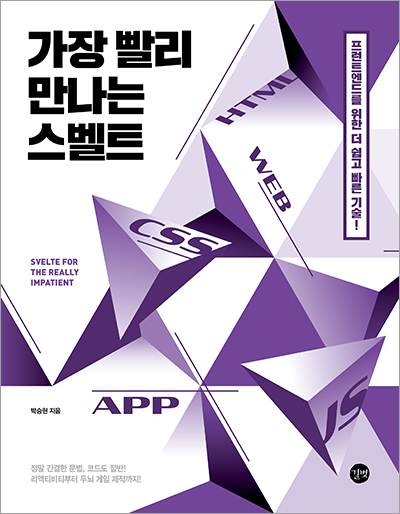
|  |  |
| --- | --- |
| 신간보도자료 | (주)도서출판 길벗  서울시 마포구 월드컵로 10길 56(서교동)  EMB00001dd01167대표전화 (02)332-0931/팩스 (02)323-0586  홈페이지 www.gilbut.co.kr  담당자 : 한동훈(02-330-9807) |

**가장 빨리 만나는 스벨트:   
프런트엔드를 위한 더 쉽고 빠른 기술!**

**▶ 발행 :** (주)도서출판 길벗

**▶ 발행일 :** 2021년 12월 29일

**▶ 지은이 : 박승현**

**▶ 쪽수** : 508

**▶ 판형** : 183\*235(mm)

**▶ 높이(두께)** : 20(mm)

**▶ 가격 :** 32,000원

**▶ ISBN :** 979-11-6521-816-4 93000

**▶ 분야 :**

**YES24 -** 국내도서 > IT 모바일 > 프로그래밍 언어 > JavaScript/CGI

국내도서 > IT 모바일 > 웹사이트 > HTML/JavaScript/CSS/jQuery

**교보문고** – 컴퓨터/IT > 웹 프로그래밍 > JavaScript

**알라딘** - 국내도서 > 컴퓨터/모바일 > 웹디자인/홈페이지 > HTML/JavaScript

**인터파크** - 국내도서 > 컴퓨터/인터넷 > IT/전문서 > 프로그래밍 언어

**[책 소개]**

**쉽고 간결한 코드로 빠르게 배우는 프런트엔드!**

**간결한 코드와 낮은 학습 곡선**

스벨트의 코드는 간결하다. 다른 프레임워크와 비교해도 매우 간결하다. 리액티비티를 위해 필요한 코드는 단 한 줄이다. 코드가 간결하기 때문에 프런트엔드 초보자도 빠르게 배울 수 있고, 경험자도 빠르게 배울 수 있다. 즉, 학습 곡선이 낮다.

**빠른 실행 속도와 작은 크기**

스벨트는 가상 돔을 사용하지 않으니 변경된 부분을 처리하는 부하가 필요하지 않다. 그래서 빠르다. 스벨트는 프레임워크가 아니라 컴파일러로 동작한다. 프레임워크를 배포하는 것이 아니라 모든 코드를 컴파일한 자바스크립트 번들로 제공하기 때문에 생성되는 코드의 크기가 작다.

**빠른 학습을 위한 주제별 분류**

프레임워크의 다양한 기능을 주제별로 분류해서 빠르게 접근한다. 기본 템플릿 문법, 반응성, props, 이벤트, 스토어, 모션, 액션 등의 기능을 주제별로 분류해서 정리했다. 또한, 매번 프로젝트를 생성하지 않고 REPL을 활용해서 바로바로 실습 결과를 확인할 수 있게 했다.

**두뇌 개발 게임 만들기로 실습하기**

두뇌 개발 게임을 만들어보면서 아키텍처 구성, 화면 구성, 라우팅 구성, DB 사용법까지 종합적으로 사용해본다. 깃허브에 소스 코드를 배포하고, 깃허브의 소스 코드를 아마존 웹 서비스에서 자동으로 빌드하여 서비스하는 과정까지 소개했다. 내가 만든 프로젝트를 웹에서 실제로 배포하고 서비스하는 것까지 경험해보자.

