


프로그래밍언어응용_보고서_정선미

2022.04.06

문항1

APP

searchImg

ID 726790

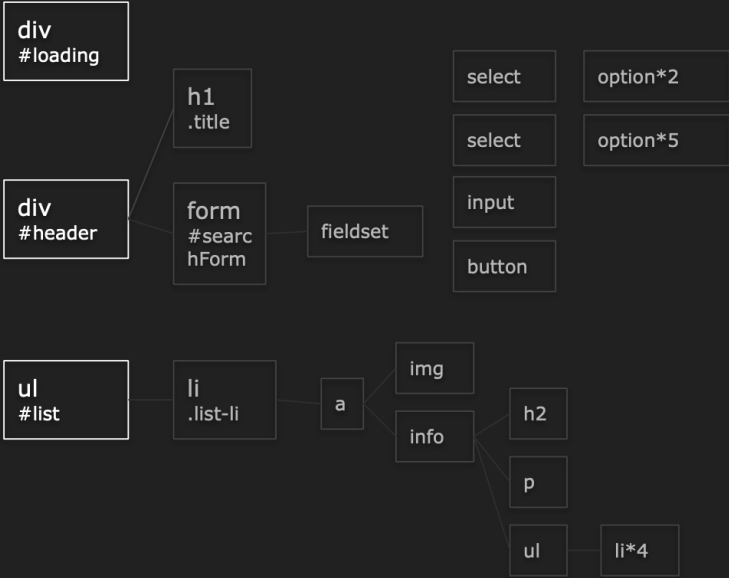
OWNER

앱 키

네이티브 앱 키	45e53247059718b5b04d2253e5fe7f59
REST API 키	9b3c76b409424925624b69c50222cfa3
JavaScript 키	217d488a4557c7647c26082ce9933ced
Admin 키	a3dcfa5741a1fc5008a3f09b4852caca

문항2

02.
DOM
DOM Structure



문항3

카카오 책 검색

정확도순 ▾ 30건 ▾ 자바스크립트 검색



Do it! HTML+CSS+자바스크립트 웹 표준의 정석

웹의 기본을 한 권으로 끝낼 수 있는 책이 탄생했다! 8년 연속 웹 분야 도서 1위인 《Do it! HTML5+CSS3 웹 표준의 정석》의 고경희 저자는 'HTML, CSS, 자바스크립트를 한 권으로' 배우고 싶다는 많은 독자의 요구를 모아 최신 경향을 반영한 이 책을 집필했다. 웹 개발을 시작하는 입문자도 쉽게 실습할 수 있고, 중요한 문법만 다시 공부하고 싶은 중·고급자에게도 도움이 된다. 친절한 설명은 기본! 핵심만 쏙쏙 뽑아서 실...

고경희 | 이지스퍼블리싱 | 30000 | 27000 |



코딩만 따라 해도 웹페이지가 만들어지는 HTML+CSS+자바스크립트

이 책은 HTML/CSS/자바스크립트를 지루한 문법 위주로 알려주는 것이 아니라 책을 따라 해보면서 바로 결과물을 확인해보고 코딩이 실제로 어떻게 구현되는지 이해할 수 있게 구성한 책이다. 책에 소개된 방법으로 코드를 실행하기만 하면 독자의 컴퓨터에서 바로 웹 페이지가 만들어진다. 이제 문법을 외우는 것이 아니라 이해해 보자!

반병현 | 생능북스 | 24000 | 21600 |



자바스크립트(프로그래밍 입문자를 위한)

『프로그래밍 입문자를 위한 자바스크립트』는 자바스크립트에 막 입문한 사람들을 위해 기초내용을 꼼꼼히 설명한 책이다. 자바스크립트 기초부터 변수와 상수, 연산자, 배열 등 프로그래밍에 필요한 내용을 수록했다.



Spring Boot

Mark Heckler | O'Reilly Media | 70080 | 67980 |



Hearing

Hejinian Lyn, Scalapino Leslie | Litmus Press | 23610 | 24190 |



안드로이드 with Kotlin 앱 프로그래밍 가이드(2판)(애프터스킬 시리즈)

"이런 앱이 있다면 좋지 않을까?"라는 가장 기본적인 아이디어로 시작되는 앱 개발! 하지만 알아야 할 자바(Java) 문법과 개념들로 인해 실제 출시까지의 앱 개발 과정은 생각보다 쉽지 않습니다. 더구나 보다 간결한 문법과 쉬운 기능 구현을 목표로 JetBrains사에서 발표된 코틀린(Kotlin)이 특유의 장점을 바탕으로 구글 안드로이드 공식 개발 언어로 채택되면서, 안드로이드 개발을 막 시작하려는 분들에게는 자칫 부담...

안귀정 | 아이코스 | 40000 | 36000 |

Request_Ajax/시험/index.html

계정 [

127.0.0.1:5500 내용:

검색어를 입력하세요.

