

SPA를 만들자 (1)

별소프트 장경순 매니저 (ksjang@byulsoft.com)



1. SPA 소개
2. 왜 SPA를 만드는가?
3. Front-end Framework
4. Vue.js 소개
5. Vue.js 시작하기

1. SPA 소개

SPA

Single Page Application

추가적인 페이지 이동없이
단일 페이지에서 구동하는 웹사이트

2. 왜 SPA를 만드는가?

기존 웹사이트 방식



SPA 방식



적은 데이터 전송

= 속도 향상

= 사이트 전체 트래픽 감소
(서버 비용 감소)

즉각적인 화면 반응

= 사용자 편의 증가

Front-end / Back-end 분리

= 개발 효율성 증가

= 유연한 서버 확장

물론 단점도 있다...

검색 엔진 노출 안됨 -> SSR로 해결

최초 로딩 지연 -> LazyLoad로 해결

3. Front-end Framework

막상 SPA를 만들려고 하니...
jQuery같은 것들로로는 한계가...

jQuery는 DOM, 이벤트, 애니메이션을
쉽게 제어해주는 도구일 뿐...
SPA같은 복잡한 구조를 구현하기엔 부족

그러다 2009년...



Misko Hevery
(구글 직원)

아오... 답답해서
내가 만든다!

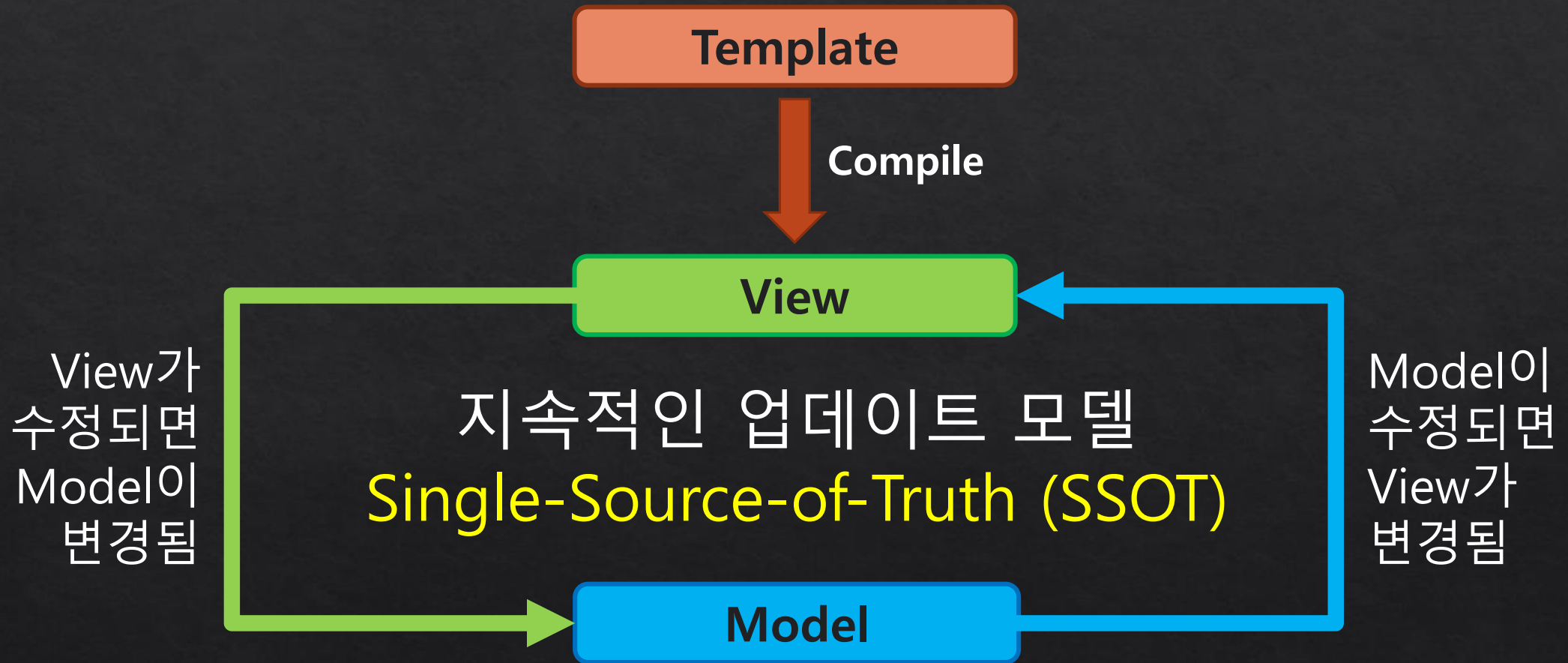
(실제로 이렇게 말하지는 않았습니다...)

