



ThingJS 离线开发 SDK 说明书

(版本 Rev. 1.6.0)

北京优锆科技有限公司

2021 年 12 月 03 日

目 录

1	前言说明.....	2
1.1	硬件配置.....	2
1.2	软件依赖.....	2
2	软件安装.....	3
2.1	安装流程.....	3
3	项目开发.....	5
3.1	下载编辑器和浏览器.....	5
3.2	配置编辑器和浏览器.....	5
3.3	离线项目开发.....	6
3.4	在线开发已有项目离线开发.....	7
4	项目打包.....	11
5	开发须知.....	13
6	修订历史.....	15

1 前言说明

本文主要介绍 ThingJS 离线开发包的使用方法。

1.1 硬件配置

客户端最低配置：

cpu: 2G 主频 4 核

内存: 8G

硬盘: 100G, 推荐 ssd

显卡: 独显 2G 显存以上, gtx750 以上

1.2 软件依赖

需要安装最新版本 CampusBuilder 客户端：

最新版本可在 <https://store.thingjs.com/tools> 下载。

推荐安装 Visual Studio Code 编辑器：

最新版本可在 <https://code.visualstudio.com/> 下载。

推荐安装 Google Chrome 浏览器来调试场景：

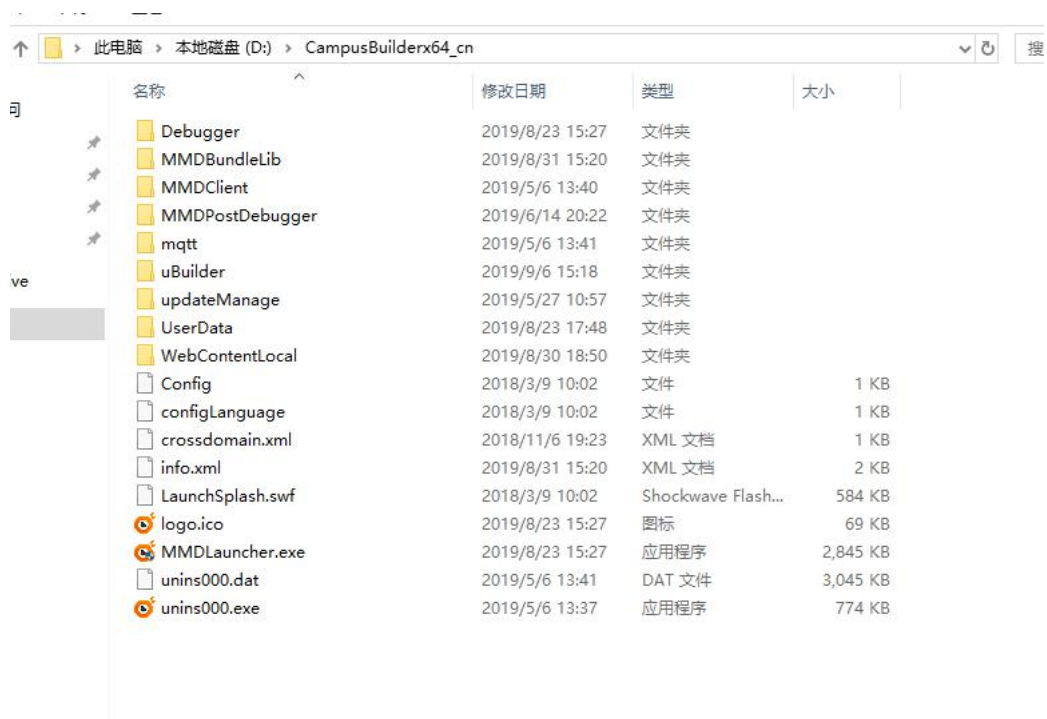
最新版本可在 <https://www.google.cn/intl/zh-CN/chrome/> 下载。

