Be Beautiful Front and Back

Vue.JS with Django

Who Am I?

- Andrew McCracken
- Theodo since August 2015
- Web Development ~1.5 years
- Python ~6 years

Why Vue.JS and Django?

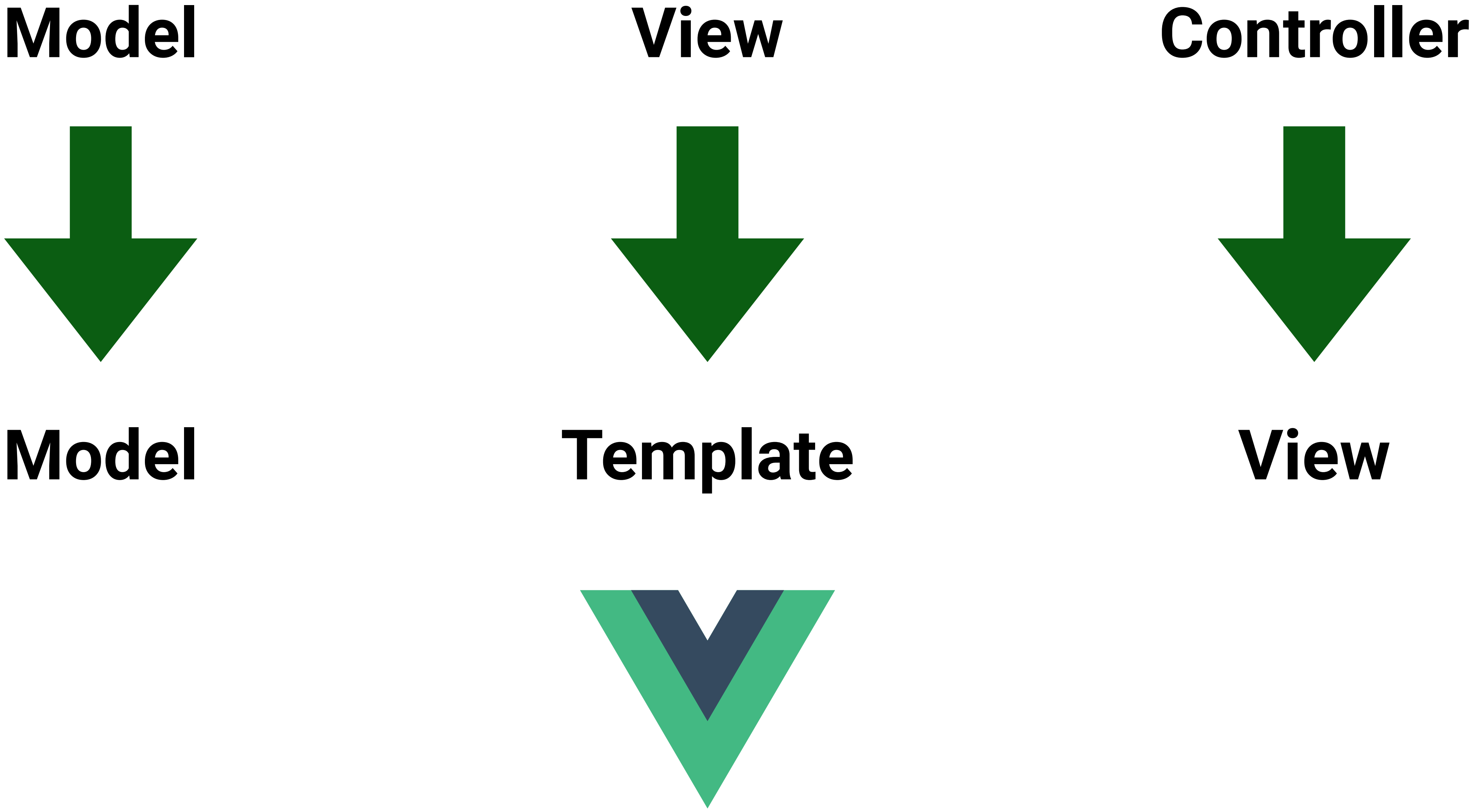
- Powerful JS framework
- Powerful backend framework
- Beautiful front-end
- Beautiful Python backend

What is Django?

- Full stack MVC framework
- Python-based
- Developed at Lawrence Journal-World, Kansas
- Public release in 2005
- v1.9.4

What is Django?