**[예제 코드]**

* 길벗 예제 소스 <https://github.com/gilbutITbook/080278>

**[상세 이미지]**

**별첨**

**[지은이의 말]**

지인 소개로 우연히 알게 된 스벨트(Svelte)! 관련된 글들을 찾아 읽다가 호기심이 생겨 튜토리얼을 차곡차곡 해보고 이거다 싶었습니다. 스벨트는 리액티비티를 충실하게 구현하고, 버추얼 돔 같은 아키텍처 이해 없이, 일반적인 자바스크립트 문법만으로 양방향 바인딩(two-way binding) 코딩을 할 수 있게 해 줍니다. 스벨트는 정말 쉬운 자바스크립트 프레임워크입니다. 쉽게 배워서 바로 코딩하기에 이보다 더 나은 프레임워크는 당분간 찾기 쉽지 않을 것입니다.

본격적으로 개발자로서 삶을 살고 싶은데 어디서부터 시작할지 모르겠다면 스벨트를 배워 프런트엔드 개발자부터 시작할 수 있습니다. 혹은 풀스택 개발자가 되고 싶은데 프런트엔드가 약하다고 느끼거나, UI를 빠르게 구현할 수 있는 프레임워크 하나쯤 익히고 싶은 분들도, 이 책으로 그 시작의 첫걸음을 내디딜 수 있습니다. 본문의 예제를 찬찬히 따라 하다 보면 스벨트라는 본인만의 무기가 생길 거라고 생각합니다.

“훌륭한 사수를 찾으면 능력 있는 개발자가 될 수 있겠지!”라는 생각을 하고 선릉역 인근 이면도로에 있는 벤처 회사의 지하 사무실에서 생계형 개발자로 세상에 첫발을 내디뎠었습니다. 그 사수는 어디에 꼭꼭 숨어 계셨는지, 아니면 제가 찾아다니길 소홀했는지, 22년째 만나 뵙지 못하고, 이제 제가 시니어 개발자가 되어 있네요. 사실, 일하면서 새로운 기술이나 흥미 있는 프레임워크가 나올 때마다 저한테 좋은 사수는 책이었던 것 같습니다. 그때의 저처럼 신입사원이 회사에 입사한다면 초급 개발자로 프런트엔드의 세상에 첫발을 내디딜 때 “개발은 재미있는 일이구나”라고 느끼게 해주고 싶다는 생각으로 이 책을 만들었습니다. 고등학교에 진학하는 우리 아들 제민이가 “아빠 개발자가 되고 싶어요”라고 말한다면 그때, 이 책을 쥐여주면 좋겠다는 바람도 글 속에 담았습니다. 그런 바람대로 이 책은 HTML과 JS로 간단한 웹페이지 정도만 만들어 봤다면 쉽게 배워서 실무에 적용할 수 있도록 예제를 구성했습니다.

반응형(reactive) 프로그래밍을 처음 접하거나, 처음 들어 보았다면 “5장 반응성” 부분을 잘 읽어보시길 추천합니다. 스벨트를 통하여 개념을 이해하고 익힌다면 반응형 프로그래밍은 아주 쉽습니다. 그래서 재미있습니다.

스벨트를 사용하여 누구나 쉽게 프런트엔드 개발자가 될 수 있습니다. 하지만 개발한 앱을 서비스까지 구동시키는 데는 개발 외적인 요소도 알아야 합니다. 그래서 20장에서는 앱 개발부터 서비스까지의 전체 과정(아키텍처, 설계, 개발, 빌드, 배포, 서버 구성 및 구동)을 차곡차곡 실습해 볼수 있게 구성했습니다. 한 땀 한 땀 따라 하다 보면 어느덧 내가 만든 앱이 인터넷에서 동작하는 것을 볼 수 있습니다.

아직 정식 버전이 나오지 않았지만, 스벨트킷은 스벨트를 풀스택 프레임워크로 만들어줍니다. 20장에서는 프로젝트를 스벨트킷으로 구성했습니다. 프로젝트 구성뿐 아니라, 빌드, 라우터 기능, 데이터 서비스 등 백 엔드의 핵심기능을 예제를 통하여 맛볼 수 있도록 구성했습니다. 따라 하다 보면 풀스택 개발자로서 알아야 할 필수적인 요소들을 스벨트킷을 통하여 배울 수 있습니다.

