



AngularJS



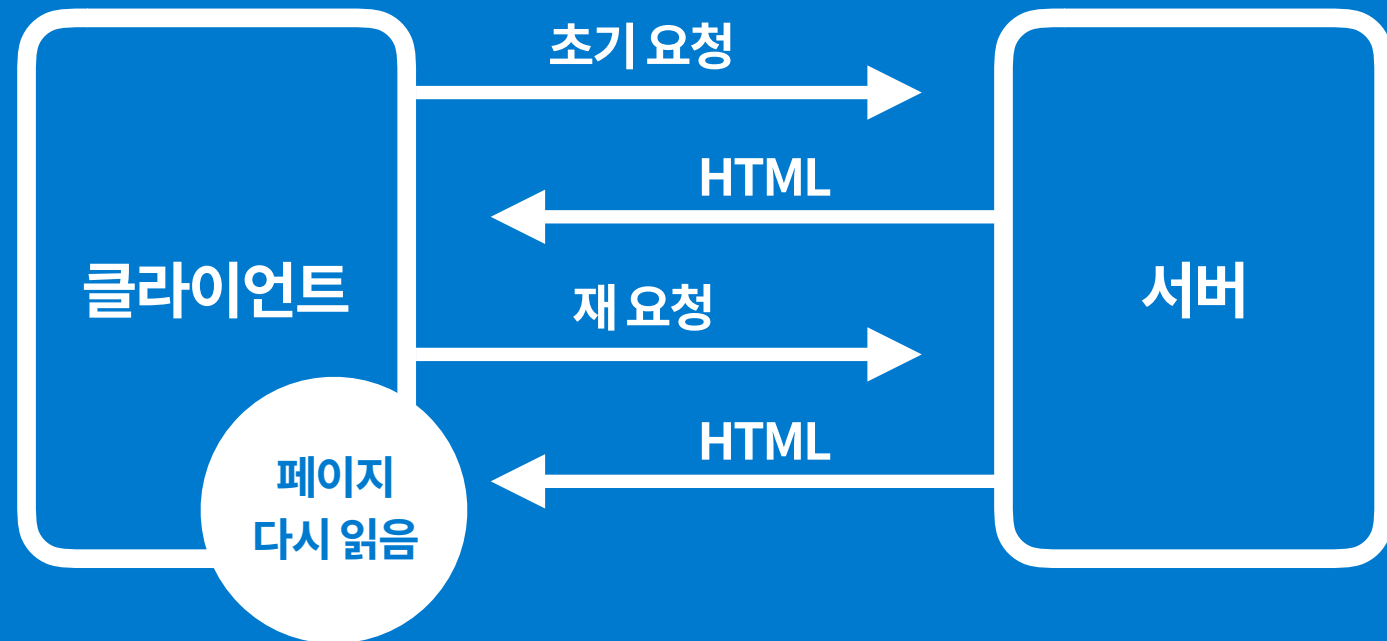
SPA

Single Page Applications

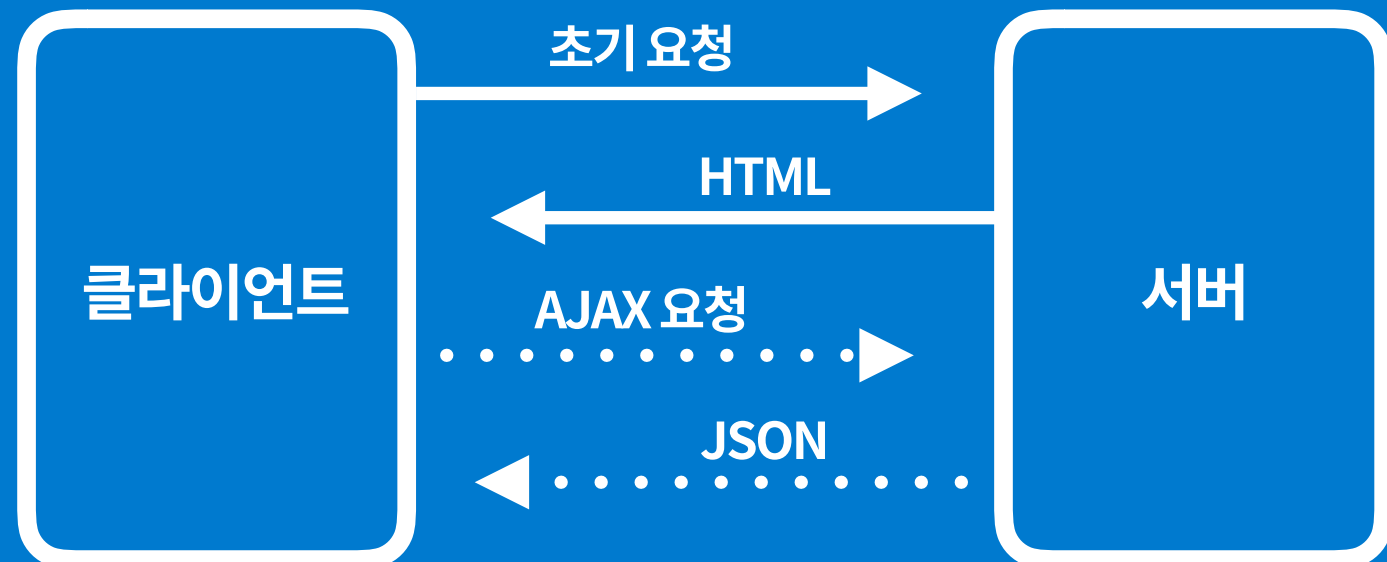
Single Page Applications



전통적인 모델



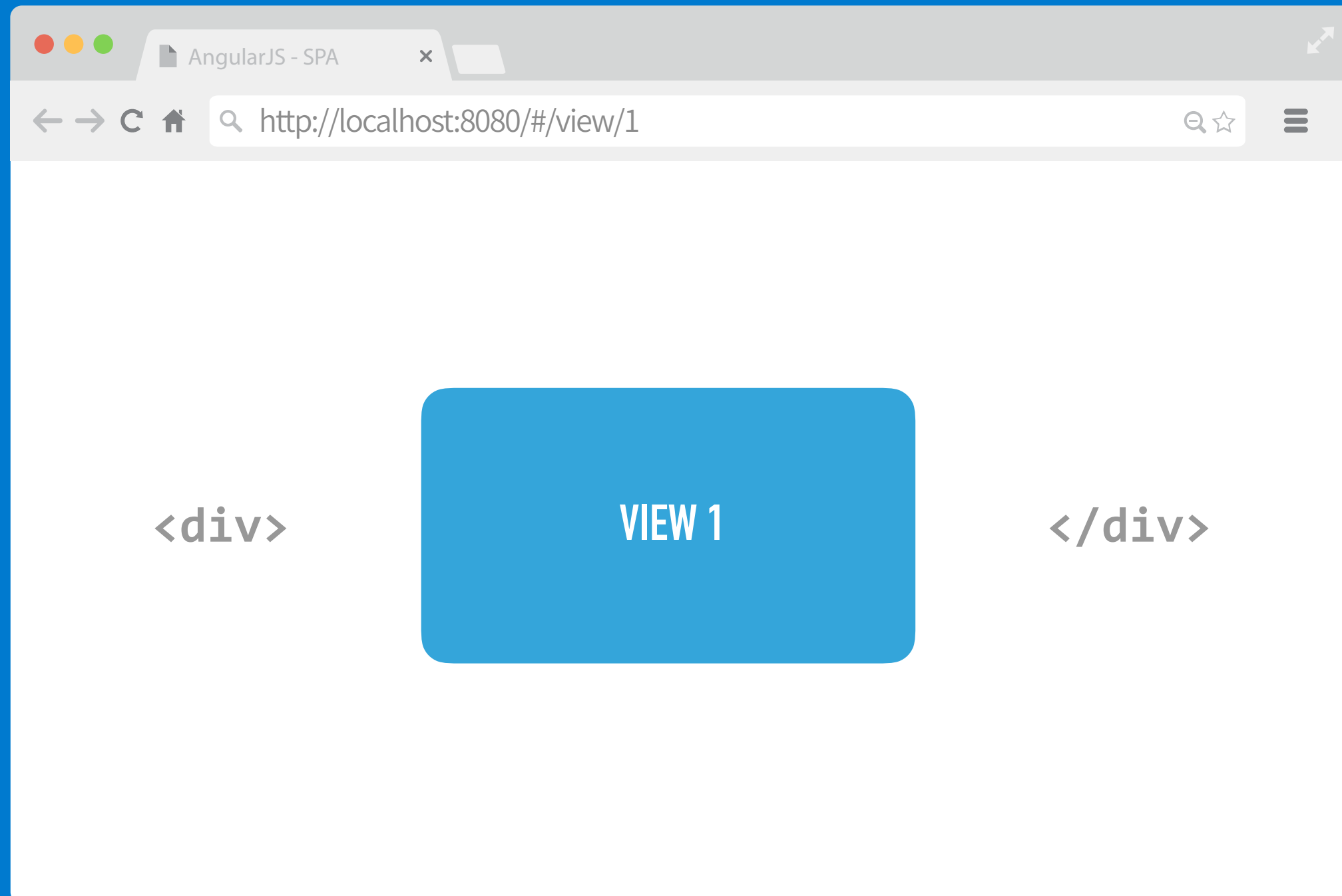
SPA 모델



Single Page Applications



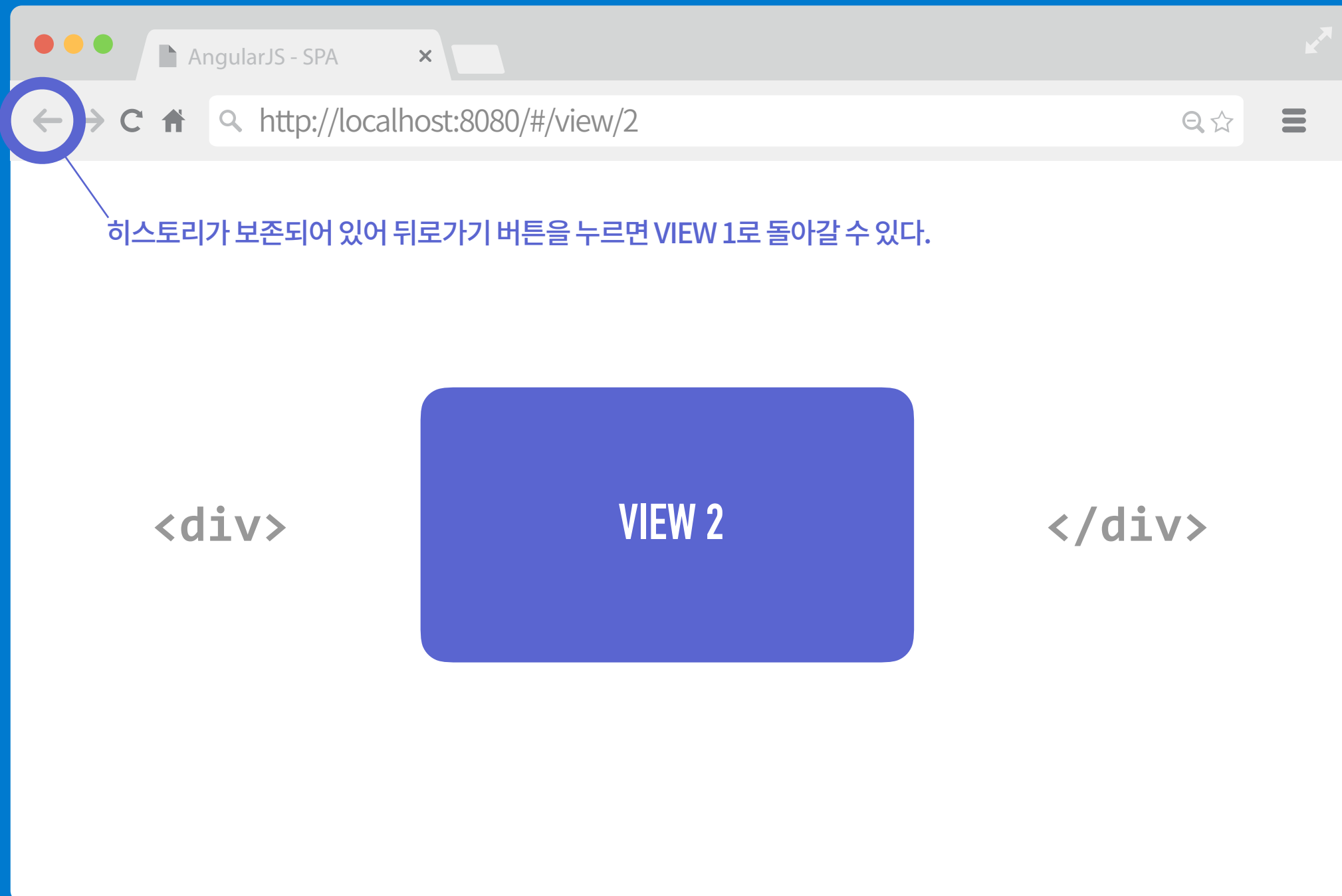
처음 페이지가 렌더링 된 이후, 페이지 전체를 다시 그리는 것이 아니라
서버로부터 전송 받은 일부분의 데이터를 변경해 현재 페이지에 다른 뷰(View)를 렌더링



Single Page Applications



처음 페이지가 렌더링 된 이후, 페이지 전체를 다시 그리는 것이 아니라
서버로부터 전송 받은 일부분의 데이터를 변경해 현재 페이지에 다른 뷰(View)를 렌더링



Single Page Applications



SPA(Single Page Applications) 구현을 위해서는 다양한 기술이 요구된다.



- 문서객체모델 조작(DOM Manipulation)
- 히스토리(기록, History)
- 라우팅(절차, Routing)
- 비동기 통신(Ajax)
- 데이터 연결(Data Binding)
- ...

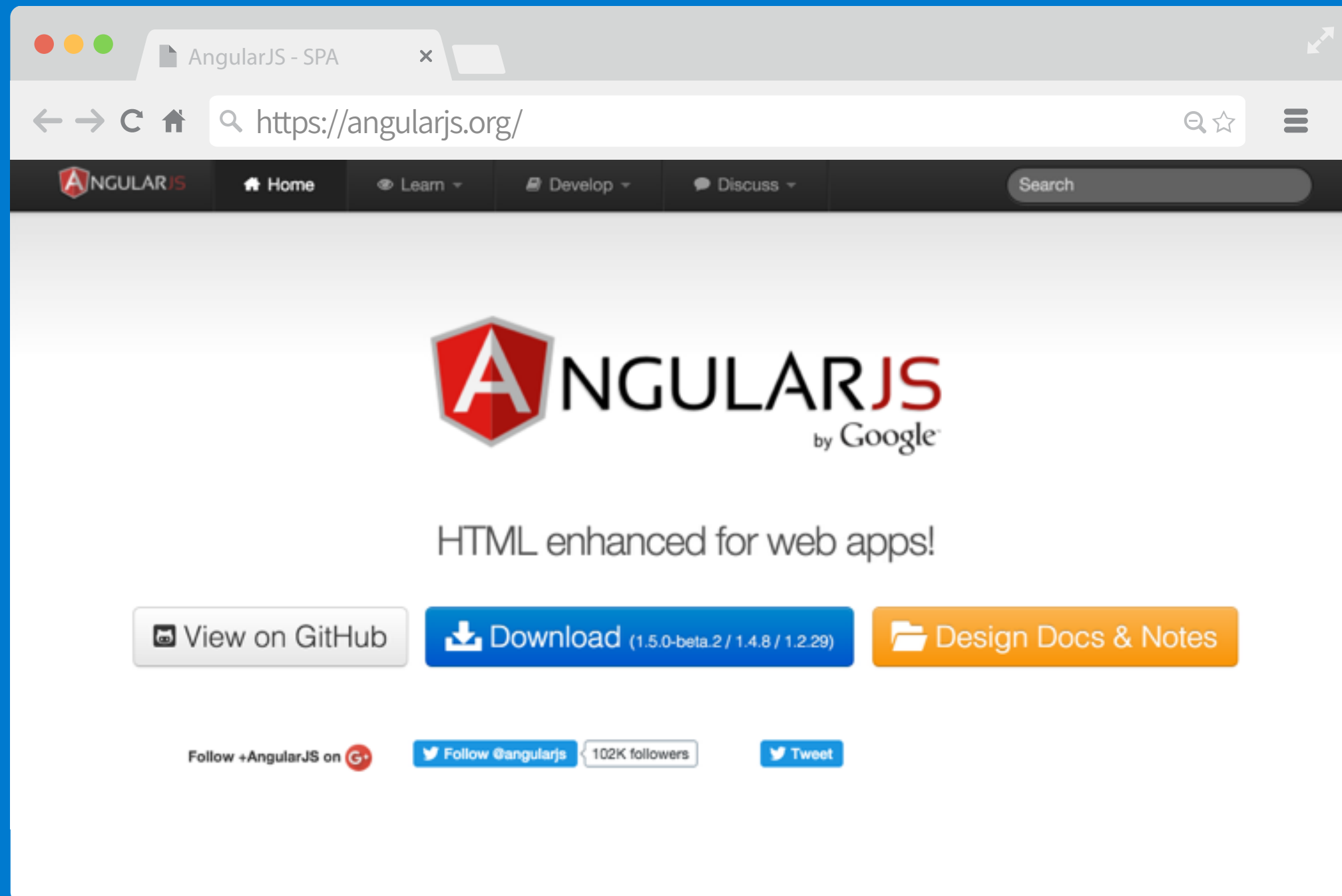


AngularJS

about SPA Framework



AngularJS는 SPA를 구현하기 위한 모든 기능을 제공하는 프레임워크



AngularJS - Full Featured SPA Framework



Model	Directives	Data Binding
View	Expression	Validation
Controller	Services	Testing
Template	Factories	Views
History	Routing	jqLite
Depedency Injection		



