

# Google Map을 위한 AngularJS 지시자 만들어 보기

AngularJS 지시자를 이용하여 재사용 가능한 구글 Map의 지시자를 만들어 본다.

## 작성자



고재도  
jeado

## 색인

1 외부 라이브러리를 이용한 지시자 개발

2 Google Map 지시자 만들기

3 속성 Observe 기능 추가

4 References

## Tags

jquery (/tags/jquery/) javascript (/tags/javascript/) html (/tags/html/) grunt (/tags/grunt/) gulp (/tags/gulp/) build (/tags/build/) yeoman (/tags/yeoman/) foundation (/tags/foundation/) bower (/tags/bower/) twitter (/tags/twitter/) framework (/tags/framework/) angularjs (/tags/angularjs/) tutorials (/tags/tutorials/) backbone (/tags/backbone/) getstarted (/tags/getstarted/) underscore (/tags/underscore/) library (/tags/library/) Ext (/tags/Ext/) JS (/tags/JS/) tutorial (/tags/tutorial/) modernizr (/tags/modernizr/) bootstrap (/tags/bootstrap/) quickstart (/tags/quickstart/) angular-di (/tags/angular-di/) animation (/tags/animation/) router (/tags/router/) promise (/tags/promise/) responsive (/tags/responsive/) form (/tags/form/) validation (/tags/validation/) map (/tags/map/) directive (/tags/directive/) file (/tags/file/) test (/tags/test/) ui-bootstrap (/tags/ui-bootstrap/) twitter-bootstrap (/tags/twitter-bootstrap/) d3.js (/tags/d3.js/) visualization (/tags/visualization/) expressjs (/tags/expressjs/) react (/tags/react/) session (/tags/session/) meteor (/tags/meteor/) SEO (/tags/SEO/) REST (/tags/REST/) API (/tags/API/) es2015 (/tags/es2015/) es6 (/tags/es6/) es7 (/tags/es7/) DDP (/tags/DDP/) ngMessages (/tags/ngMessages/) jsx (/tags/jsx/) publish (/tags/publish/) subscribe (/tags/subscribe/) meteor.js (/tags/meteor.js/) node.js (/tags/node.js/) backend (/tags/backend/) reactive (/tags/reactive/) package (/tags/package/) static (/tags/static/) assets (/tags/assets/) public (/tags/public/) flux (/tags/flux/) redux (/tags/redux/) server (/tags/server/) distribution (/tags/distribution/) webpack (/tags/webpack/) development (/tags/development/) optimization (/tags/optimization/) immutable (/tags/immutable/) orm (/tags/orm/) database (/tags/database/) hapi.js (/tags/hapi.js/) hapi (/tags/hapi/)

