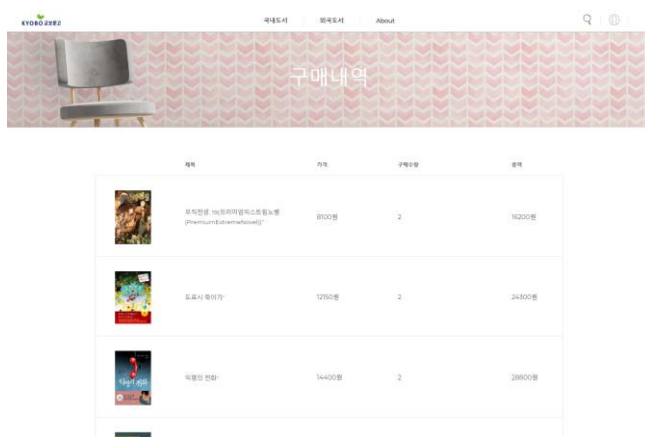
product-detail.php 설명완료




11.purchase.php

product-detail.php에서 구매하기 버튼을 누르면 get방식으로 책번호, 책제목, 구매자id, 수량이 purchase.php로 전송이 되고 이 데이터들을 이용해서 DB의 orders 테이블에 데이터를 추가하는 php파일입니다.

앞에서 설명했던 내용들이기에 자세한 설명은 생략하겠습니다.

12.order_list.php



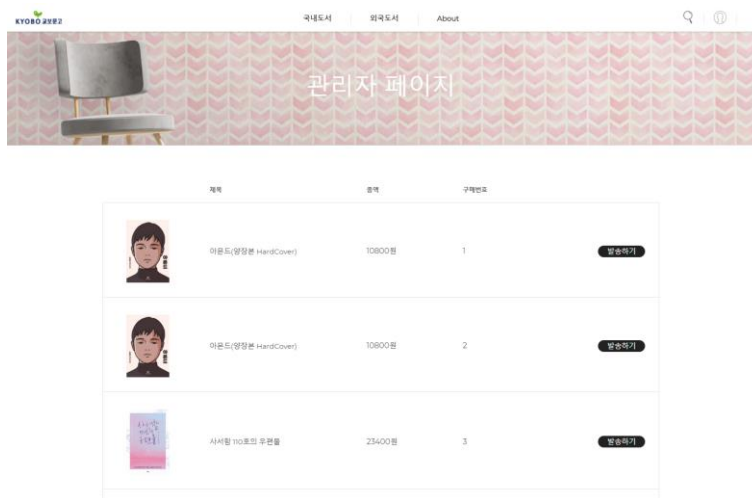
책자	가격	구매수량	총액
 유치전생 (유치전생의 유혹) ([Yuchijeonseong (Yuchijeonseong-ui Yuhok)])	8100원	2	16200원
 도토리 목탁가	12100원	2	24200원
 유령의 전당	14400원	2	28800원

로그인한 상태의 상단바에서 접근 가능한 구매내역 페이지입니다. 해당 유저의 구매한 도서 내역을 최신순으로 출력해줍니다.

52~58번이 해당 테이블의 칼럼이 무엇인지를 나타내고 68~80번이 각 레코드들을 나타내는 html을 생성합니다.

order_list.php 설명완료

13.admin.php



admin 계정으로 로그인했을 때 상단바에서 접근할 수 있는 관리자 페이지입니다.

배송하기를 눌러서 해당 주문을 완료처리 할 수 있습니다.

81번줄에 배송하기 버튼의 a태그에 **deliver.php**로 해당 제품의 주문번호를 **get**방식으로 보낸다는 점만 빼면 전반적으로 코드가 **order_list.php**와 같습니다.

admin.php 설명완료

14.deliver.php

admin.php에서 배송하기 버튼을 누르면 **order_no**를 **get**방식으로 **deliver.php**에 보내는데 이 php는 **order_no**를 이용해서 해당 번호의 주문을 완료처리(삭제)하는 **php**입니다.

그동안 설명한 php파일들과 흡사하기에 자세한 설명은 생략하도록 하겠습니다.