

1. Install Development Enviroment

1.1 Windows

Python
Node.JS
PyCharm

1.2 Debian or Ubuntu

Python
Java SDK
PostgreSQL + PostGIS
GeoServer
nginx

1.3 Python + Packages

Django
psycpg
psycpg-binary
whitenoise
django-cors-headers

```
postgres@debianos:~$ pip list
Package            Version
-----
asgiref            3.6.0
Django             4.2
django-cors-headers 3.14.0
pip                23.0.1
psycpg             3.1.8
psycpg-binary      3.1.8
setuptools         67.6.1
sqlparse           0.4.3
typing_extensions  4.5.0
whitenoise         6.4.0
```

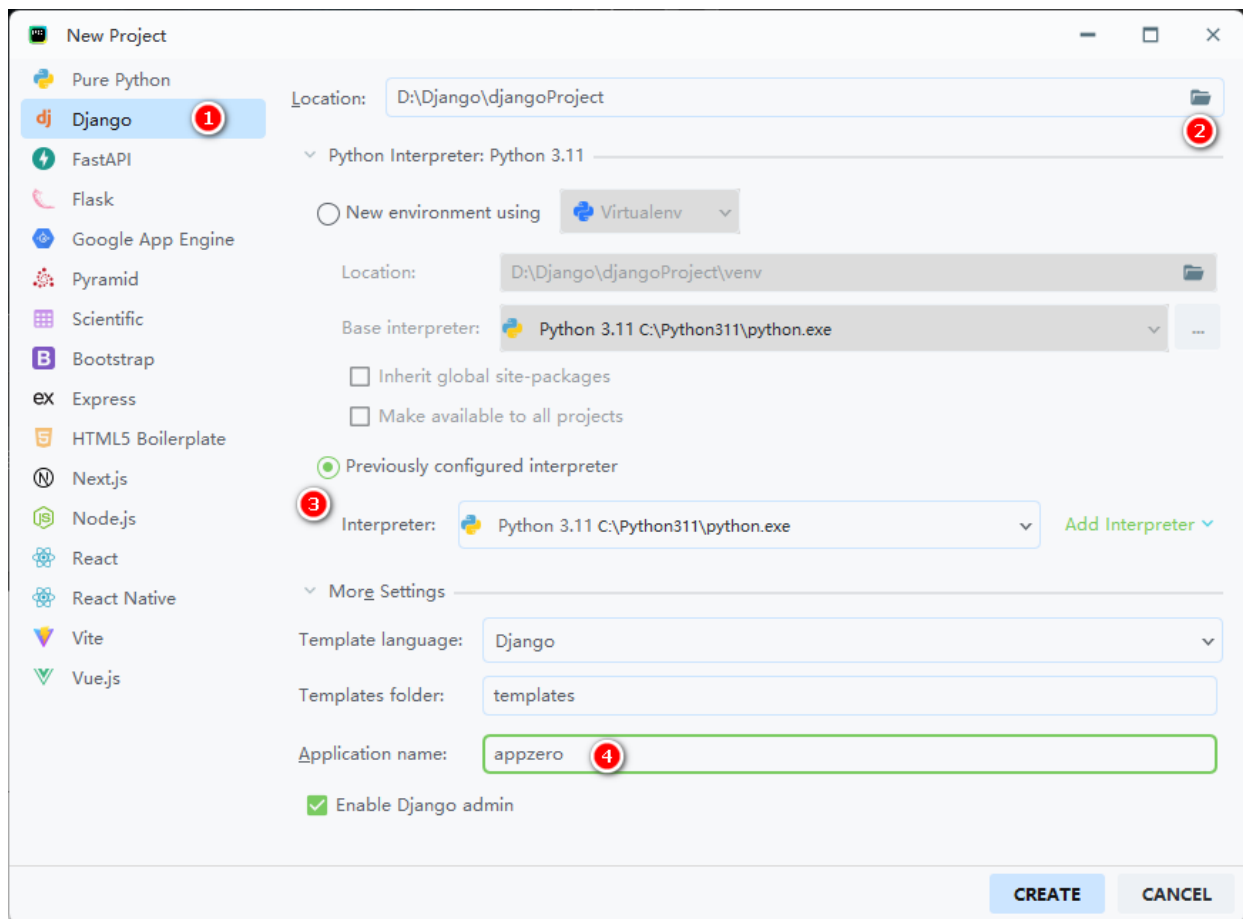
1.4 npm + Packages

```
vue
axios
cesium
bootstrap
vue3-sfc-loader
```

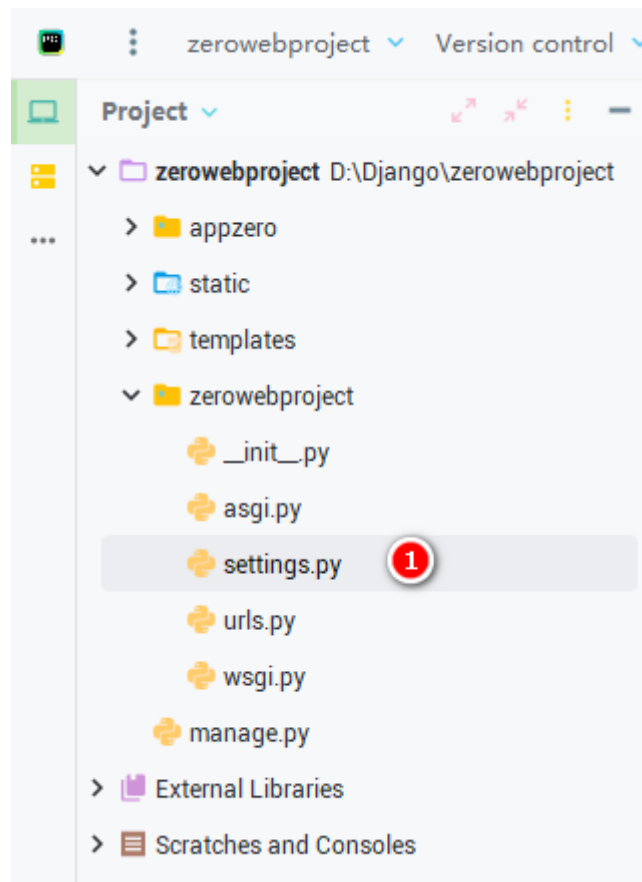
package.json

```
{
  "name": "web",
  "version": "1.0.0",
  "description": "",
  "main": "index.js",
  "scripts": {
    "test": "echo \"Error: no test specified\" && exit 1"
  },
  "author": "",
  "license": "ISC",
  "dependencies": {
    "axios": "^1.3.4",
    "bootstrap": "^5.2.3",
    "cesium": "^1.104.0",
    "vue": "^3.2.47",
    "vue3-sfc-loader": "^0.8.4"
  }
}
```

2. New Django Project



2.1 settings.py



settings.py

```
from pathlib import Path

# Build paths inside the project like this: BASE_DIR / 'subdir'.
BASE_DIR = Path(__file__).resolve().parent.parent

# Quick-start development settings - unsuitable for production
# See https://docs.djangoproject.com/en/4.2/howto/deployment/checklist/

# SECURITY WARNING: keep the secret key used in production secret!
SECRET_KEY = 'django-insecure-kui*f3i7dmb-e+*#2%=@fq#*21#u7b7d39#8l2h8_7od@at@_w'

# SECURITY WARNING: don't run with debug turned on in production!
DEBUG = True

ALLOWED_HOSTS = ['*'] # 1

# Application definition

INSTALLED_APPS = [
    'django.contrib.admin',
    'django.contrib.auth',
    'django.contrib.contenttypes',
    'django.contrib.sessions',
    'django.contrib.messages',
    'django.contrib.staticfiles',
    'corsheaders', # 2
    'appzero.apps.AppzeroConfig',
]

MIDDLEWARE = [
    'django.middleware.security.SecurityMiddleware',
    'whitenoise.middleware.WhiteNoiseMiddleware', # 3
    'corsheaders.middleware.CorsMiddleware', # 4
    'django.contrib.sessions.middleware.SessionMiddleware',
    'django.middleware.common.CommonMiddleware',
    'django.middleware.csrf.CsrfViewMiddleware',
    'django.contrib.auth.middleware.AuthenticationMiddleware',
    'django.contrib.messages.middleware.MessageMiddleware',
    'django.middleware.clickjacking.XFrameOptionsMiddleware',
]

ROOT_URLCONF = 'zerowebproject.urls'

TEMPLATES = [
    {
        'BACKEND': 'django.template.backends.django.DjangoTemplates',
        'DIRS': [BASE_DIR / 'templates'],
        'APP_DIRS': True,
        'OPTIONS': {
```

```

        'context_processors': [
            'django.template.context_processors.debug',
            'django.template.context_processors.request',
            'django.contrib.auth.context_processors.auth',
            'django.contrib.messages.context_processors.messages',
        ],
    },
},
]

WSGI_APPLICATION = 'zerowebproject.wsgi.application'

# 5 Database
# https://docs.djangoproject.com/en/4.2/ref/settings/#databases
SQLITE = False
if SQLITE:
    DATABASES = {
        'default': {
            'ENGINE': 'django.db.backends.sqlite3',
            'NAME': BASE_DIR / 'db.sqlite3'
        }
    }
else:
    DATABASES = {
        'default': {
            'ENGINE': 'django.db.backends.postgresql',
            'NAME': 'db4test',
            'USER': 'postgres',
            'PASSWORD': 'masterkey',
            'HOST': '127.0.0.1',
            'PORT': '5432',
            'OPTIONS': {
                'options': '-c search_path=public,workspace'
            }
        }
    }

# Password validation
# https://docs.djangoproject.com/en/4.2/ref/settings/#auth-password-validators

AUTH_PASSWORD_VALIDATORS = [
    {
        'NAME':
'django.contrib.auth.password_validation.UserAttributesSimilarityValidator',
    },
    {
        'NAME': 'django.contrib.auth.password_validation.MinimumLengthValidator',
    },
    {
        'NAME': 'django.contrib.auth.password_validation.CommonPasswordValidator',
    },
    {

