## 스토리북이란

스토리북(Storybook)은 UI 컴포넌트를 위한 유저 인터페이스 개발 환경 및 자유롭게 컴포넌트를 테스트할 수 있는 도구입니다. 스토리북을 활용하면 개발자는 독립적인 환경에서 컴포넌트를 개발할 수 있고, 개발환경에서 컴포넌트들을 인터랙티브하게 보여줄 수 있습니다.

스토리북(Storybook)은 기본적으로 애플리케이션 외부(독립적인 개발환경)에서 실행되므로 사용자는 애플리케이션별 종속성 및 요구 사항에 대해서 걱정할 필요 없이 UI 컴포넌트를 독립적으로 개발할 수 있습니다.

스토리북은 다양한 부가기능(addons)를 지원하며 커스터마이징하기 쉽게 굉장히 유연한 API를 제공합니다. 스토리북의 정적 버전(a Static version of Storybook)은 빌드하여 HTTP 서버에 배포할 수 있습니다.

## 스토리북 설치

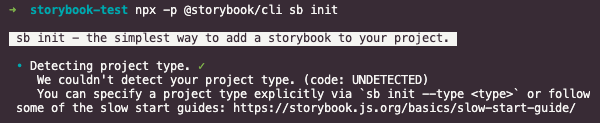
스토리북은 굉장히 많은 프론트엔드 view 라이브러리를 지원하고 있습니다. 스토리북은 현재 React, Vue, Angular, Mithril, Marko, HTML, Svelte, Meteor, Ember, Riot, Preact를 지원하고 있습니다. 스토리북을 시작하기 위해서 아래 단계들을 참고해주세요.

스토리북 CLI(@storybook/cli) 를 활용해서 시작해보겠습니다. 명령어는 프로젝트에 스토리북과 관련된 boilerplate 파일들을 생성합니다.

cd my-project-directory // 현재 자신의 프로젝트 디렉토리로 이동합니다.

npx -p @storybook/cli sb init // 스토리북 설치

스토리북 CLI는 package.json 파일을 검사하여 프로젝트에서 사용하고 있는 프론트엔드 view 라이브러리를 결정합니다.

스토리북 CLI에서 프로젝트 view 타입을 감지하는 모습

만약 스토리북에서 HTML 스니펫을 개발하고 싶다면, 자동으로 이 부분을 처리할 수 있습니다. HTML 유형을 용도로 설정하고 싶다면 --type 플래그를 활용해서 스토리북을 설치해야됩니다.