2 软件安装

本章节中，将详细说明 ThingJS 离线开发包的安装方法。

2.1 安装流程

首先关闭 CampusBuilder 客户端，右键桌面图标打开文件所在位置即找到摸摸搭客户端安装位置，如图所示。



将拿到的开发包解压到当前目录中，双击离线开发包内的 ThingJS_SDK.bat。

名称	修改日期	类型	大小
Debugger	2021/9/1 20:01	文件夹	
MMDBundleLib	2021/11/26 16:43	文件夹	
MMDClient	2020/7/7 12:01	文件夹	
MMDLauncher-v1.2.2.169	2021/11/26 15:00	文件夹	
MMDPostDebugger	2020/7/7 12:01	文件夹	
mqtt	2020/7/7 12:01	文件夹	
uBuilder	2021/11/26 15:05	文件夹	
updateManage	2020/9/27 9:10	文件夹	
UserData	2021/6/17 16:12	文件夹	
WebContentLocal	2021/11/26 15:03	文件夹	
离线开发包	2021/12/1 16:44	文件夹	
Contig	2020/9/27 21:02	文件	1 KB
Config_	2018/3/9 10:02	文件	1 KB
configLanguage	2018/3/9 10:02	文件	1 KB
crossdomain.xml	2018/11/6 19:23	XML 源文件	1 KB
info.xml	2021/11/26 15:01	XML 源文件	2 KB
LaunchSplash.swf	2018/3/9 10:02	SWF - File	584 KB
logo.ico	2018/10/22 10:37	ICO 图片文件	69 KB
MMDLauncher.exe	2021/4/28 18:44	应用程序	3,407 KB
unins000.dat	2020/7/7 12:01	DAT 文件	7,563 KB
unins000.exe	2020/7/7 11:52	应用程序	774 KB
unins001.dat	2021/11/26 15:01	DAT 文件	2 KB
unins001.exe	2021/11/26 15:00	应用程序	774 KB

安装过程中请勿关闭窗口，如下图所示：

```

C:\Windows\system32\cmd.exe
===== 离线开发SDK安装终端 =====
请确保摸摸搭客户端已经关闭!!!
=====
按任意键进行离线开发SDK安装...
===== START
请勿关闭该终端窗口!!!
正在为您安装离线开发包...
复制了 21212 个文件
-- 生成了备份文件 /t_bak/Debugger_20211203113410_bak
复制了 9 个文件
-- 生成了备份文件 /t_bak/WebContentLocal_20211203113410_bak
复制了 21212 个文件
复制了 17961 个文件
复制了 4514 个文件
复制了 5274 个文件
离线开发包安装步骤完成。
===== END
双击桌面快捷方式启动客户端。
=====
按任意键将退出...
    
```

双击桌面快捷方式启动客户端，至此离线开发包安装步骤完成。

3 项目开发

3.1 下载编辑器和浏览器

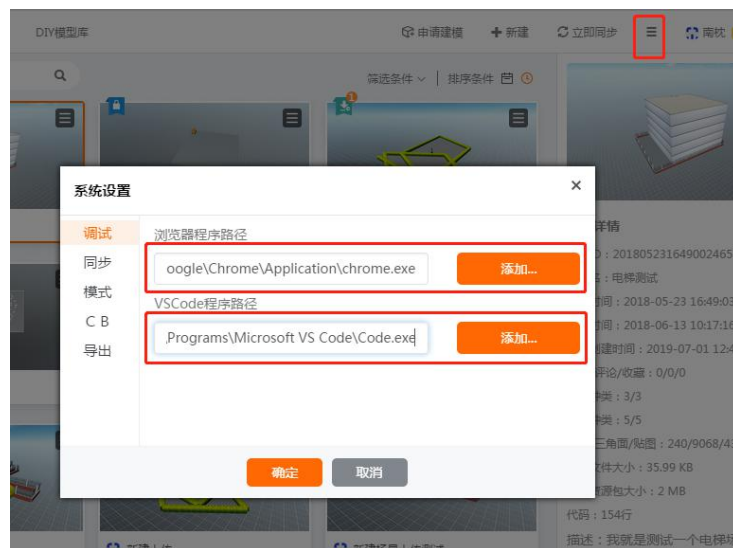
离线开发包 推荐使用的编辑器是 Visual Studio Code， 推荐使用的浏览器是 chrome。使用前，需先安装，安装路径如下：

Visual Studio Code 编辑器（<https://code.visualstudio.com/>）；

Chrome 浏览器（<https://www.google.cn/intl/zh-CN/chrome/>）。

3.2 配置编辑器和浏览器

启动客户端之后应先配置 Visual Studio Code 路径以及默认浏览器 chrome 路径，保证程序可正常调用。在客户端菜单中“系统设置”中的调试中配置正确安装路径。如图所示。

