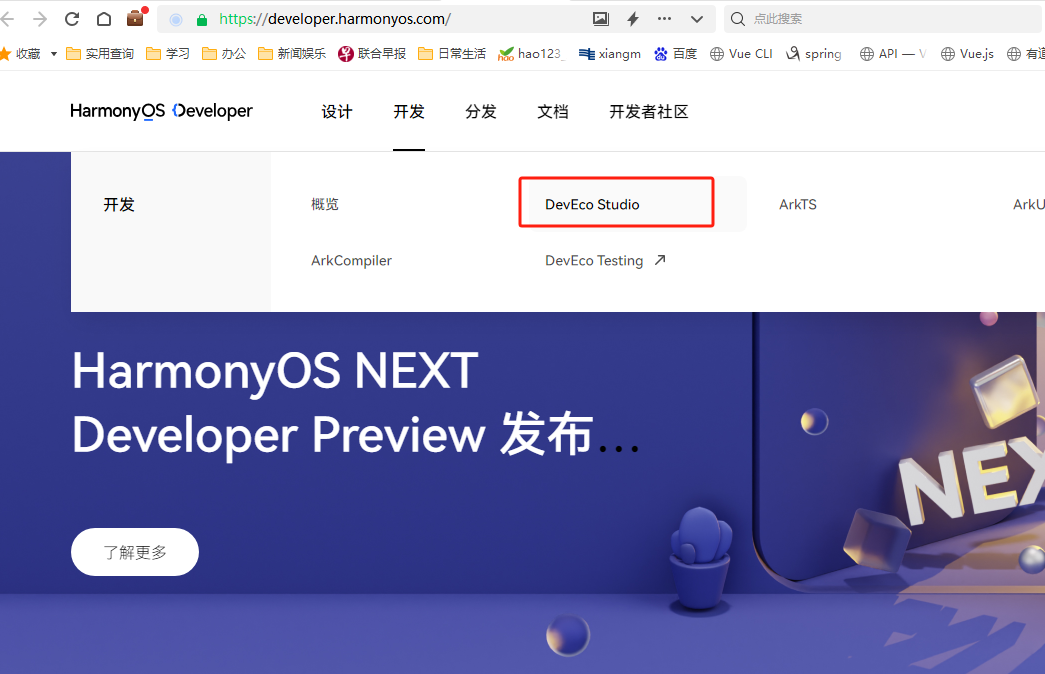
# 环境搭建

## 下载开发工具

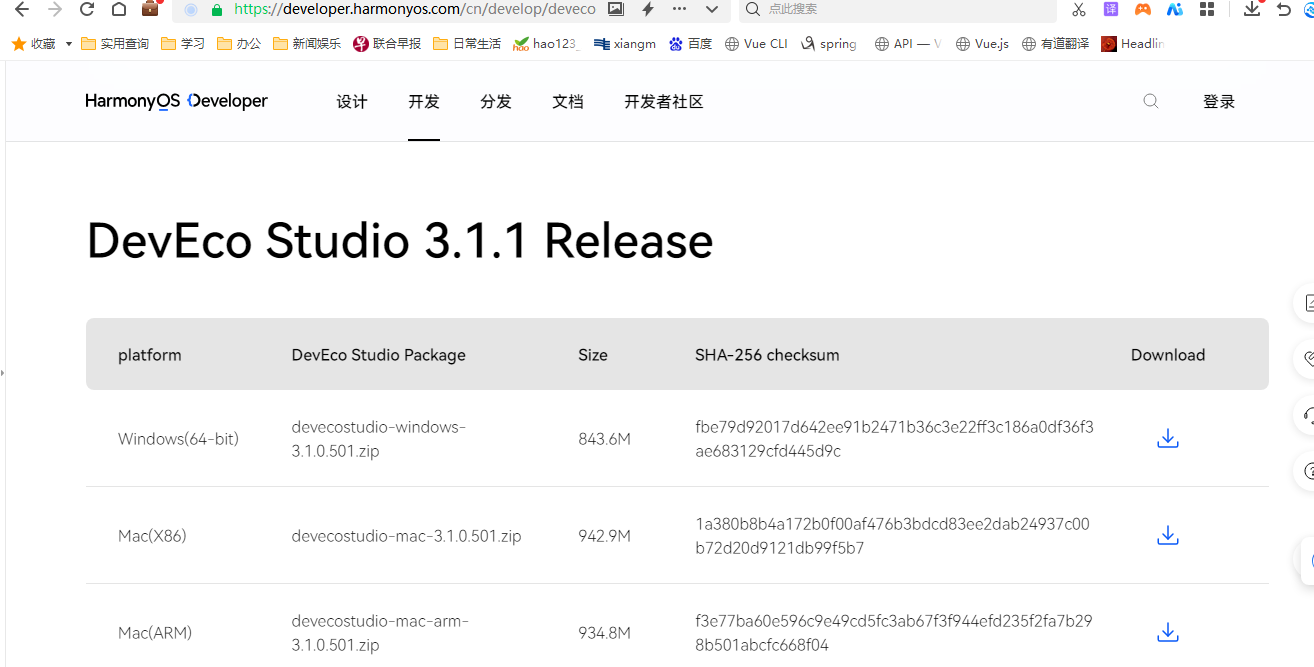
进入鸿蒙官网，下载https://www.harmonyos.com/，下载







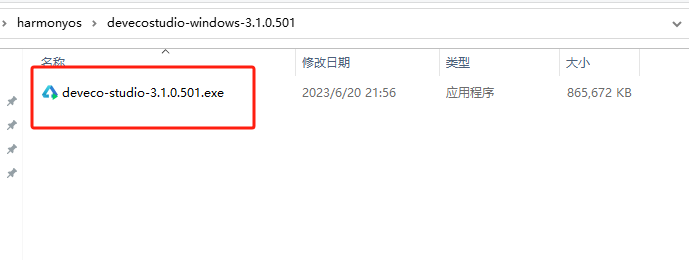
下载地址为：https://developer.harmonyos.com/cn/develop/deveco-studio/#download

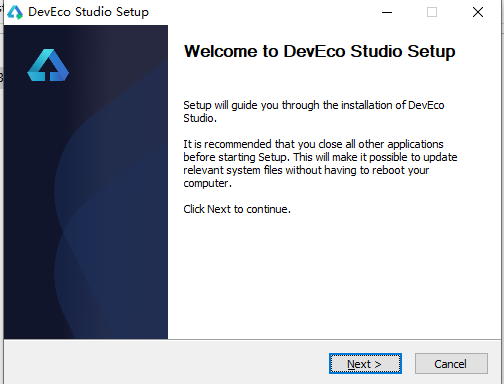


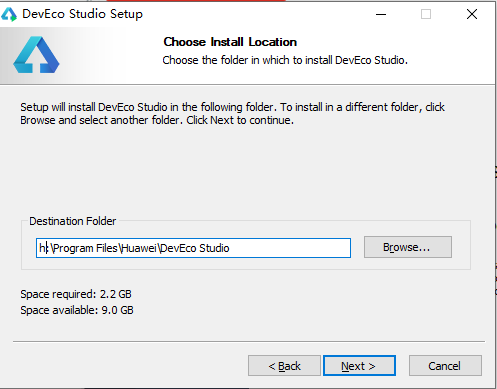
下载成功后，得如下文件devecostudio-windows-3.1.0.501.zip

