

EverAPI 2.0 快速入门

配置环境

安装 Node.js

访问 [Node.js 官网](#)，安装 Node.js 环境，外网访问速度缓慢可以访问 [Node.js 国内网站](#)


检查 Node.js 是否安装成功

打开 **终端** 或 **命令提示符**，输入以下命令，检查 Node.js 是否安装成功

提示：如何打开 **终端** 或 **命令提示符**？请参阅 FAQ

```
$ node -v
```

如下图所示成功显示版本号，则表示 Node.js 已安装成功

A terminal window with a dark background and three colored window control buttons (red, yellow, green) in the top-left corner. The prompt is '\$ node -v' and the output is 'v12.14.0'.

```
$ node -v  
v12.14.0
```

设置 NPM 源

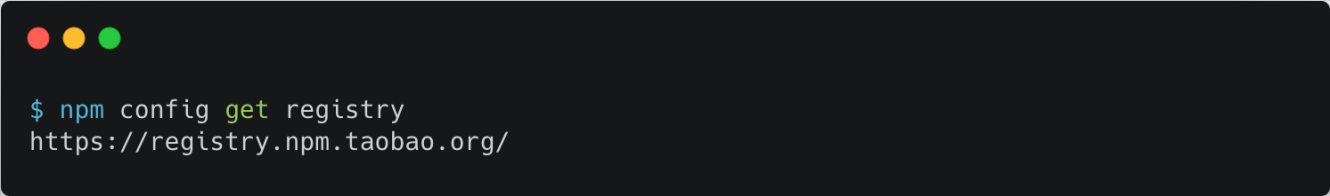
打开 **终端** 或 **命令提示符**，输入以下命令，将 NPM 源设置为国内镜像

```
$ npm config set registry https://registry.npm.taobao.org/
```

检查是否设置成功

```
$ npm config get registry
```

若结果如下图所示，则表示成功将 NPM 源设置为国内镜像

A terminal window with a dark background and three colored window control buttons (red, yellow, green) in the top-left corner. The prompt is '\$ npm config get registry' and the output is 'https://registry.npm.taobao.org/'.

```
$ npm config get registry  
https://registry.npm.taobao.org/
```

安装 http-server

打开 **终端** 或 **命令提示符**，输入以下命令，安装 http-server

```
$ npm install http-server -g
```

使用方法

创建 index.html 文件

如果正在使用 Windows 操作系统，此步骤请参阅 **Windows** 条目。如果使用 macOS 或 Linux 操作系统，此步骤请参阅 **macOS & Linux** 条目

Windows

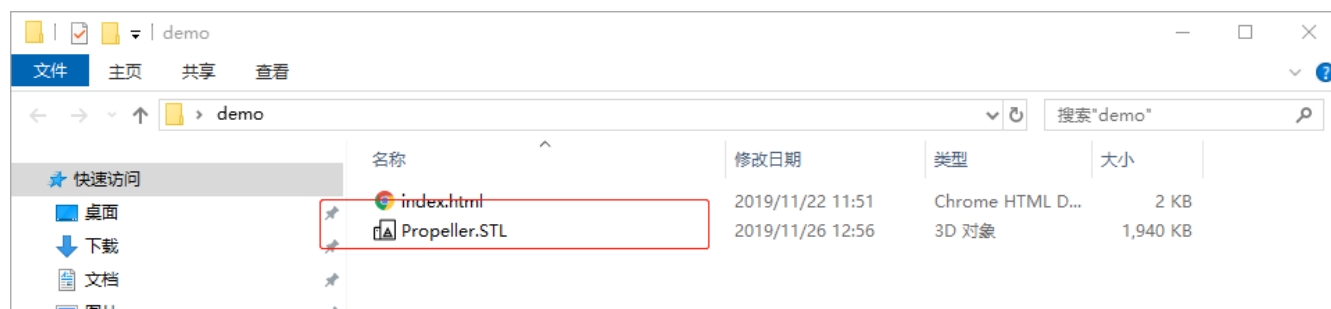
第一步：在桌面上创建文件夹 demo，然后打开 demo 并新建一个文本文档，windows 可点击右键新建文本文档，需要显示文件扩展名。