스벨트가 너무 재미있고 쉬운 프레임워크라서, 얼마 안 있으면, 프런트엔드 개발의 춘추 전국시대 같은 지금을 평정할 것 같습니다. 이런 이유로, 일과 함께 책을 쓴다는 것이 쉽지 않다는 것을 『Meteor 코딩 공작소』(길벗, 2017)를 쓰면서 뼈저리게 느꼈지만, 다시 용기를 내었습니다. 글쓰기에 어설픈 제가 책을 다시 완성할 수 있도록, 항상 조언과 도움을 주신 길벗 출판사 한동훈 님

께 진심으로 감사드립니다.

부모님 사랑합니다. 항상 건강하셨으면 좋겠습니다. 우리 아들 제민이, 왕딸 지수, 막내 지윤아 사랑한다! 항상 옆에서 응원해주는 윤용정 님도 사랑합니다!

코로나가 빨리 종식돼서, 여러 개발자들과 스벨트를 주제로 수다를 떨 수 있는 날이 하루빨리 오기를 기원합니다. 모두 건강하세요. 고맙습니다.

**[지은이 소개]**

**박승현**

제민이, 지수, 그리고 지윤이 아빠입니다. 어렸을 때부터 프로그래밍을 직업으로 가지고 싶었고, 매일 매일 그 꿈을 이루고 사는 생계형 개발자입니다. 자바스크립트를 좋아하고 관련 기술에 대한 잡담을 좋아합니다.

**[목차]**

1장 스벨트 소개

\_\_1.1 적은 코드

\_\_1.2 순수한 리액티비티(반응성)

2장 개발 환경 구성하기

\_\_2.1 REPL로 시작해보기

\_\_2.2 로컬에서 프로젝트 만들어 코드 작성하기

\_\_2.3 폴더 및 파일 구성

3장 Hello, Svelte~!

\_\_3.1 <script>

\_\_3.2 HTML 마크업

\_\_3.3 <style>

\_\_3.4 {} 표현식

\_\_3.5 스벨트 컴포넌트를 document에 포함시키기

4장 템플릿 기본 표현식

\_\_4.1 {#if ...} 블록

\_\_4.2 {:else if ...} 블록

\_\_4.3 {#each ...}를 사용해 배열 표시하기

\_\_4.4 {#each} 루프 안에서 인덱스 사용하기

\_\_4.5 {#each}로 JSON 객체 다루기, 다차원 배열 표시하기

\_\_4.6 Promise와 {#await ...} 블록 사용하기

\_\_4.7 fetch 결과를 {#await ...} 블록으로 표시하기

\_\_4.8 {#key 표현식} HTML {/key}

\_\_4.9 {#key 표현식} 하위 컴포넌트 {/key}

\_\_4.10 {@html}로 돔에 HTML 엘리먼트 생성하기

\_\_4.11 {@debug}로 디버깅, 리액티브하게 디버깅하기

5장 반응성

\_\_5.1 스벨트의 반응성이란?