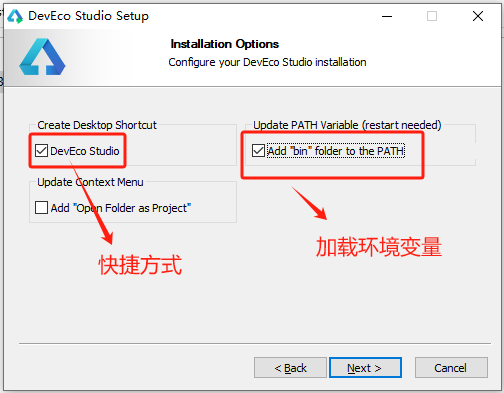
## 安装

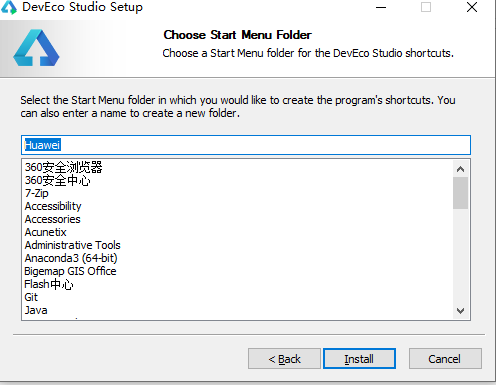
解压下载得devecostudio-windows-3.1.0.501.zip文件，执行exe



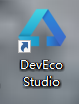


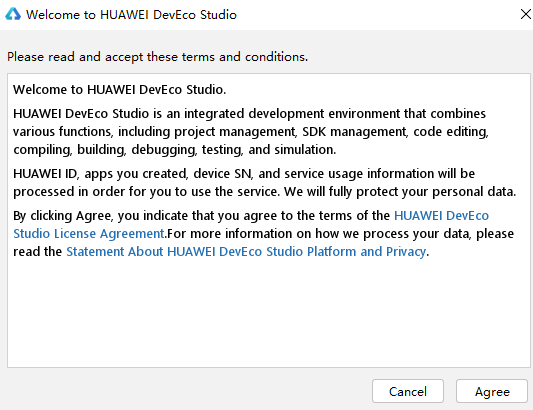






Next一直继续，直到安装完成。

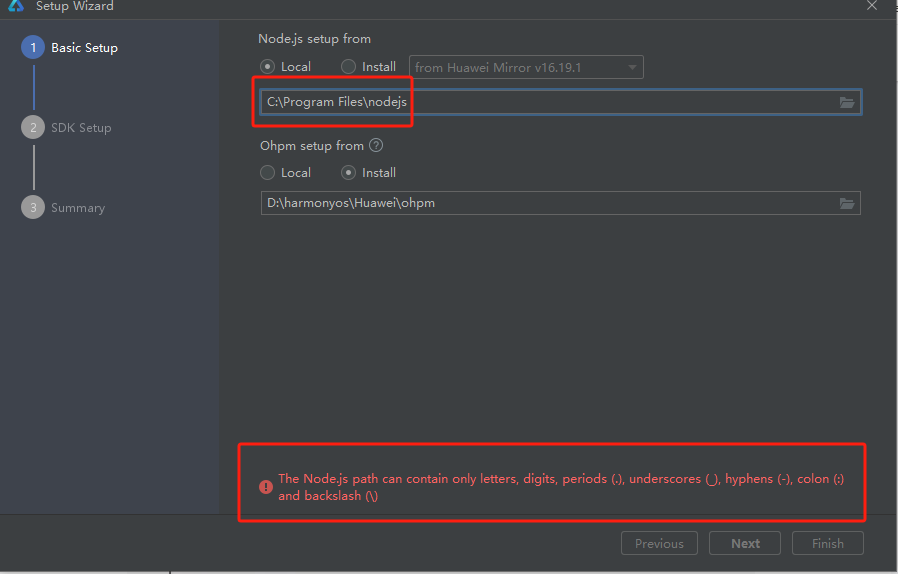
安装成功后，双击进入



进入后配置

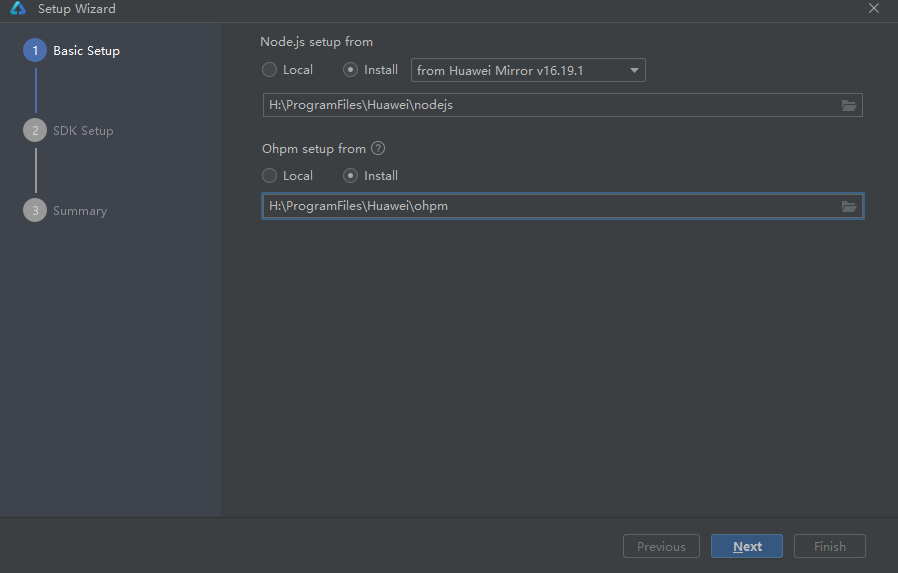
提示如下错误，不能Next

The [Node.js](https://so.csdn.net/so/search?q=Node.js&spm=1001.2101.3001.7020) path can contain only letters,digits,periods(.),underscores(\_), hyphens (-), colon (:) and backslash(\);