Miško Hevery의 개인 프로젝트로 시작된 AngularJS는 초기에 웹 개발자가 아닌, HTML을 사용할 줄 아는 디자이너를 대상으로 개발되었다. (2009년)

Google Feedback

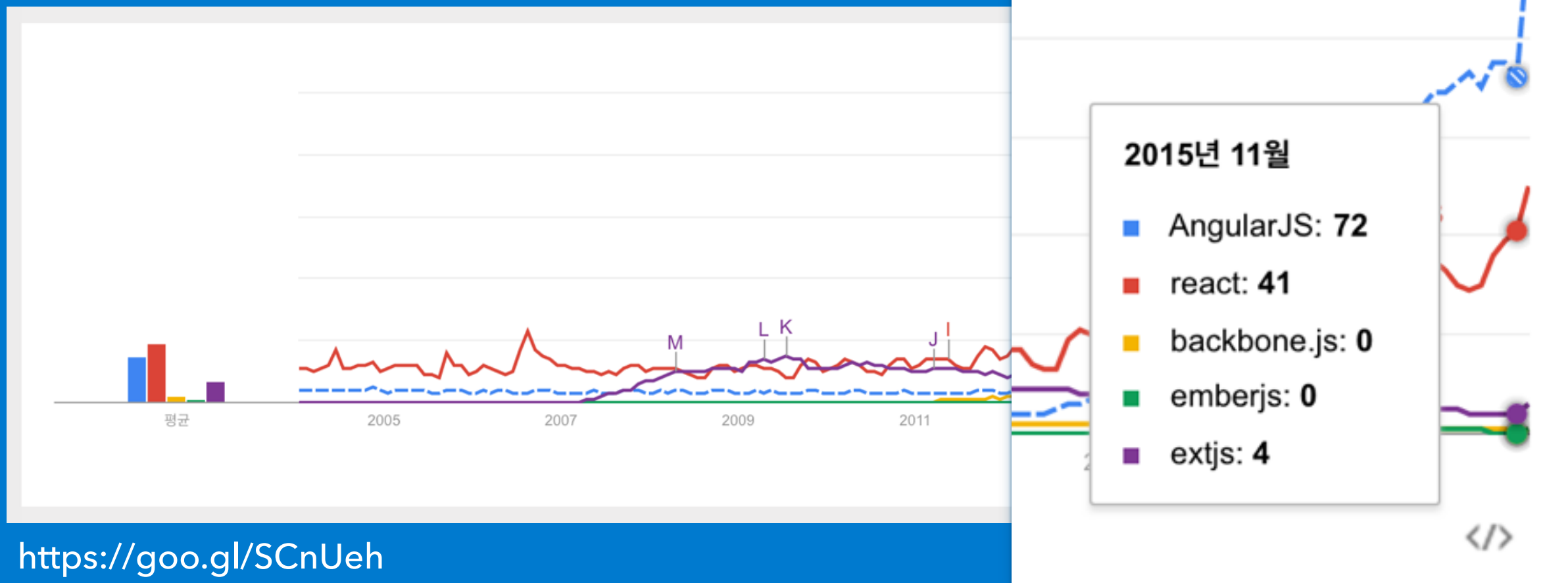
		기간	코드
	GWT	3명 x 6개월	약 17,000 라인
	AngularJS	1명 x 3주	약 1,000 라인

AngularJS - Growth



Google의 후광에 힘입은 강력한 마케팅으로 2012년 이후 가파르게 급성장

Google



<https://goo.gl/SCnUeh>



프론트-엔드/백-엔드 개발자가 AngularJS 프레임워크를 선호하는 이유



- 코드 일관성 유지 (Maintain Coding Style Consistency)
- MVC 구조 (MVC Structure Pattern)
- 코드양 감소 (Less Code)
- 코드 재사용 (Re-Useable Pattern)
- 양방향 데이터 바인딩 (Two-way Data Binding)
- 다양한 모듈 확장 (Modules Extension)



AngularJS

Getting Started

AngularJS - STEP 1



AngularJS 파일은 로컬/호스팅(CDN) 중 선택하여 호출

Local

```
<script src="js/vender/angular-1.5.1.min.js"></script>
```

Hosted Version (CDN)

```
<script src="http://ajax.googleapis.com/ajax/libs/angularjs/1.5.1/angular.min.js"></script>
```

Bower Management 

AngularJS - STEP 2-1



HTML 문서(<html> 요소)에 ngApp 디렉티브(지시자, Directive)를 추가

※ data-* 형식으로 디렉티브 추가 권장(웹표준 준수)

Web Non-Standard

```
<html lang="ko-KR" ng-app>
```

Error Attribute `ng-app` not allowed on element `html` at this point.