[ionic \(/tags/ionic/\)](/tags/ionic/) [hybrid-app \(/tags/hybrid-app/\)](/tags/hybrid-app/) [django \(/tags/django/\)](/tags/django/) [fullstack \(/tags/fullstack/\)](/tags/fullstack/) [spa \(/tags/spa/\)](/tags/spa/) [ExpressJS \(/tags/ExpressJS/\)](/tags/ExpressJS/) [NodeJS \(/tags/NodeJS/\)](/tags/NodeJS/) [Sequelize \(/tags/Sequelize/\)](/tags/Sequelize/) [Mocha \(/tags/Mocha/\)](/tags/Mocha/) [Supertest \(/tags/Supertest/\)](/tags/Supertest/) [UnitTest \(/tags/UnitTest/\)](/tags/UnitTest/) [reactjs \(/tags/reactjs/\)](/tags/reactjs/) [this \(/tags/this/\)](/tags/this/) [dom \(/tags/dom/\)](/tags/dom/) [es5 \(/tags/es5/\)](/tags/es5/) [virtualdom \(/tags/virtualdom/\)](/tags/virtualdom/) [modules \(/tags/modules/\)](/tags/modules/) [asynchronous \(/tags/asynchronous/\)](/tags/asynchronous/) [semicolon \(/tags/semicolon/\)](/tags/semicolon/) [nodejs \(/tags/nodejs/\)](/tags/nodejs/) [ECMAScript2015 \(/tags/ECMAScript2015/\)](/tags/ECMAScript2015/) [babel \(/tags/babel/\)](/tags/babel/) [angular \(/tags/angular/\)](/tags/angular/) [electron \(/tags/electron/\)](/tags/electron/) [docker \(/tags/docker/\)](/tags/docker/) [D3 \(/tags/D3/\)](/tags/D3/) [node \(/tags/node/\)](/tags/node/) [express \(/tags/express/\)](/tags/express/) [aws \(/tags/aws/\)](/tags/aws/) [dynamoDB \(/tags/dynamoDB/\)](/tags/dynamoDB/) [chrome \(/tags/chrome/\)](/tags/chrome/) [native \(/tags/native/\)](/tags/native/) [markdown \(/tags/markdown/\)](/tags/markdown/) [vue \(/tags/vue/\)](/tags/vue/) [middleware \(/tags/middleware/\)](/tags/middleware/) [IoT \(/tags/loT/\)](/tags/loT/) [serverless \(/tags/serverless/\)](/tags/serverless/) [python \(/tags/python/\)](/tags/python/) [socket.io \(/tags/socket.io/\)](/tags/socket.io/) [typescript \(/tags/typescript/\)](/tags/typescript/) [laravel \(/tags/laravel/\)](/tags/laravel/) [express.js \(/tags/express.js/\)](/tags/express.js/) [next \(/tags/next/\)](/tags/next/) [nuxt \(/tags/nuxt/\)](/tags/nuxt/) [java \(/tags/java/\)](/tags/java/) [spring \(/tags/spring/\)](/tags/spring/) [github \(/tags/github/\)](/tags/github/) [slack \(/tags/slack/\)](/tags/slack/) [react-native \(/tags/react-native/\)](/tags/react-native/) [data \(/tags/data/\)](/tags/data/) [opensource \(/tags/opensource/\)](/tags/opensource/) [pwa \(/tags/pwa/\)](/tags/pwa/) [php \(/tags/php/\)](/tags/php/) [vuejs \(/tags/vuejs/\)](/tags/vuejs/) [nextjs \(/tags/nextjs/\)](/tags/nextjs/) [react\\_admin \(/tags/react\\_admin/\)](/tags/react_admin/) [webRTC \(/tags/webRTC/\)](/tags/webRTC/) [cd \(/tags/cd/\)](/tags/cd/) [pm2 \(/tags/pm2/\)](/tags/pm2/) [docker\\_registry \(/tags/docker\\_registry/\)](/tags/docker_registry/) [ecr \(/tags/ecr/\)](/tags/ecr/) [twilio \(/tags/twilio/\)](/tags/twilio/) [json\\_schema \(/tags/json\\_schema/\)](/tags/json_schema/) [sql \(/tags/sql/\)](/tags/sql/) [json \(/tags/json/\)](/tags/json/) [web \(/tags/web/\)](/tags/web/) [sdk \(/tags/sdk/\)](/tags/sdk/) [ssl \(/tags/ssl/\)](/tags/ssl/) [portainer \(/tags/portainer/\)](/tags/portainer/) [jwt \(/tags/jwt/\)](/tags/jwt/) [mailu \(/tags/mailu/\)](/tags/mailu/) [sam \(/tags/sam/\)](/tags/sam/) [lambda \(/tags/lambda/\)](/tags/lambda/) [thumbnail \(/tags/thumbnail/\)](/tags/thumbnail/) [redis \(/tags/redis/\)](/tags/redis/) [hyperledger \(/tags/hyperledger/\)](/tags/hyperledger/) [fabric \(/tags/fabric/\)](/tags/fabric/)