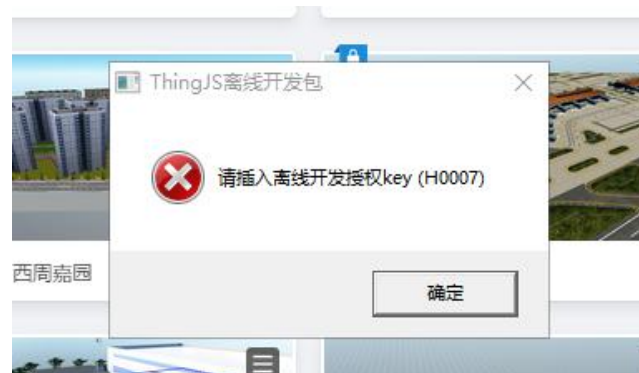


3.3 离线项目开发

配置正确之后，在插入秘钥 U 盘的情况下，便可以使用离线开发。开发方式为：点击场景菜单中的 ThingJS 第三方调试。如图所示。



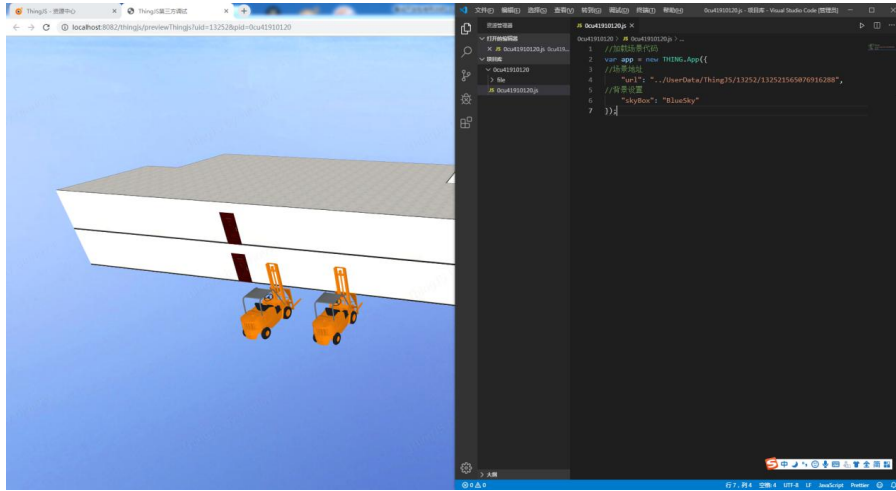
请确保程序运行时，秘钥 U 盘一直插在电脑中，如果拔出会出现如下错误：



秘钥 U 盘认证通过之后，会出现如下新建项目的页面，如图所示。



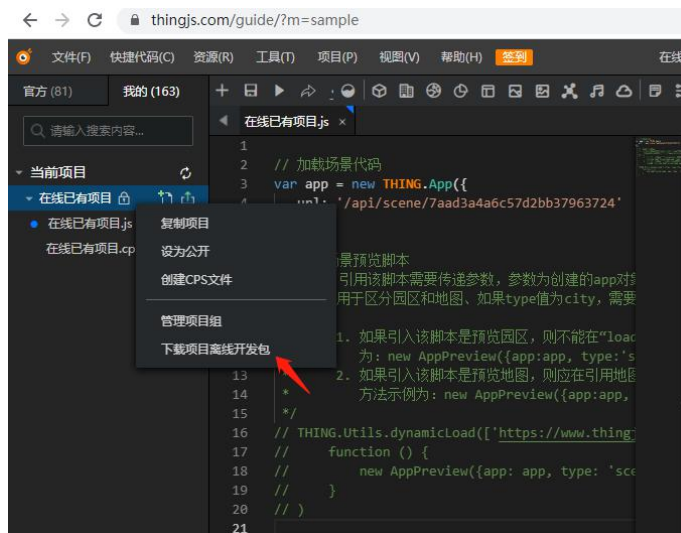
点击确定，项目新建完成。会默认弹出配置好的 Chrome 浏览器窗口，预览地址为本地服务器预览地址，同时还会打开 Visual Studio Code 编辑器，在 Visual Studio Code 编辑器中打开项目脚本。左侧为浏览器窗口，右侧为 Visual Studio Code 编辑器窗口。如图所示。