[From line 1, column 16; to line 2, column 26](#)

```
TYPE html>↵<html lang="ko-KR" ng-app>↵<head
```

Attributes for element [html](#):

[Global attributes](#)

[manifest](#)

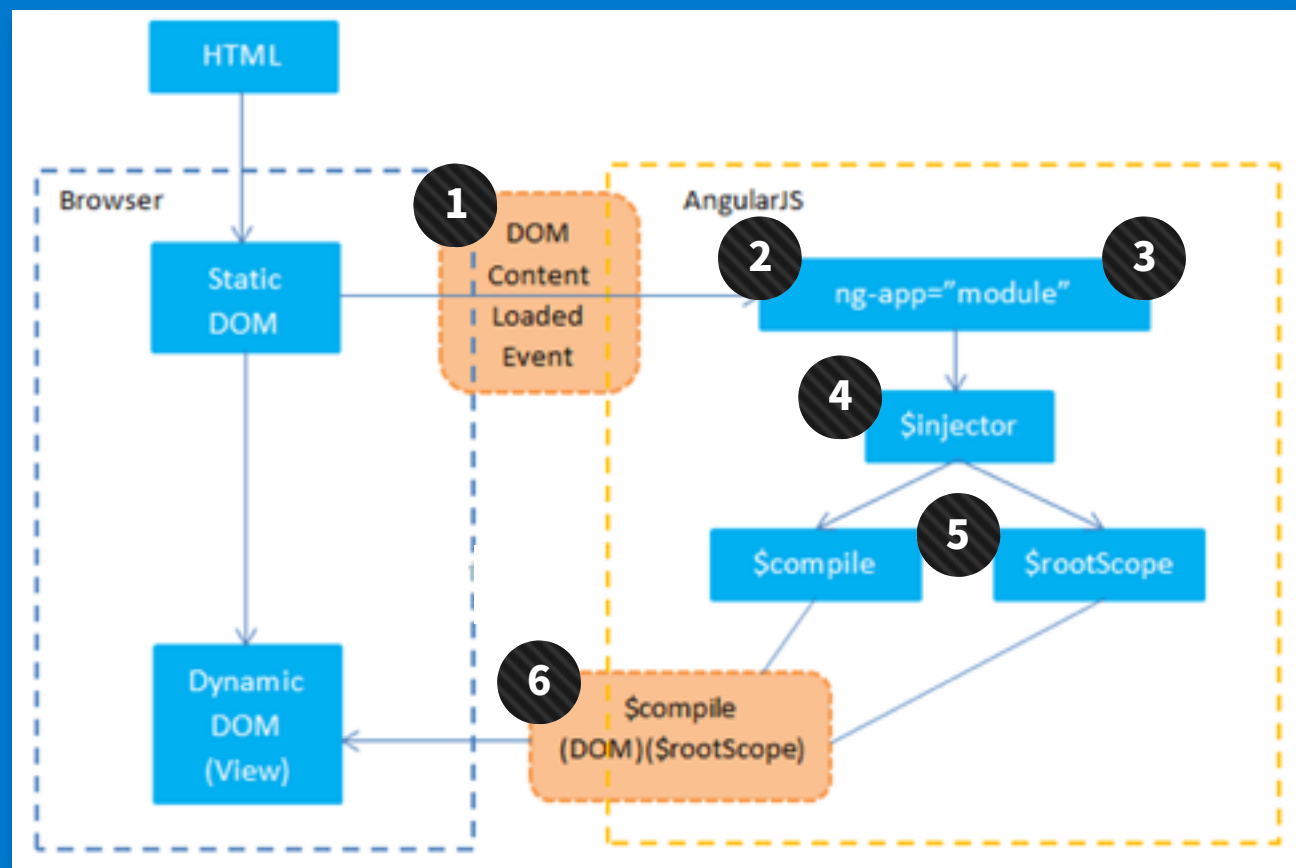
Web Standard

```
<html lang="ko-KR" data-ng-app>
```



AngularJS 부트스트랩(Bootstrap) 프로세스

정적인 HTML 문서를 AngularJS 웹 애플리케이션으로 동작시키는 일련의 과정



docs.angularjs.org/guide/bootstrap

1. angularJS 실행
DOMContentLoaded Event
`document.readyState === "complete"`
2. ngApp 디렉티브 탐색 (애플리케이션 초기화)
auto-bootstrap
3. 모듈(Module)이 있을 경우, 읽어들이м (옵션)
4. \$Injector 설정 (하나만 존재)
5. \$Compile / \$rootScope 생성
6. DOM 렌더링 후, View 생성

AngularJS - STEP 3-1



ngModel, ngBind 디렉티브를 사용하여 데이터를 양방향(Two-Way Data Binding)으로 연결한다.

Directive

```
<input type="text" data-ng-model="two_way_binding">  
<p data-ng-bind="two_way_binding"></p>
```

Expression

```
<input type="text" data-ng-model="two_way_binding">  
<p>{{two_way_binding}}</p>
```

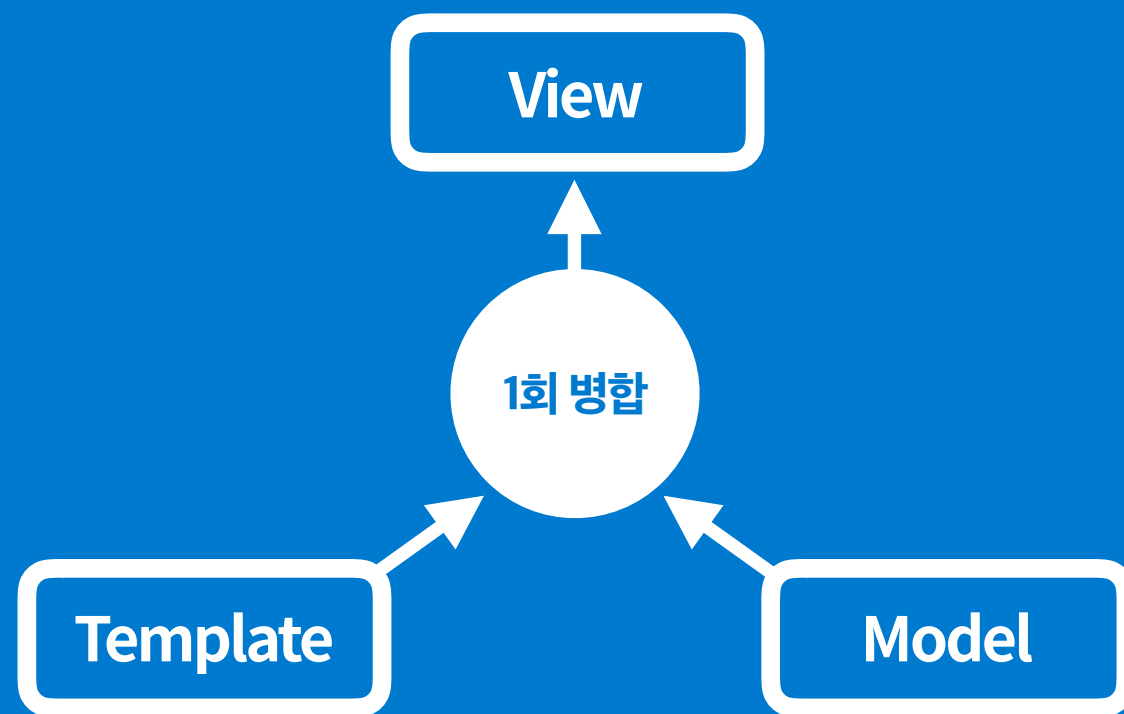
ngBind 디렉티브 대신
익스프레션(표현식, Expression)을
사용할 수도 있다.

데이터 바인딩 익스프레션
Data Binding Expression

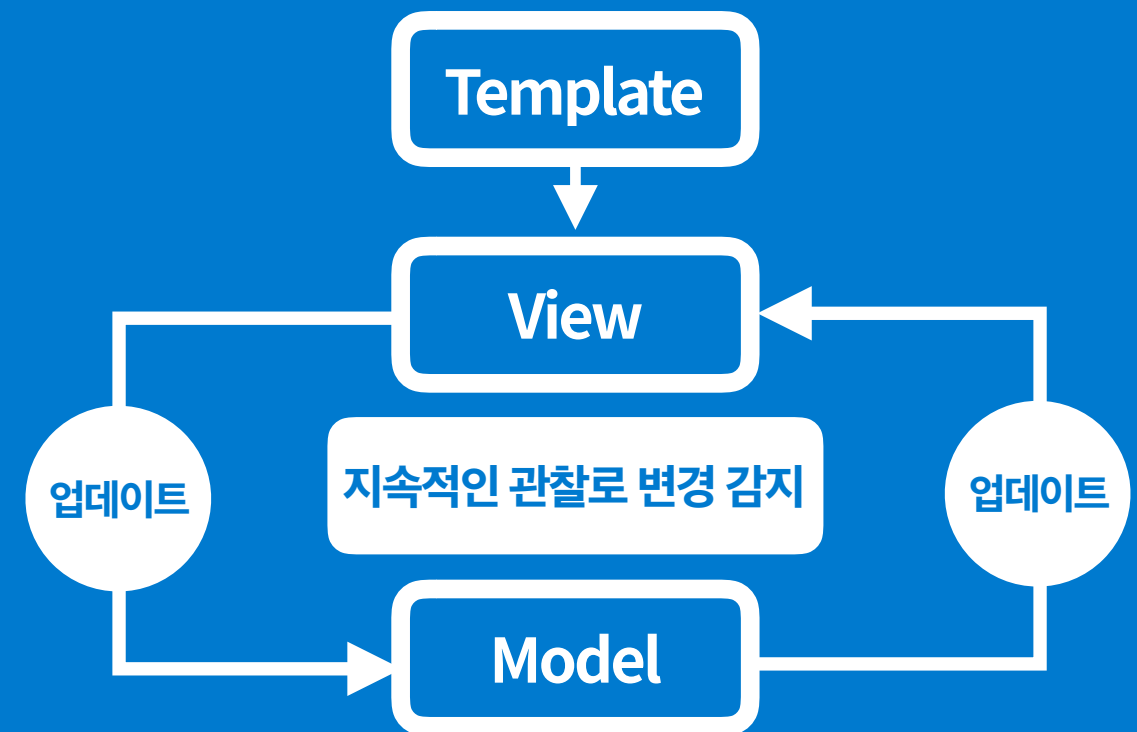


AngularJS는 Two-Way Binding 기술을 사용하여 Model ↔ View 업데이트

One-Way Data Binding



Two-Way Data Binding





DEMO



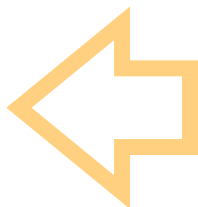
jQuery DEMO

```
yamoo9_example.html
+ yamoo9_example.html
1
<!DOCTYPE html>
<html lang="ko-KR">
<head>
  <meta http-equiv="X-UA-Compatible" content="IE=Edge">
  <meta charset="UTF-8">
  <title>AngularJS Start</title>
  <script src="//code.jquery.com/jquery.min.js"></script>
  <script>
    function init($) {
      var $twb      = $('#twb'),
          $twb_binding = $('#twb-binding');

      $twb.on('keyup', function(event) {
        $twb_binding.text( event.target.value );
      });
    }
    jQuery.noConflict(true)(init);
  </script>
</head>
<body>

  <input type="text" id="twb">
  <p id="twb-binding"></p>

</body>
</html>
```