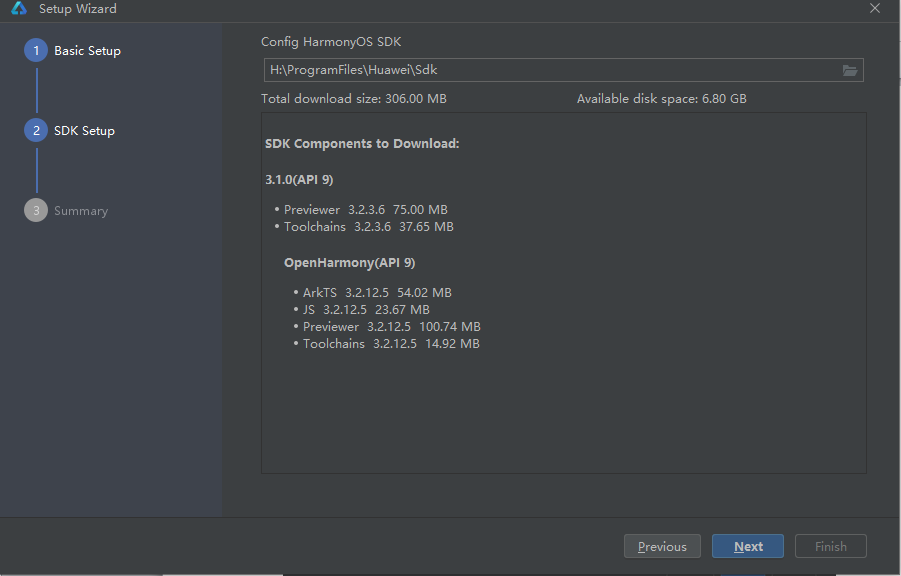


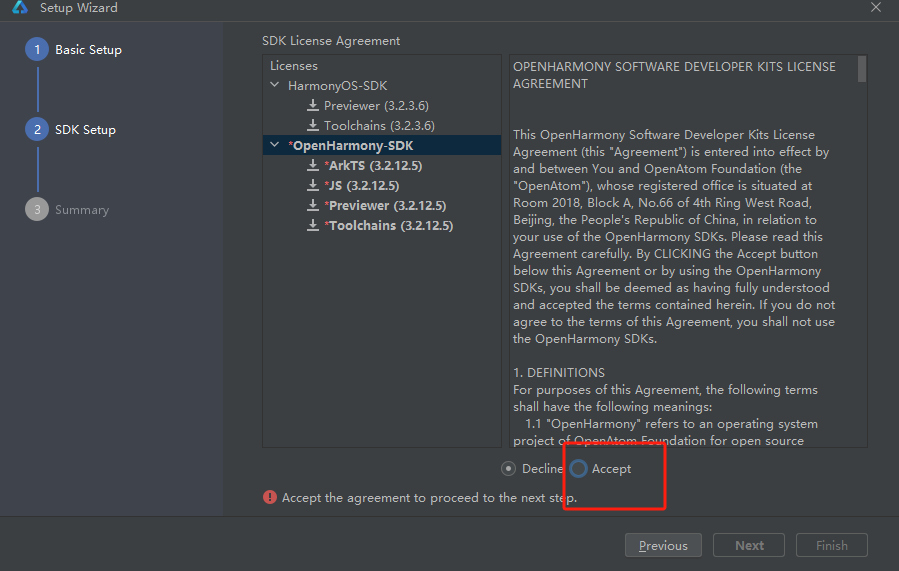
解决办法：

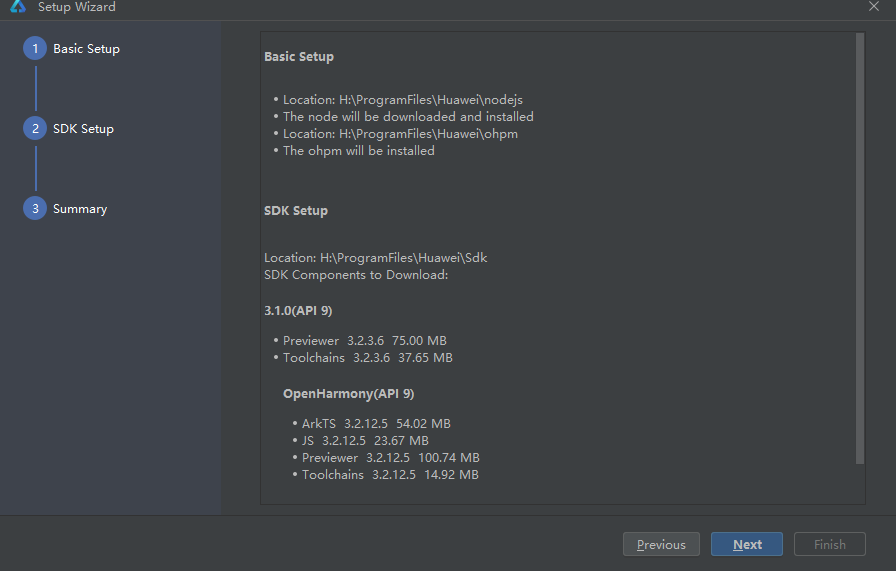
原因是nodejs安装路径不能有空格，在安装一个针对鸿蒙系统得nodejs

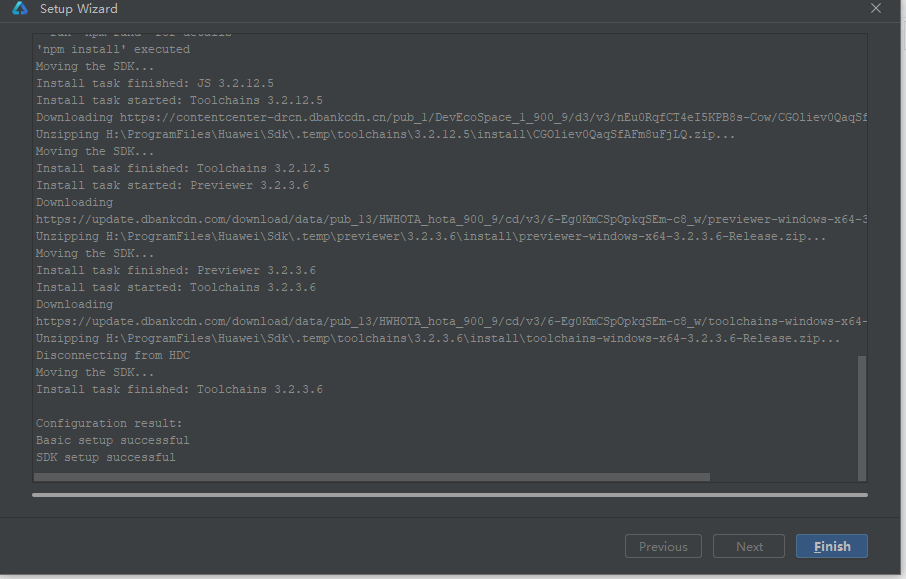


接下来安装sdk