至此，便在 Visual Studio Code 编辑器中编写代码，保存。在 chrome 浏览器中刷新，即可预览当前项目的执行效果，开始项目的离线开发。

3.4 在线开发已有项目离线开发

如果要对“在线开发”里已有项目进行离线开发，需要去在线开发里找到项目并下载该项目的离线开发包。首先找到项目，然后右键项目找到“下载项目离线开发包”的入口。



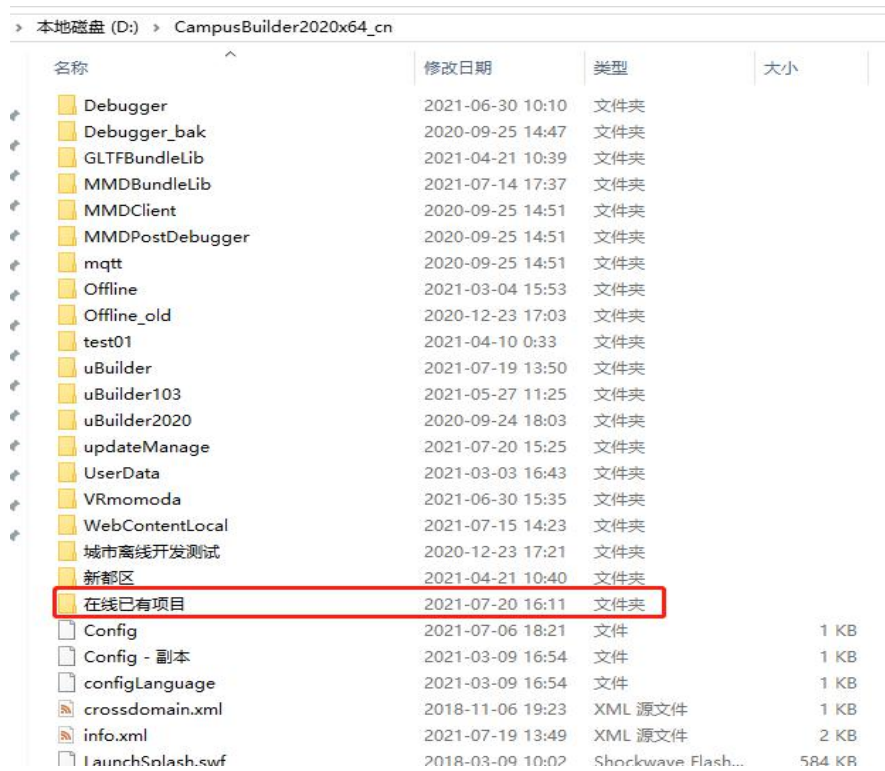
在弹出页面点击“申请打包下载”。



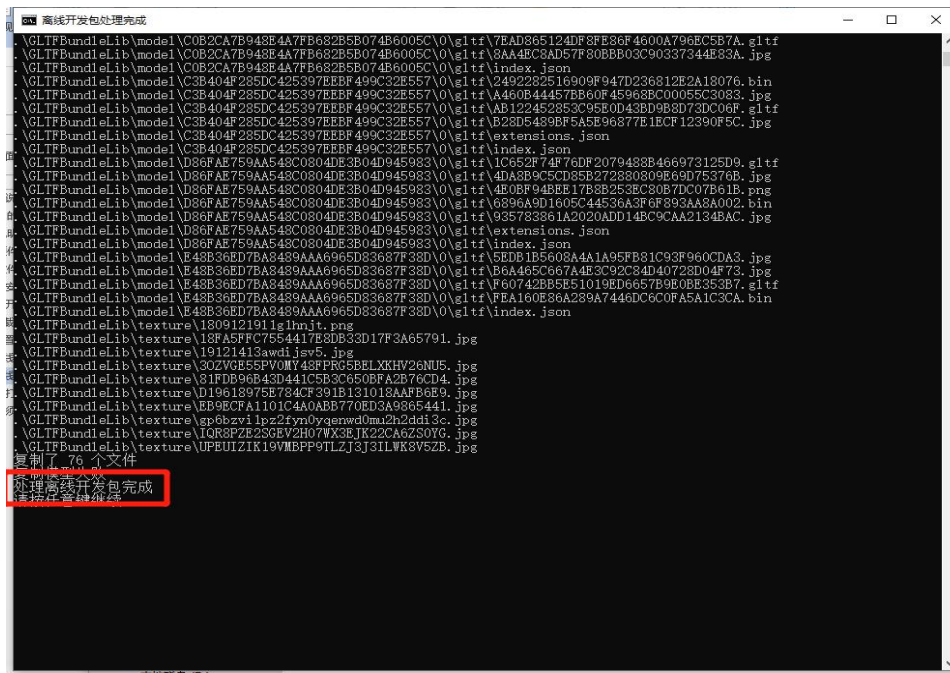
稍等后台打包完成之后可以在列表上看到一条下载记录，点击下载按钮，方可自动下载一个项目名命名的离线开发 zip 包。