AngularJS DEMO

```
yamoo9_example.html
+ yamoo9_example.html
1
<!DOCTYPE html>
<html lang="ko-KR" data-ng-app>
<head>
  <meta http-equiv="X-UA-Compatible" content="IE=Edge">
  <meta charset="UTF-8">
  <title>AngularJS Start</title>
  <script src="//ajax.googleapis.com/ajax/libs/angularjs/1.4.8/angular.min.js"></script>
</head>
<body>

  <input type="text" data-ng-model="twb">
  <p data-ng-bind="twb"></p>
  <p>{{twb}}</p>

</body>
</html>
```



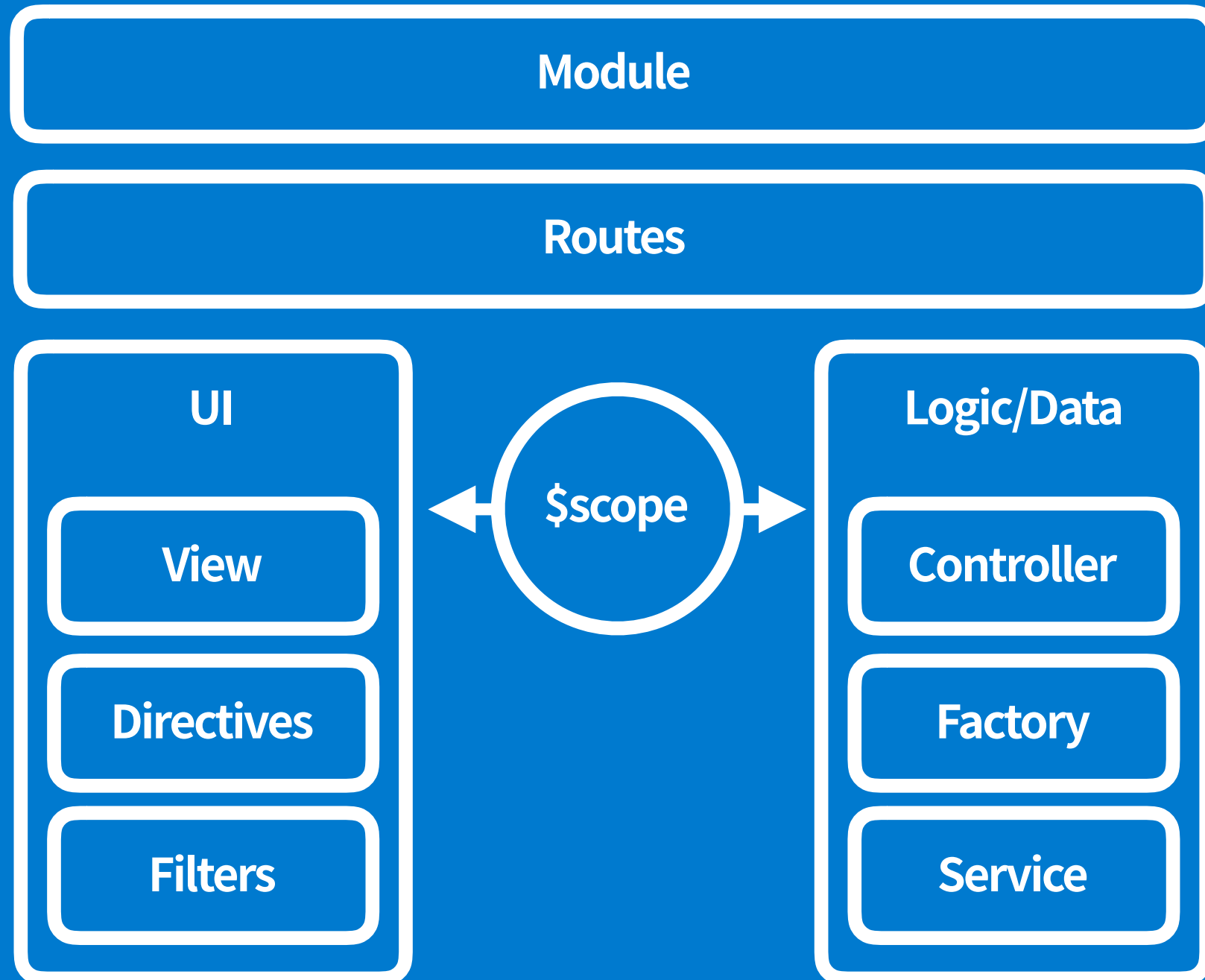
AngularJS

Big Picture

AngularJS - Key Players

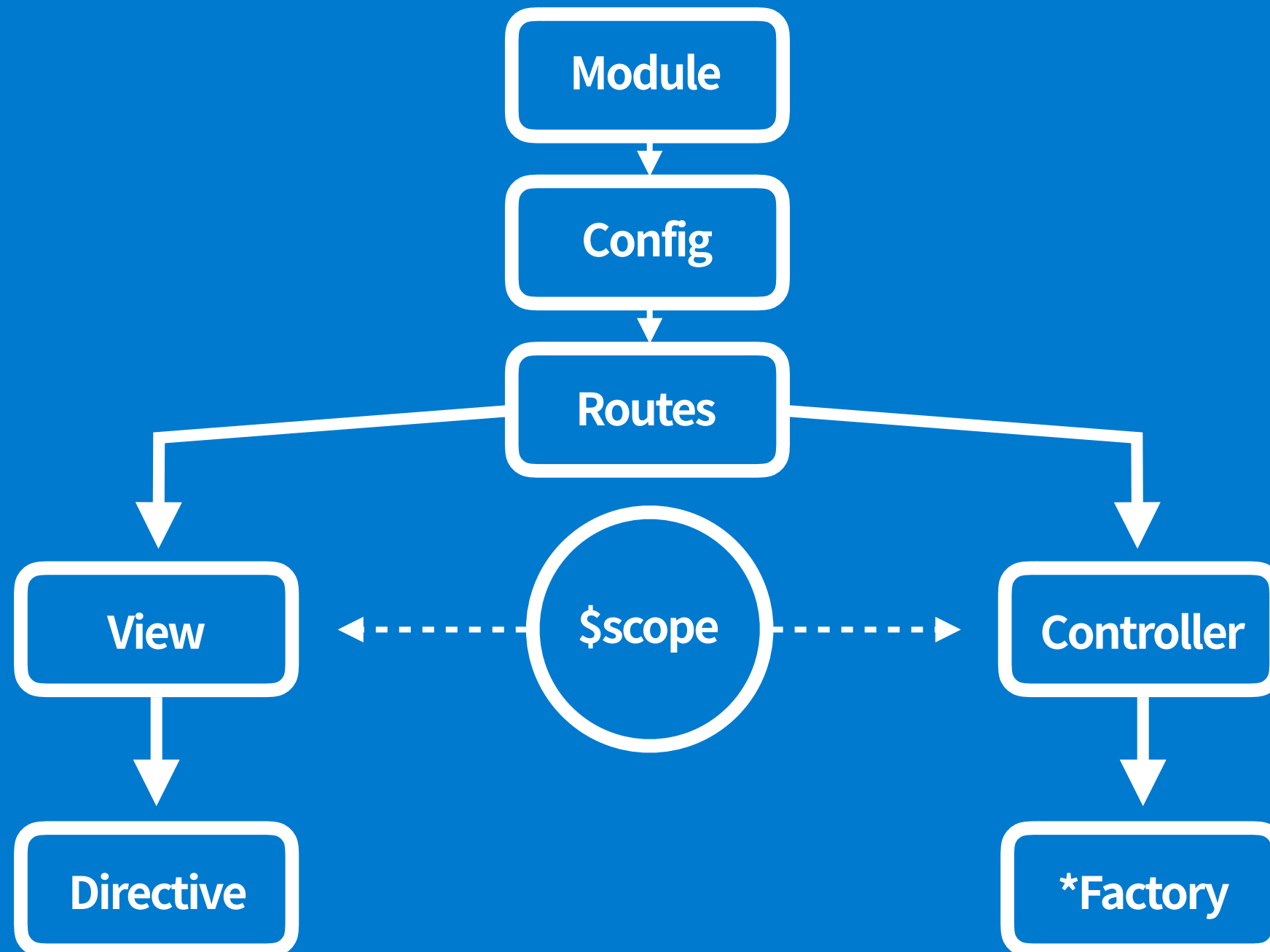


AngularJS를 구성하는 애플리케이션 핵심 구조





AngularJS를 구성하는 애플리케이션 핵심 구조





모듈(Module)은 컨테이너(Container)

Module

- 컨트롤러(Controllers)
- 라우트(Routes)
- 팩토리(Factories)/서비스(Services)
- 디렉티브(Directives)
- 필터(Filters)