MVC -> MTV



Model

View

Controller

Model

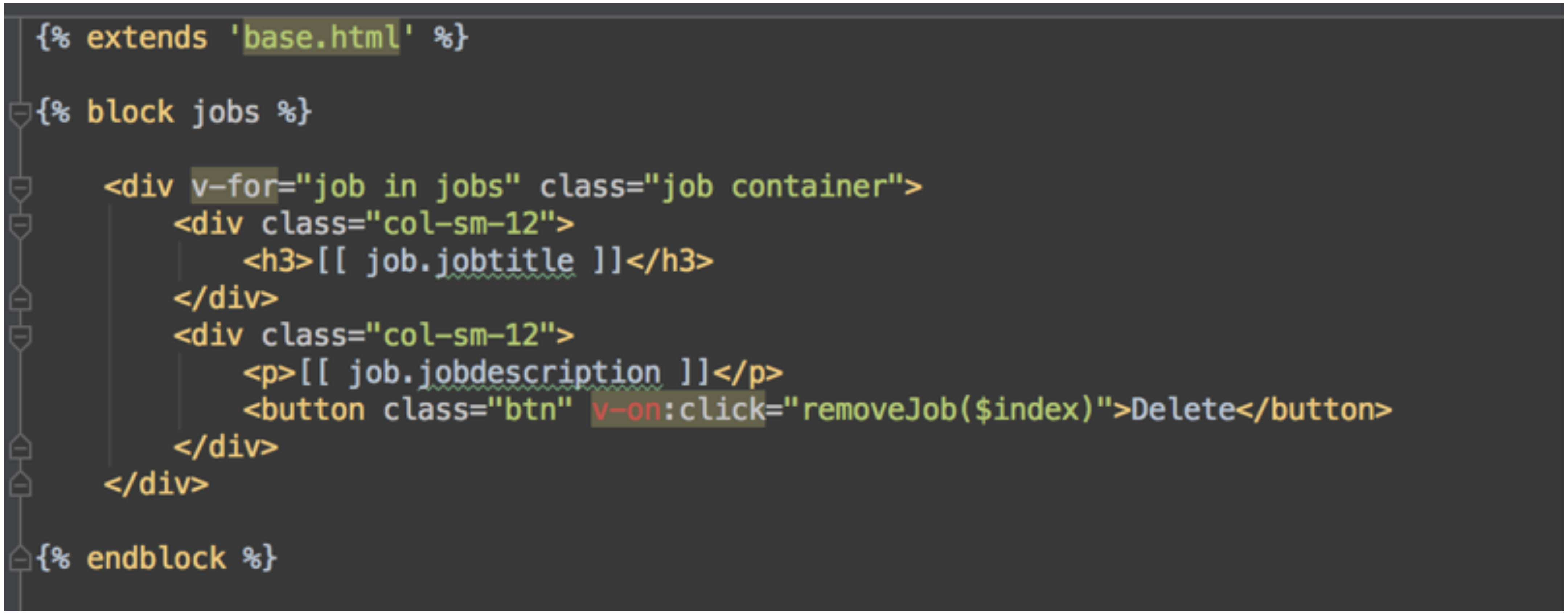
Template

View

How to combine the two?

- Components in Django templates
- Django REST API
- Interact with API in Vue

Django Templates

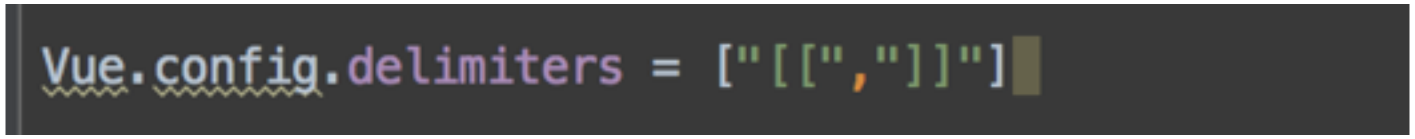


```
{% extends 'base.html' %}

{% block jobs %}

    <div v-for="job in jobs" class="job container">
        <div class="col-sm-12">
            <h3>[[ job.jobtitle ]]</h3>
        </div>
        <div class="col-sm-12">
            <p>[[ job.jobdescription ]]</p>
            <button class="btn" v-on:click="removeJob($index)">Delete</button>
        </div>
    </div>

{% endblock %}
```