将 zip 包放在客户端的根目录下，并右键解压到项目名文件夹下，解压之后目录结构如图所示。

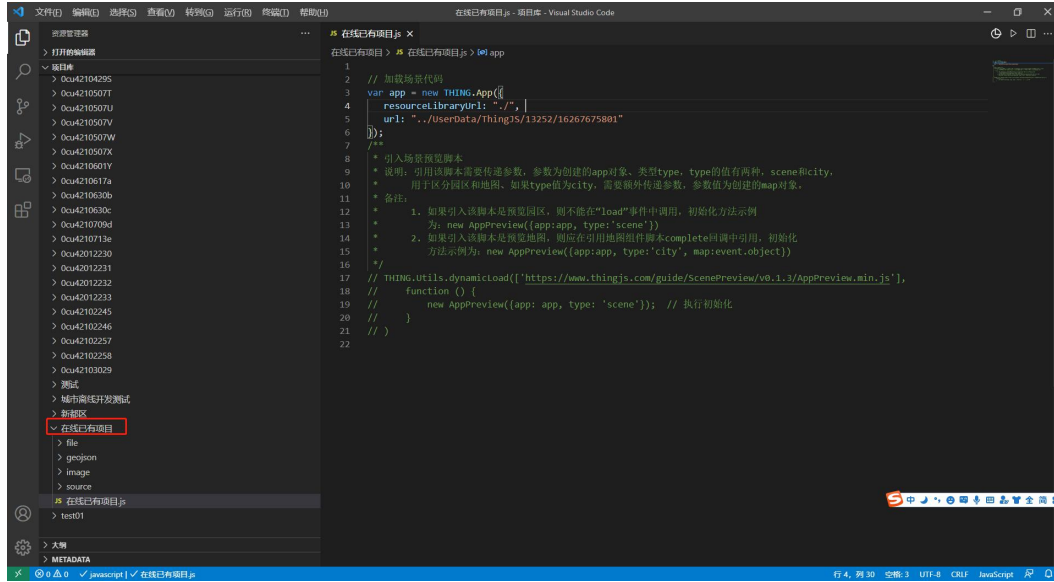


双击进去目录，运行文件夹同名的.bat 文件完成项目迁移到本地开发项目库下。



处理完成之后，可以在本地项目库“安装目录\UserData\Projects\id\项目库\”中看到同名的项目。项目的调试地址为：

<http://localhost:8082/thingjs/previewThingjs?uid=用户 id&pid=项目名>



```

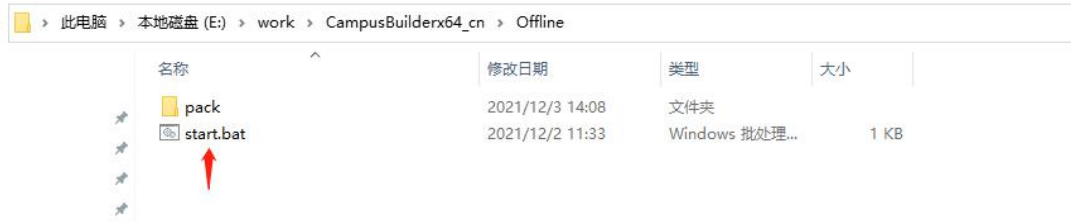
1 // 加载场景代码
2
3 var app = new THING.App({
4   resourceLibraries: [],
5   url: "../UserData/ThingJS/13252/16267675801"
6 });
7 /**
8  * 引入场景预览脚本
9  * 说明：引用脚本需要传递参数，参数为创建的app对象、类型type，type的值有两种，scene和city，
10  * 用于区分园区和地图，如果type值为city，需要额外传递参数，参数值为创建的map对象。
11  * 备注：
12  *   1. 如果引入该脚本是预览园区，则不能在“load”事件中调用，初始化方法示例
13  *     为：new AppPreview({app:app, type:'scene'})
14  *   2. 如果引入该脚本是预览地图，则应在引用地图组件脚本complete回调中引用，初始化
15  *     方法示例为：new AppPreview({app:app, type:'city', map:revent.object})
16  */
17 // THING.Utils.dynamicLoad(['https://www.thingjs.com/guide/scenePreview/v0.1.3/AppPreview_min.js'],
18 //   function () {
19 //     new AppPreview({app: app, type: 'scene'}); // 执行初始化
20 //   }
21 // )
22
    
```