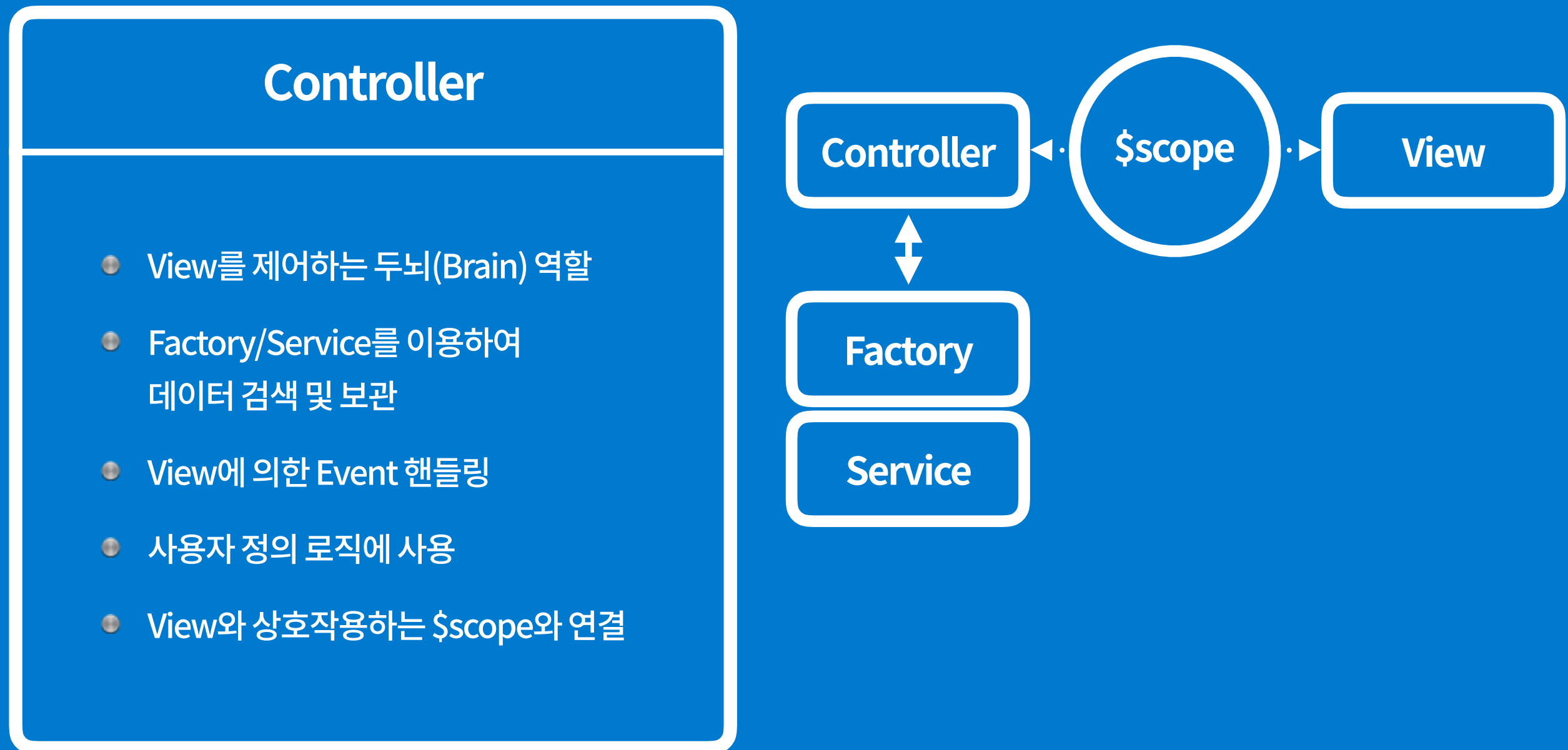
팩토리(Factories)/서비스(Services)는 싱글톤 객체

Factories / Services

- RESTful 서비스에 사용
- Controller 사이 데이터 공유에 사용
- 사용자 정의 로직에 사용
- 싱글톤 객체(Singleton Object)

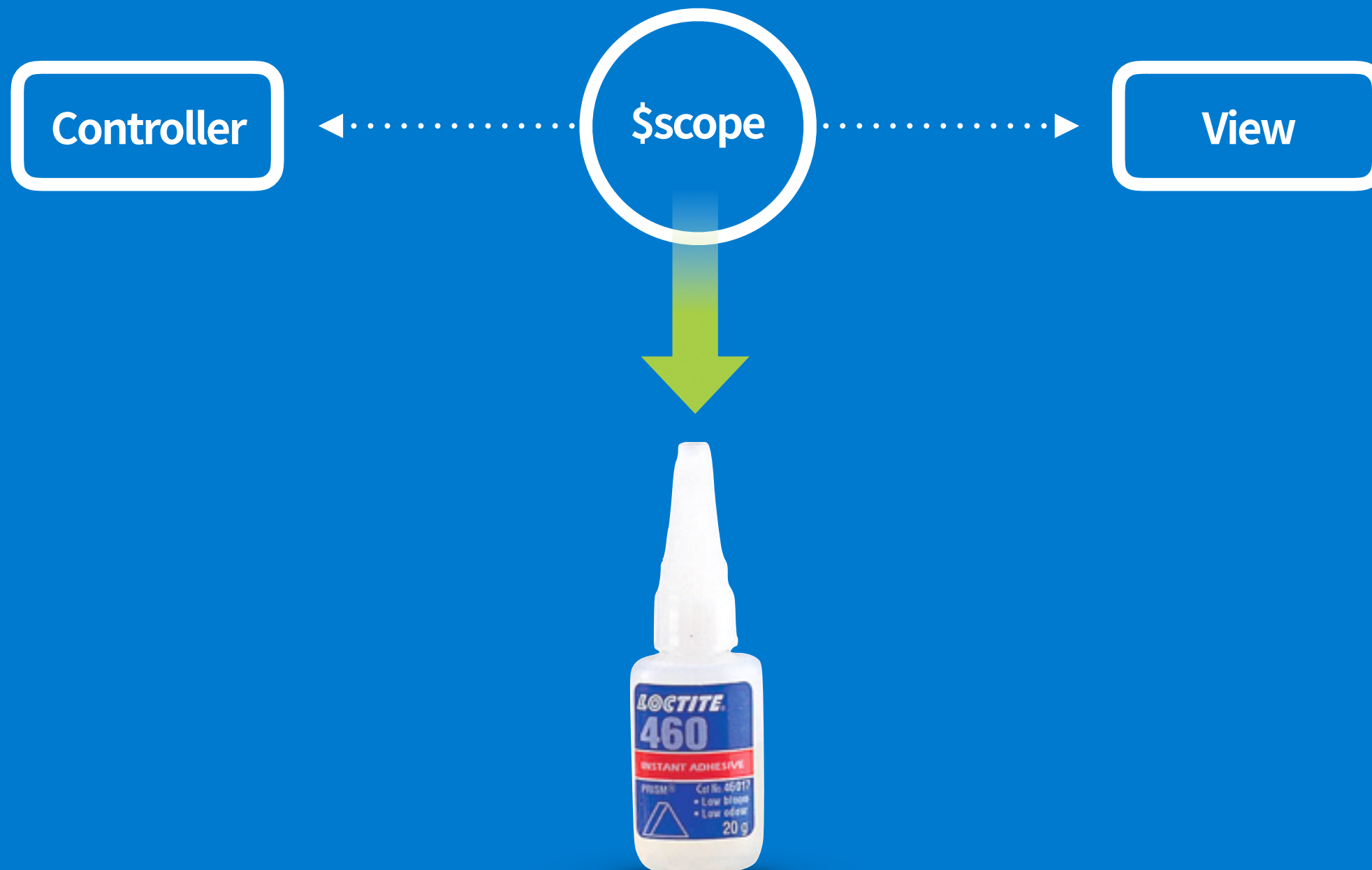


컨트롤러(Controller)는 데이터(Model)/뷰(View) 제어



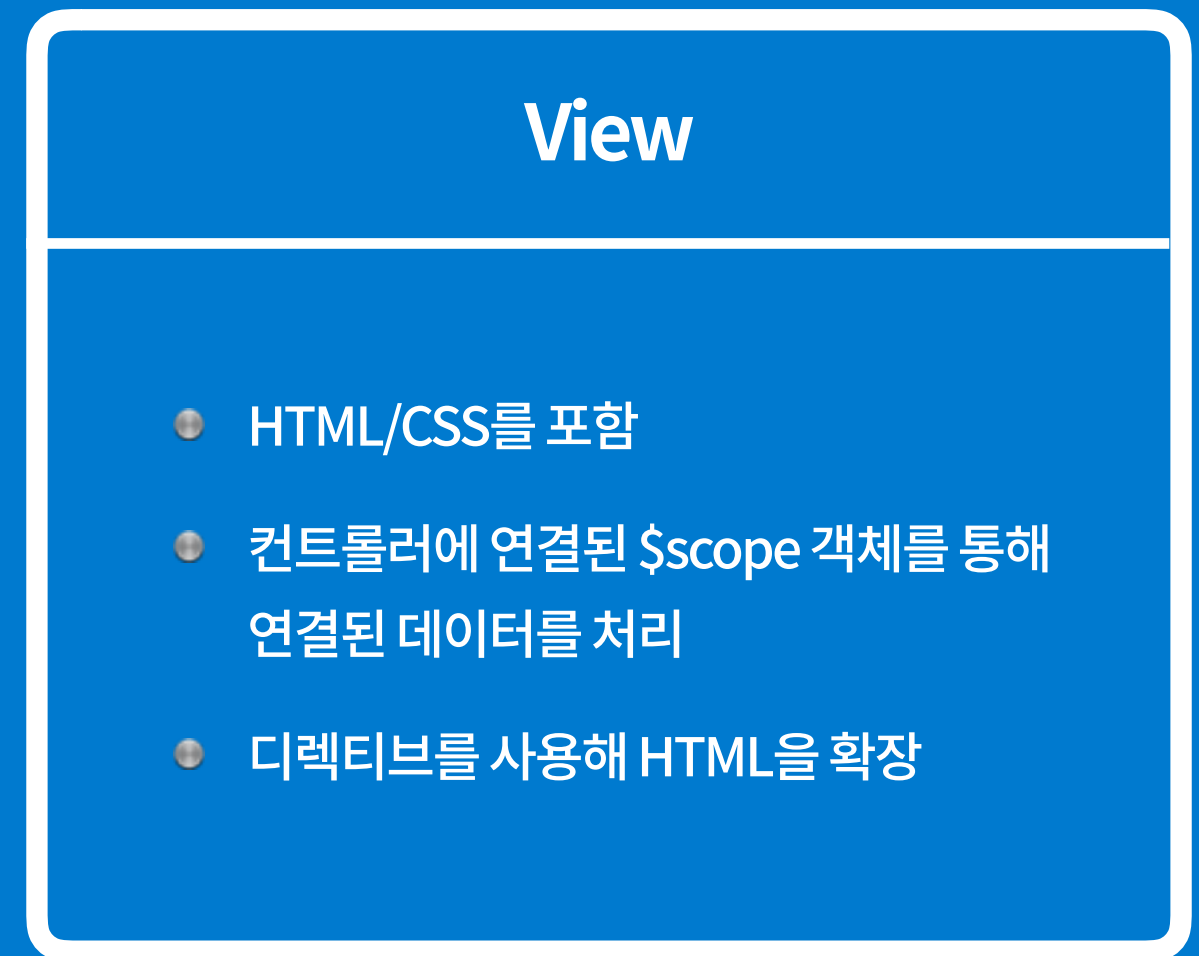
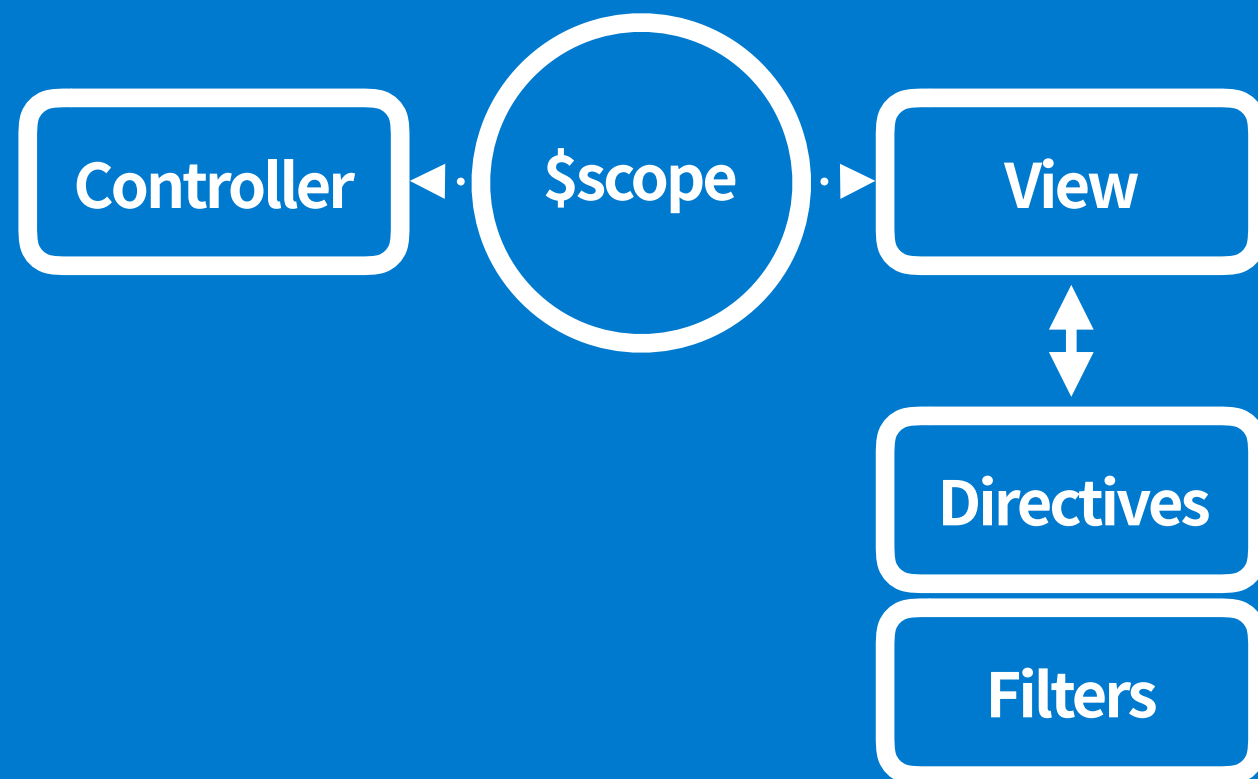