```

```

        'NAME': 'django.contrib.auth.password_validation.NumericPasswordValidator',
    },
]

# Internationalization
# https://docs.djangoproject.com/en/4.2/topics/i18n/

LANGUAGE_CODE = 'en-us'
TIME_ZONE = 'UTC'
USE_I18N = True
USE_TZ = True

# 6 Static files (CSS, JavaScript, Images)
# https://docs.djangoproject.com/en/4.2/howto/static-files/
STATIC_URL = 'static/'
STATIC = 'static'
if DEBUG:
    STATICFILES_DIRS = [BASE_DIR / STATIC]
else:
    STATIC_ROOT = STATIC

# Default primary key field type
# https://docs.djangoproject.com/en/4.2/ref/settings/#default-auto-field

DEFAULT_AUTO_FIELD = 'django.db.models.BigAutoField'

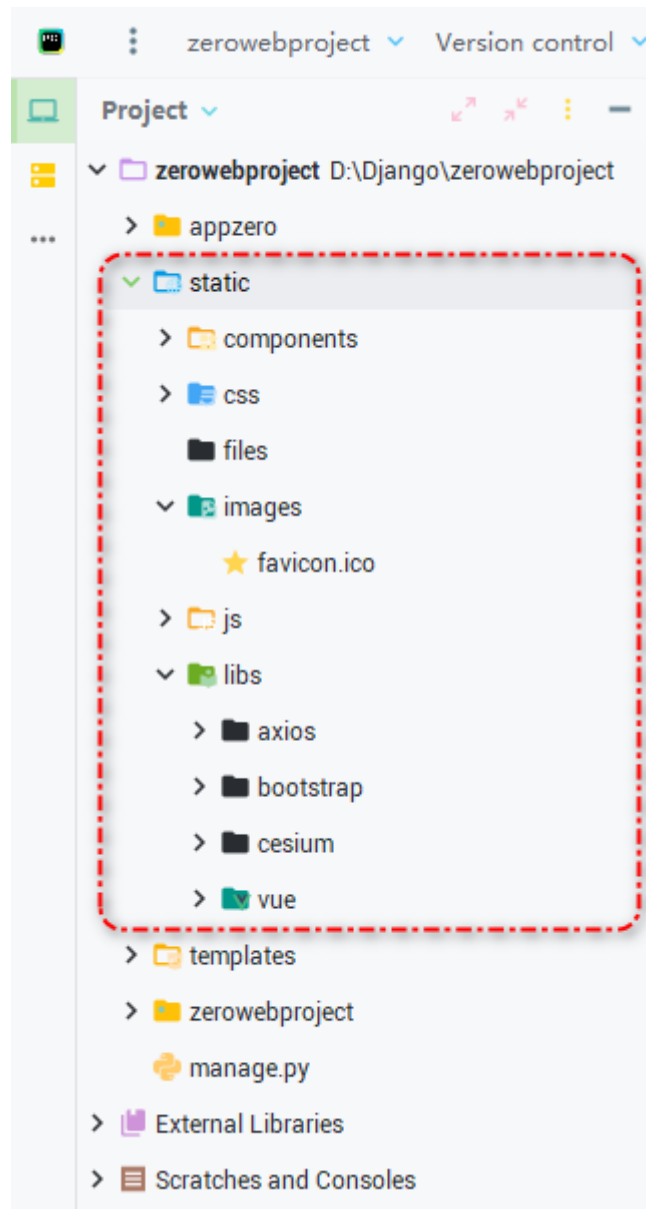
# 7 Session
SESSION_COOKIE_DOMAIN = None
SESSION_COOKIE_SECURE = False
SESSION_COOKIE_HTTPONLY = True
SESSION_COOKIE_AGE = 3 * 60 * 60
SESSION_EXPIRE_AT_BROWSER_CLOSE = True
SESSION_SAVE_EVERY_REQUEST = False

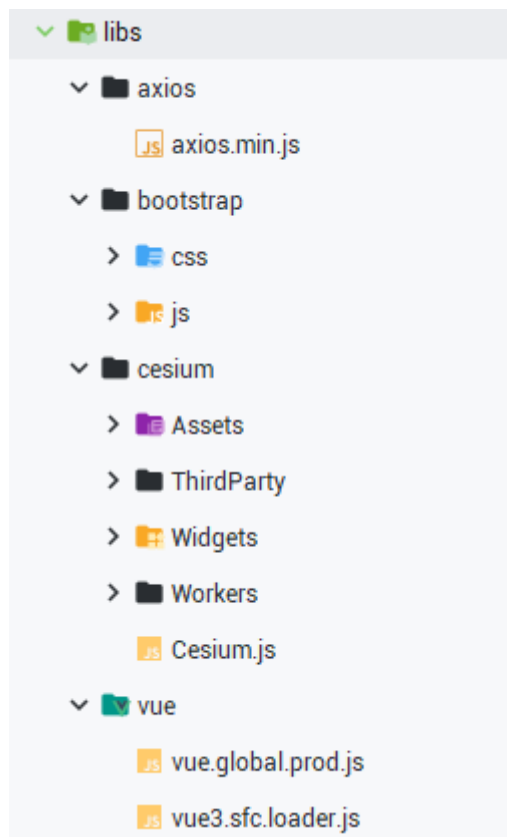
# 8 Cors
CORS_ALLOW_HEADERS = ['*']
CORS_ORIGIN_ALLOW_ALL = True
CORS_ALLOW_CREDENTIALS = True
CORS_ALLOW_ALL_ORIGINS = True
X_FRAME_OPTIONS = 'SAMEORIGIN'
SECURE_CONTENT_TYPE_NOSNIFF = True

# 9
APPEND_SLASH = False
LOGIN_URL = 'login.html'

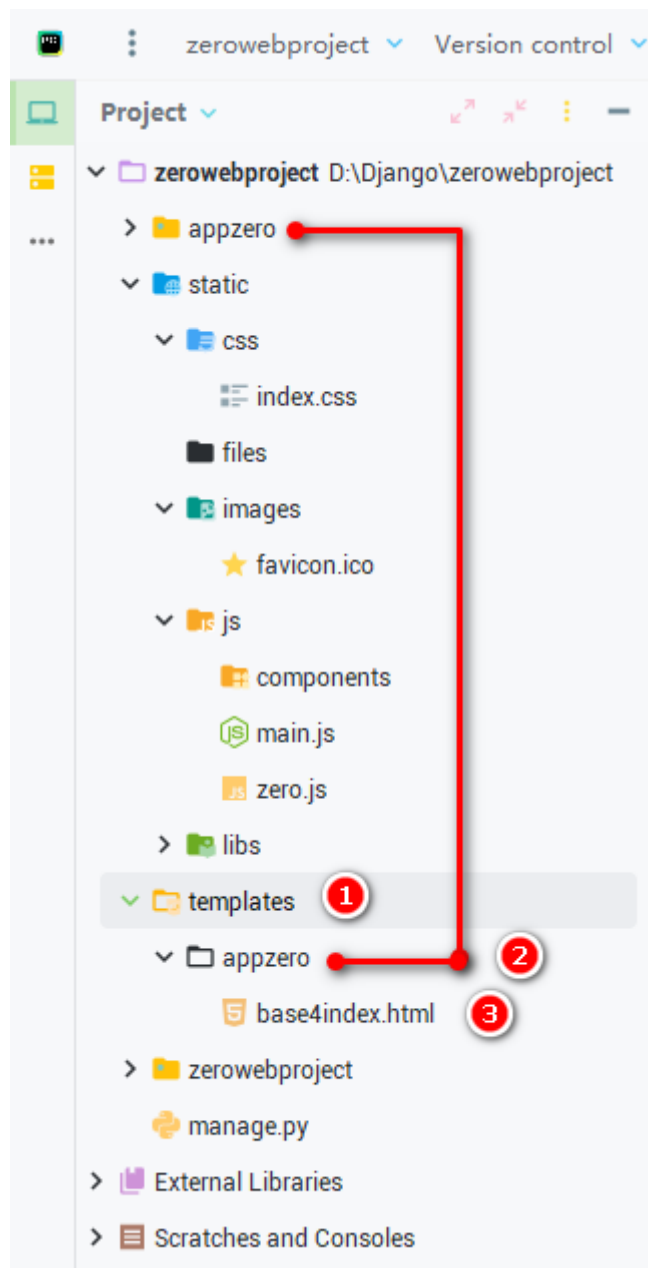
```

2.2 static





2.3 templates



base4index.html

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8"/>
  <meta name="author" content="ljs"/>
  <meta name="keywords" content="django,cesium,vue"/>
  <meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-
fit=no"/>

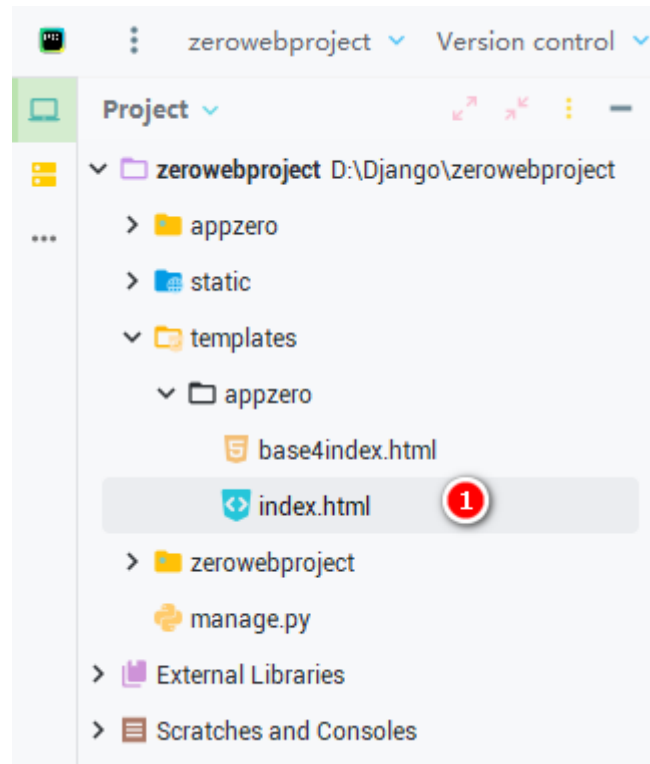
  <link rel="icon" href="/static/images/favicon.ico">
  <link rel="stylesheet" href="/static/css/index.css">
  <link rel="stylesheet" href="/static/libs/cesium/widgets/widgets.css">
  <link rel="stylesheet" href="/static/libs/bootstrap/css/bootstrap.min.css">
```

```

<script src="/static/libs/cesium/Cesium.js"></script>
<script src="/static/libs/axios/axios.min.js"></script>
<script src="/static/libs/vue/vue.global.prod.js"></script>
<script src="/static/libs/vue/vue3.sfc.loader.js"></script>
<script src="/static/libs/bootstrap/js/bootstrap.min.js"></script>