# Google Map을 위한 AngularJS 지시자 만들어 보기

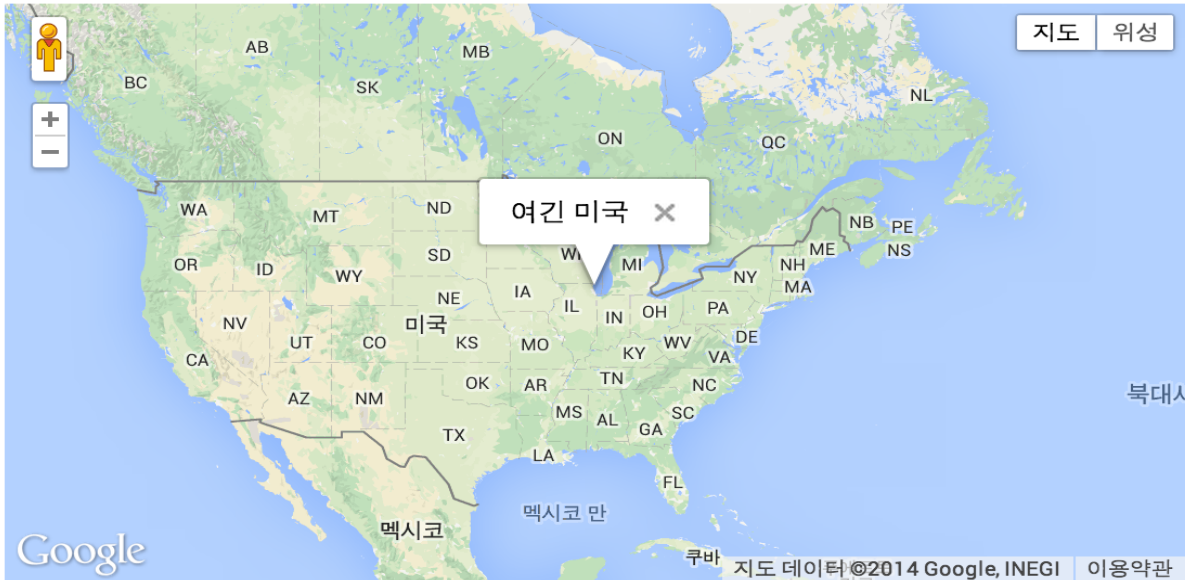
이전에 UI 컴포넌트를 위한 지시자 알아보기 (/tutorials/angularjs/directive\_for\_ui\_component)에서 우리는 지시자를 만드는 법을 간단히 알아 보았다면 이번 튜토리얼은 좀 다른 시각으로 지시자로 UI 컴포넌트를 만드는 법에 대하여 살펴보겠다.

## 외부 라이브러리를 이용한 지시자 개발

AngularJS의 지시자를 이용하여 UI 컴포넌트 개발 방식은 크게 2가지로 볼 수 있다. 하나는 이전 튜토리얼에서 한 방법과 같이 템플릿을 이용하여 처음부터 끝까지 직접 개발하는 방식이고 다른 하나는 바퀴를 맨땅에서부터 다시 발명할 필요 없이<sup>Don't reinvent the wheel</sup> 이미 만들어진 라이브러리를 활용하여 AngularJS 프레임워크와 잘 맞춰 돌아가게 만드는 것이다.

## Google Map 지시자 만들기

Google Map은 자바스크립트를 이용하여 특정 좌표를 중심으로 지도를 보여주거나 다음 그림과 같이 지도안에 별도의 윈도우창을 만들어 정보를 보여줄 수 있다. 우리보고 이런 지도를 만들라면..... 할 말이 없다.



다음 코드는 자바스크립트로 google map을 초기화하고 지도에 서울을 표시하는 코드이다.

```
var mapOptions = {
  center: new google.maps.LatLng(37.561192, 127.030487),
  zoom: 8,
  mapTypeId: google.maps.MapTypeId.ROADMAP
};
var map = new google.maps.Map(document.getElementById("map_canvas"),mapOptions);
```