스코프(Scope)는 컨트롤러(Controller)와 뷰-모델(ViewModel)을 연결하는 `접착제`



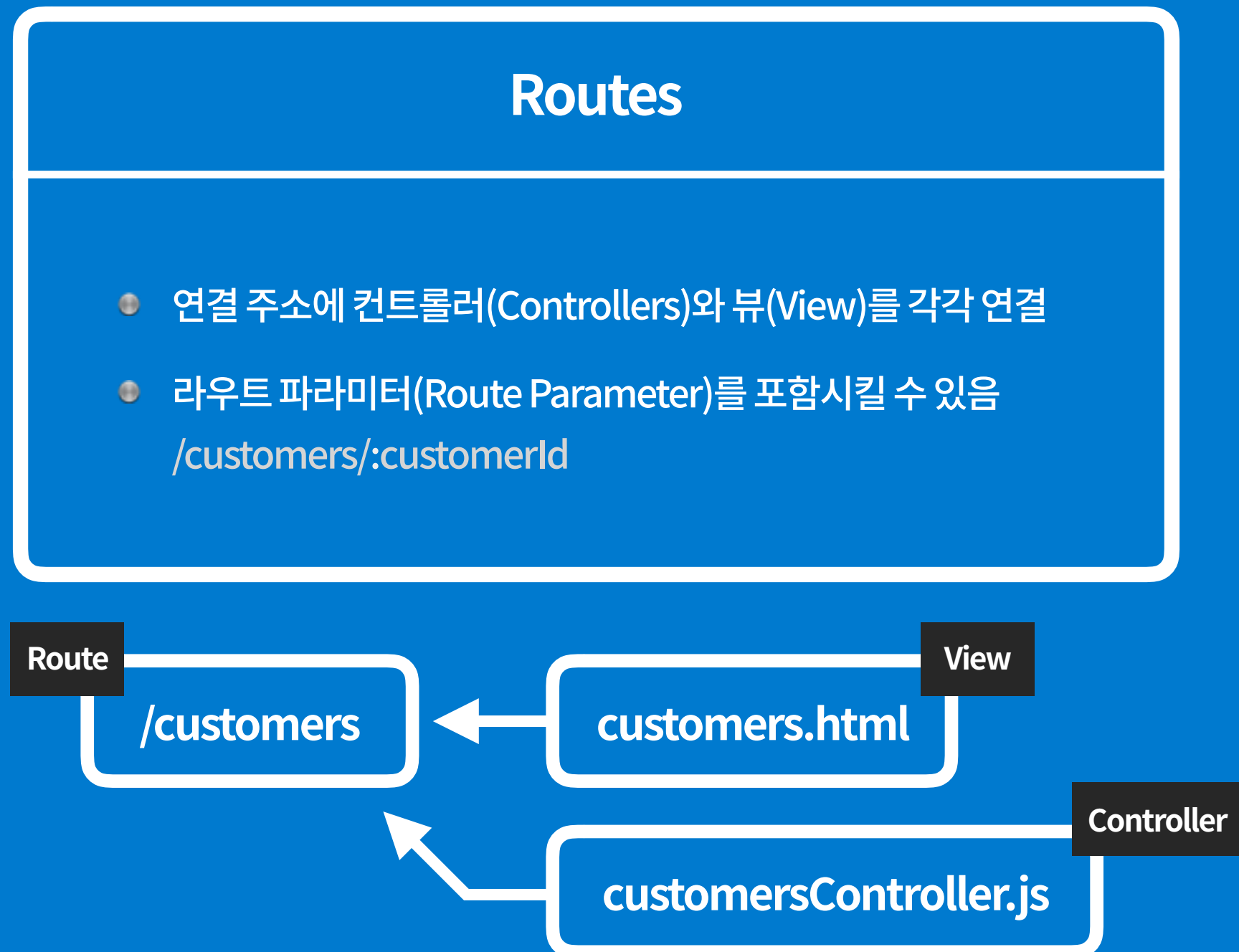


뷰(View)는 사용자 인터페이스(UI)를 렌더링





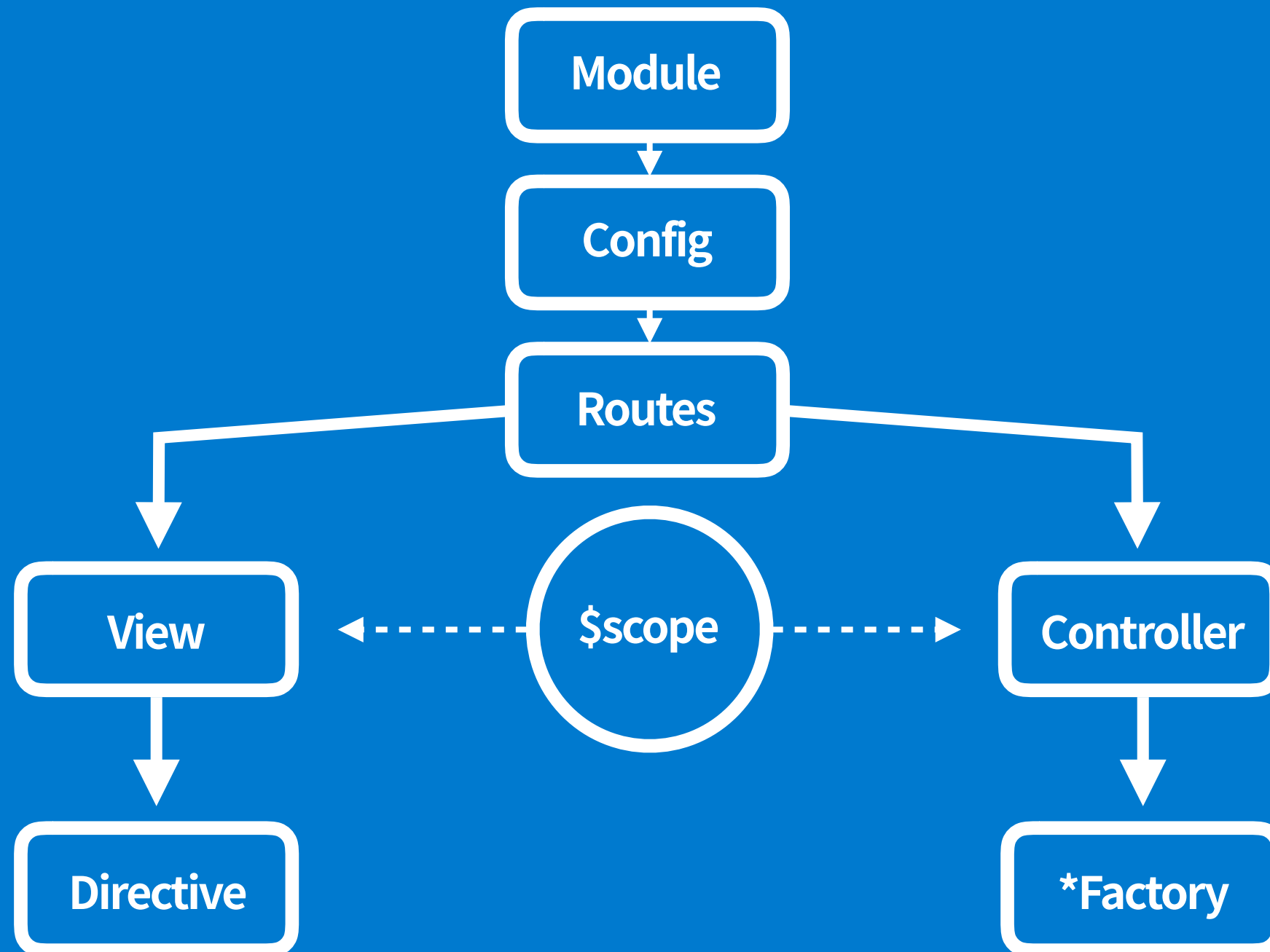
라우트(Routes)는 고유한 연결 주소에 컨트롤러/뷰를 연결



AngularJS - Big Picture



AngularJS를 구성하는 애플리케이션 핵심 구조





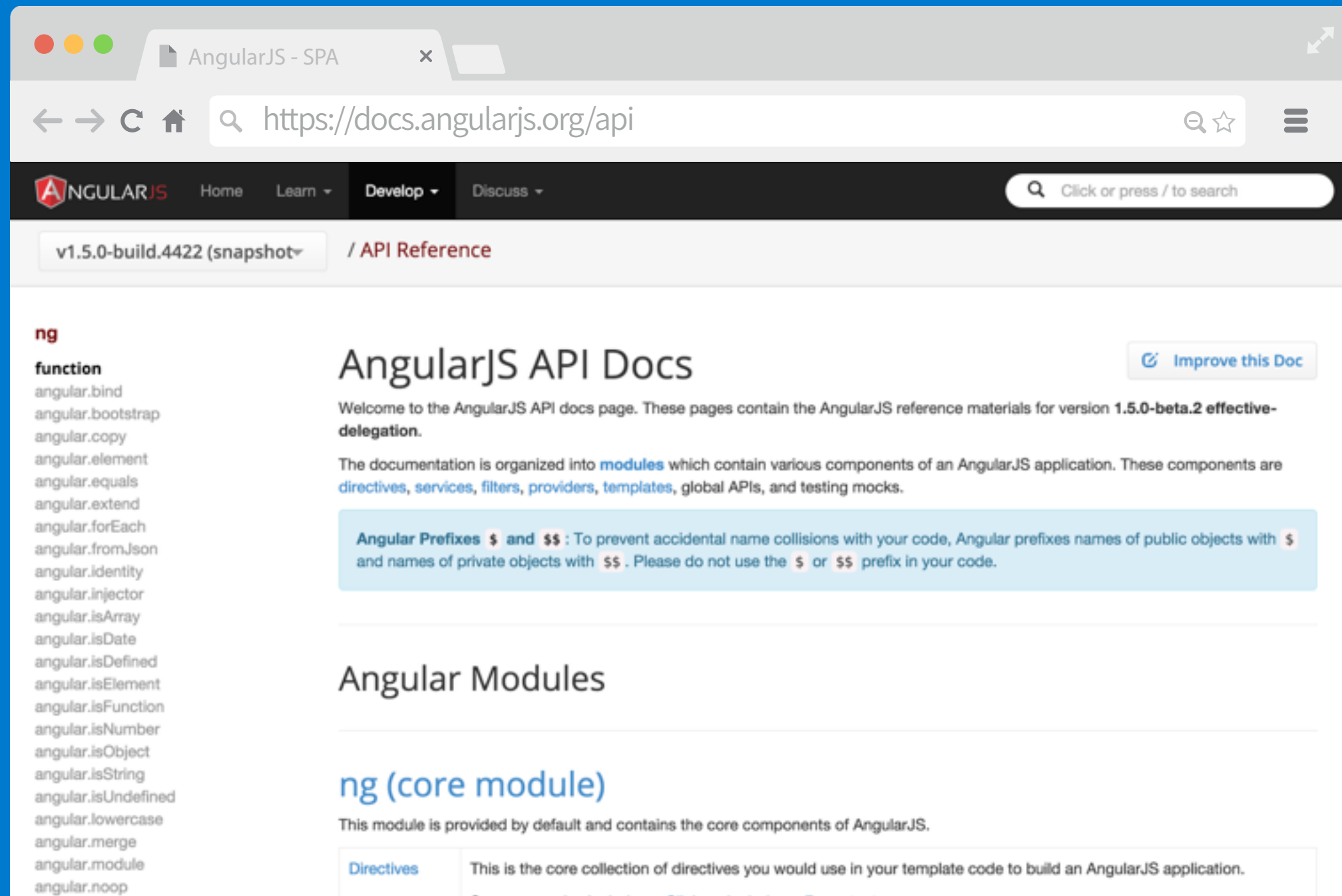
AngularJS

Documentation & Summary

AngularJS - Documentation



AngularJS API Docs: 사용 방법 참고 문서



The screenshot shows a web browser window with the URL `https://docs.angularjs.org/api`. The page title is "AngularJS API Docs". The navigation bar includes links for "Home", "Learn", "Develop", and "Discuss". The version is "v1.5.0-build.4422 (snapshot)". The page content includes a sidebar with a list of functions under the "ng" namespace, such as `angular.bind`, `angular.bootstrap`, `angular.copy`, `angular.element`, `angular.equals`, `angular.extend`, `angular.forEach`, `angular.fromJson`, `angular.identity`, `angular.injector`, `angular.isArray`, `angular.isDate`, `angular.isDefined`, `angular.isElement`, `angular.isFunction`, `angular.isNumber`, `angular.isObject`, `angular.isString`, `angular.isUndefined`, `angular.lowercase`, `angular.merge`, `angular.module`, and `angular.noop`. The main content area has a heading "AngularJS API Docs" and a subheading "Angular Modules". It includes a welcome message and a list of modules, with "ng (core module)" highlighted. A blue box contains a note about