<title>{{ pagetitle }}</title>
</head>
<body>
<main id="comroot" class="vh-100">
    {% block main %}
    {% endblock %}
</main>
{% block script %}
{% endblock %}
<script src="/static/js/main.js" type="module"></script>
</body>
</html>

```



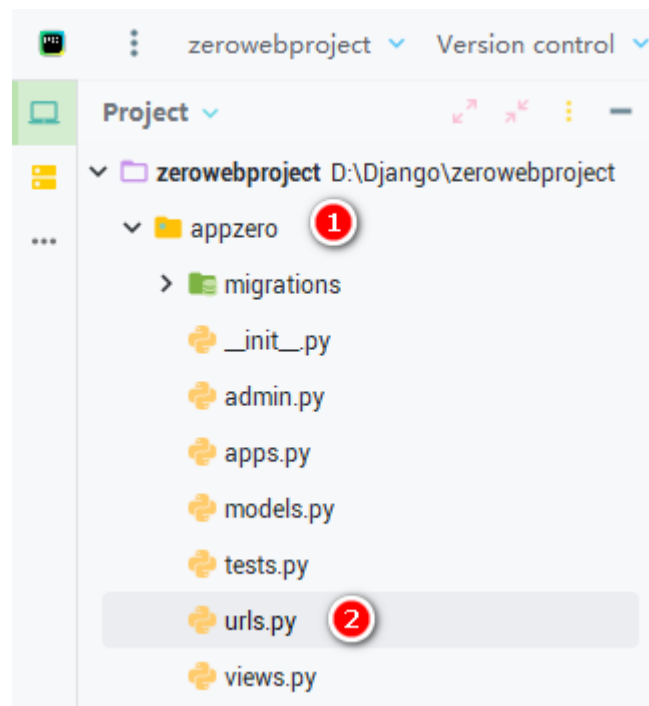
index.html

```
{% extends 'appzero/base4index.html' %}

{% block main %}
    <div class="text-info text-center">
        This is Main Page!
    </div>
{% endblock %}

{% block script %}
    <script>
        const rootcomfile = 'App.vue';
    </script>
{% endblock %}
```

2.4 new app's urls.py



urssl.py

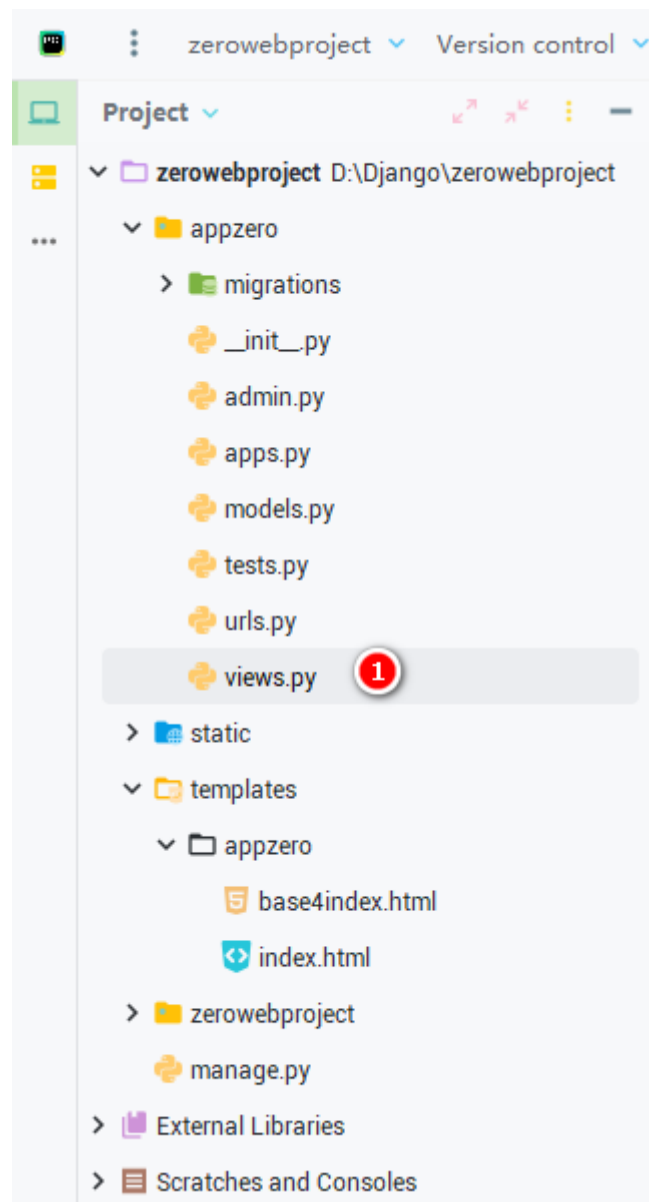
```
from django.urls import path
from django.urls import re_path

from . import apps
from . import views

app_name = apps.AppzeroConfig.name

urlpatterns = [
    path('', views.begin, name='begin'),
]
```

2.5 def view function



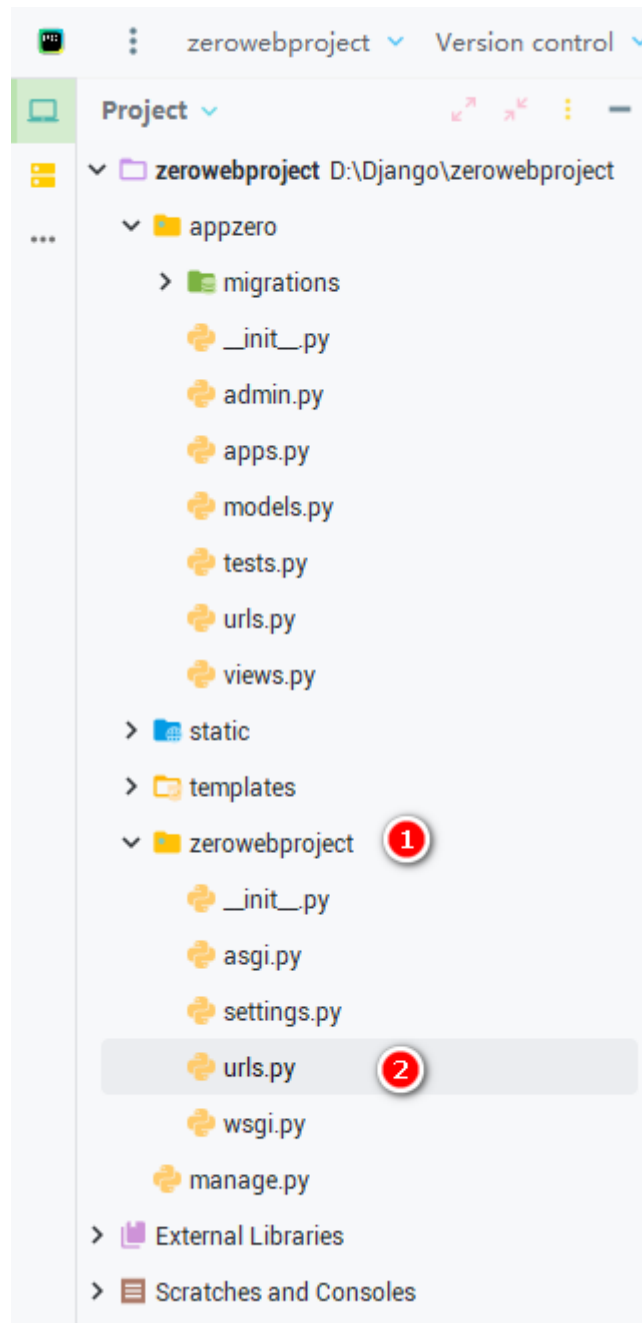
views.py

```
from django.http import HttpResponseRedirect
from django.http import JsonResponse

from django.shortcuts import render
from django.shortcuts import reverse
from django.shortcuts import redirect

# Create your views here.
def begin(request):
    pagetitle = 'MainPage'
    return render(request, 'appzero/index.html', locals())
```

2.6 set project urls.py

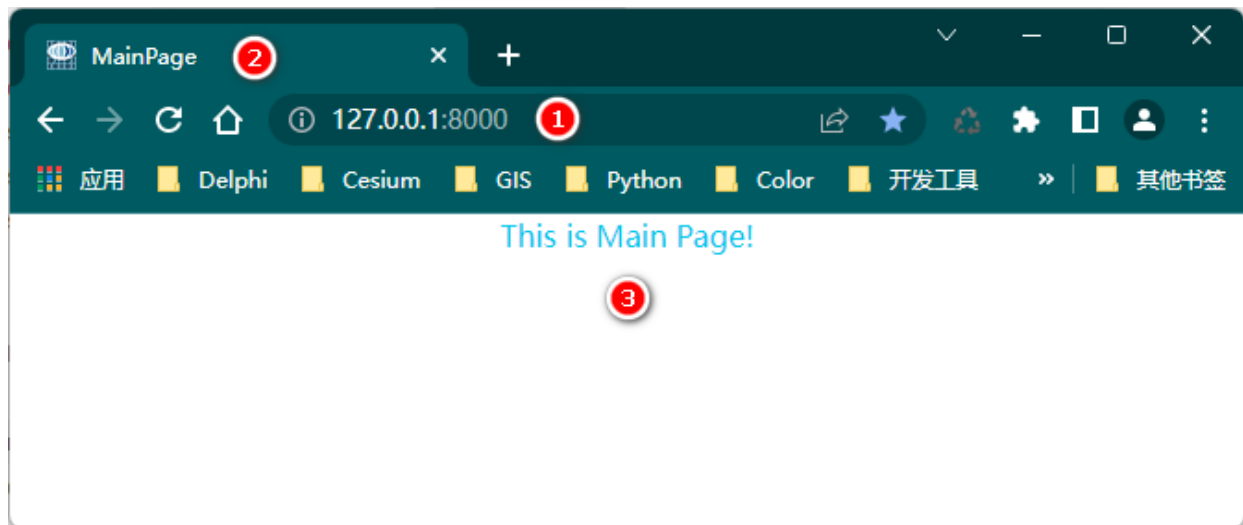
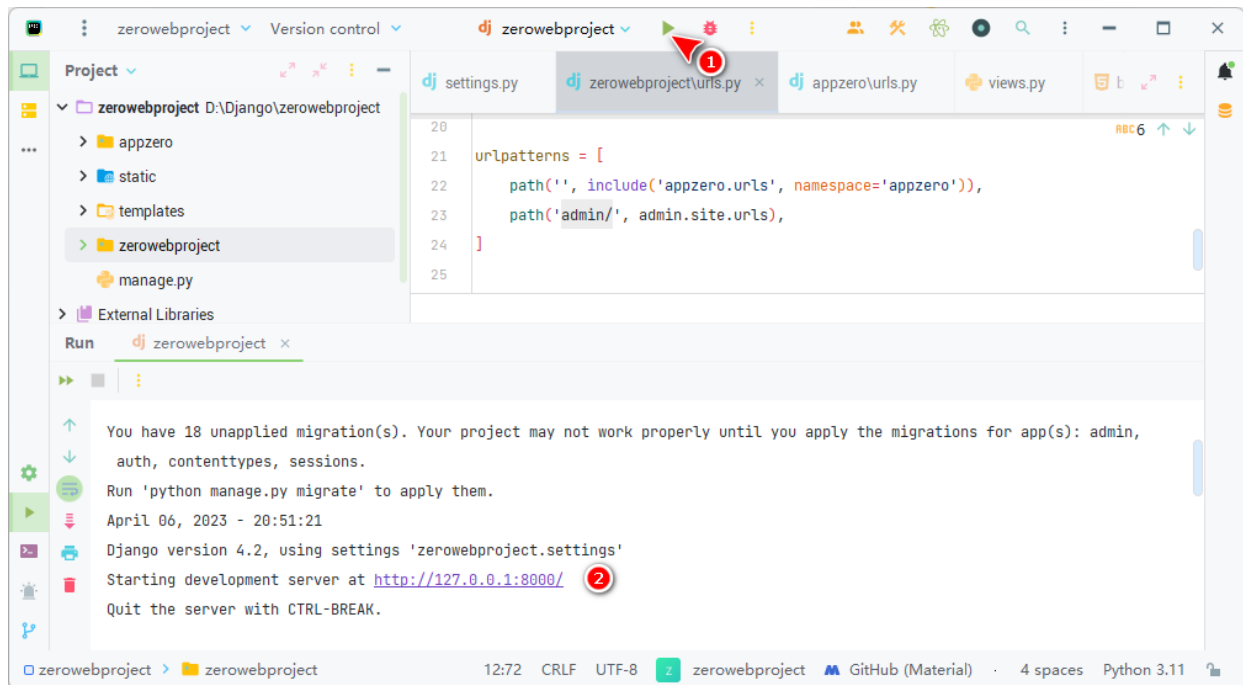


urls.py

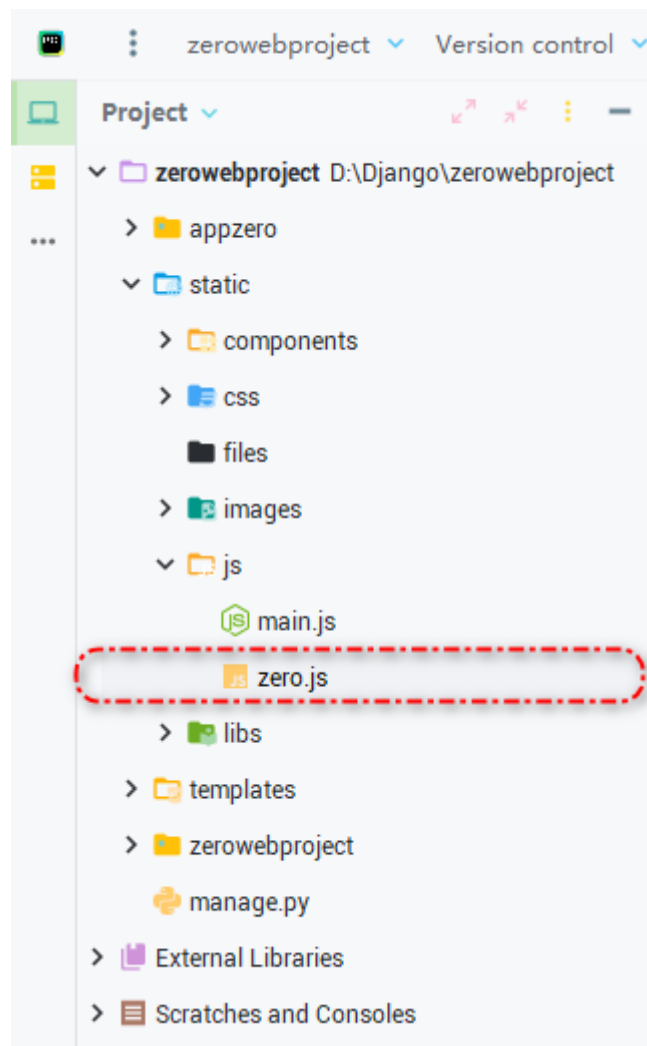
```
from django.contrib import admin
from django.urls import include
from django.urls import path