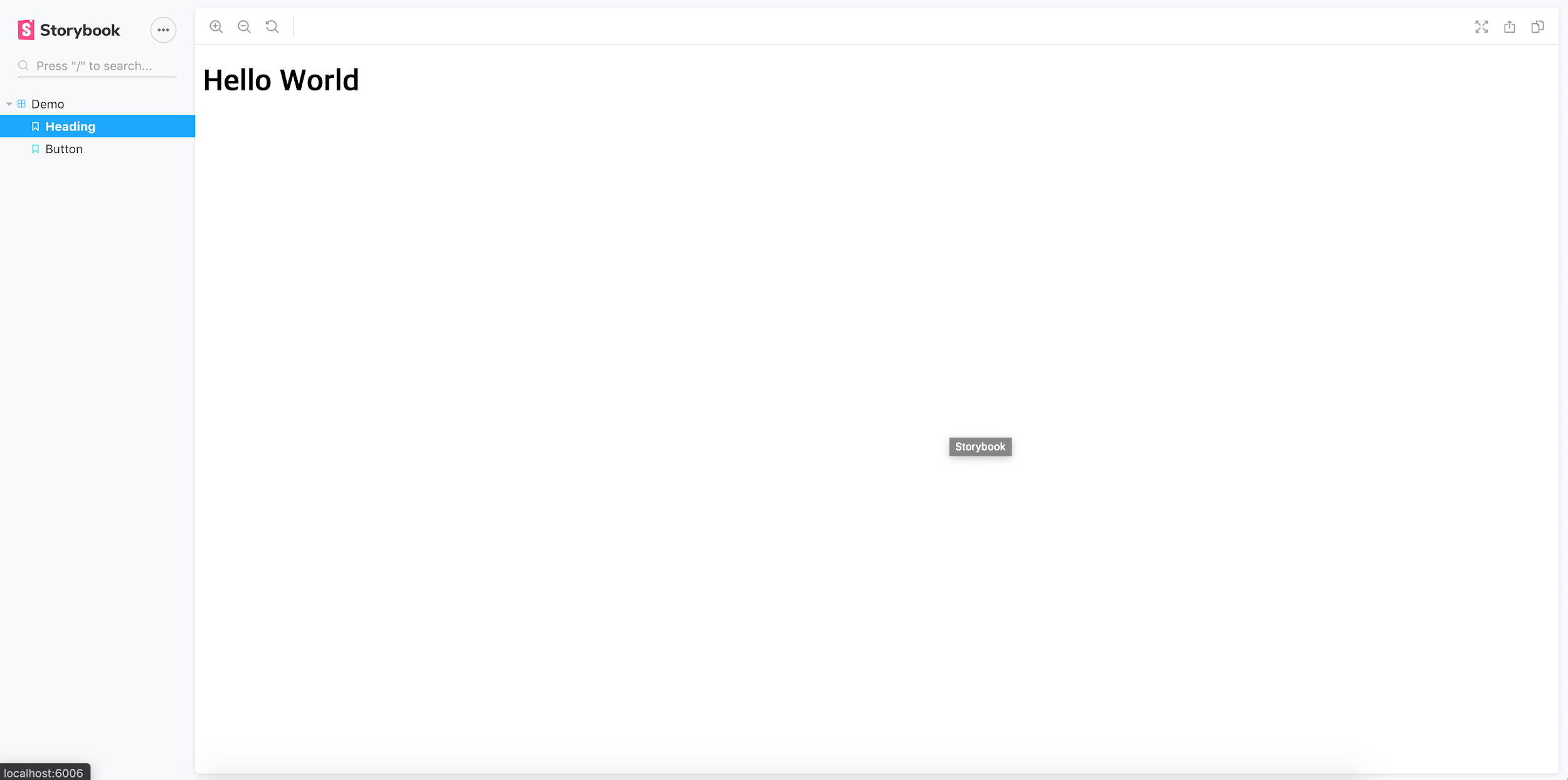
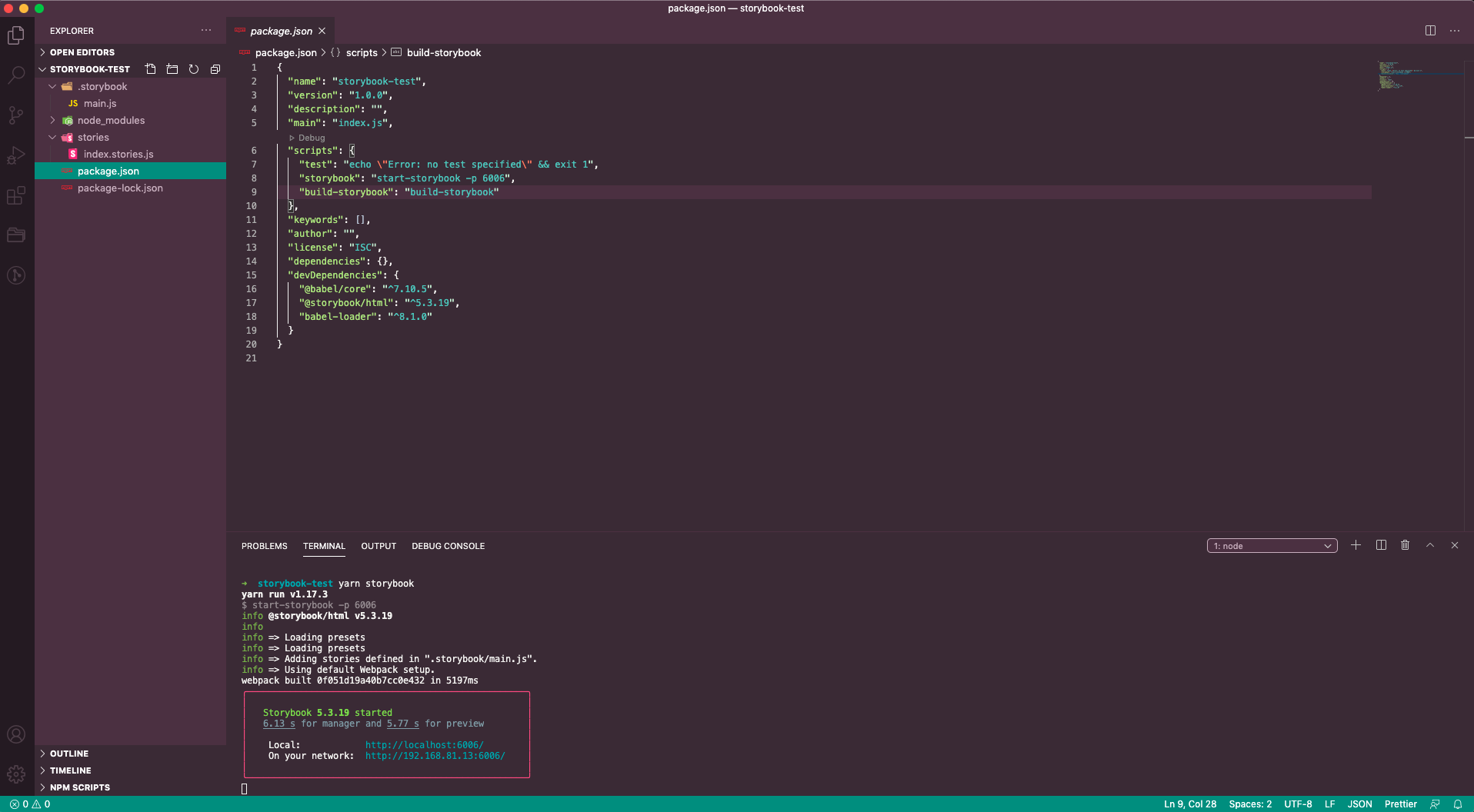
npx -p @storybook/cli sb init --type html // --type 플래그 추가