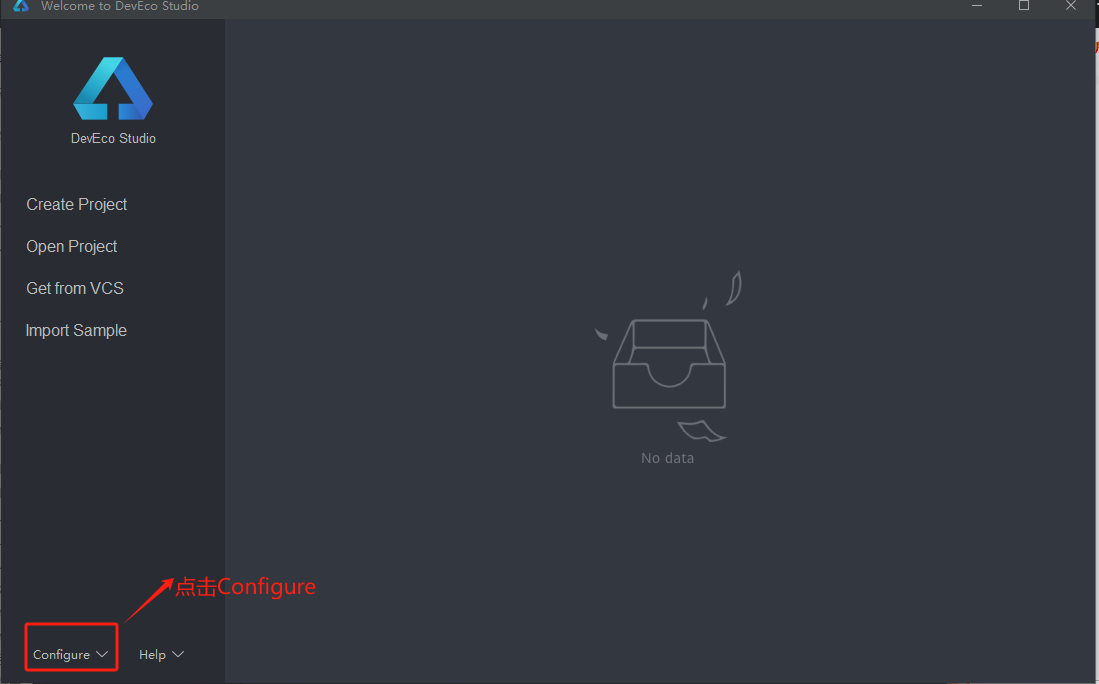




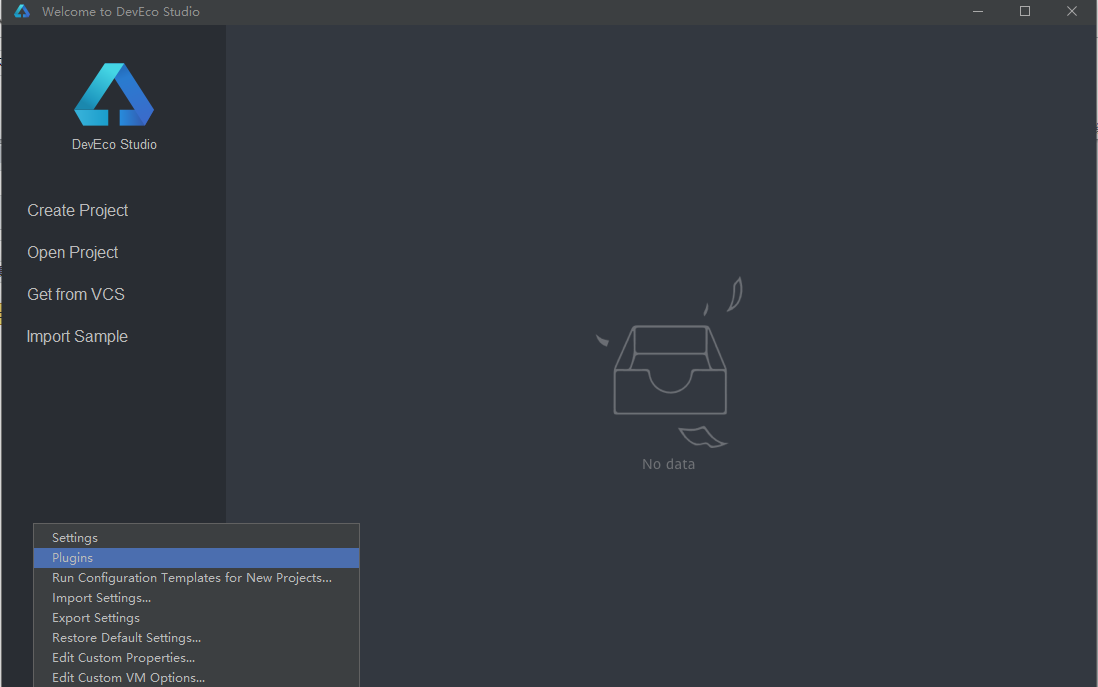
安装完成。

## 汉化DevEco Studio

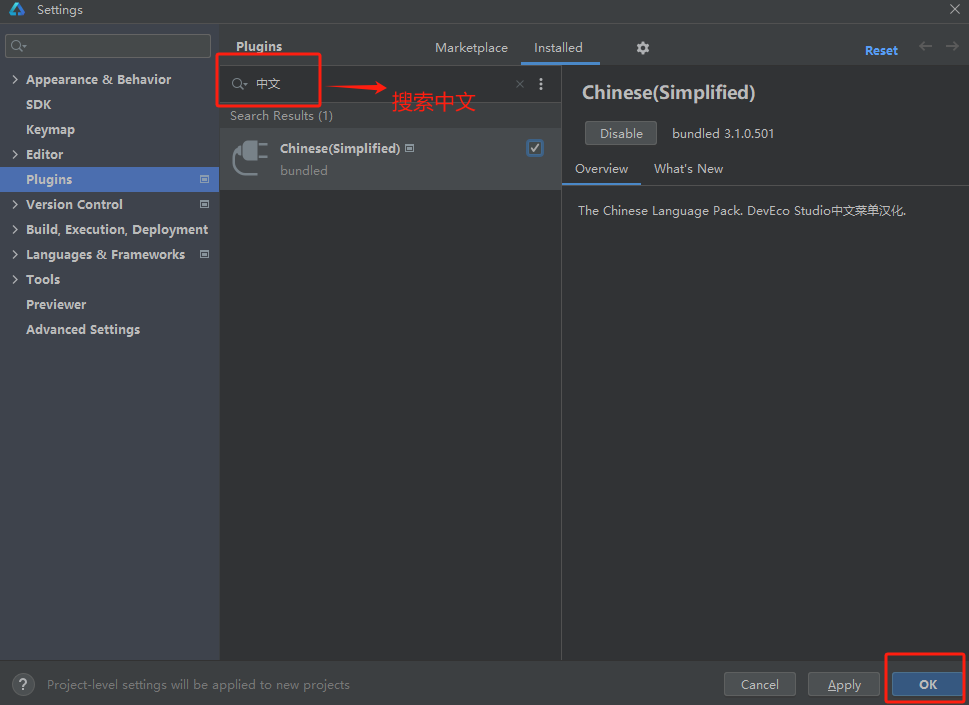
打开DevEco Studio，点击左下方得Configure（如果这里没有Configure，可能没有加载，点击这里得警告加载完成后再点击即可），