urlpatterns = [
    path('', include('appzero.urls', namespace='appzero')),
    path('admin/', admin.site.urls),
]
```

2.7 run



3. Vue SFC loader



zero.js

```
const {loadModule} = window['vue3-sfc-loader'];
const {defineAsyncComponent} = Vue;
/*-- modules --*/

/* --stores-- */

/*-- moduleCaches --*/
const moduleCache = {
  "vue": Vue,
  "axios": axios,
};
/*-- comoptions --*/
const comoptions = {
  //delimiters: ['[[', ']]'],
  moduleCache: moduleCache,
  async getFile(url) {
    const res = await fetch(url);
    if (!res.ok) {
      throw Object.assign(new Error(res.statusText + ' ' + url), {res,});
    }
  }
}
```

```

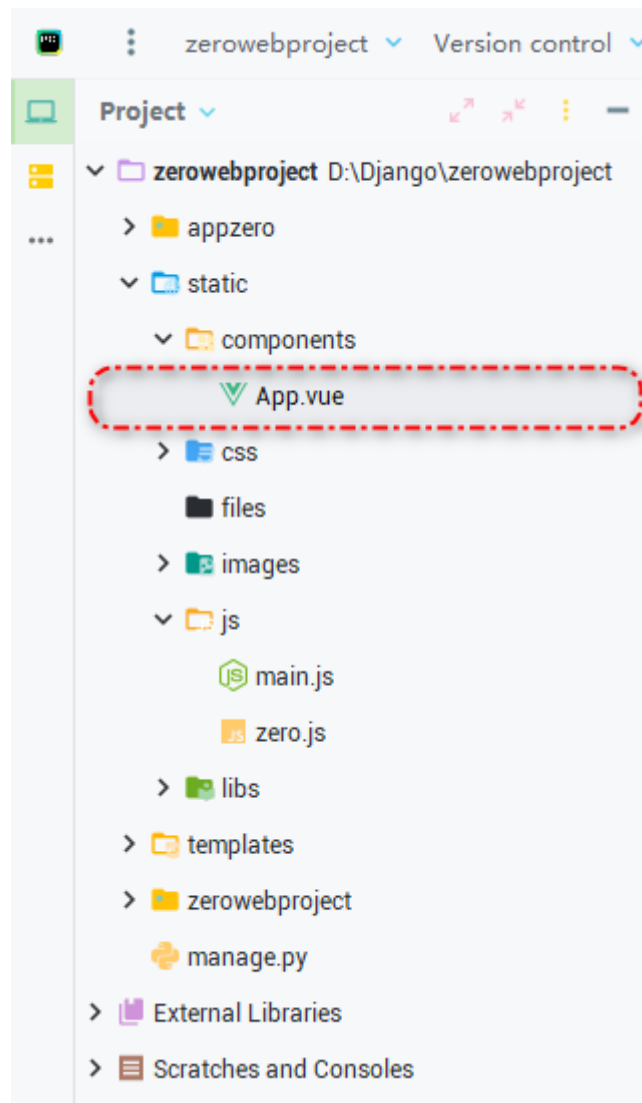
        return {getContentData: asBinary => asBinary ? res.arrayBuffer() :
res.text()};
    },
    addStyle(textContent) {
        const style = Object.assign(document.createElement('style'),
{textContent,});
        const ref = document.head.getElementsByTagName('style')[0] || null;
        document.head.insertBefore(style, ref);
    },
};

/*-- loadvuecom --*/
function loadVueCOM(vueurl, options = comoptions) {
    return defineAsyncComponent(() => loadModule(vueurl, options));
}

export {loadVueCOM};

```

4. new App.vue



Vue Component Structure

```
<template>
  <!--HTML元素-->
</template>

<script setup>
  //JavaScript脚本
</script>

<style scoped>
  /*本地CSS*/
</style>
```

App.vue

```
<template>
  <section id="idapp" v-show="visible" class="">
    This is App.vue.
  </section>
</template>

<script setup>
  /*-- imports --*/
  import {defineComponent, defineProps,
    onBeforeMount,onMounted,onBeforeUnmount,onUnmounted} from 'vue';
  import axios from "axios";

  const name = defineComponent({
    name: 'app',
  });

  /*-- props --*/
  const props = defineProps({
    visible: {
      type: String,
      required: false,
      default: 'show',
    },
  });

  /*-- stores --*/

  /*-- vars --*/

  /*-- methods --*/

  /*-- events --*/
  onBeforeMount(() => {
    console.log('App.onBeforeMount');
  });
```

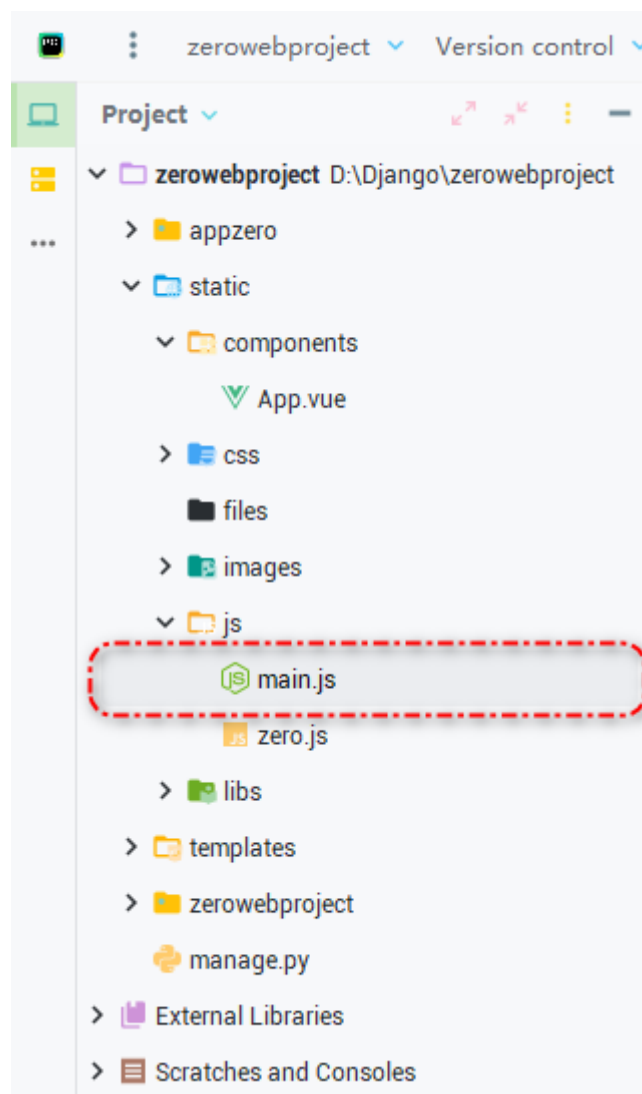
```
onMounted(() => {
  console.log('App.Mounted');
});

onBeforeUnmount(() => {
  console.log('App.onBeforeUnmount');
});