4 项目打包

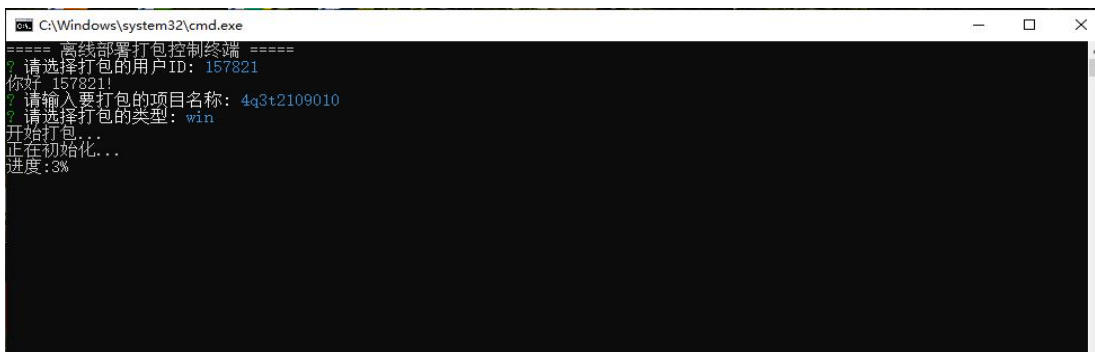
开发完成后，需要将项目打包输出为离线部署包，才能在项目的实际环境中部署。

打包步骤如下：

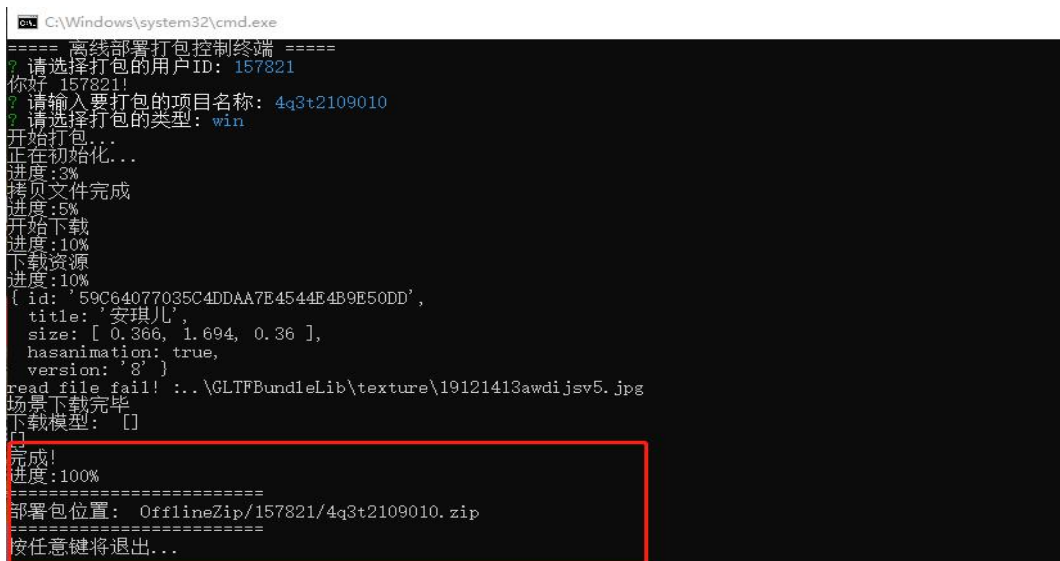
1. 进入安装目录下 Offline 文件夹

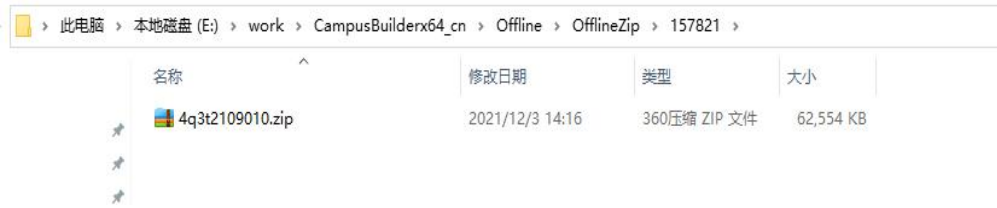


2. 双击 start.bat, 按提示选择要打包的用户 id、项目名称和打包类型，如下图所示：



3. 提示进度 100% 即为完成，生成的部署包在当前文件的 OfflineZip 目录对应的的用户 id 下，如下图所示。