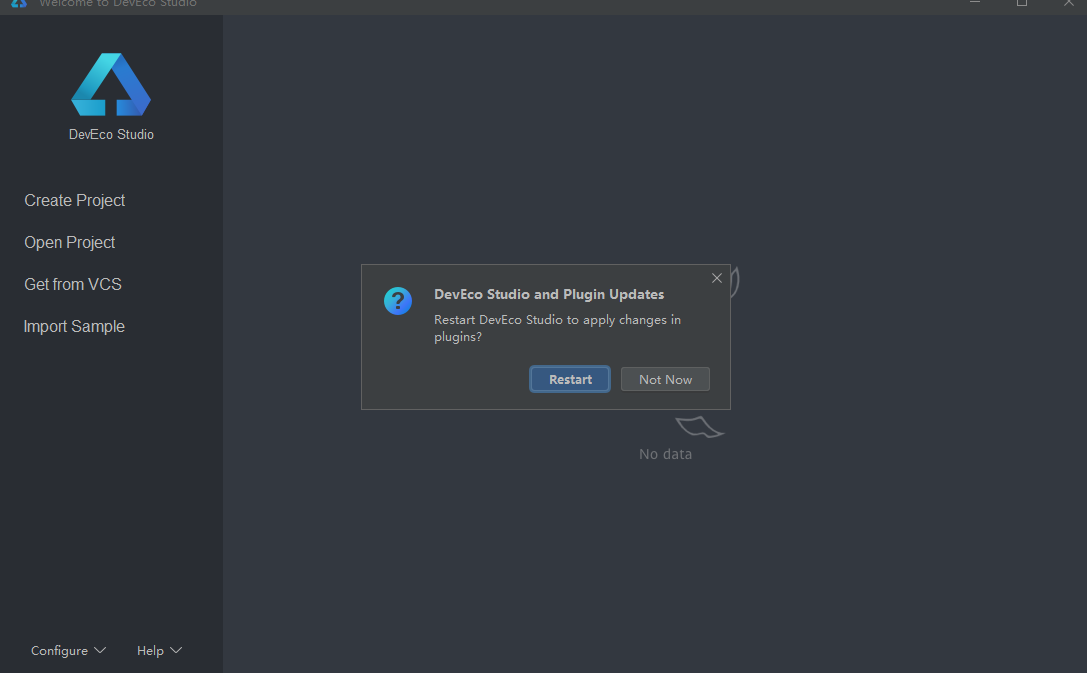
选择plugins



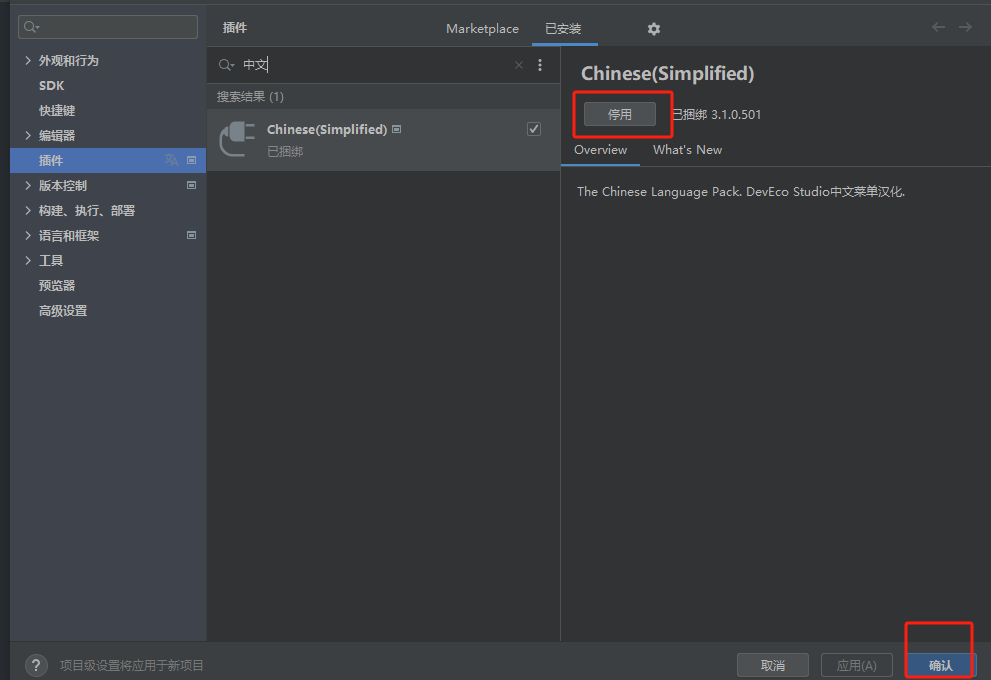
搜索中输入“中文”