3주 만에 똑딱똑딱 만들어 낸 것이...!



(1.0 정식 버전 발표는 2010년)

AngularJS의 특징 : Two-Way Data binding



View와 Model을 분리하고,
그 변화를 실시간으로 감지하기 때문에
적은 코드로도 쉽게 화면 개발이 가능

그러나 치명적인 단점이 존재
→ 느낌!

느린 속도의 원인 : **watch**와 **digest loop**

Two-way Data Binding을 구현하기 위해
Model의 변화를 감지하는 **watch**가 추가되고,
이를 주기적으로 실행하기 위한 **digest loop**가 존재함

→ **watch**의 수가 늘어날수록 속도가 느려짐

세월이 흘러 2013년...

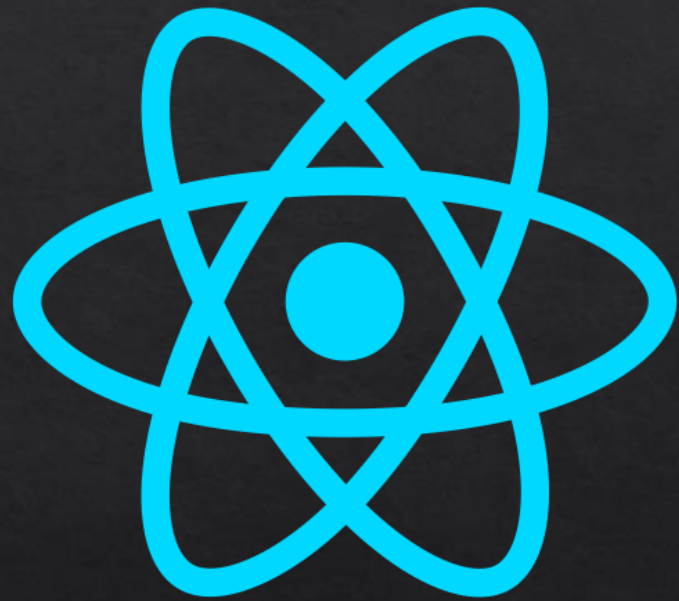


(Facebook)

(Instagram)

더 좋은
프레임워크를
만들자!