\_\_5.2 반응형 데이터 만들기

\_\_5.3 센서 생성과 동작 정의

\_\_5.4 할당을 통한 트리거 만들기

\_\_5.5 센서의 사용하는 네 가지 패턴

\_\_5.6 배열의 변화 감지하기

\_\_5.7 JSON 객체의 변화 감지하기

6장 HTML 엘리먼트에 데이터 바인딩하기

\_\_6.1 텍스트 / 텍스트 입력값 바인딩하기

\_\_6.2 input에 숫자 값 바인딩하기

\_\_6.3 체크박스 바인딩하기

\_\_6.4 다중 객체(radio, check)에 그룹 바인딩하기

\_\_6.5 textarea 바인딩하기

\_\_6.6 단일 select, 다중 select 바인딩하기

\_\_6.7 contenteditable 바인딩

\_\_6.8 {#each ...} 블록에서 속성값 바인딩하기

\_\_6.9 media의 속성값 바인딩하기

\_\_6.10 크기 값 바인딩

\_\_6.11 this 바인딩

\_\_6.12 하위 컴포넌트 데이터 바인딩

7장 하위 컴포넌트 사용

\_\_7.1 예제 프로젝트 생성하기

\_\_7.2 하위 컴포넌트 작성하기

\_\_7.3 하위 컴포넌트 불러오기

\_\_7.4 HTML에서 사용하기

8장 props

\_\_8.1 props 선언하기

\_\_8.2 props 사용하기

\_\_8.3 spread 활용하기

9장 이벤트

\_\_9.1 돔 이벤트 처리하기

\_\_9.2 인라인으로 이벤트 처리하기

\_\_9.3 이벤트 제한자 사용하기

\_\_9.4 하위 컴포넌트의 커스텀 이벤트 처리하기

\_\_9.5 이벤트 포워딩하기

\_\_9.6 돔 이벤트 포워딩하기

10장 라이프사이클

\_\_10.1 onMount

\_\_10.2 onDestroy

\_\_10.3 beforeUpdate와 afterUpdate

\_\_10.4 tick

11장 스토어 사용하기

\_\_11.1 writable 스토어로 시작하기

\_\_11.2 readable 스토어 사용하기

\_\_11.3 derived 스토어 사용하기

\_\_11.4 사용자 정의 스토어 만들기

\_\_11.5 get 함수 사용하기

12장 모션

\_\_12.1 트윈

\_\_12.2 스프링

13장 트랜지션

\_\_13.1 트랜지션 지시자

\_\_13.2 파라미터 이용하기

\_\_13.3 in/out 사용하기

\_\_13.4 트랜지션 이벤트 정의하기

\_\_13.5 로컬 트랜지션

14장 애니메이션

15장 액션

\_\_15.1 use 지시자 사용하기

\_\_15.2 파라미터 사용하기

16장 클래스

\_\_16.1 class 속성을 확장하여 사용하기

\_\_16.2 class 지시자 사용하기

17장 slot 사용하기

\_\_17.1 slot이란?

\_\_17.2 slot 기본값 사용하기

\_\_17.3 named slot 사용하기

\_\_17.4 $$slots을 이용한 콘텐츠 체크

\_\_17.5 slot의 props 사용하기

18장 컨텍스트 API

\_\_18.1 setContext와 getContext

\_\_18.2 hasContext

\_\_18.3 컨텍스트와 스토어 차이점 비교

19장 특수 엘리먼트

\_\_19.1 <svelte:self>

\_\_19.2 <svelte:component>

\_\_19.3 <svelte:window>

\_\_19.4 <svelte:body>

\_\_19.5 <svelte:head>

\_\_19.6 <svelte:options>

\_\_19.7 <svelte:fragment>

20장 두뇌 개발 게임만들기

\_\_20.1 게임 화면 미리 보기

\_\_20.2 애플리케이션 아키텍처

\_\_20.3 배포 아키텍처

\_\_20.4 스벨트킷으로 프로젝트 생성

\_\_20.5 코드 작성 준비하기

\_\_20.6 라우터로 페이지 구성하기

\_\_20.7 레이아웃으로 메뉴 구성하기

\_\_20.8 스토어 구성하기

\_\_20.9 홈(이번 주의 점수) 구현

\_\_20.10 게임 구현

\_\_20.11 점수 저장 구현

\_\_20.12 몽고디비 연결하기

\_\_20.13 REST API 구현

\_\_20.14 이번 주의 점수를 REST로 구현

\_\_20.15 점수 저장을 REST로 구현

\_\_20.16 깃허브 클라이언트 설치하기

\_\_20.17 SSH로 깃허브에 연결하기

\_\_20.18 깃허브로 소스 배포하기

\_\_20.19 AWS 서버 만들기

\_\_20.20 터미널에서 접속하기

\_\_20.21 AWS 앱 서버로 설정하기

\_\_20.22 로컬 개발 환경에서 빌드

\_\_20.23 운영(AWS) 환경에서 빌드 및 실행