Angular prefixes: "Angular Prefixes \$ and \$\$: To prevent accidental name collisions with your code, Angular prefixes names of public objects with \$ and names of private objects with \$\$. Please do not use the \$ or \$\$ prefix in your code." A button "Improve this Doc" is visible in the top right corner.

AngularJS API Docs

Welcome to the AngularJS API docs page. These pages contain the AngularJS reference materials for version **1.5.0-beta.2 effective-delegation**.

The documentation is organized into **modules** which contain various components of an AngularJS application. These components are **directives**, **services**, **filters**, **providers**, **templates**, global APIs, and testing mocks.

Angular Prefixes \$ and \$\$: To prevent accidental name collisions with your code, Angular prefixes names of public objects with \$ and names of private objects with \$\$. Please do not use the \$ or \$\$ prefix in your code.

Angular Modules

ng (core module)

This module is provided by default and contains the core components of AngularJS.

Directives	This is the core collection of directives you would use in your template code to build an AngularJS application.



AngularJS는 SPA 프레임워크



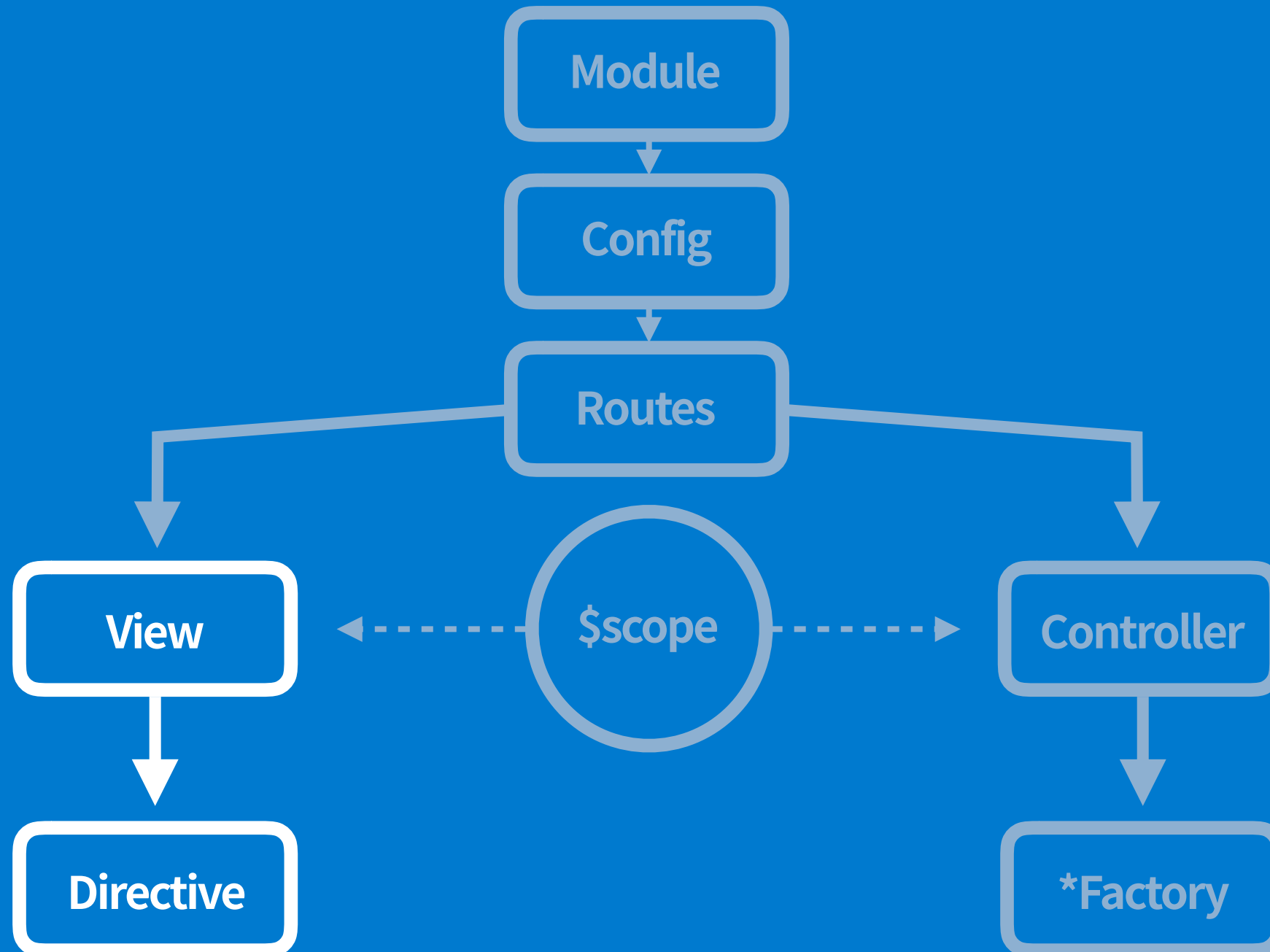
- SPA(Single Page Application) 제작에 최적화된 프레임워크
- 애플리케이션은 한 개 이상의 컨테이너(Modules)로 구성
- 각기 다른 여러 개의 모듈로 구성
 - Controllers
 - Factories/Services
 - View
 - Directives
 - Filters