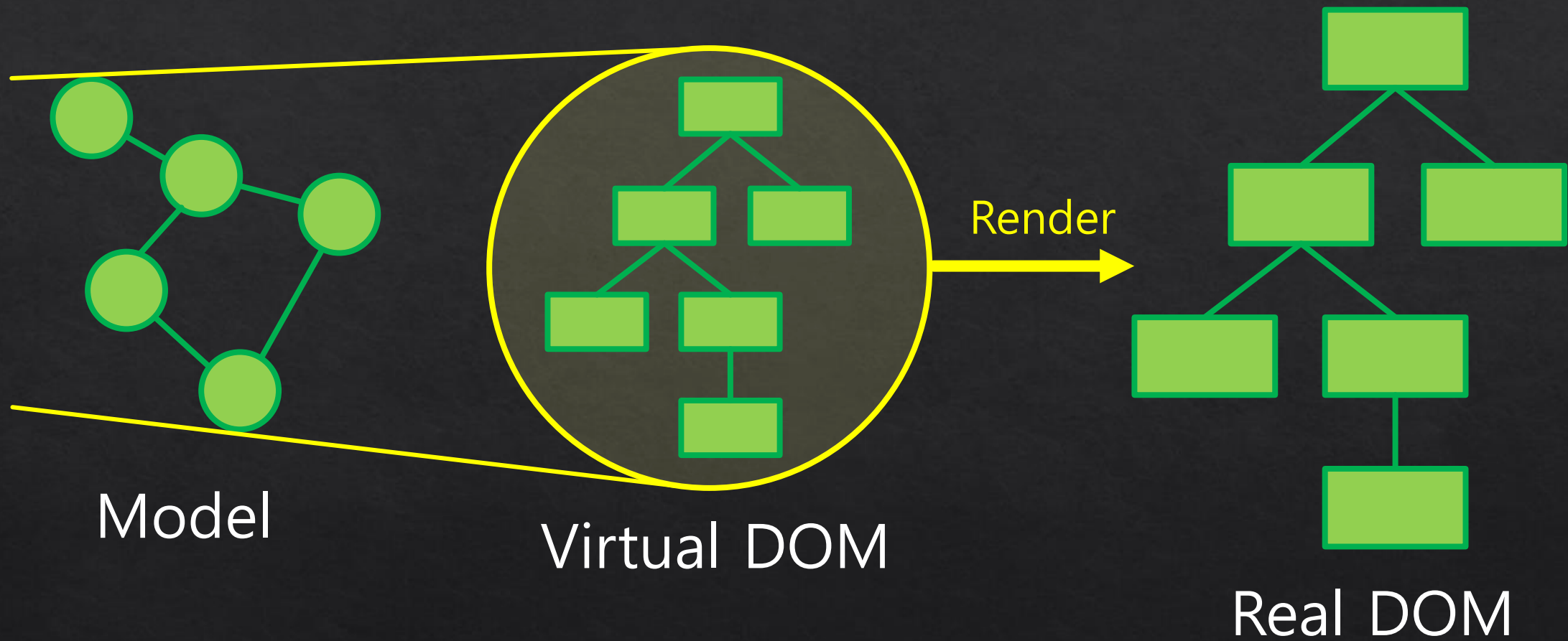
그래서 나온게...