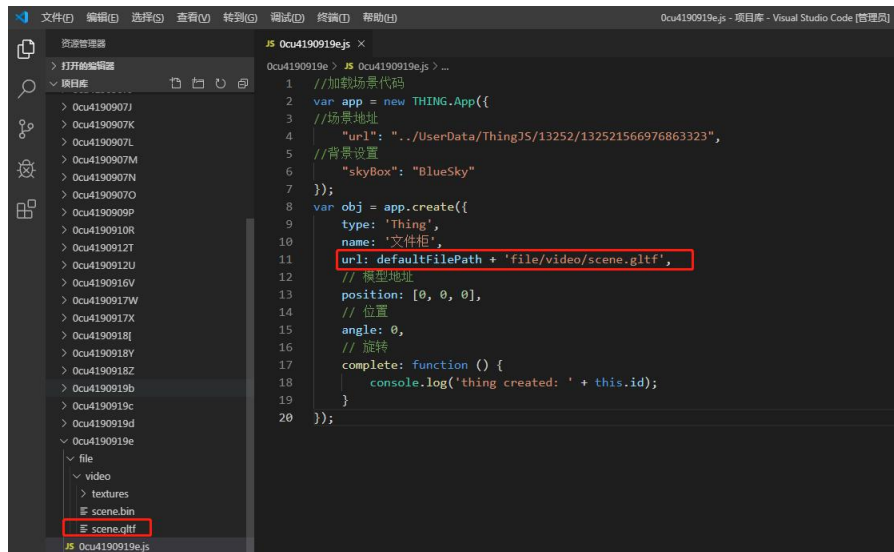




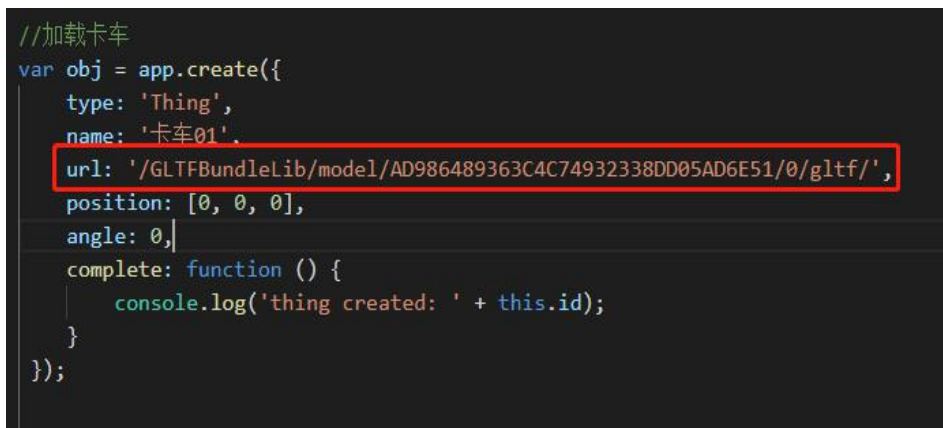
4. 离线部署流程请参考离线部署包中说明手册。

5 开发须知

1. 项目文件夹名字需要和主文件名字.js 保持一致。
2. 项目中的其他资源需要放在项目文件夹下的默认的文件中做为引用路径，保证打包之后可正常使用,项目引用 file 资源时应填写 defaultFilePath + 路径。注意该路径前不需要“/”从 file 文件夹开始写即可。引用路径写法如图所示。