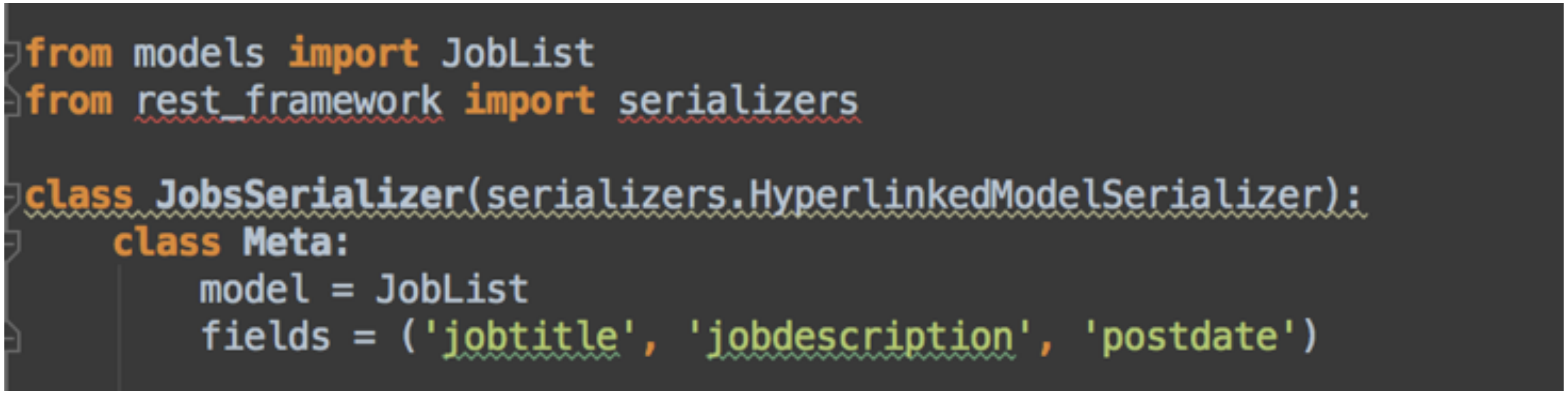
```
Vue.config.delimiters = ["[[","]]"]
```

Django REST

- REST API for Django projects
- Simple to integrate
- Very well documented
- Numerous return types - JSON, XML …

Setup Endpoint

- Serialise model

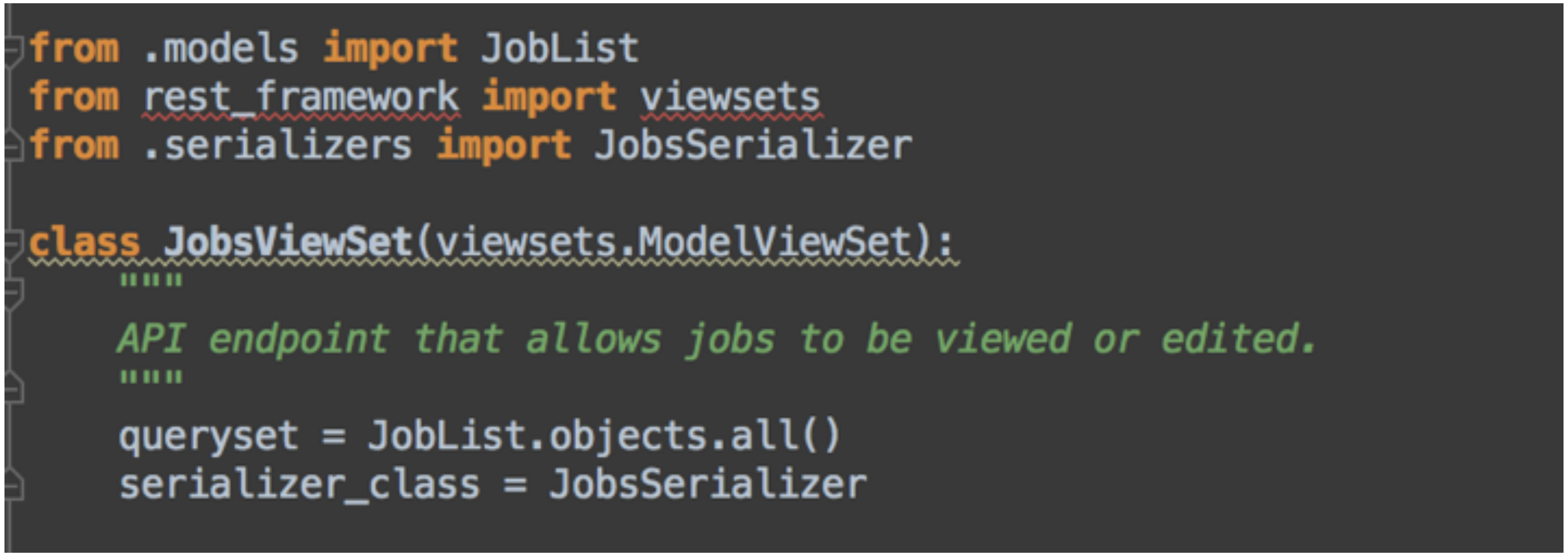


```
from models import JobList
from rest_framework import serializers

class JobsSerializer(serializers.HyperlinkedModelSerializer):
    class Meta:
        model = JobList
        fields = ('jobtitle', 'jobdescription', 'postdate')
```