React

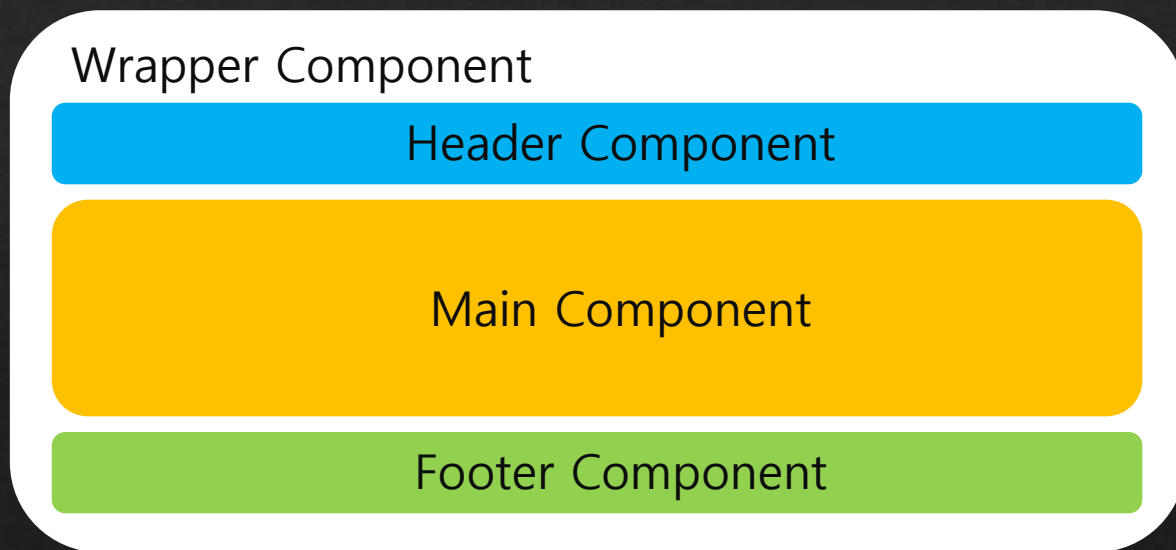
React의 특징 1 : Virtual DOM

추가/수정이 느린 DOM 대신
추상화된 JavaScript Object인
Virtual DOM을 사용



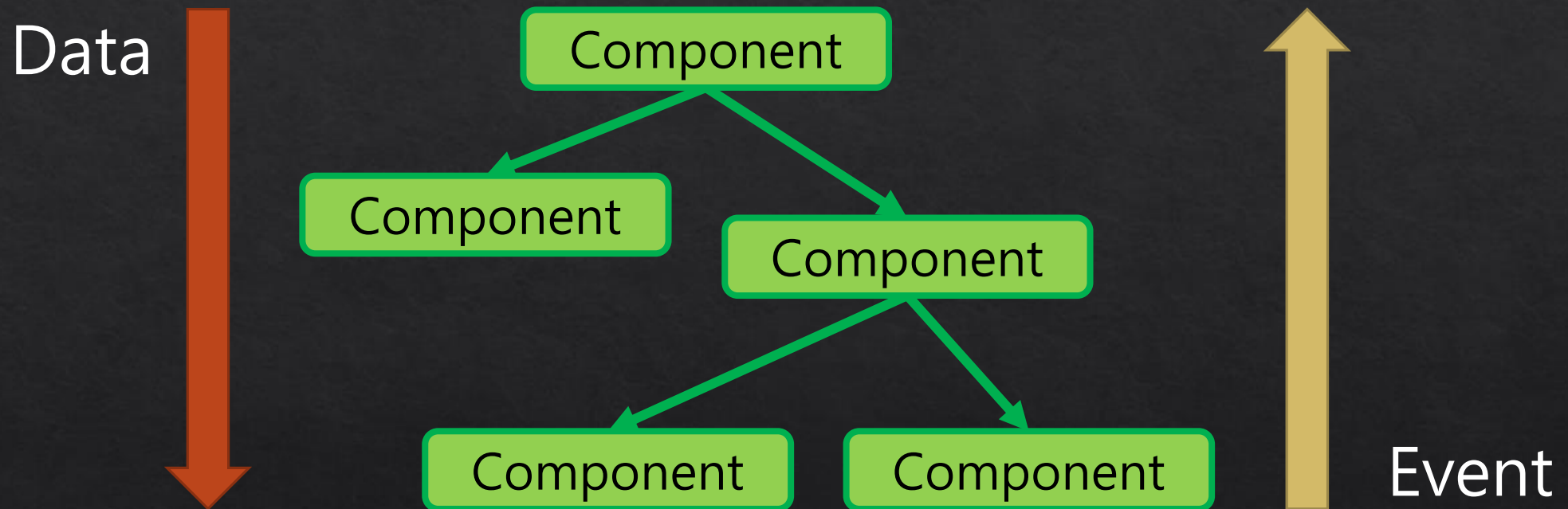
React의 특징 2 : Component 개발 방식

모든 것을 Component로!