onUnmounted(() => {
  console.log('App.onUnmounted');
});
</script>

<style scoped>
</style>
```

5. main.js



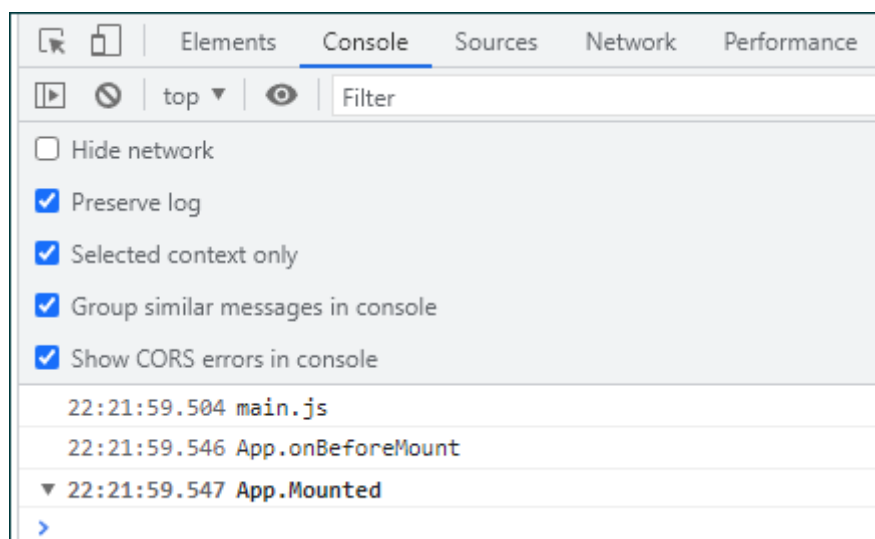
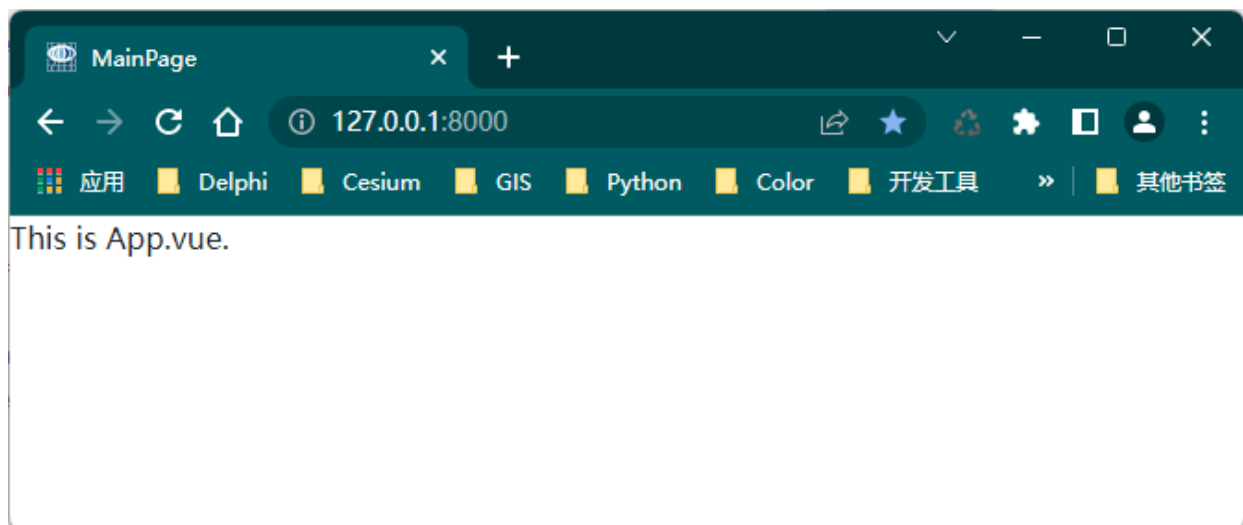
main.js

```
const {createApp} = Vue;
import {loadVueCOM} from "./zero.js";

class ComRoot {
  static begin(rootcomfile) {
    const AppPath = '/static/components/';
    const AppUrl = AppPath + rootcomfile;
    const App = loadVueCOM(AppUrl);
    const app = createApp(App);
    //

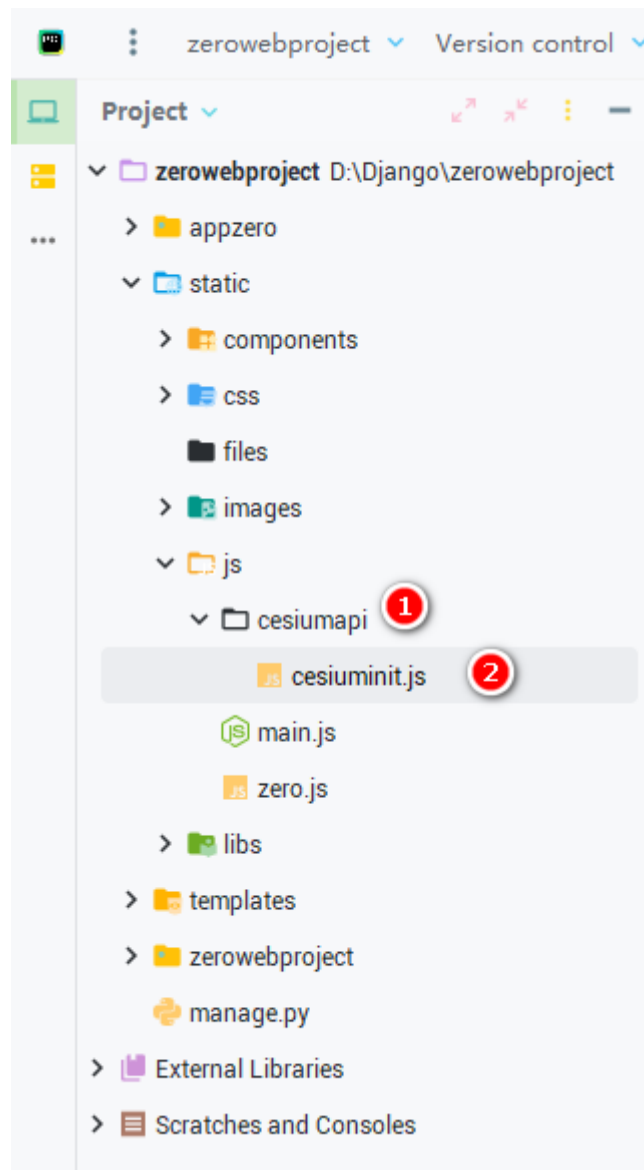
    //
    const appid = '#' + 'comroot';
    const vueapp = app.mount(appid);
    console.log('main.js');
  }
}

ComRoot.begin(rootcomfile);
```



6. New Vue Component

6.1 cesiumapi



6.1.1 cesiuminit.js

```
const {Viewer, Camera, Rectangle, Color, FeatureDetection} = Cesium;

class CesiumInit {
  static begin(container = 'idcomcesium', extend = false) {
    const options = {
      infoBox: false,
      vrButton: false,
      geocoder: false,
      timeline: false,
      animation: false,
      homeButton: true,
      scene3DOnly: true,
      shouldAnimate: false,
```

```

        imageryProvider: false,
        baseLayerPicker: false,
        sceneModePicker: false,
        fullscreenButton: false,
        selectionIndicator: true,
        navigationHelpButton: false,
        sceneMode: Cesium.SceneMode.SCENE3D,
        navigationInstructionsInitiallyVisible: false,
    };

    const viewer = new Viewer(container, options);
    viewer.cesiumWidget.creditContainer.style.display = 'none';

    const scene = viewer.scene;
    scene.fxaa = true;
    scene.sun.show = false;
    scene.skyBox.show = false;
    scene.skyAtmosphere.show = false;
    scene.pickTranslucentDepth = true;
    scene.postProcessStages.fxaa.enabled = true;
    scene.screenSpaceCameraController.enableCollisionDetection = false;

    const globe = scene.globe;
    globe.show = true;
    globe.translucency.enabled = false;
    globe.depthTestAgainstTerrain = false;
    globe.baseColor = Color.TRANSPARENT;

    if (extend) {
        viewer.extend(Cesium.viewerDragDropMixin);
        viewer.extend(Cesium.viewerPerformanceWatchdogMixin);
        viewer.extend(Cesium.viewerCesiumInspectorMixin);
    }

    if (FeatureDetection.supportsImageRenderingPixelated()) {
        viewer.resolutionScale = window.devicePixelRatio;
    }

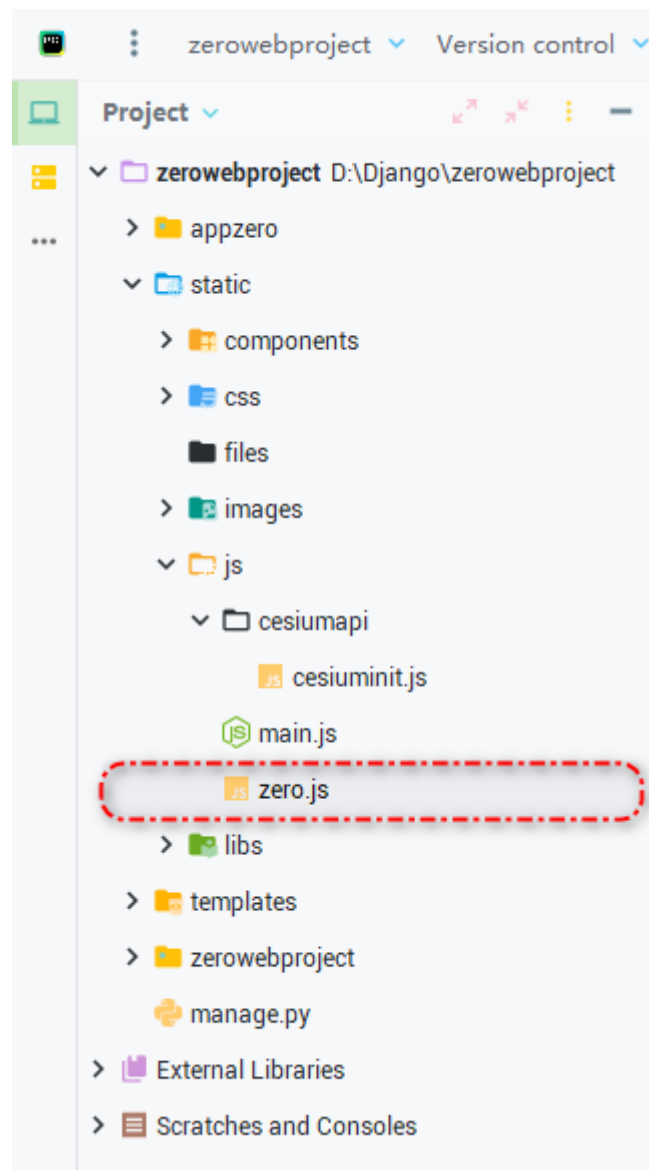
    const xmin = 119.065117829867;
    const ymin = 45.8955968828478;
    const xmax = 119.150342668864;
    const ymax = 45.9698186561991;
    Camera.DEFAULT_VIEW_RECTANGLE = Rectangle.fromDegrees(xmin, ymin, xmax,
ymax);
    Camera.DEFAULT_VIEW_FACTOR = 0.001;

    return viewer;
}

export {CesiumInit};

```

6.2 moduleCache



zero.js

```
const {defineAsyncComponent} = Vue;
/*-- modules --*/
import {CesiumInit} from "../cesiumapi/cesiuminit.js"; // 1