View

View

Directives / Expressions / Filters





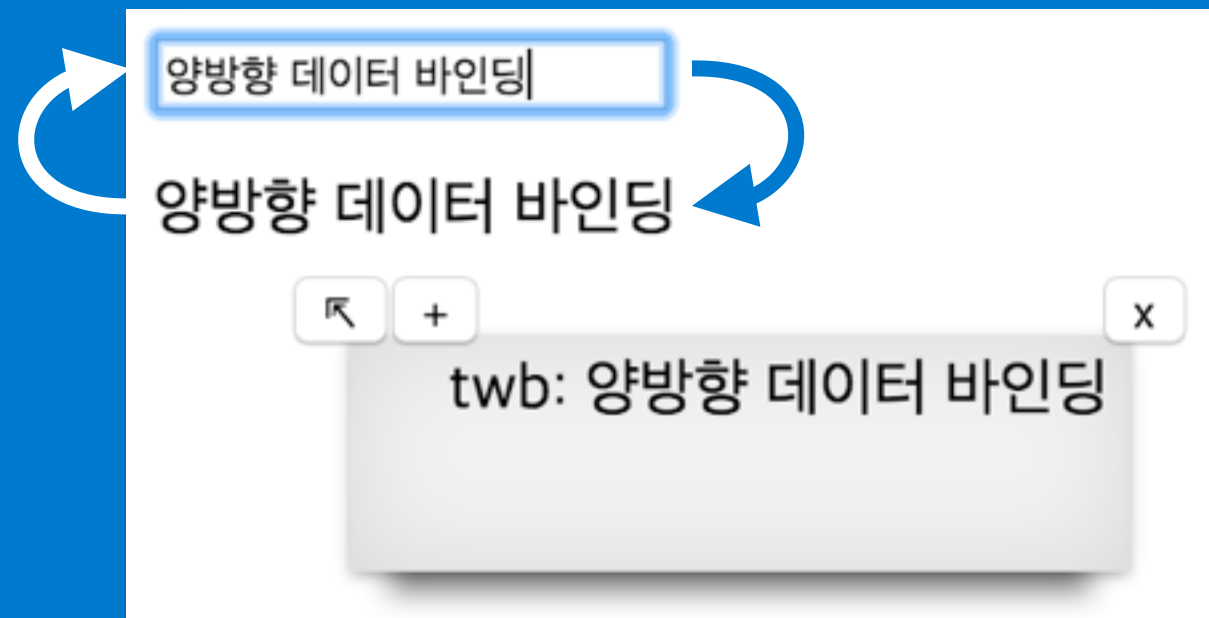
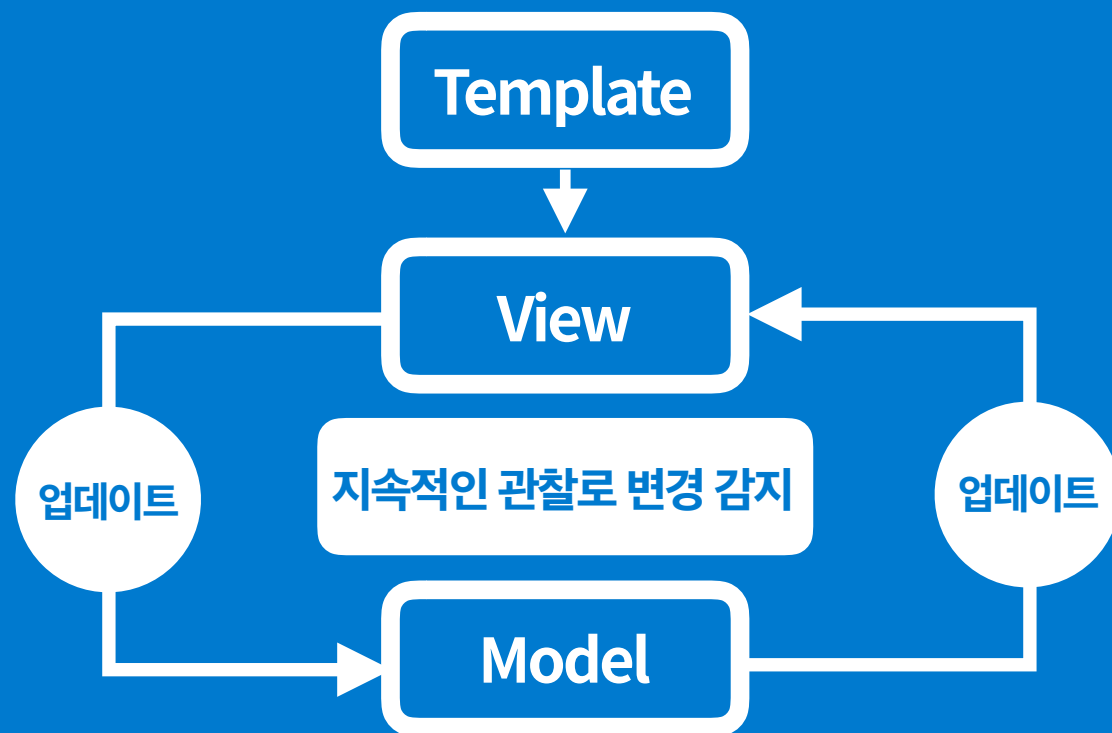
Data Binding

Control-Oriented vs Data-Oriented

Data Binding



Javascript는 데이터 바인딩 기능이 내장(Native Support)되어 있지 않다.
AngularJS의 양방향 데이터 바인딩은 코딩 양을 상당히 줄여 준다.



Control-Oriented vs Data-Oriented



컨트롤 중심 개발 vs 데이터 중심 개발

ID

1. `<input>` 요소를 선택해서 초기 값 입력
`document.querySelector('#user-id').value = 'yamoo9';`
2. 사용자가 데이터 입력(수정)
3. 사용자 '전송' 버튼 클릭
4. `<input>` 요소에 입력된 값을 코드를 통해 가져 오
`var u_id = document.querySelector('#user-id').value;`

1. `<input>` 요소에 속성 추가
`<input data-ng-model='u_id'>`
2. 사용자가 데이터 입력(수정)
3. 속성 값 자동 업데이트!

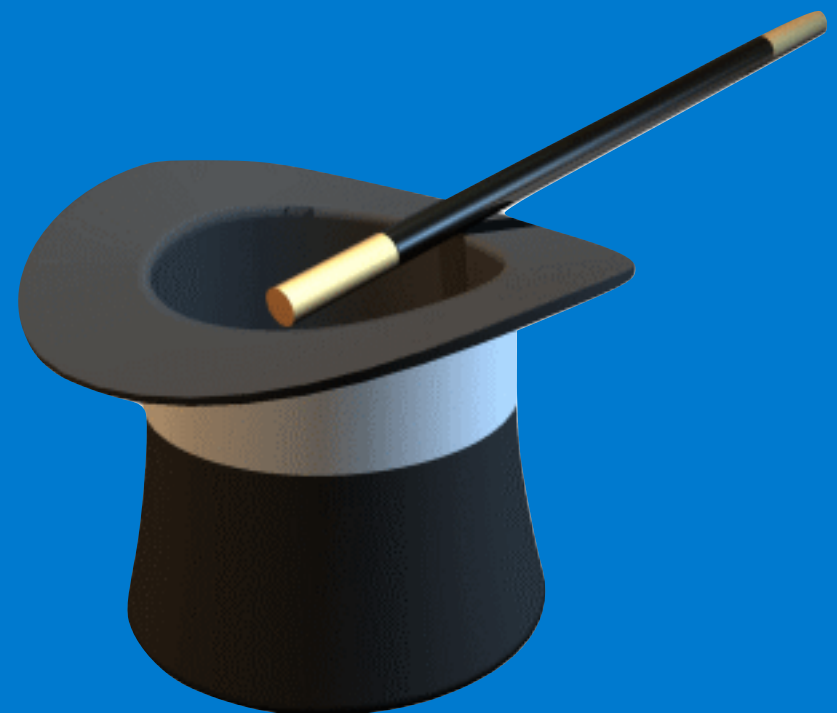


Directives & Expressions



디렉티브(지시어, Directives)는 HTML을 확장한다.
디렉티브는 다음과 같은 일을 수행할 수 있다.

- 문서객체모델 조작 (DOM Manipulation)
- 데이터 연결 (Data Binding)
- 컨트롤러/모듈 (Controllers/Modules)
- 뷰 로딩 (view Loading)
- 스타일링 (CSS)
- 이벤트 (Events)





문서객체모델(DOM) 조작

- ngShow
- ngHide
- ngView
- ngRepeat
- ...

모듈(Modules)/컨트롤러(Controllers)

- ngApp
- ngController
- ...

데이터 바인딩(Data Binding)

- ngInit
- ngModel
- ngBind
- ...

이벤트(Events)

- ngClick
- ngMouseenter
- ngKeydown
- ...

Directives



디렉티브(지시자, Directives)를 정의하는 방법은 다양하다.
하지만.. 웹표준을 준수 하기 위해서는 data-* 접두사를 붙여야 한다.

```
<div ng-hide="is_hidden"></div>
```

```
<div data-ng-hide="is_hidden"></div>
```

```
<ng-view></ng-view>
```



Expressions



익스프레션(표현식, Expression)으로 연결된 마크업({}) 내부에
코드 조각(Code Snippet)이 배치된다.

{{ 'yamoo9' + 9 }}

바인딩 마크업
Binding Markup

익스프레션
Expression



DEMO

Directives & Expressions



ngInit / ngModel / ngClick / ngShow / ngHide

{{...}}

모델 연결
input, textarea, select

```
1  <!DOCTYPE html>
2  <html lang="ko-KR" data-ng-app>
3  <head>
4    <meta http-equiv="X-UA-Compatible" content="IE=Edge">
5    <meta charset="UTF-8">
6    <title>Directives & Expressions - AngularJS</title>
7    <style>
61  </style>
62  <script src="http://ajax.googleapis.com/ajax/libs/angularjs/1.4.8/angular.min.js"></script>
63 </head>
64 <body data-ng-init="bg_color='#5549a6'; is_hidden=true">
65
66   <!--
67   Directives & Expressions
68   ○ ng-init
69   ○ ng-show
70   ○ ng-hide
71   ○ ng-model
72   ○ ng-click
73   -->
74
89
90 </body>
91 </html>
```

Directives & Expressions



ngCloak / ngSwitch / ngClass

은폐: {{}} 코드 노출 방지

FOUC(Flash Of Unstyled Content)

```
yamoo9_example.html
+ yamoo9_example.html
1 <body data-ng-init="member={name: 'yamoo9', log_in: true, is_visible: true, status: 'failed'}">

  <!--
    Directives & Expressions
    ◉ ng-cloak [FOUC(Flash Of Unstyled Content)]
    ◉ ng-switch
    ◉ ng-class
    ◉ ng-show
  -->
  <div class="header-bar" data-ng-cloak data-ng-switch="member.is_visible">
    <div class="notice" data-ng-show="member.is_visible">
      공지사항을 알려드립니다.
    </div>
    <div class="log_in_success" data-ng-switch-when="success">
      <p>안녕하세요. {{member.name}}님. 오늘 하루도 행복하세요.</p>
    </div>
    <div class="log_in_failed" data-ng-switch-when="failed">
      <p>로그인에 실패하셨습니다. :( 다시 로그인해보시길 바랍니다.</p>
    </div>
    <div class="sign_up" data-ng-switch-default>
      <p>회원 가입을 환영합니다. :-)</p>
    </div>
  </div>

</body>
```



<https://gist.github.com/82bc2abf21c6b88b83d1>