第二步：将新建的文件名修改为 index.html



第三步：将 3D 文件拷贝到 demo 目录下：



第四步：使用文本编辑器（使用记事本即可）打开 index.html，将下述内容拷贝到文件内，并保存关闭

请注意！请下述内容中的 **Propeller.stl** 替换为你放到 demo 目录的 3D 文件名称 + 扩展名

```
<!doctype html>
<html lang="en">

<head>
  <meta charset='utf8'>
  <title>EverAPI Demo</title>
  <script src="https://unpkg.com/axios/dist/axios.min.js"></script>
  <!-- 引入 EverAPI 脚本文件 -->
```

```

<script src="https://evercraft.co/api/everapi.js"></script>
</head>

<body>
  <h1>EverAPI 2.0 快速入门</h1>
  <div id="root"></div>
  <script>
    // EverAPI 配置
    const config = {}

    // api 是否已经加载成够的标志位
    let apiLoaded = false

    // 新建 api 实例
    // 构造函数的前两个参数分别是 AppID 和 AppKey, 这里的参数可用于本地 localhost:9000 调试
    const api = new EverAPI('6103ec4a2189', 'cc239b39cd09383aba9b1310db8cf334', config)

    // api 加载成功后调用 loadApi 函数
    window.addEventListener('oneverapiloading', () => {
      // 此处放需要渲染的文件
      const files = [{ filename: 'Propeller.stl', url: './Propeller.stl' }]
      loadApi(files)
    })

    // 此函数中调用 api 的 render 方法, 在网页中加载出 api 窗口
    function loadApi (files) {
      // 此方法的第一个参数为放置 dom 元素的 id, 第二个参数为需要在 api 中显示的模型的数组
      api.render('root', files)
      apiLoaded = true
    }
  </script>
</body>

</html>

```

macOS & Linux

第一步：打开终端，输入以下命令在桌面创建 **demo** 文件夹，并在此文件夹下创建 **index.html**

```

$ cd ~/Desktop
$ mkdir demo && cd demo
$ touch index.html

```

第二步：将 3D 文件拷贝到 **demo** 文件夹下

第三步：检查目录下的文件，输入 `ls` 命令，显示应如下图

```

$ ls
index.html Propeller.stl

```

第四步：使用文本编辑器打开 **index.html**，将下述内容拷贝到文件内，并保存关闭

请注意！请下述内容中的 **Propeller.stl** 替换为你放到 demo 目录的 3D 文件名称 + 扩展名

```
<!doctype html>
<html lang="en">

<head>
  <meta charset='utf8'>
  <title>EverAPI Demo</title>
  <script src="https://unpkg.com/axios/dist/axios.min.js"></script>
  <!-- 引入 EverAPI 脚本文件 -->
  <script src="https://evercraft.co/api/everapi.js"></script>
</head>

<body>
  <h1>EverAPI 2.0 快速入门</h1>
  <div id="root"></div>
  <script>
    // EverAPI 配置
    const config = {}

    // api 是否已经加载成够的标志位
    let apiLoaded = false

    // 新建 api 实例
    // 构造函数的前两个参数分别是 AppID 和 AppKey，这里的参数可用于本地 localhost:9000 调试
    const api = new EverAPI('6103ec4a2189', 'cc239b39cd09383aba9b1310db8cf334', config)

    // api 加载成功后调用 loadApi 函数
    window.addEventListener('oneverapiloaded', () => {
      // 此处放需要渲染的文件
      const files = [{ filename: 'Propeller.stl', url: './Propeller.stl' }]
      loadApi(files)
    })

    // 此函数中调用 api 的 render 方法，在网页中加载出 api 窗口
    function loadApi (files) {
      // 此方法的第一个参数为放置 dom 元素的 id，第二个参数为需要在 api 中显示的模型的数组
      api.render('root', files)
      apiLoaded = true
    }
  </script>
</body>

</html>
```