캡슐화
확장성
결합성
재사용성

React의 특징 3 : One-Way Data binding



Component One-Way Data binding

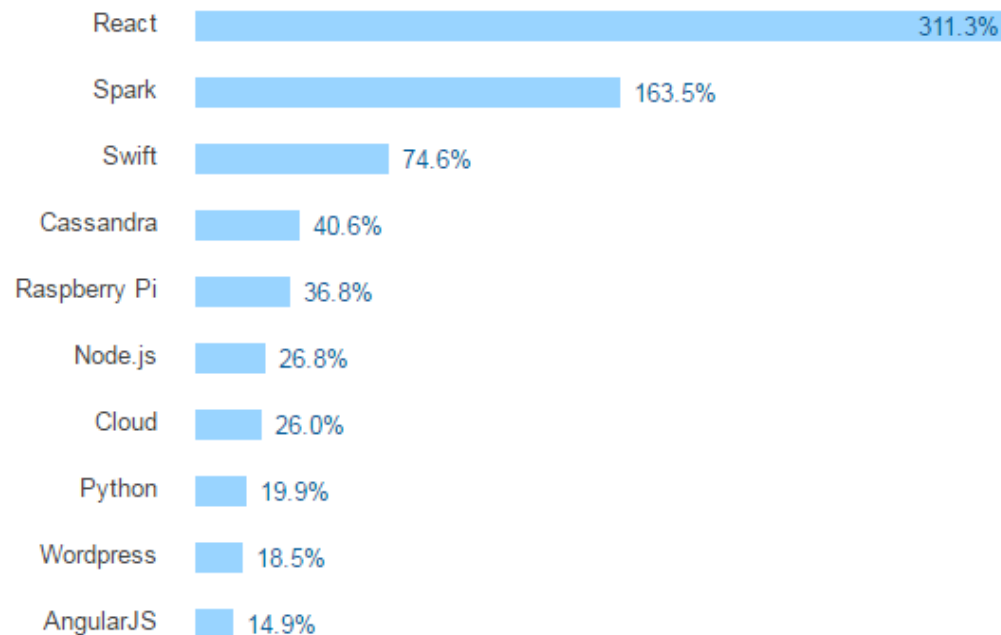
1. 불필요한 watch가 없어서 **빠름**
2. 데이터 흐름 파악이 **쉬움**

결국 React는 왕이 되었습니다...

IV. Trending Tech on Stack Overflow

Winners

Losers



(2015년 기준)

4. Vue.js 소개






2013년 Evan You라는 중국인 개발자가 발표 (개인주도)




2015년 10월 27일 1.0 정식 릴리즈 (2020년 1월 기준 2.7)

킹왕짱 React의 자리를 위협하는 중...



 Watch	6.6k	 Star	143k	 Fork	27.5k
---	------	--	------	--	-------



 Watch	6k	 Star	156k	 Fork	23.5k
---	----	--	------	--	-------

Vue.js의 특징 1 : React와 비슷...

Virtual DOM
One-Way Data Binding
Component 개발 방식
기타 등등...

Vue.js의 특징 2 : 배우기 쉽다

매우 단순하면서도 직관적인 구조
한글 공식 문서 지원

<https://kr.vuejs.org/v2/guide/index.html>

Vue.js의 특징 3 : Made in China(?)