//2
const cesiumapi = {
  CesiumInit: CesiumInit,
};
/* --stores-- */

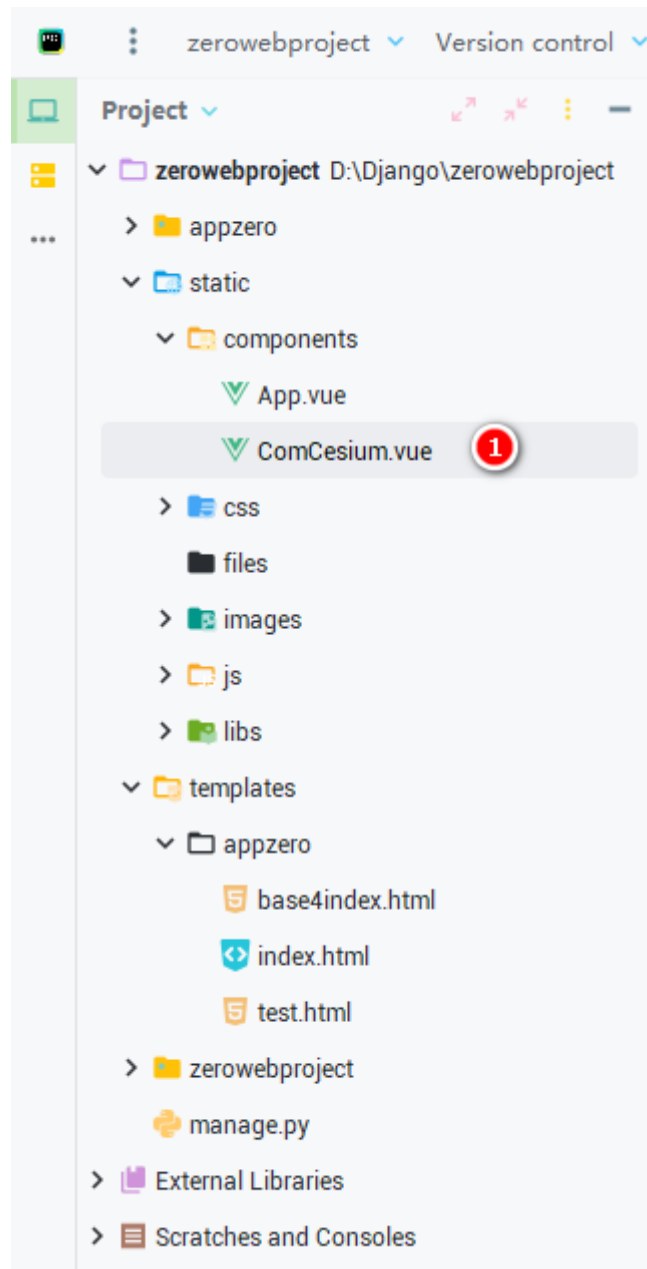
/*-- moduleCaches --*/
const moduleCache = {
  "vue": Vue,
  "axios": axios,
```

```

"cesium": cesiumapi,// 3
};
/*-- comoptions --*/
/*-- ..以下省略.. --*/

```

6.3 ComCesium



ComCesium.vue

```

<template>
  <section id="idcomcesium" v-show="visible" class="">
  </section>
</template>

<script setup>
/*-- imports --*/

```

```

import {defineComponent, defineProps, onBeforeMount, onBeforeUnmount, onMounted,
onUnmounted} from "vue";
import axios from "axios";
import {CesiumInit} from "cesium"; // 1

const name = defineComponent({
  name: 'comcesium',
});

/*-- props --*/
const props = defineProps({
  visible: {
    type: String,
    required: false,
    default: 'show',
  },
});
/*-- stores --*/

/*-- vars --*/

/*-- methods --*/

/*-- events --*/
const beforemounted = onBeforeMount(() => {
});

onMounted(() => {
  CesiumInit.begin();// 2
});

onBeforeUnmount(() => {
});

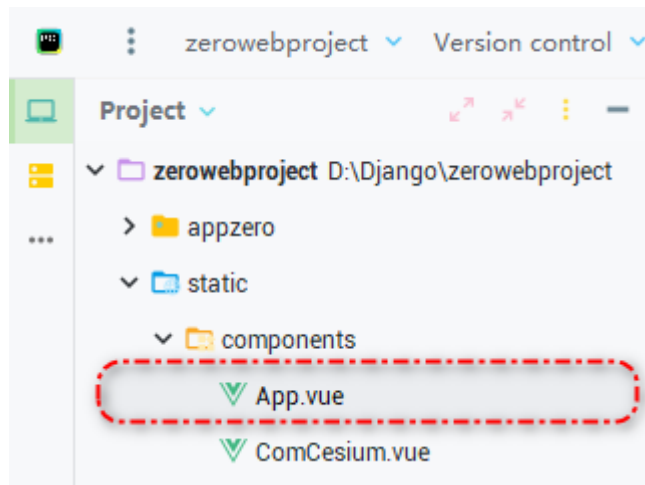
onUnmounted(() => {
});
</script>

<style scoped>
</style>

```

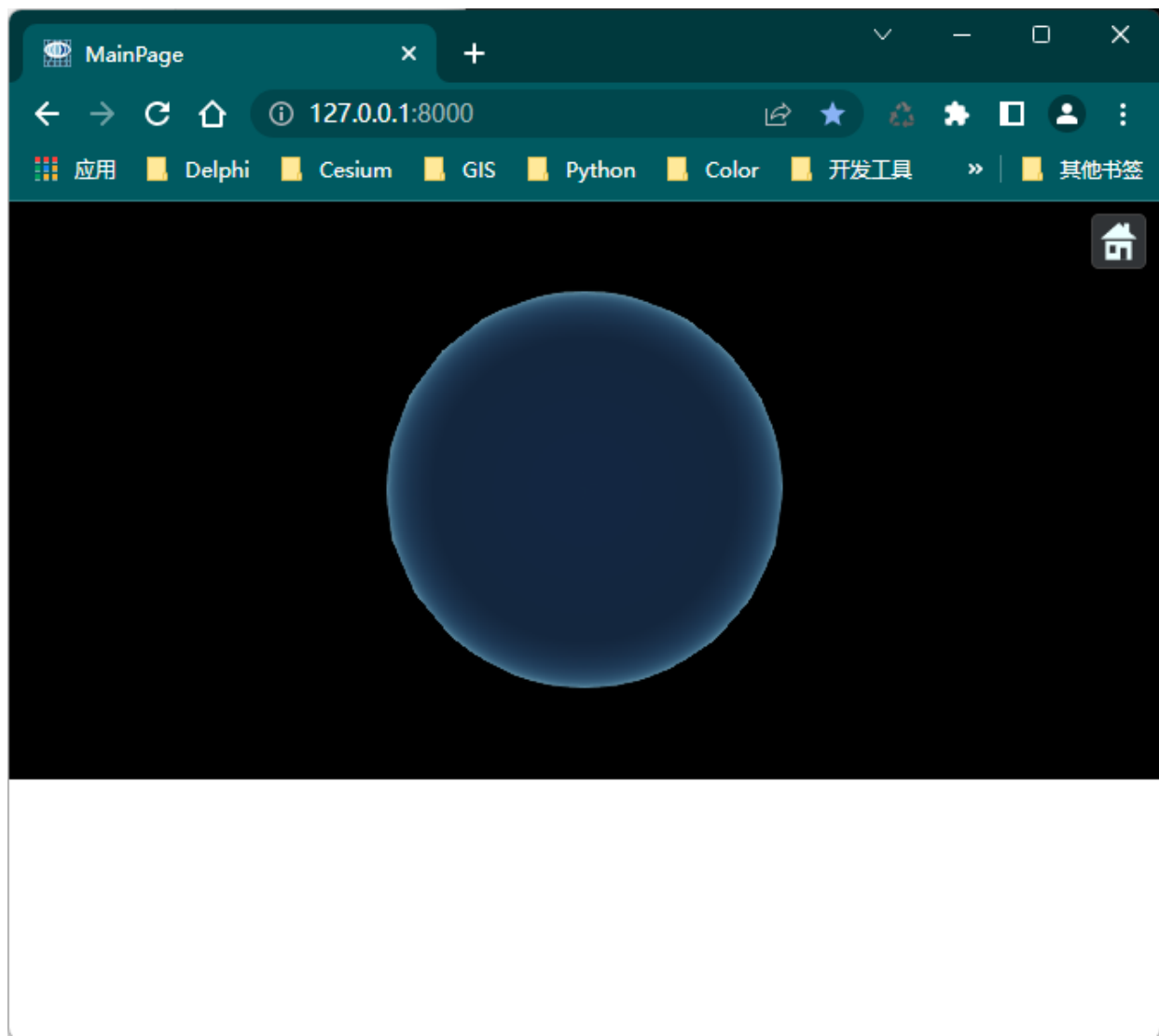
7. Use Component

7.1 App.vue



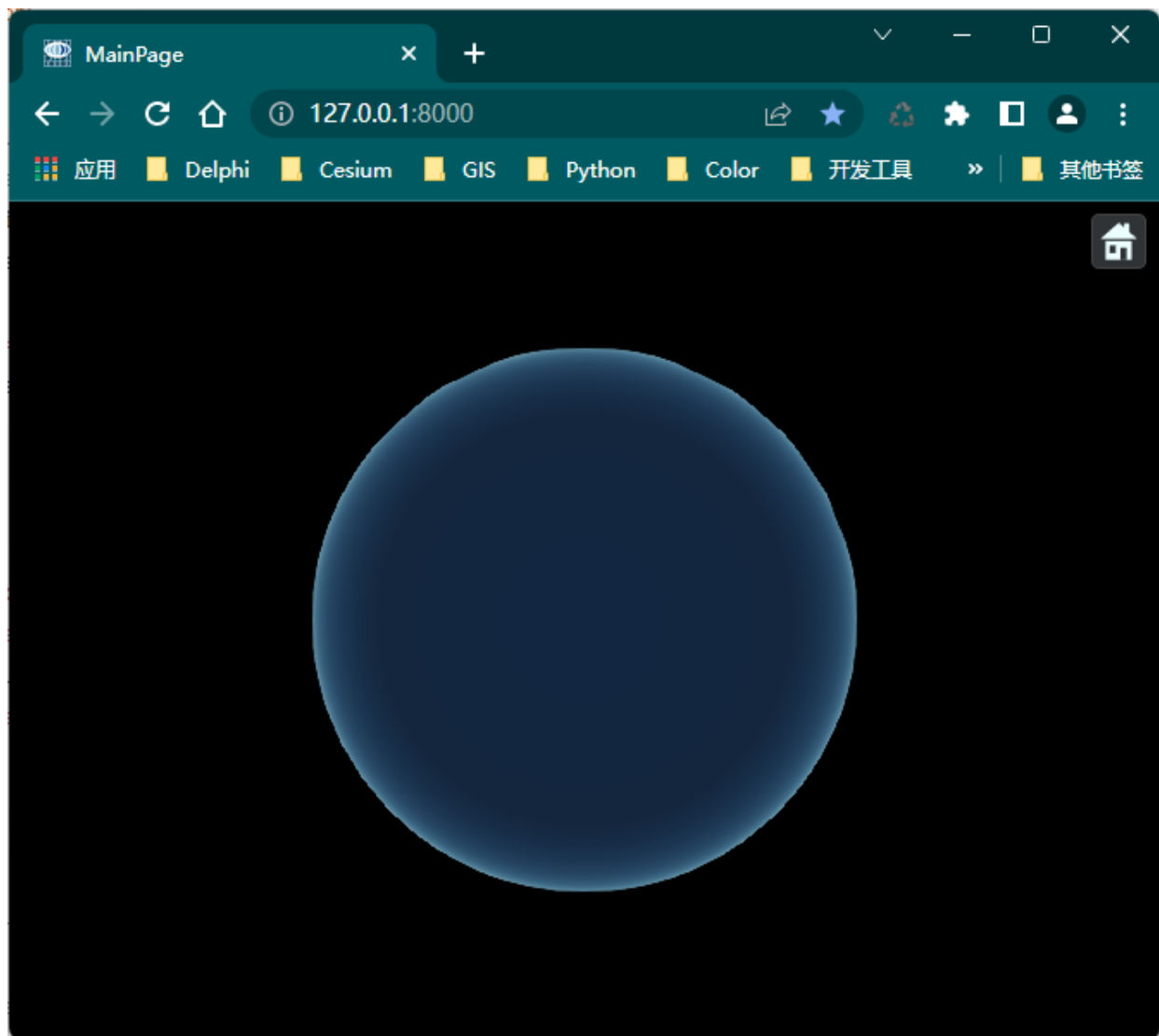
```
<template>
  <section id="idapp" v-show="visible" class="">
    <comcesium></comcesium> <!--1-->
  </section>
</template>