重启



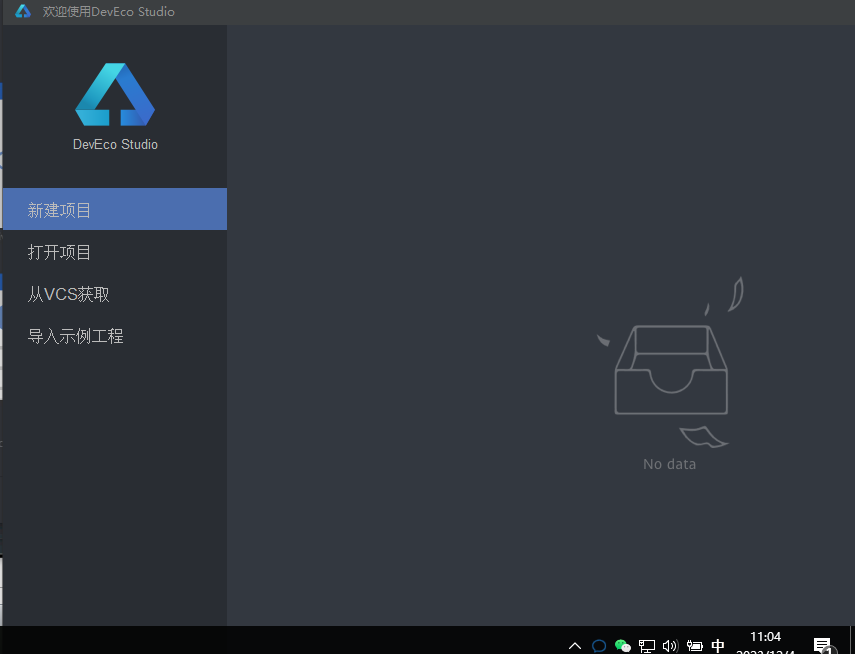
如果想恢复英文，则点击配置-插件 停用即可。



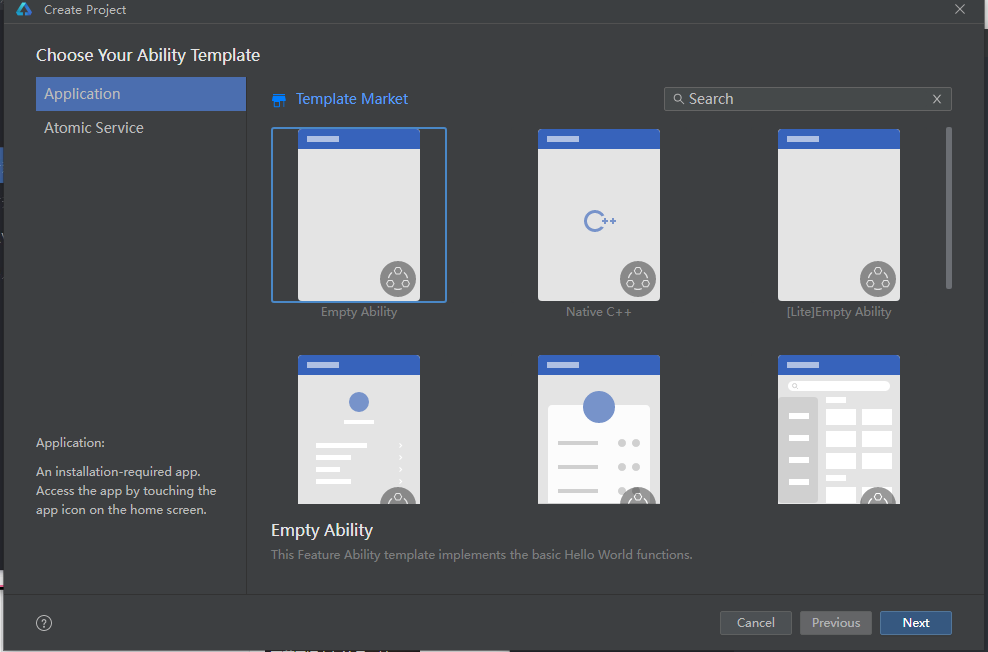
## 项目创建介绍和DevEco Studio使用

新建项目：

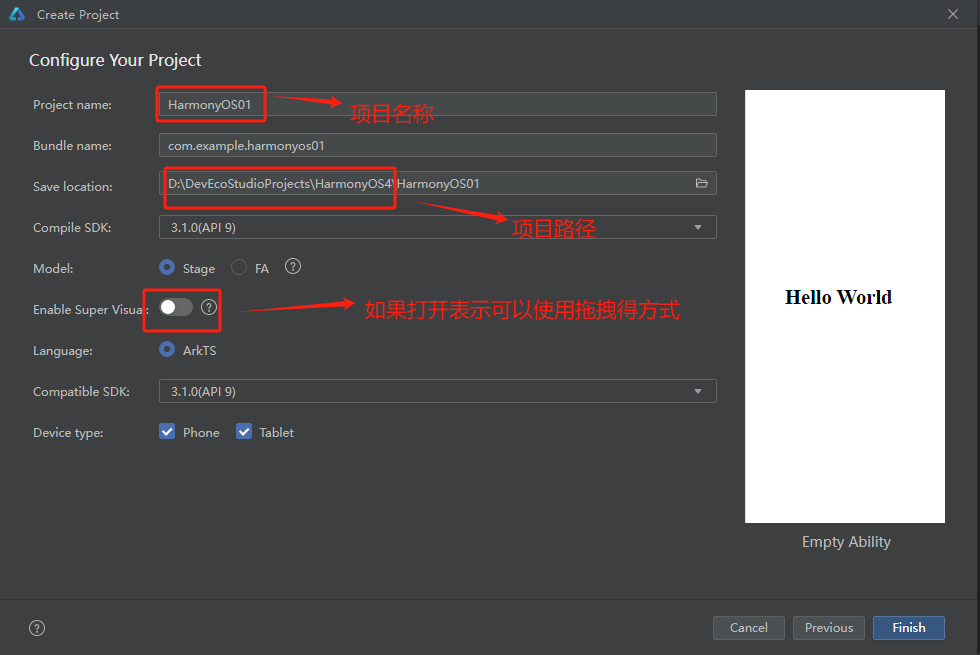
点击左侧“新建项目”，



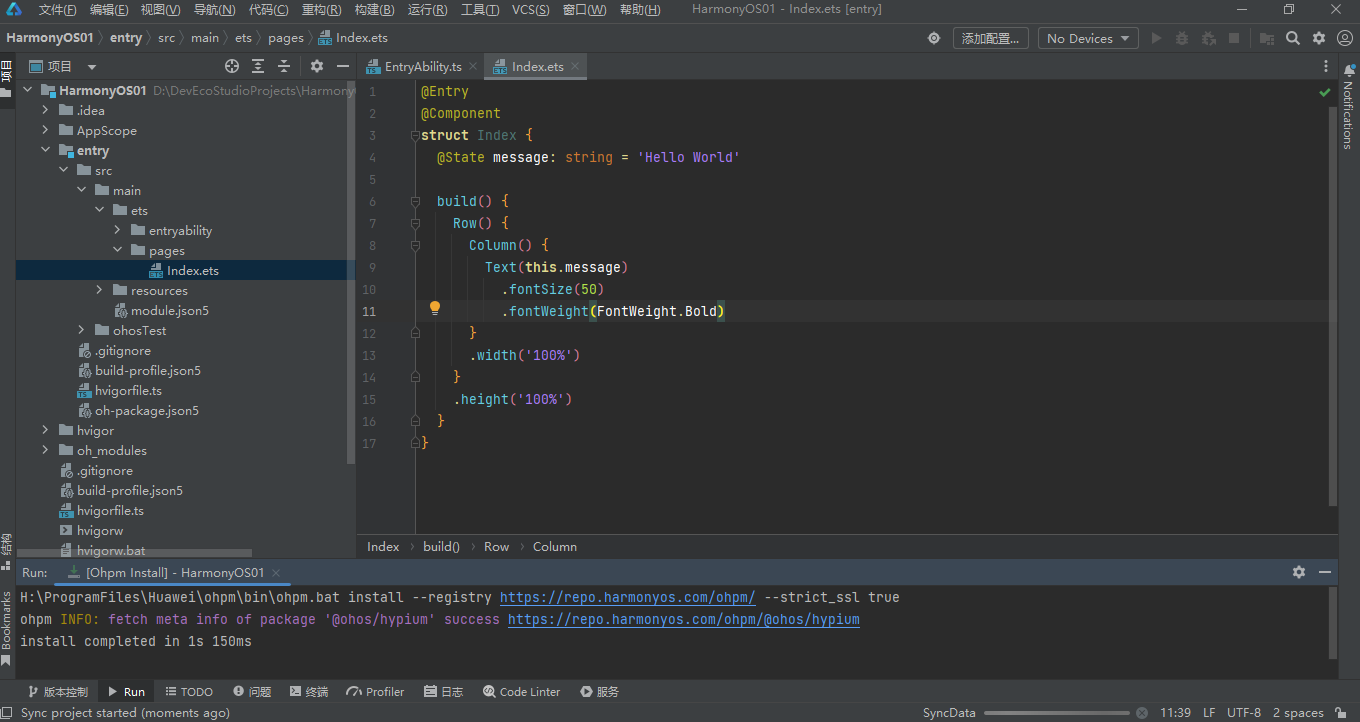
选择默认得“Template market”