쏟아지는 레퍼런스

빠른 속도로 커져가는 커뮤니티

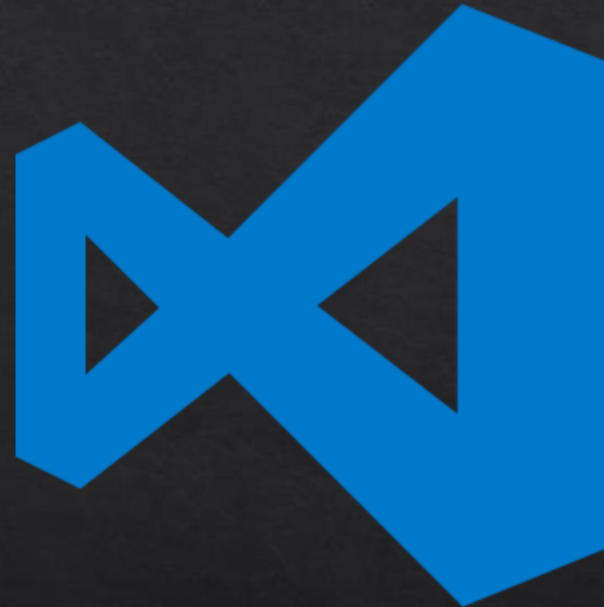
5. Vue.js 시작하기

Vue.js 준비 1 : 툴 설정



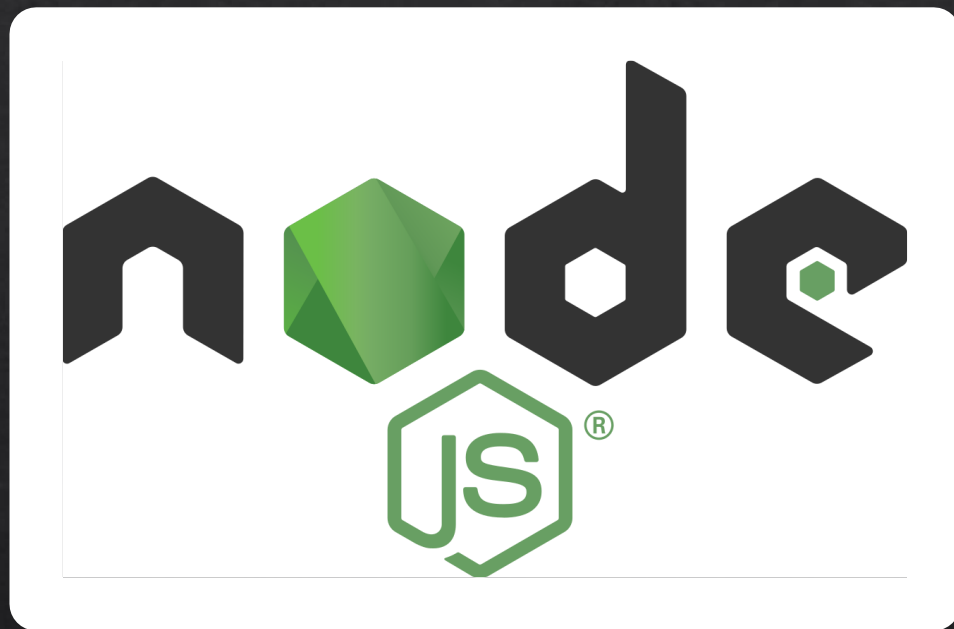
WebStorm
(유료)