확인

HTML

```
<!DOCTYPE html>
<html lang="ko">

<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>카카오 책 검색</title>
  <link rel="stylesheet" href="./main.css">
</head>

<body>
  <div id="loading"></div>

  <div id="header">
    <h1>카카오 책 검색</h1>
    <form id="searchForm">
      <fieldset>
        <select name="sort" id="sort">
          <option value="accuracy">정확도순</option><!--기본값-->
          <option value="latest">발간일순</option>
        </select>
        <select name="size" id="size">
          <option value="10">10건</option><!--기본값-->
          <option value="20">20건</option>
          <option value="30">30건</option>
          <option value="40">40건</option>
          <option value="50">50건</option>
        </select>
        <input type="search" id="query" placeholder="책을 검색해보세요!">
        <button type="submit">검색</button>
      </fieldset>
```

```
    </form>
  </div>

  <ul id="list">
  </ul>

  <script src="https://cdn.jsdelivr.net/npm/axios/dist/axios.min.js">
</script>
  <script src="./main.js"></script>
</body>

</html>
```

CSS

```
* {
  margin: 0;
  padding: 0;
  box-sizing: border-box;
}

a {
  text-decoration: none;
  color: inherit;
}

body {
  background: #fff;
  color: #333
}

ul {
  list-style: none;
}

#loading {
  display: block;
  position: absolute;
  left: 50%;
  top: 50%;
  transform: translate(-50%, -50%);
  z-index: 99999;
  display: none;
  margin: 40px auto;
  width: 8px;
  height: 8px;
  background-color: rgba(62, 74, 249, 0.5);
  ;
  box-shadow: -14px 0px 0px rgb(119, 190, 232);
  border-radius: 50%;
  -webkit-animation: circle_classic 1s ease-in-out infinite alternate;
```

```
-moz-animation: circle_classic 1s ease-in-out infinite alternate;
animation: circle_classic 1s ease-in-out infinite alternate;
}

@-webkit-keyframes circle_classic {
  0% {
    opacity: 0.1;
    -webkit-transform: rotate(0deg) scale(0.5);
  }

  100% {
    opacity: 1;
    -webkit-transform: rotate(360deg) scale(1.2);
  }
}

@-moz-keyframes circle_classic {
  0% {
    opacity: 0.1;
    -moz-transform: rotate(0deg) scale(0.5);
  }

  100% {
    opacity: 1;
    -moz-transform: rotate(360deg) scale(1.2);
  }
}

@keyframes circle_classic {
  0% {
    opacity: 0.1;
    transform: rotate(0deg) scale(0.5);
  }

  100% {
    opacity: 1;
    transform: rotate(360deg) scale(1.2);
  }
}

#loading.active {
  display: block;
}

#header {
  margin-bottom: 12px;
}

#header h1 {
  text-align: center;
  padding: 16px 0;
  font-size: 24px;
}
```

```
#searchForm fieldset {
  border: 1px solid rgb(0 0 0 / 0.3);
  text-align: center;
  padding: 16px 0;
}

#list>li {
  margin-bottom: 12px;
  padding: 12px;
  border-bottom: 1px dotted rgb(66, 66, 66)
}

#list li a {
  display: flex;
  align-items: center;
}

#list li a img {
  display: block;
  width: 120px;
  max-height: 174px;
  overflow: hidden;
  object-fit: cover;
  margin-right: 16px;
}

#list li a .info {
  width: calc(100% - 136px);
}

.info h2 {
  font-size: 18px;
  overflow: hidden;
  text-overflow: ellipsis;
  text-align: left;
  word-wrap: break-word;
  display: -webkit-box;
  -webkit-line-clamp: 1;
  -webkit-box-orient: vertical;
  margin: 0 0 10px;
}

.info p {
  font-size: 14px;
  line-height: 1.4;
  height: 4.2em;
  overflow: hidden;
  text-overflow: ellipsis;
  text-align: left;
  word-wrap: break-word;
  display: -webkit-box;
  -webkit-line-clamp: 3;
  -webkit-box-orient: vertical;
  margin-bottom: 10px;
}
```

```

}

.info .detail {
  display: flex;
  align-items: center;
}

.info .detail li {
  margin-right: 10px;
  padding-right: 10px;
  border-right: 1px solid #333;
  line-height: 1;
  font-size: 12px;
}

```

JS

```

/*
@Filename : main.js
@author : Jeong Seon-mi
@Description : 카카오 책 검색 api 연결

< 파일 구성 >
- setting key, parameter
- 검색폼의 submit 이벤트(신규검색)
- 스크롤 이벤트(추가검색)
- Ajax요청 후 결과를 화면에 HTML로 출력하는 함수

*/

/* setting key, parameter */
// kakao rest api key
const KAKAO_REST_KEY = '9b3c76b409424925624b69c50222cfa3';

// 페이지 번호
let currentPage = 1;

// 검색어
let queryKeyword = null;

// 마지막페이지인지 검사
let isEnd = false;

// 결과 문서 정렬 방식
let selectSort = 'accuracy';

// 한 페이지에 보여질 문서 수
let selectSize = 10;

/* 검색폼의 submit 이벤트(신규검색) */