启动 http-server 服务

在 **终端** 或 **命令提示符** 中，切换到上文创建的 **demo** 目录，执行以下命令，启动 http-server 服务

提示：如何在 **终端** 或 **命令提示符** 中打开指定的目录？请参阅 FAQ

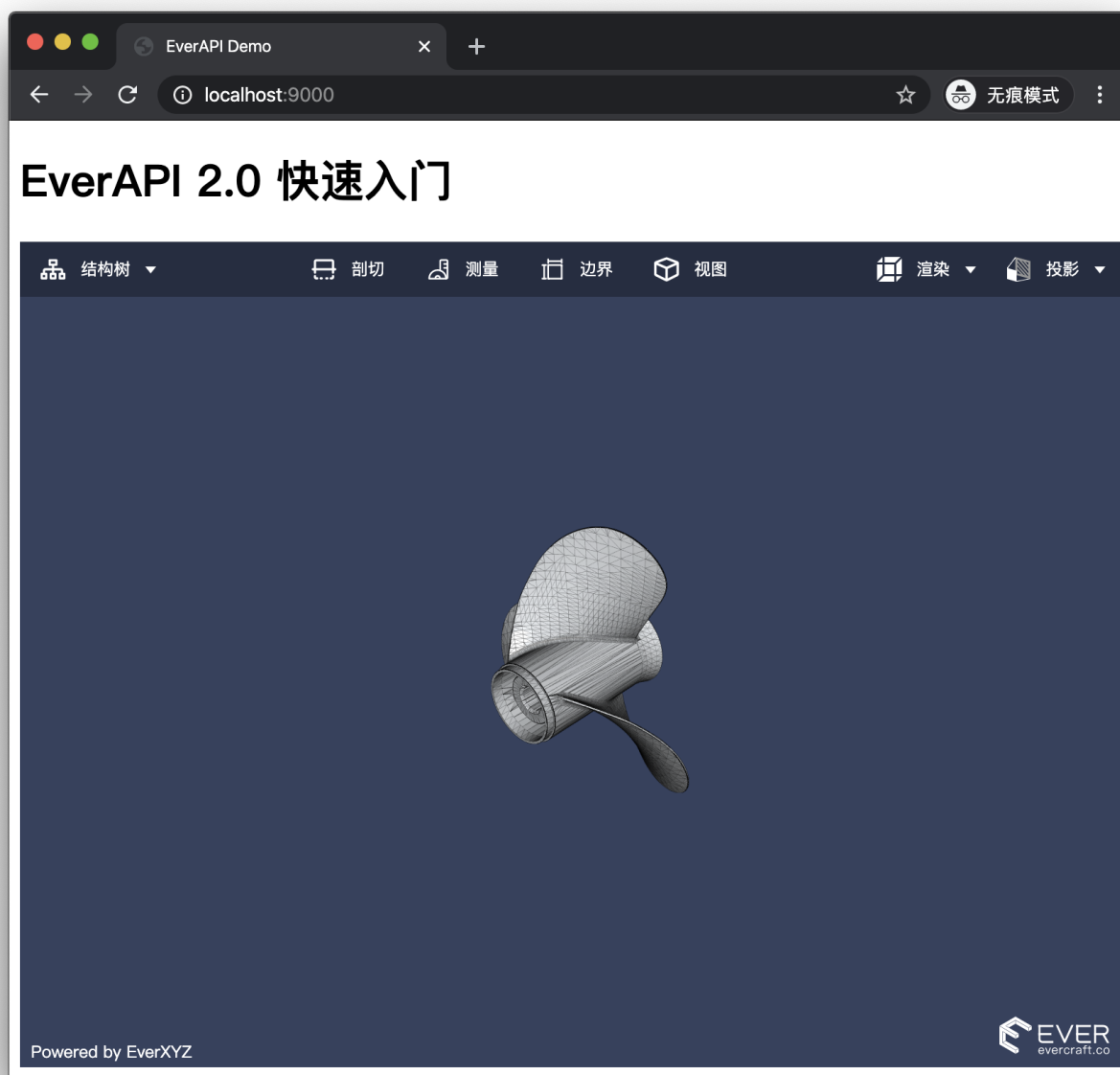
```
$ http-server -p 9000
```

若显示如下图，则表示服务启动成功

```
$ http-server -p 9000
Starting up http-server, serving ./
Available on:
  http://127.0.0.1:9000
  http://192.168.0.50:9000
Hit CTRL-C to stop the server
```

在浏览器中查看 3D 模型

打开浏览器，在地址栏中输入 <http://127.0.0.1:9000>，就可以显示出 3D 模型了



FAQ

如何在 macOS 上打开终端？

在 Dock 中打开应用程序 **启动台**，输入 **终端** 即可看到搜索结果。

如何在 Windows 上打开命令提示符？

在 **任务栏**（Windows 10）或 **开始菜单**（Windows 7）的搜索中搜索 **cmd** 或 **命令提示符**，即可看到搜索结果。

如何在 macOS 的终端中切换到指定目录？

输入以下命令，即可切换到 **桌面** 的 **demo** 目录中

```
$ cd ~/Desktop/demo
```

如何将 Windows 中的文件夹在命令提示符中打开？

首先在 **我的电脑** / **计算机** / **资源管理器** 中导航到目标目录，然后在地址栏中输入 **cmd**，就可以将目标目录在命令提示符中打开了。