<script setup>
/*-- imports --*/
import {defineComponent, defineProps, onBeforeMount, onBeforeUnmount, onMounted,
onUnmounted} from "vue";
import Comcesium from "../ComCesium.vue"; // 2
/*-- ..以下省略.. --*/
```



```
<template>
  <section id="idapp" v-show="visible" class="">
    <!--3-->
    <comcesium class="vh-100"></comcesium> <!--1-->
  </section>
</template>

<script setup>
/*-- imports --*/
import {defineComponent, defineProps, onBeforeMount, onBeforeUnmount, onMounted,
onUnmounted} from "vue";
import Comcesium from "./ComCesium.vue"; // 2
/*-- ..以下省略.. --*/
```



8. Django Data API

8.1 model.py (for example)

```
from django.db import models

# Create your models here.
class Targometa(models.Model):
    platformname = models.IntegerField(primary_key=True, db_comment='浮标编号')
    transmissionsys = models.CharField(max_length=8, blank=True, null=True,
db_comment='传输系统')
    postioningsys = models.CharField(max_length=16, blank=True, null=True,
db_comment='定位系统')
    platformmodel = models.CharField(max_length=16, blank=True, null=True,
db_comment='浮标型号')
    platformmaker = models.CharField(max_length=30, blank=True, null=True,
db_comment='生产厂商')
    firmwareversion = models.CharField(max_length=32, blank=True, null=True,
db_comment='固件版本')
    serialnumber = models.CharField(max_length=32, blank=True, null=True,
db_comment='序列号码')
```

```

    manualversion = models.CharField(max_length=16, blank=True, null=True,
db_comment='手控版本')
    instrumenttype = models.CharField(max_length=3, blank=True, null=True,
db_comment='设备类型')
    projectname = models.CharField(max_length=64, blank=True, null=True,
db_comment='项目名称')
    datacentre = models.CharField(max_length=2, blank=True, null=True,
db_comment='数据中心')
    piname = models.CharField(max_length=60, blank=True, null=True, db_comment='负责
人员')
    startupdate = models.CharField(max_length=14, blank=True, null=True,
db_comment='启动日期')
    launchdate = models.CharField(max_length=14, blank=True, null=True,
db_comment='投放日期')
    latitude = models.DecimalField(max_digits=6, decimal_places=3, blank=True,
null=True, db_comment='投放纬度')
    longitude = models.DecimalField(max_digits=6, decimal_places=3, blank=True,
null=True, db_comment='投放经度')
    deployplatform = models.CharField(max_length=40, blank=True, null=True,
db_comment='部署平台')
    parkpressure = models.IntegerField(blank=True, null=True, db_comment='滞留压力')
    profilepressure = models.IntegerField(blank=True, null=True, db_comment='剖面压
力')
    geom = models.TextField(blank=True, null=True) # This field type is a guess.

class Meta:
    managed = False
    db_table = 'targometa'
    db_table_comment = 'Argo浮标元数据表'

```

8.2 views.py

```

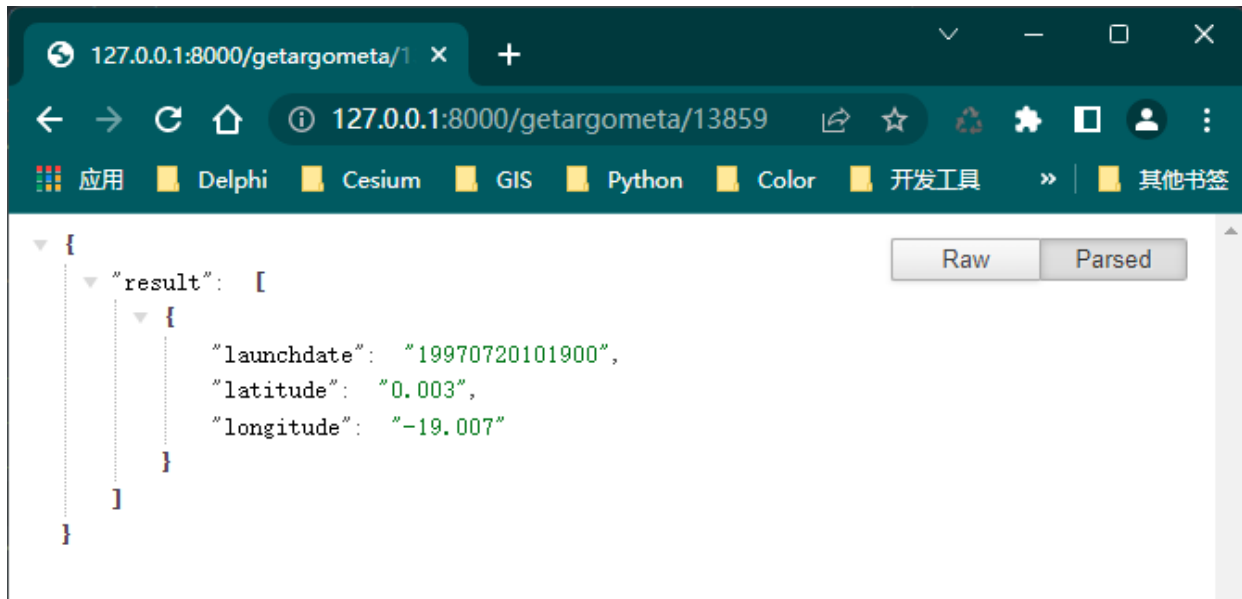
from . import models

def gettargometa(request, platformname):
    data = {}
    ###
    result = models.Targometa.objects #
    result = result.filter(platformname=platformname) #
    result = result.values('launchdate', 'latitude', 'longitude', ) #
    ###
    data['result'] = list(result)
    return JsonResponse(data)

```

8.3 app'urls.py

```
urlpatterns = [
    path('', views.begin, name='begin'),
    ###
    path('getargometa/<platformname>', views.getargometa, name='getargometa'), #
    ###
]
```



9. Axios

App.vue

```
<template>
  <section id="idapp" v-show="visible" class="">
    <div class="fixed-top w-50 bg-light opacity-75">{{ latitude }} , {{
longitude }}</div> <!-- 1 vue 变量 -->
    <comcesium class="vh-100"></comcesium>

  </section>
</template>

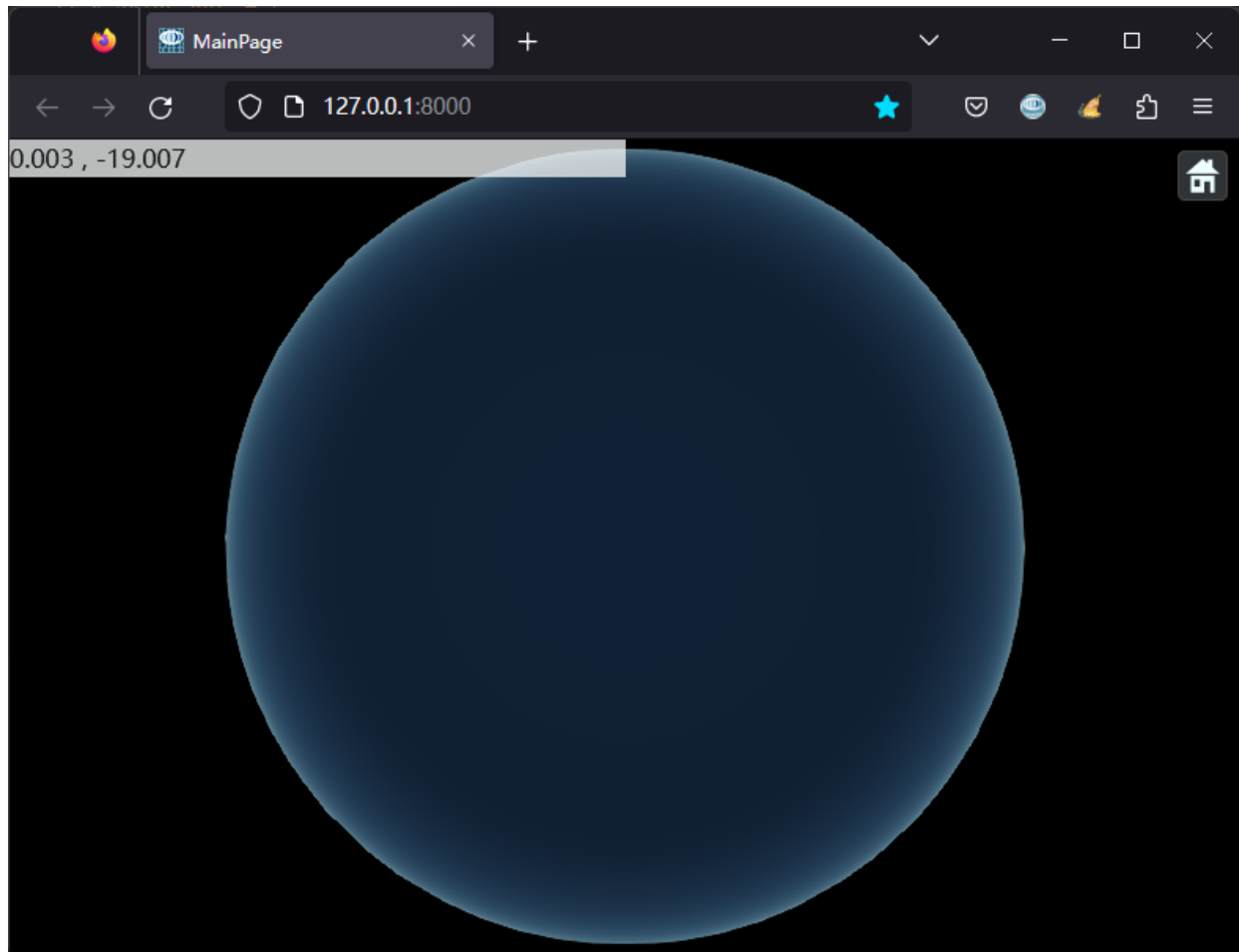
<script setup>
/*-- ..以上省略.. --*/
/*-- vars --*/
const latitude = ref(0); // 2 声明 vue 变量 , 用ref()初始化
const longitude = ref(0); // 3