3. 如果想用代码创建模型,可以去客户端右侧列表找到要加载的模型 id,然后模型地址为:!/GLTFBundleLib/model/AD986489363C4C74932338DD05AD6E51/0/gltf/', 其中的 AD986489363C4C74932338DD05AD6E51 为模型 id。如图所示。

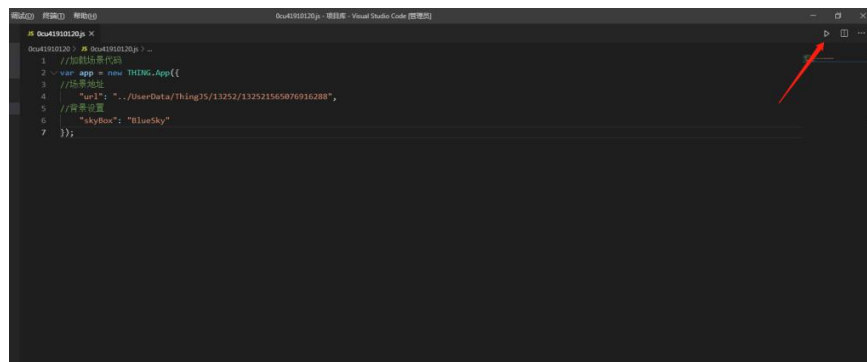


4. 如果想用代码创建 CB 中上传的模型，需要先去搭建环境找到模型 id，然后模型地址为：
 '/GLTFBundleLib/api/models/objmodel/d5958c76928a41e4a2f4f2740147e0f7/UnityToGLTF.gltf' 其中的 d5958c76928a41e4a2f4f2740147e0f7 为模型 id。如图所示。

```

// 加载CB上传的模型
var obj = app.create({
  type: 'Thing',
  name: '工地建筑_L_34',
  url: '/GLTFBundleLib/api/models/objmodel/d5958c76928a41e4a2f4f2740147e0f7/UnityToGLTF.gltf',
  position: [0, 50, 0],
  angle: 0,
  complete: function () {
    console.log('thing created: ' + this.id);
  }
});
    
```

5. 地图离线开发组件地址引用“/tjs/js/uearth.min.js”。
6. 全景图离线开发组件地址引用“/panooss/pano/js/thing.pano.min.js”
7. 该开发包中现仅含有官方授权的免费模型,个人模型和收费模型需要联系官方,将模型下载到本地,才可使用。
8. 若想要在本机修改其他已存在项目，项目库的路径是客户端的安装目录下的“\UserData\Projects\用户 id\项目库\”。这里每一个文件夹代表一个项目，项目主文件.js 需要和项目文件夹的名字保持一致，项目的调试地址为：
<http://localhost:8082/thingjs/previewThingjs?uid=用户 id&pid=项目名>。
9. 提供一个 Visual Studio Code 的小插件，可以快速打开项目浏览地址，文件在 Visual Studio Code 文件夹中的 vscode-plugin-thingjs-0.1.6.vsix，安装方法打开 Visual Studio Code 左侧面板最后一项插件项。然后菜单项选择从 vsix 安装选择该文件即可，安装完成之后可点击文件栏右上角执行项目或者邮件菜单打开本项目的浏览地址。



6 修订历史

版本	日期	更新内容描述
Rev. 1.0.0	2019/09/19	第一版
Rev. 1.1.0	2019/09/30	完善安装流程并上线
Rev. 1.2.0	2019/10/28	新增项目离线部署包部分
Rev. 1.3.0	2020/05/29	离线部署包方式升级修改打包方式
Rev. 1.4.0	2021/03/04	完善部分开发须知
Rev. 1.5.0	2021/06/30	文档样式调整，新增项目离线开发包用法
Rev. 1.6.0	2021/12/03	升级 SDK 安装、部署包下载过程