

Ask Django

여러분의 파이썬/장고 페이스메이커가 되겠습니다.
EP 7. 위젯 만들기 (구글맵 위젯)

지도 API

- 구글 지도
 - JavaScript v3, Android, iOS, Static 지도 등
- 네이버 지도
 - JavaScript v3, Android, iOS
- 다음 지도
 - Web, Android, iOS

지도 사용을 위해서, 각 API 마다 **api key**가 필요합니다. 각 개발자 센터에서 발급받으실 수 있습니다.

코드

blog/forms.py

```
class PostForm(forms.ModelForm):
    class Meta:
        model = Post
        fields = '__all__'
        widgets = {
            'country': AutoCompleteWidget ajax_url=reverse_lazy('country_list')),
            'when': DatePickerWidget,
            'rating': RateItJsWidget,
            'photo': PreviewClearableFileInput,
            'location': LocationWidget,
        }
```

blog/widgets.py

추가 라이브러리없이 직접 구현

```
class LocationWidget(forms.TextInput):
    template_name = 'widgets/location_widget.html'

class Media:
    js = [
        '//code.jquery.com/jquery-2.2.4.min.js',
        '//maps.googleapis.com/maps/api/js', # FIXME: Google Maps JavaScript API 키 적용
    ]

    def build_attrs(self, *args, **kwargs):
        attrs = super().build_attrs(*args, **kwargs)
        attrs['readonly'] = True
        attrs['style'] = 'background-color: #eee; border: 0;'
        return attrs
```

템플릿 1단계 : 지도/마커 띄우기

blog/templates/widgets/location_widget.html

```
{% include "django/forms/widgets/input.html" %}
```

```
<div id="map_{{ widget.attrs.id }}" style="width: 100%; height: 300px;"></div>
```

```
<script>
```

```
$(function() {  
    var adjust = function(latlng) {  
        var pos = latlng.lat() + ',' + latlng.lng();  
        var selector_id = '{{ widget.attrs.id }}';  
        $('#'+ selector_id).val(pos);  
    };  
};
```

```
{% if widget.value %}  
    var center = new google.maps.LatLng('{{ widget.value }}');  
{% else %}  
    var center = new google.maps.LatLng(37.55515726813202, 126.97069409811479); {# default center #}  
    adjust(center);  
{% endif %}
```

```
var map_selector_id = 'map_{{ widget.attrs.id }}';
var map = new google.maps.Map(document.getElementById(map_selector_id), {
    zoom: 4,
    center: center
});

var marker = new google.maps.Marker({
    position: center,
    map: map
});

map.addListener('click', function(e) {
    marker.setPosition(e.latLng);
    adjust(e.latLng);

    // FIXME: 위도, 경도를 통한 주소 찾기
});
});
</script>
```

템플릿 2단계

```
{% include "django/forms/widgets/input.html" %}
<span id="address_{{ widget.attrs.id }}"></span>

<div id="map_{{ widget.attrs.id }}" style="width: 100%; height: 300px;"></div>
<p style="text-align: right;">
    <input id="search_{{ widget.attrs.id }}" type="text" placeholder="위치검색" />
</p>

<script>
function geocode(address, callback) {
    $.get('https://maps.googleapis.com/maps/api/geocode/json', {
        address: address
        // key: '' // FIXME: Google Maps JavaScript API 키 적용
    }).done(function(res) {
        callback(res.results);
    }).fail(function() {
        alert('error');
    });
};
</script>
```



```
<script>
$(function() {
    /* 종락 */

    var marker = new google.maps.Marker({
        position: center,
        map: map
    });

    map.addListener('click', function(e) {
        marker.setPosition(e.latLng);
        adjust(e.latLng);

        var address = e.latLng.lat() + ',' + e.latLng.lng();
        geocode(address, function(results) {
            if ( results.length == 0 ) {
                $('#address_{{ widget.attrs.id }}').html('주소 검색결과가 없습니다.');
```

```
            }
            else {
                $('#address_{{ widget.attrs.id }}').html(results[0].formatted_address);
            }
        });
    });

    /*
     * 보안은 중요하므로 특히 사용자 위치와 같은 민감한 사용자 데이터를 요청에 포함하는 애플리케이션의 경우,
     * 가급적 HTTPS를 사용하는 것이 좋습니다. HTTPS 암호화를 사용하면 애플리케이션의 보안이 강화되고 스누핑이나 변조에 대한 저항력도 강화됩니다.
     */
    $('#search_{{ widget.attrs.id }}').keypress(function(e) {
        if ( e.keyCode == '13' ) { // Enter 입력일 경우
            e.preventDefault();

            var q = $(this).val();
            if ( q.length > 0 ) {
                geocode(q, function(results) {
                    if ( results.length == 0 ) {
                        $('#address_{{ widget.attrs.id }}').html('검색결과가 없습니다.');
```

```
                    }
                    else {
                        $('#address_{{ widget.attrs.id }}').html(results[0].formatted_address);

                        var loc = new google.maps.LatLng(results[0].geometry.location.lat, results[0].geometry.location.lng);
                        map.setCenter(loc);
                        map.setZoom(10);
                        marker.setPosition(loc);
                        adjust(loc);
                    }
                });
            }

            $(this).val(''); // clear input
        }
    });
});
</script>
```

API KEY 적용

API KEY (1)

- key 설정이 없어도 사용할 수는 있으나, 사용한도가 아주 적으며, 금세 제한을 받습니다.
- 발급순서
 - Google Cloud Platform 콘솔에서 프로젝트 생성
 - 좌측 "라이브러리" 메뉴 선택
 - "Google Maps JavaScript API" 선택 => "사용설정" 클릭 => "사용자 인증 정보 만들기" 클릭 => "Google Map JavaScript API" Key 생성
 - "키 제한하기" 를 통해 "HTTP 리퍼러 (웹사이트)" 제한 => 필요한 웹사이트 주소 추가
 - localhost:8000/*
 - 실제 서비스 주소/*
 - 참고) 서비스 주소에는 호스트명과 포트번호가 포함됩니다.

코드 적용

- askdjango/settings.py

GOOGLE_MAP_API_KEY = 'AIzaSyAJs8v8p2JmakkJr9TohAcEw0rIfzPLuOM' # ex) 직접 정의한 설정

- Key 지정 포인트
 - 위젯 Media JavaScript
 - 위젯 템플릿 Geocode

A scenic view of a white-washed town, likely Santorini, with a church in the background. The text "Life is short, use Python 3/Django." is overlaid in a large, elegant, black script font.

*Life is short,
use Python 3/Django.*