/*-- methods --*/
// 4 axios get 方法
function getargometainfo(url) {
  axios.get(url).then(response => {
    const result = response.data.result;
    latitude.value = result[0].latitude; // 传递给 vue 变量 必须.value
    longitude.value = result[0].longitude; // 传递给 vue 变量
  }).catch(error => {
```

```

        console.log('getargometainfo=', error);
    });
}
onMounted(() => {
    console.log('App.Mounted');
    getargometainfo(`http://127.0.0.1:8000/getargometa/13859`); // 传递 api 调用 url
});
/*-- ...以下省略... --*/
</script>

```



10. Vue Show Data

10.1 {{ }}

10.2 v-model & click event

```

<template>
  <section id="idapp" v-show="visible" class="">
    <div class="fixed-top w-25 opacity-75">
      <!-- {{ latitude }} , {{ longitude }} 已被注释 -->
      <!-- 元素设计 3 -->
      <div class="input-group">
        <input type="text" class="form-control" readonly v-
model="longitude">

```

```

        <span class="input-group-text"></span>
        <input type="text" class="form-control" readonly v-model="latitude">
        <span class="input-group-text"></span>
        <input type="text" class="form-control" placeholder="编号" v-
model="platformname">
        <button class="btn btn-success" @click="querybyid">查询</button>
    </div>
</div>
<comcesium class="vh-100"></comcesium>
</section>
</template>

<script setup>
/*-- imports --*/
import {defineComponent, defineProps, onBeforeMount, onBeforeUnmount, onMounted,
onUnmounted, ref} from "vue";
import axios from "axios";
import Comcesium from "../ComCesium.vue";

const name = defineComponent({
    name: 'app',
});

/*-- props --*/
const props = defineProps({
    visible: {
        type: String,
        required: false,
        default: 'show',
    },
});

/*-- stores --*/

/*-- vars --*/
// 规划需要使用的变量 ref 初始化 1
const latitude = ref('');
const longitude = ref('');
const platformname = ref('');

/*-- methods --*/
function getargometainfo(url) {
    axios.get(url).then(response => {
        const result = response.data.result;
        latitude.value = result[0].latitude;
        longitude.value = result[0].longitude;
    }).catch(error => {
        console.log('getargometainfo=', error);
    });
}

/*-- events --*/
// 声明单击事件响应函数 2

```

```

function querybyid() {
  const url = `http://127.0.0.1:8000/getargometa/${platformname.value}`;
  getargometainfo(url);
}

onBeforeMount(() => {
  console.log('App.onBeforeMount');
});

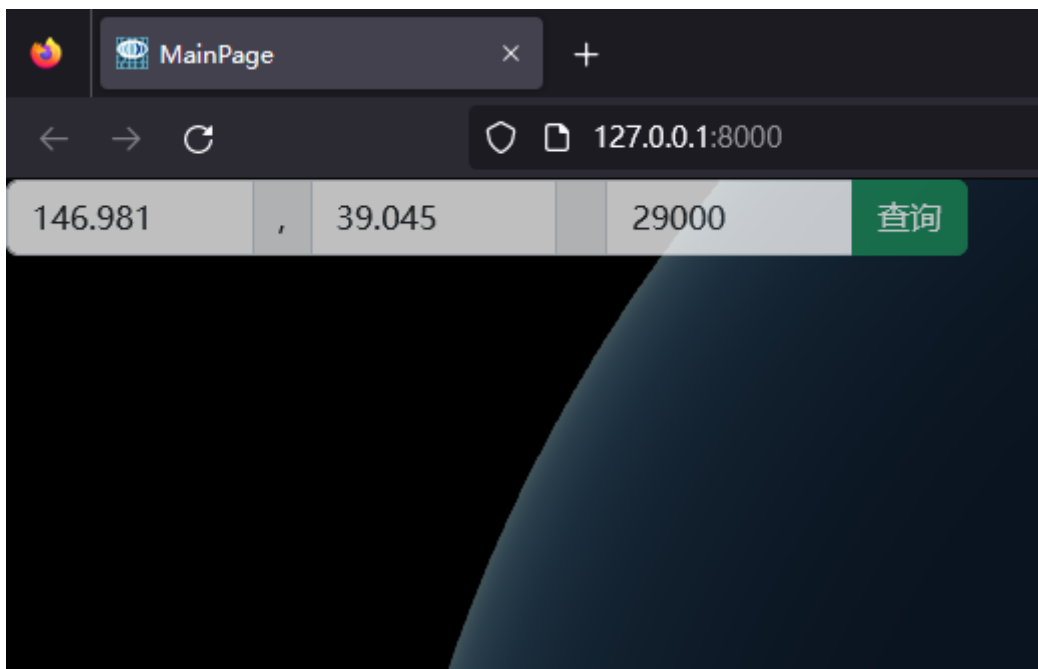
onMounted(() => {
  console.log('App.Mounted');
  getargometainfo(`http://127.0.0.1:8000/getargometa/13859`);
});

onBeforeUnmount(() => {
  console.log('App.onBeforeUnmount');
});

onUnmounted(() => {
  console.log('App.onUnmounted');
});
</script>

<style scoped>
</style>

```



10.3 v-for

```

<template>
  <section id="idapp" v-show="visible" class="">
    <div class="fixed-top w-25 opacity-75">
      <!-- {{ latitude }} , {{ longitude }} -->
      <div class="input-group">

```



```

        <input type="text" class="form-control text-success" readonly v-
model="longitude">
        <span class="input-group-text text-success">,</span>
        <input type="text" class="form-control text-success" readonly v-
model="latitude">
        <span class="input-group-text"></span>
        <input type="text" class="form-control" placeholder="编号" v-
model="platformname">
        <button class="btn btn-success" @click="querybyid">查询1</button>
        <button class="btn btn-primary" @click="querybyid2">查询2</button>
    </div>
    <hr>
    <table class="table table-primary">
        <thead>
            <tr>
                <th>number</th>
                <th>launchdate</th>
                <th>longitude</th>
                <th>latitude</th>
            </tr>
        </thead>
        <tbody>
            <tr v-for="(item,index) in rows">
                <td>{{ index + 1 }}</td>
                <td>{{ item.launchdate }}</td>
                <td>{{ item.longitude }}</td>
                <td>{{ item.latitude }}</td>
            </tr>
        </tbody>
    </table>
</div>
<comcesium class="vh-100"></comcesium>
</section>
</template>

<script setup>
/*-- imports --*/
import {defineComponent, defineProps, onBeforeMount, onBeforeUnmount, onMounted,
onUnmounted, ref} from "vue";
import axios from "axios";
import Comcesium from "../ComCesium.vue";

const name = defineComponent({
    name: 'app',
});

/*-- props --*/
const props = defineProps({
    visible: {
        type: String,
        required: false,
        default: 'show',
    },

```

```

    },
  });
  /*-- stores --*/

  /*-- vars --*/
  const latitude = ref('');
  const longitude = ref('');
  const platformname = ref(13859);
  // const result = ref(['Apple', 'Dell']);
  const rows = ref([
    // {"launchdate": "19970720101900", "latitude": "0.003", "longitude":
    "-19.007"},
    // {"launchdate": "19970720101900", "latitude": "0.003", "longitude": "-19.007"}
  ]);

  /*-- methods --*/
  function getargometainfo(url) {
    axios.get(url).then(response => {
      const result = response.data.result;
      latitude.value = result[0].latitude;
      longitude.value = result[0].longitude;
    }).catch(error => {
      console.log('getargometainfo=', error);
    });
  }

  function getargometainfo2(url) {
    axios.get(url).then(response => {
      const result = response.data.result;
      rows.value = result;
    }).catch(error => {
      console.log('getargometainfo2=', error);
    });
  }

  /*-- events --*/
  function querybyid() {
    const url = `http://127.0.0.1:8000/getargometa/${platformname.value}`;
    console.log(url);
    getargometainfo(url);
  }

  function querybyid2() {
    const url = `http://127.0.0.1:8000/getargometa2/${platformname.value}`;
    console.log(url);
    getargometainfo2(url);
  }

  onBeforeMount(() => {
    console.log('App.onBeforeMount');
  });

```

```

onMounted(() => {
  console.log('App.Mounted');
  // getargometainfo(`http://127.0.0.1:8000/getargometa/13859`);
});

onBeforeUnmount(() => {
  console.log('App.onBeforeUnmount');
});

onUnmounted(() => {
  console.log('App.onUnmounted');
});
</script>

<style scoped>
</style>

```

views.py

```

def getargometa2(request, platformname):
    data = {}
    ###
    result = models.Targometa.objects
    result = result.filter(platformname__lte=platformname)
    result = result.values('launchdate', 'latitude', 'longitude', )
    ###
    data['result'] = list(result)
    return JsonResponse(data)

```

urls.py

```

from django.urls import path

from . import apps
from . import views

app_name = apps.AppzeroConfig.name

urlpatterns = [
    path('', views.begin, name='begin'),
    ###
    path('getargometa/<platformname>', views.getargometa, name='getargometa'),
    path('getargometa2/<platformname>', views.getargometa2, name='getargometa'),
    ###
]

```