혹은

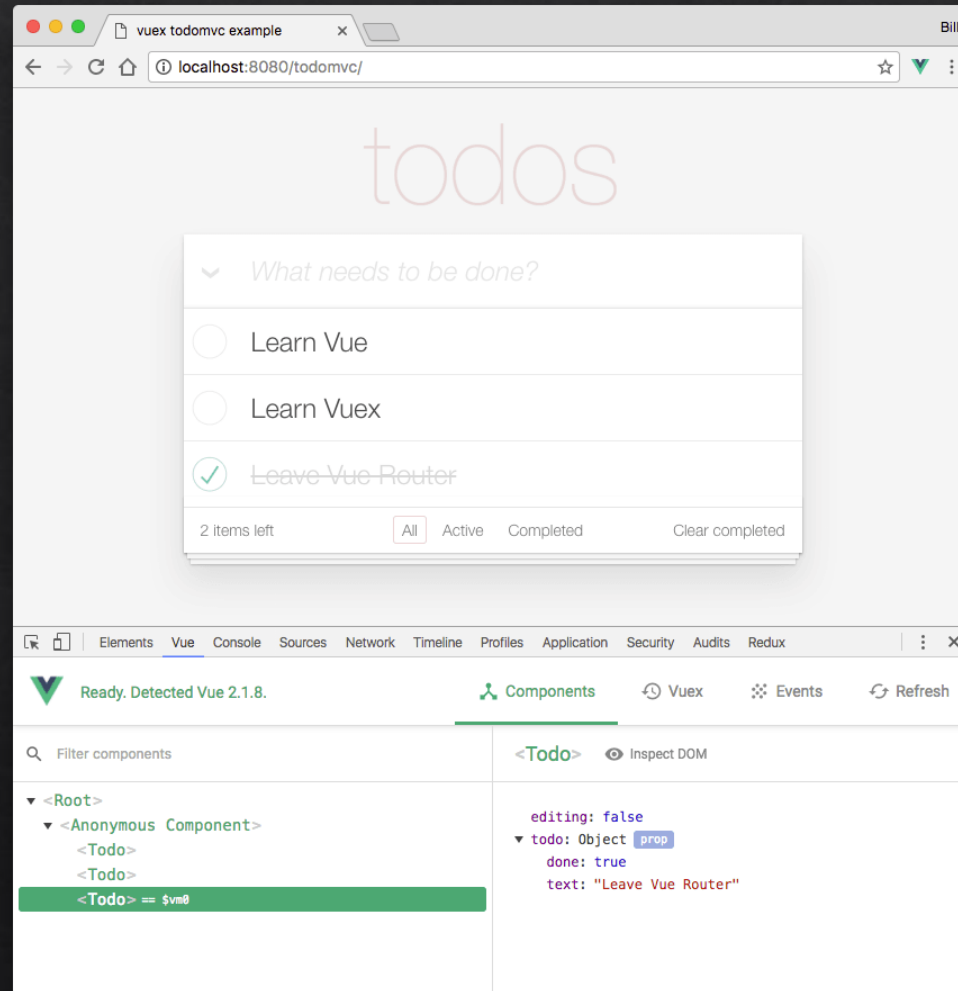


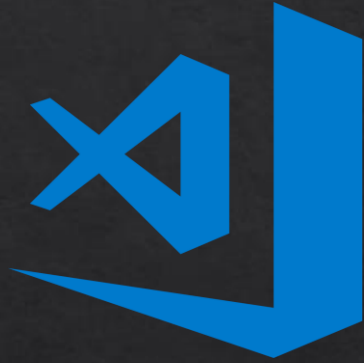
VS Code
(무료)

Vue.js 준비 2 : Node.js 설치



Vue.js 준비 3 : Vue DevTools 설치





VS Code 실행!

< 이번 시간의 목표 >

기본 문법 익히기

이번 시간은 간단한 예제이므로 CDN에서 가져옵니다~

```
<script src="https://cdn.jsdelivr.net/npm/vue"></script>
```

Vue Instance 생성

```
var vm = new Vue({  
  // Option  
});
```

Option : el

```
<div id="app"></div>
```

```
var vm = new Vue({  
  el: '#app',  
});
```


Option : **data**

```
var vm = new Vue({  
  ...  
  data: function() {  
    return {  
      msg: 'Hello World'  
    }  
  }  
});
```

Interpolation (보간법)

```
<div id="app">
```

```
{{ msg }}
```

```
</div>
```

Directive

v-* (v-bind, v-if...)

Vue에서 제공하는 Element 특수 속성

Directive : **v-bind** (데이터 바인딩)

```
<div id="app">  
  <span v-bind:title="msg">  
    여기에 마우스를 올려보세요  
  </span>  
</div>
```

Directive : **v-model** (양방향 바인딩)

```
<div id="app">  
  <p>{{ msg }}</p>  
  <input v-model="msg" />  
</div>
```

Directive : **v-if** (조건문)

```
<div id="app">  
  <span v-if="true">  
    조건이 참일 때 보입니다.  
  </span>  
</div>
```


Directive : **v-show** (조건문)

```
<div id="app">  
  <span v-show="true">  
    조건이 참일 때 보입니다.  
  </span>  
</div>
```

v-if vs v-show

v-if : 진짜 조건 렌더링. 전환 비용 높음.

V-show : 가짜 조건 렌더링. 초기 렌더링 비용 높음.

Directive : **v-for** (반복문)

```
<div id="app">  
  <span v-for="num in [1,2,3]">  
    {{ num }}번째 문장입니다.  
  </span>  
</div>
```

Option : **methods**

```
var vm = new Vue({  
  ...  
  methods: {  
    say: function() {  
      alert(this.msg);  
    }  
  }  
});
```

Directive : **v-on** (이벤트 청취)

```
<div id="app">  
  <p>{{ msg }}</p>  
  <button v-on:click="say()">  
    말하기  
  </button>  
</div>
```

Option : **computed**

```
var vm = new Vue({  
  ...  
  computed: {  
    msg2: function() {  
      return this.msg + '!!!';  
    }  
  }  
});
```


Option : **watch**

```
var vm = new Vue({  
  ...  
  watch: {  
    msg: function() {  
      alert('changed!');  
    }  
  }  
});
```

computed vs watch

computed : 대상 값이 계산된 속성 (property)

watch : 대상 값이 변경될 때 실행되는 함수 (method)

Thank you!