选择项目路径

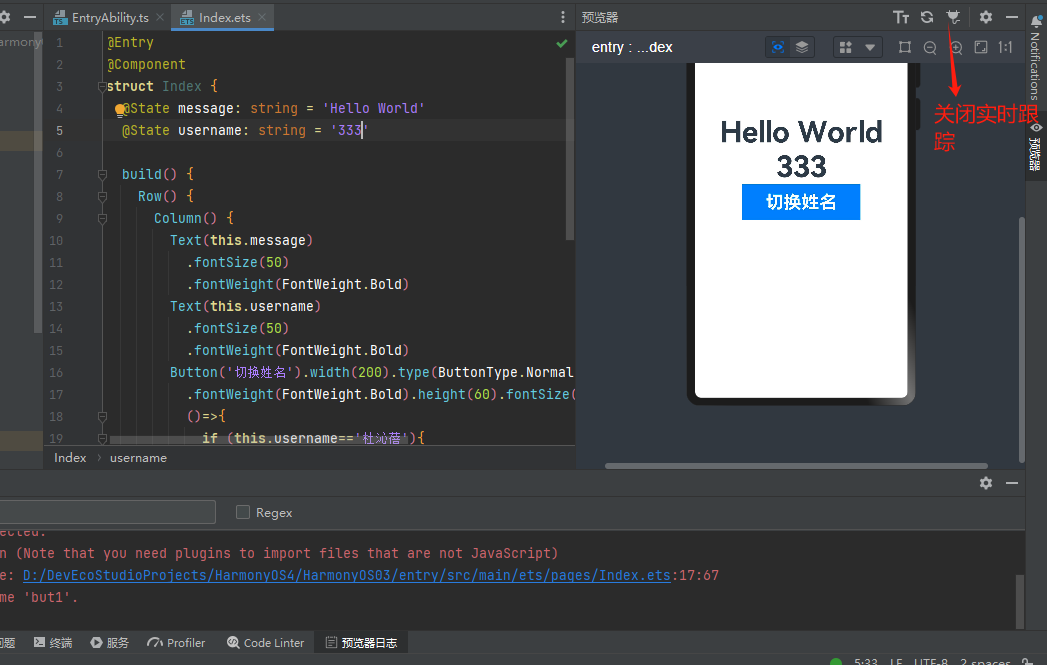


创建完成后项目如下：

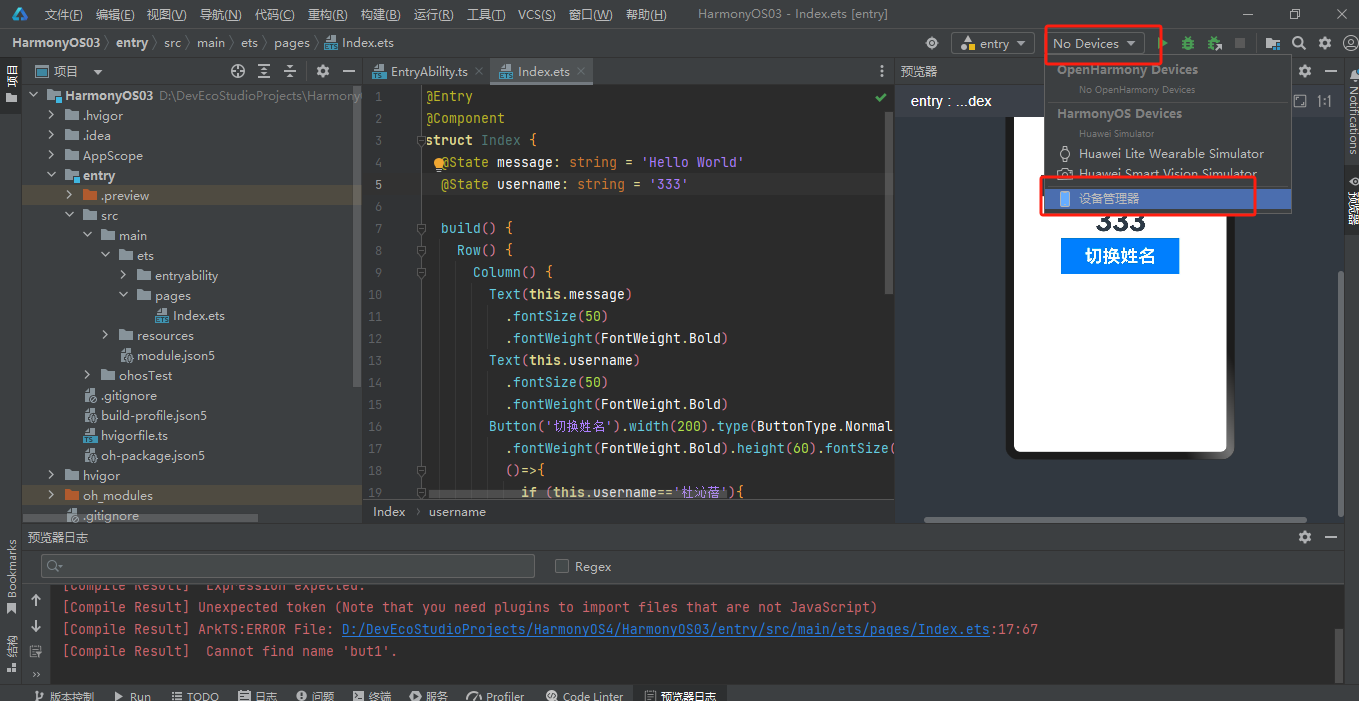


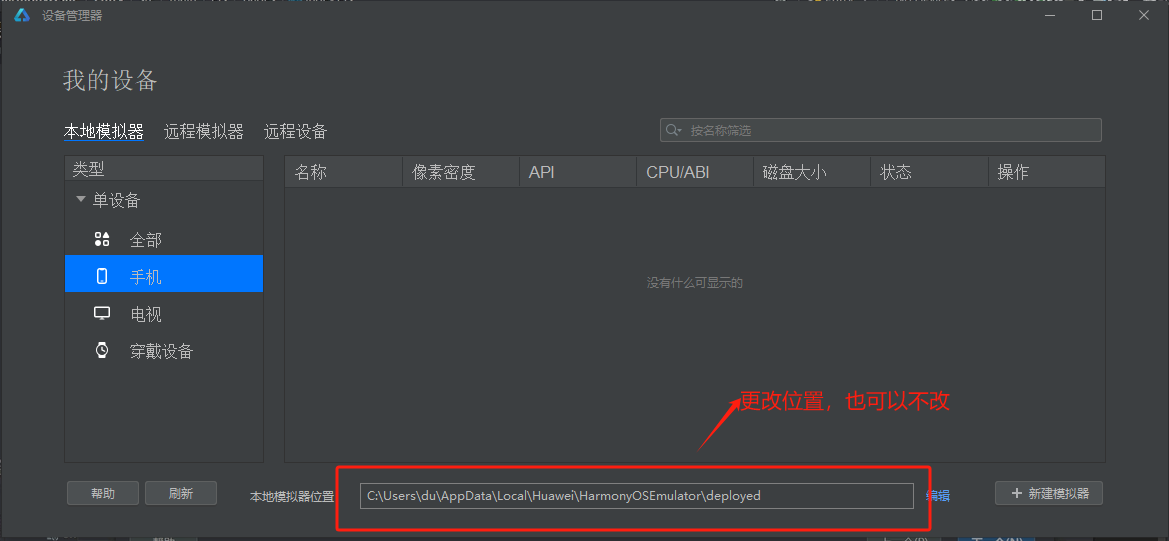
## 预览器

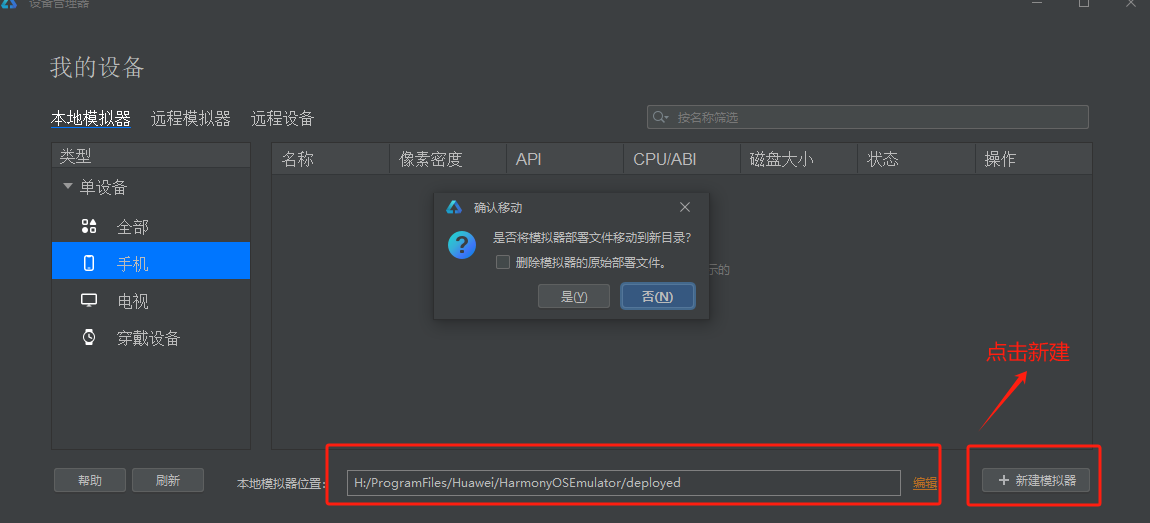