스토리북은 아래의 명령어와 함께 실행 가능합니다.

npm run storybook

// or

yarn storybook



스토리북 실행 명령어를 통해 스토리북을 실행한 예시

해당 포스팅에서는 Vue를 활용해서 스토리북을 간단히 사용하는 방법에 대해서 알아봅니다. 특정 프론트엔드 view 라이브러리와 관련된 사용법이 궁금하면 아래를 참고하세요.

* Vue

## 스토리북 사용법

스토리북은 스토리들의 모음입니다. 각 스토리는 컴포넌트의 개별 시각 상태를 나타냅니다.

기술적으로 스토리는 화면에 렌더링 할 수있는 것을 반환하는 함수입니다.

### 기본 스토리

Button 컴포넌트를 생성합니다. 스토리북 예시를 위해 간단한 컴포넌트를 하나 만들어줍니다.

<template>

<button><slot></slot></button>

</template>

<script>

export default {

name: 'BasicButton'

}

</script>

<style>

</style>

Button 컴포넌트를 위한 스토리를 작성합니다.

../src/BasicButton/BasicButton.stories.js 파일을 생성하고, 첫 스토리를 작성합니다.

import Vue from 'vue';

import BasicButton from './BasicButton';

export default { title: 'BasicButton' };

// 스토리 1

export const withText = () => '<BasicButton>스토리북 이동</BasicButton>'

// 스토리 2

export const withEmoji = () => '<BasicButton>😎</BasicButton>'

// 스토리 3

export const withTextAndEmoji = () => '<BasicButton>Sotrybook! 😎</BasicButton>'