그럼 우선 지시자를 설계해보자. 지시자의 이름은 gMap으로하고 지도의 중심점은 center 그리고 줌레벨은 zoom으로하겠다. 그리고 각 이름을 태그 속성명으로 하고 속성값을 통하여 값을 받겠다. 그러면 다음과 같이 구글맵 지시자를 정의할 수 있을 것이다.

```
<g-map center="[37.561192, 127.030487]" zoom="8"></g-map>
```

이제 실제 지시자를 구현해보자. google-map.js라는 파일을 만들어 다음 코드와 같이 작성한다. 전체 코드는 GitHub web-angular-sample 프로젝트의 google-map Branch (<https://github.com/jead0/web-angular-sample/tree/google-map>)에서 google-map.html과 gmap 폴더를 확인하면 된다.

```
angular.module('ngGMap', [])
.directive('gMap', ['$timeout', function ($timeout) {
  return {
    restrict: 'EA',
    link: function (scope, iElement, iAttrs) {
      // 지도를 담고 있는 div요소 생성과 스타일 지정
      var el = document.createElement("div");
      el.style.width = "100%";
      el.style.height = "100%";
      iElement.prepend(el);

      // 구글 맵을 만든다.
      var map = new google.maps.Map(el, {});
      // 옵션을 정의
      var mapOptions = {
        center: new google.maps.LatLng(37.561192, 127.030487),
        zoom: 8,
        mapTypeId: google.maps.MapTypeId.ROADMAP
      };

      // 부분적으로만 그려지는 것을 방지하기 위한 코드
      $timeout(function() {
        google.maps.event.trigger(map, "resize");
      });

      // 옵션 적용
      map.setOptions(mapOptions);
    }
  };
}]);
```

이제 위 코드를 사용할 html 파일을 다음과 같이 만든다.

```
<!DOCTYPE html>
<html ng-app="demoApp">
  <head>
    <meta charset="UTF-8">
    <title>AngularJS Google Map Directive 데모 앱</title>
    <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/3.2.0/css/bootstrap.min.css (https://maxcdn.bootstrapcdn.com/bootstrap/3.2.0/css/bootstrap.min.css)">
    <style>
      .map-container { height: 300px; width: 300px; margin: 20px; padding: 0; border: 1px solid #000; }
    </style>
    <script src="https://maps.googleapis.com/maps/api/js?v=3.exp&libraries=weather,visualization,panoramio (https://maps.googleapis.com/maps/api/js?v=3.exp&libraries=weather,visualization,panoramio)"></script>
    <script type="text/javascript" src="libs/angular/angular.min.js"></script>
    <script src="gmap/directive/google-map.js"></script>
    <script>
      angular.module('demoApp', ['ngGMap']);
    </script>
  </head>
  <body>
    <div class="map-container">
      <g-map></g-map>
    </div>
  </body>
</html>
```