Setup Endpoint

- Set up end point

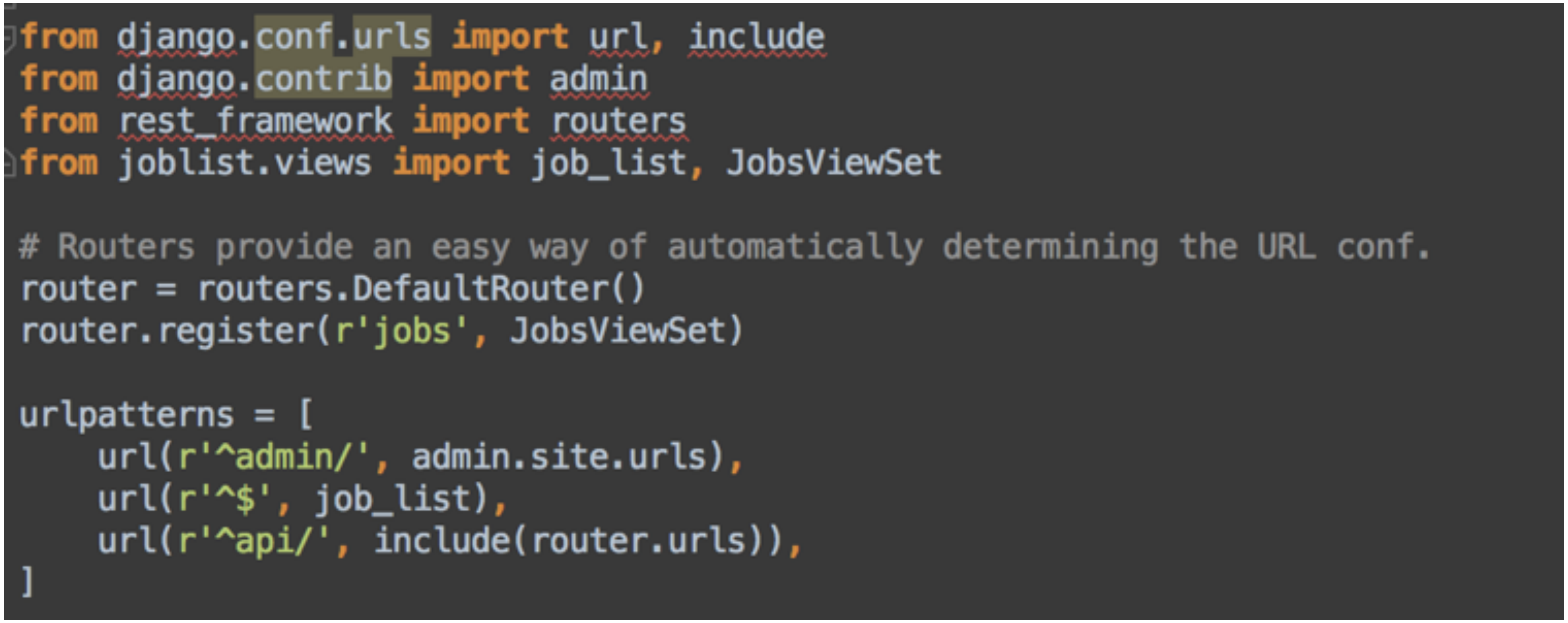


```python
from .models import JobList
from rest_framework import viewsets
from .serializers import JobsSerializer

class JobsViewSet(viewsets.ModelViewSet):
    """
    API endpoint that allows jobs to be viewed or edited.
    """
    queryset = JobList.objects.all()
    serializer_class = JobsSerializer
```