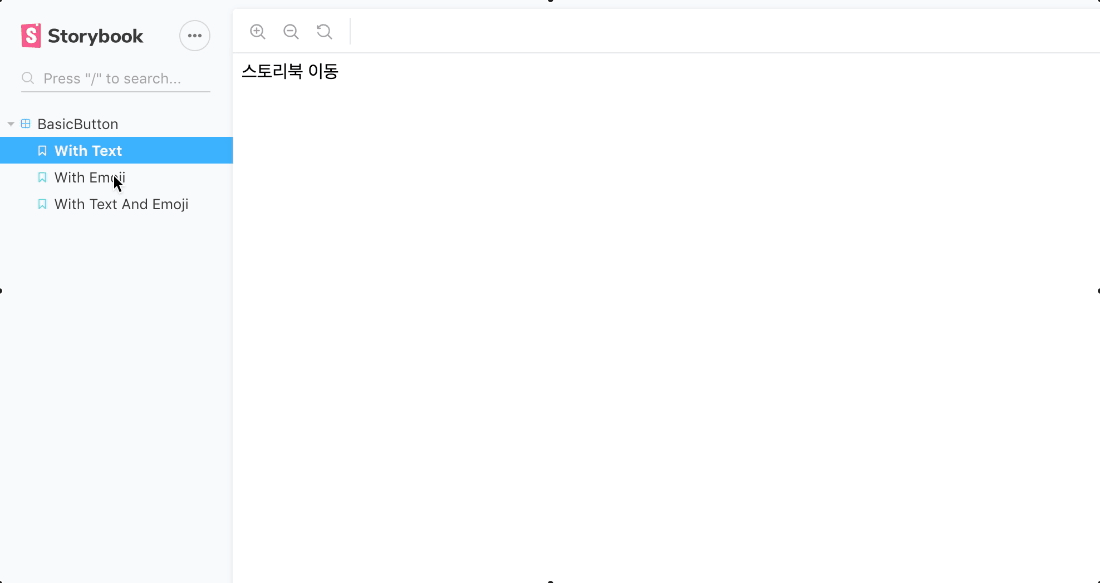
스토리북을 실행합니다!

npm run storybook

// or

yarn storybook

스토리북에 저희가 작성한 스토리가 모두 렌더링 되었습니다.



각 스토리는 컴포넌트의 개별적인 상태값을 가지는 예시

### 배포해보기

스토리북은 웹팩(Webpack)의 HMR(Hot Module Replacement, 브라우저를 새로 고치지 않아도 웹팩으로 빌드한 결과물이 웹 애플리케이션에 실시간으로 반영될 수 있게 도와주는 설정)을 통해 개발자들이 개발하는 시간을 단축할 수 있게 도와줌으로써 굉장히 좋은 개발자 경험을 제공합니다.

그러나 여기에 그치지 않고, 스토리북은 작성한 스토리들을 다른 사람에게 보여줄 수있는 기능을 제공합니다. 스토리북에는 스토리들을 정적 웹 앱으로 내보내는 도구가 제공됩니다. 그런 다음 GitHub 페이지 또는 정적 호스팅 서비스에 배포 할 수 있습니다.

NPM 스크립트(script)를 넣어봅시다.

// package.json

{

"scripts": {

"build-storybook": "build-storybook -c .storybook -o .out"

}

}

NPM script를 넣으셨다면 아래 구문을 실행할 수 있습니다.