```



```
document.querySelector('#searchForm').addEventListener('submit', e => {
    e.preventDefault();

    // 입력된 검색어를 가져온다.
    const queryField = document.querySelector('#query');
    queryKeyword = queryField.value.trim();

    // 검색어가 입력되지 않은 경우에 대한 예외처리
    if (!queryKeyword) {
        alert('검색어를 입력하세요. ');
        queryField.focus();
        return;
    }

    // 선택된 셀렉트 박스 값을 가져온다.
    const sizeBox = document.querySelector('#size')
    const sizeBoxSelected =
    Number(sizeBox.options[sizeBox.selectedIndex].value);

    // 선택된 셀렉트 박스 값을 가져온다.
    const sortBox = document.querySelector('#sort')
    const sortBoxSelected = sortBox.options[sortBox.selectedIndex].value;

    // 신규검색
    currentPage = 1;
    selectSize = sizeBoxSelected;
    selectSort = sortBoxSelected;
    get_book_search();
})

/* 스크롤 이벤트(추가검색) */
window.addEventListener('scroll', e => {
    // 마지막 페이지이거나 이미 로딩바가 화면에 표시되고 있다면 처리 중단
    if (isEnd ||
    document.querySelector('#loading').classList.contains('active')) {
        return ;
    }

    // 스크롤바의 Y좌표
    const scrollTop = window.scrollY;
    // 웹브라우저의 창 높이
    const windowHeight = window.screen.availHeight;
    // HTML 문서의 높이
    const documentHeight = document.body.scrollHeight;

    // 스크롤바의 반동 효과를 고려해서 scrollTop + windowHeight가 실제 화면 크기보다 커 질
    수도 있다.
    if (scrollTop + windowHeight >= documentHeight) {
        // 2페이지 이후는 추가 검색
        currentPage++;
        get_book_search();
    }
});
```

```
/* Ajax요청 후 결과를 화면에 HTML로 출력하는 함수 */
async function get_book_search() {
  // 로딩바 객체
  const loading = document.querySelector('#loading');

  // 로딩바 화면에 표시하기
  loading.classList.add('active');

  // 검색 결과가 표시될 영역
  const list = document.querySelector('#list');

  // 1페이지에 대한 요청일 경우 기존에 표시되고 있던 검색결과가 있다면 삭제한다.
  if (currentPage == 1) {
    Array.from(list.querySelectorAll('.list-li')).forEach((v, i) => {
      list.removeChild(v);
    })
  }

  // 검색결과를 저장할 빈 변수
  let json = null;

  // 데이터 받아오기
  try {
    json = await axios.get(`https://dapi.kakao.com/v3/search/book`, {
      params: {
        query: queryKeyword,
        page: currentPage,
        sort : selectSort,
        size : selectSize,
      },
      headers: {
        Authorization: `KakaoAK ${KAKAO_REST_KEY}`
      }
    });
    // 응답 결과 확인
    console.log(json)
  } catch (e) {
    console.error(err);
    alert('요청을 처리하는데 실패했습니다. ');
    return;
  } finally {
    // 로딩바 닫기
    loading.classList.remove('active');
  }

  if (json != null) {
    const { data } = json;

    // 다음 페이지를 요청할 수 있는지를 판단하기 위한 값.
    isEnd = data.meta.is_end;

    data.documents.map((v, i) => {
      // li-1dep 추가
      const listLi = document.createElement('li');
```

```
listLi.classList.add('list-li')

// link a 추가
const a = document.createElement('a');
a.setAttribute('href', v.url);
a.setAttribute('target', '_blank');

// 이미지 추가
const img = document.createElement('img');
if(v.thumbnail == ''){
    img.setAttribute('src', './img/noimg.jpeg');
}else{
    img.setAttribute('src', v.thumbnail);
}

// info
const info = document.createElement('div');
info.classList.add('info')

// 도서제목
const title = document.createElement('h2');
title.innerHTML = v.title;

// 도서소개
const contents = document.createElement('p');
contents.innerHTML = v.contents;

// 상세정보
const detail = document.createElement('ul')
detail.classList.add('detail')
const detailLi1 = document.createElement('li') // 저자 authors
detailLi1.innerHTML = v.authors;
const detailLi2 = document.createElement('li') // 출판사 publisher
detailLi2.innerHTML = v.publisher;
const detailLi3 = document.createElement('li') // 정가 price
// 판매가가 -1이 아닌 경우에
// 정가가 0일 경우 판매가를 정가로 처리
if(v.sale_price !== -1){
    detailLi3.innerHTML = v.price == 0 ? v.sale_price : v.price;
}else{
    detailLi3.innerHTML = v.price
}

const detailLi4 = document.createElement('li') // 판매가 sale_price
// 판매가 -1 일 경우 정가와 동일하게 처리
detailLi4.innerHTML = v.sale_price == -1 ? v.price : v.sale_price;

detail.appendChild(detailLi1);
detail.appendChild(detailLi2);
detail.appendChild(detailLi3);
detail.appendChild(detailLi4);

info.appendChild(title);
```

```
        info.appendChild(contents);
        info.appendChild(detail);

        a.appendChild(img);
        a.appendChild(info);

        listLi.appendChild(a);
        list.appendChild(listLi);
    })
}
}
```