Setup Endpoint

- Add API route



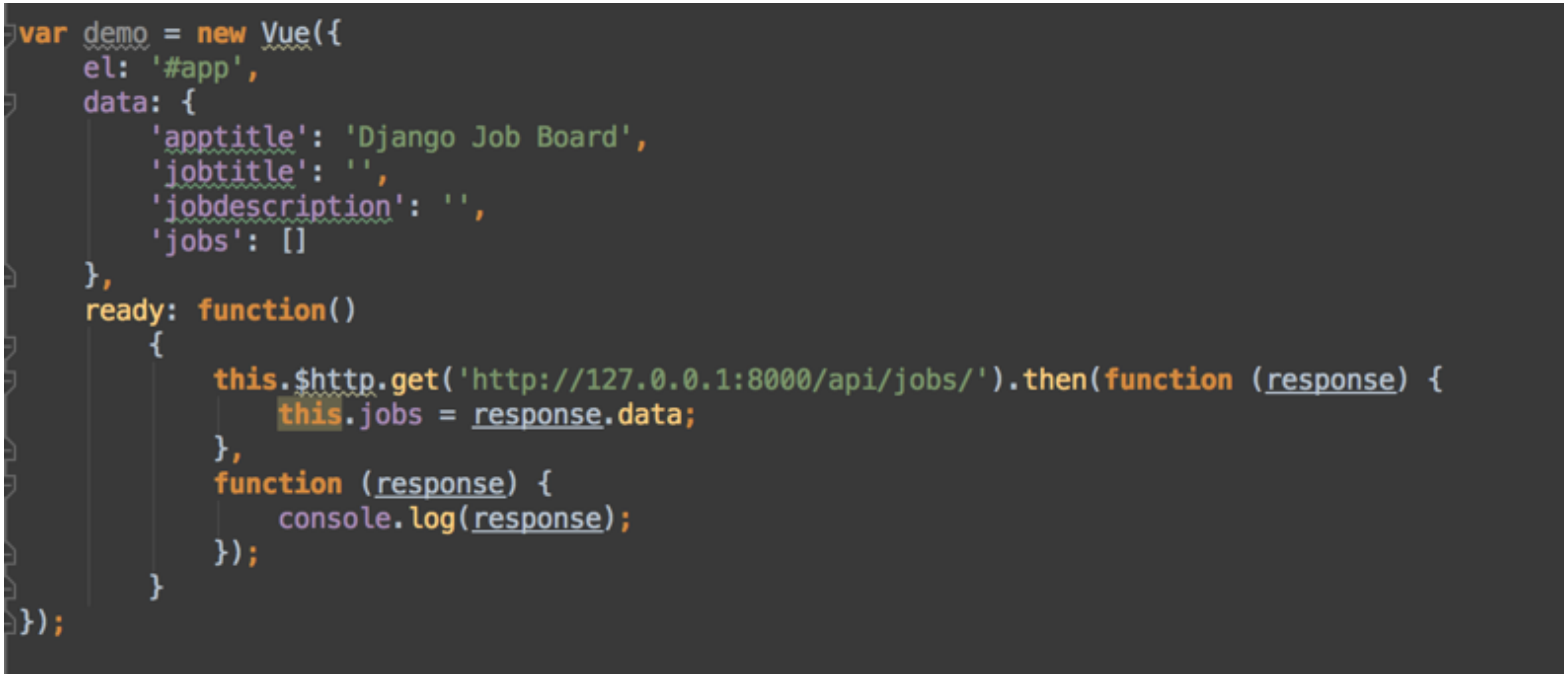
```python
from django.conf.urls import url, include
from django.contrib import admin
from rest_framework import routers
from joblist.views import job_list, JobsViewSet

# Routers provide an easy way of automatically determining the URL conf.
router = routers.DefaultRouter()
router.register(r'jobs', JobsViewSet)

urlpatterns = [
    url(r'^admin/', admin.site.urls),
    url(r'^$', job_list),
    url(r'^api/', include(router.urls)),
]
```

Call Endpoint

- API calls
- Vue-Resource



```
var demo = new Vue({
    el: '#app',
    data: {
        'apptitle': 'Django Job Board',
        'jobtitle': '',
        'jobdescription': '',
        'jobs': []
    },
    ready: function()
        {
            this.$http.get('http://127.0.0.1:8000/api/jobs/').then(function (response) {
                this.jobs = response.data;
            },
            function (response) {
                console.log(response);
            });
        }
});
```

Example

- Job board
- Post
- Job list

Benefits

- Fancy javascript frontends made easy
- Shallow learning curve
- Easily integrate into framework of choice

Difficulties

- Fair amount of set up
- Large applications can get messy
- Web-pack not easy to configure

Framework of your Choice

- <Insert your framework here>
- See you next time?

?

https://github.com/andrewm-theodo/djangovuejs