npm run build-storybook

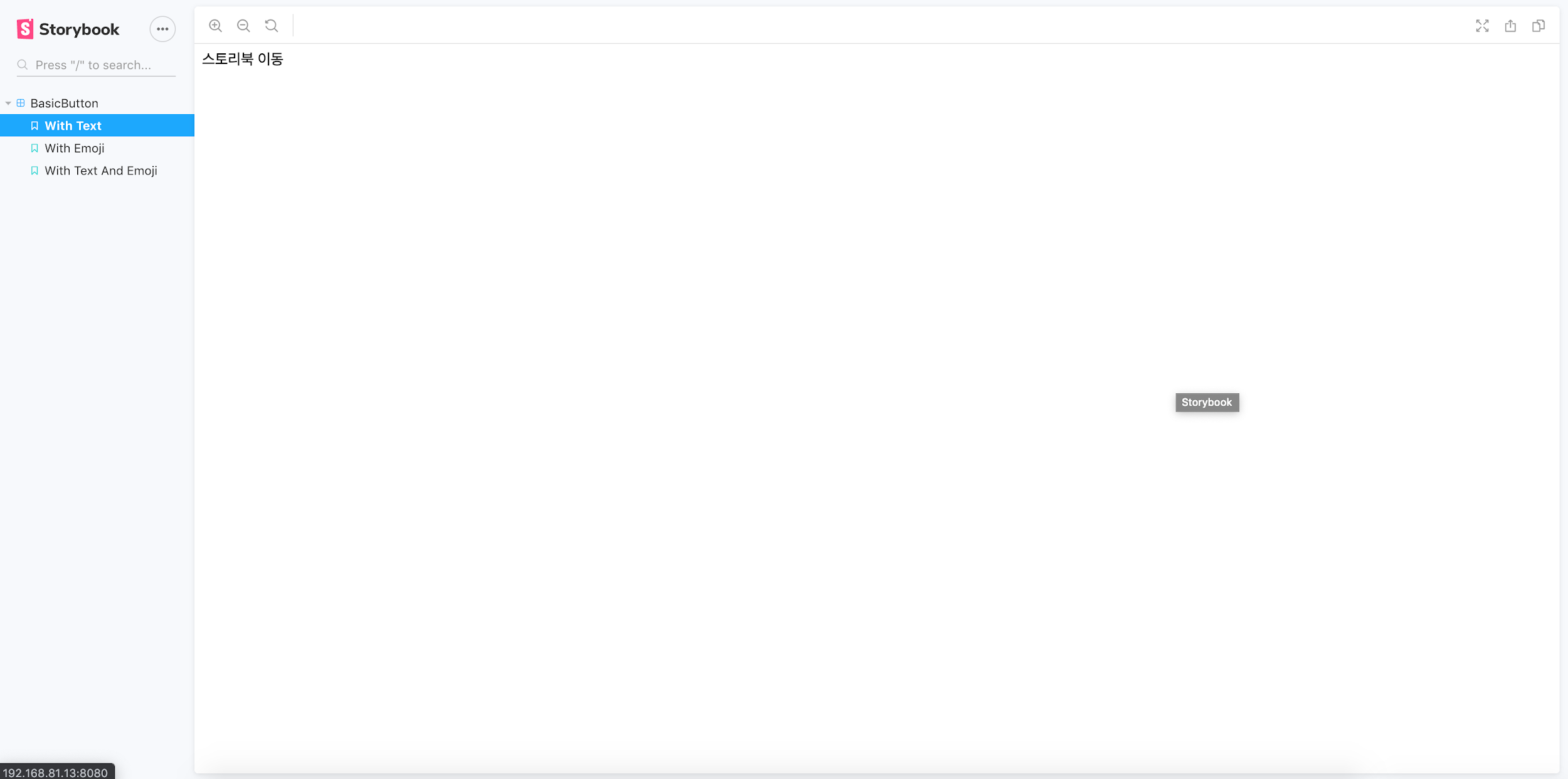
// or

yarn build-storybook

스토리북 빌드 스크립트를 실행하면 스토리북 디렉토리에 구성된 스토리들이 정적 웹 앱으로 빌드되어 .out 디렉토리에 배치됩니다. 이제 원하는 위치에 .out 디렉토리에 컨텐츠를 배치 할 수 있습니다.

#### 로컬환경에서 테스트

npx http-server .out

로컬환경에서 스토리북이 실행된 예시

## 마무리

이번 포스팅에서는 스토리북의 정의부터 간단한 설치 그리고 사용법에 대해서 다뤄봤습니다. 오늘 소개해드린 부분은 빙산의 일각일 뿐 실제 프로젝트에서 사용할 때는 좀 더 디테일한 부분들(설정(Configurations), 확장 기능(Addons), 테스팅(Testing))을 다룰줄 알아야 합니다. 해당하는 부분들은 프론트엔드 view에 따라서 달라지는 부분이 아닌 공통적인 부분임으로 개념을 알아두시면 프로젝트에서 사용하고 있는 라이브러리에 상관없이 해당 부분들을 반영할 수 있습니다.