브라우저에서 확인하면 확인을 하면 서울을 나타내는 구글 지도가 잘 보일 것이다. 하지만 위 지시자는 앞에서 설계한 것과는 다르게 특정 위치와 줌레벨에 소스에 밝혀있다. 그럼 외부에서 위치와 줌레벨을 받을 수 있도록 아래와 같이 지시자 코드를 수정하자.

```
angular.module('ngGMap', [])
.directive('gMap', ['$timeout', function ($timeout) {
  return {
    restrict: 'EA',
    link: function (scope, iElement, iAttrs) {
      var el = document.createElement("div");
      el.style.width = "100%";
      el.style.height = "100%";
      iElement.prepend(el);

      //iAttrs.center를 이용하여 지시자가 적용된 DOM의 center 속성값을 가지고 온다. 하지만 해당 값이 문자열이라서 JSON.parse를 이용해 배열로 변환한다. zoom은 숫자여야 하기에 Number를 이용하여 숫자로 변환한다.
      var cordi = (iAttrs.center !== undefined) ? JSON.parse(iAttrs.center) : [37.561192, 127.030487],
        zoom = (iAttrs.zoom !== undefined) ? Number(iAttrs.zoom) : 8;

      var map = new google.maps.Map(el, {}),
        mapOptions = {
          center: new google.maps.LatLng(cordi[0], cordi[1]),
          zoom: zoom,
          mapTypeId: google.maps.MapTypeId.ROADMAP
        };

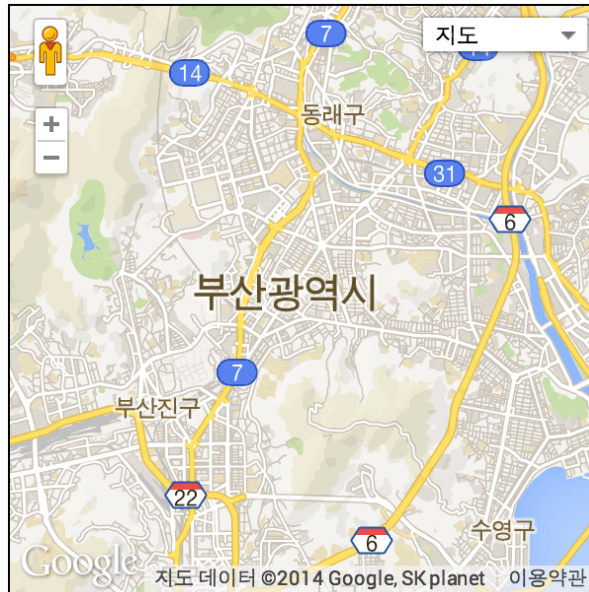
      //생략...

    }
  };
}]);
```

이제 지시자 사용하는 코드를 아래와 같이 부산의 좌표로 변경하고 줌레벨값을 더 높게하여 수정하고 브라우저에서 확인하자.

```
<g-map center="[35.178202, 129.078666]" zoom="12"></g-map>
```

그럼 아래 부산이 더 줌인되서 보이는 구글지도를 확인할 수 있을 것이다.



## 속성 Observe 기능 추가

우리가 만든 gMap 지시자를 좀 더 동적으로 사용되는 시나리오를 생각해보자. 여러 좌표정보를 포함하는 배열이 있고 이를 콤보박스로 표현했다. 해당 콤보박스를 선택하면 선택한 좌표로 지도의 중심점이 변경되는 것이다.

단순히 제이쿼리를 이용해 구현한다고 생각하면 컨트롤 요소에 변경 이벤트 리스너를 구현하고 그때마다 지도의 중심 위치를 변경하는 코드를 작성할 것이다. 하지만 AngularJS의 데이터 바인딩 기능과 지시자 기능이 있기 때문에 쉽게 사용할 수 있는 gMap 지시자를 만들 수 있다. 그럼 google-map.js 파일을 다음과 같이 수정한다.

```
//생략
var map = new google.maps.Map(e1, {}),
    mapOptions = {
        center: new google.maps.LatLng(cordi[0], cordi[1]),
        zoom: zoom,
        mapTypeId: google.maps.MapTypeId.ROADMAP
    };

//center 속성을 감시하고 있다 값이 변경하면 해당 콜백함수를 호출한다. 이때 바뀐 값이 전달된다.
iAttrs.$observe("center",function(value) {
    var latlng = JSON.parse(value);
    //구글 맵의 setCenter 메소드로 바뀐 값으로 맵의 중앙위치를 변경한다.
    map.setCenter({lat:latlng[0], lng:latlng[1]});
});
//위와 마찬가지로 zoom 속성을 감시한다.
iAttrs.$observe("zoom",function(value) {
    map.setZoom(Number(value));
});
//생략
```

이제 콤보박스과 zoom을 제어하는 요소들을 추가한 구글맵지시자 데모페이지를 아래와 같이 수정한다.

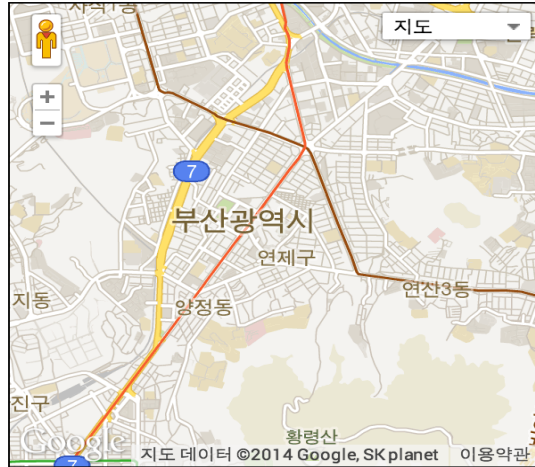
```
<!-- 생략 -->
<script>
angular.module('demoApp', ['ngGMap']).
  controller('demoCtrl', ['$scope', function ($scope) {
    //서울과 부산의 좌표를 가지고 있는 목록이다.
    $scope.cordiList = [
      {
        name : '서울',
        center : [37.561192, 127.030487]
      },
      {
        name : '부산',
        center : [35.178202, 129.078666]
      }
    ];
    //콤보박스에서 선택한 좌표 정보이다.
    $scope.selectedCord = $scope.cordiList[0];
    //지도의 줌레벨이다.
    $scope.zoom = 12;
  }]);
</script>
</head>
<body ng-controller="demoCtrl">
  <form class="form-inline search-container">
    <div class="form-group">
      <select ng-model="selectedCord" class="form-control" ng-options="cord.name for cord
in cordiList" >
        <input type="number" class="form-control" ng-model="zoom">
      </select>
    </div>
  </form>
  <div class="map-container">
    <!-- {{ }} 를 이용하여 데이터 바인딩 처리를 한다. -->
    <g-map center="{{ selectedCord.center }}" zoom="{{ zoom }}"></g-map>
  </div>
</body>
<!-- 생략 -->
```

브라우저에서 데모 페이지를 읽으면 아래 그림과 같은 화면을 볼 수 있다.



부산

13



지금까지 구글 지도 지시자를 만들어 보았는데 사실 더 많은 기능을 추가해야 된다. 이벤트 처리와 인포메이션 윈도우 처리 등... 정말 많은 기능이 필요로 한다. 우리가 원하는 데로 만들수도 있지만 벌써 훌륭한 오픈소스 개발자들이 만들어 공개하고있다. 그래서 몇 가지 지시자를 추천하며 튜토리얼을 맞추겠다.

- Angularjs-Google-Maps (<http://ngmap.github.io/>)
- Angular UI팀의 Google Maps (<https://angular-ui.github.io/angular-google-maps/>)

## References

- angularjs official document (<https://docs.angularjs.org/>)
- 시작하세요 angularjs 프로그래밍 (<http://wikibook.co.kr/beginning-angularjs/>)
- Google Maps JavaScript API v3 (<https://developers.google.com/maps/documentation/javascript/>)

Tags : javascript (/tags/javascript) framework (/tags/framework) angularjs (/tags/angularjs) tutorials (/tags/tutorials) map (/tags/map) directive (/tags/directive)

[Comments](#)[Community](#)[Login](#) 1[Recommend](#)[Tweet](#)[Share](#)[Sort by Best](#)

LOG IN WITH

OR SIGN UP WITH DISQUS ?

Be the first to comment.

ALSO ON HTML5KR FOR KISA

### Mock API를 만들기 위해 json-server 이용하기 (1/2)

1 comment • 2 years ago

[쌤닭](#) — 와...감사합니다.

### PHP 라라벨 5 On Mac - 2.Composer

1 comment • a year ago

[쌤닭](#) — 잘 보고 갑니다~~~

### AWS CloudFormation으로 개발 스택 만들기

1 comment • 2 years ago

[쌤닭](#) — 도움 많이 되었습니다. 감사합니다.

### WebFrameworks.kr - React 소개 및 맛보기

4 comments • 3 years ago

[윤효연](#) — 좋은 글 감사합니다. 질문이 있는데, 단, 브라우저 단에서 DOM의 변화가 일어[Subscribe](#) [Add Disqus to your site](#)[Add Disqus](#)[Add](#)[Disqus' Privacy Policy](#)[Privacy Policy](#)[Privacy Policy](#)