创建完项目后，打开页面，在右侧加载预览器，可以实时查看界面效果。预览器与显卡有很大关系，我在调试得时候，预览器显示不出来，提示无法连接，最后升级显卡驱动后正常。

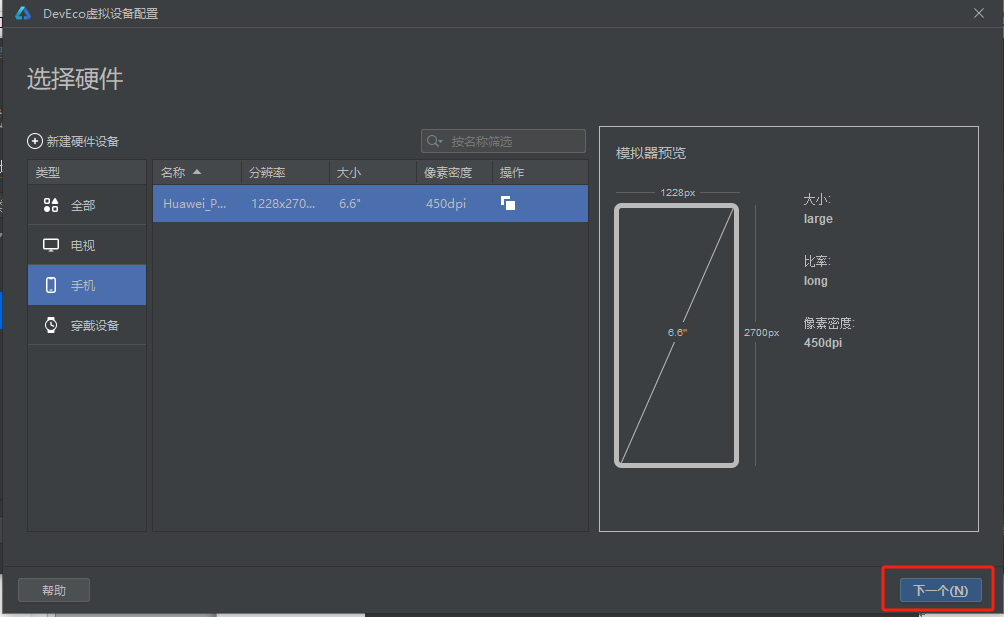


## 模拟器

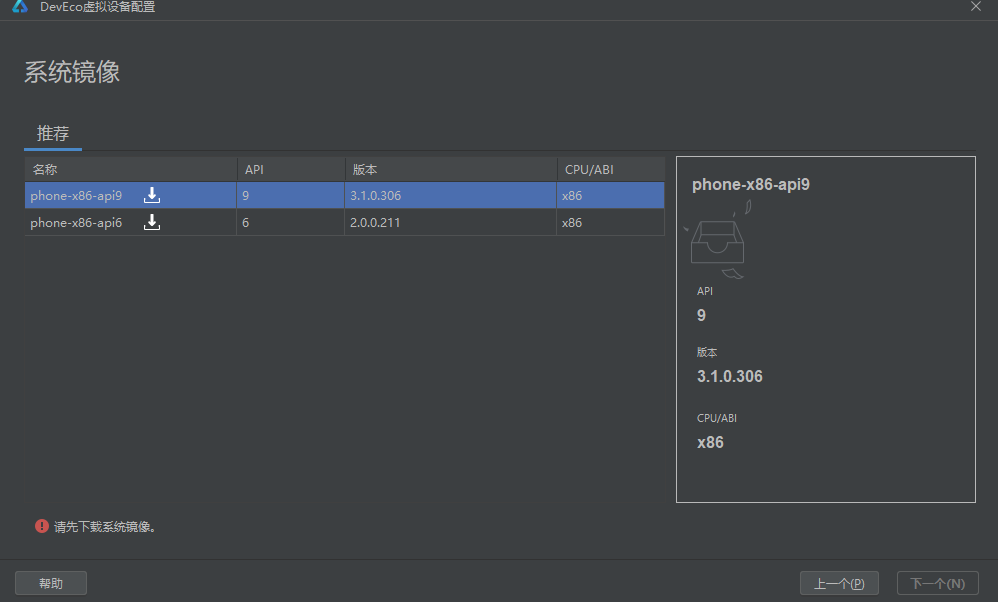


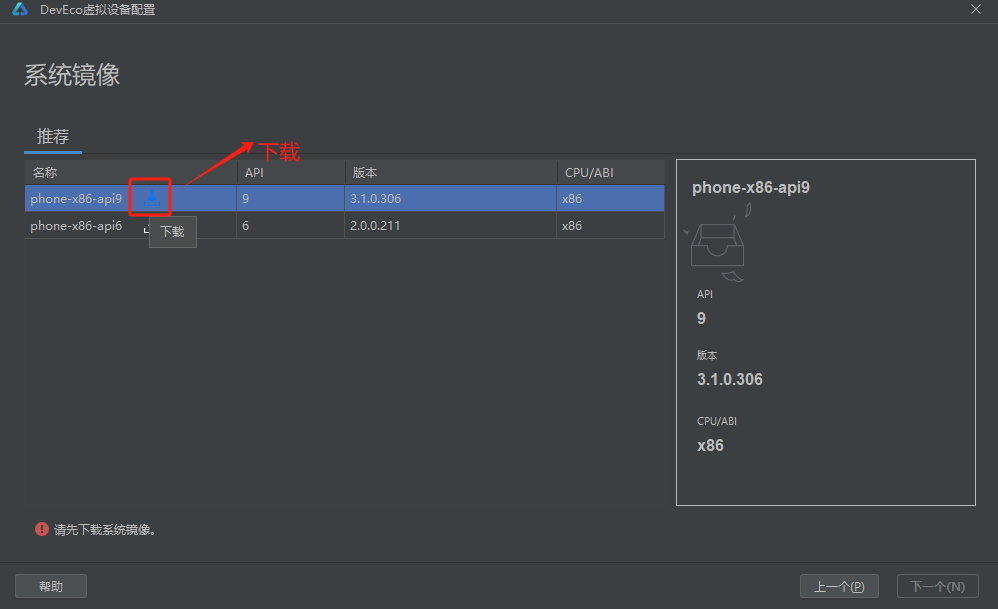






先点击下载图标下载



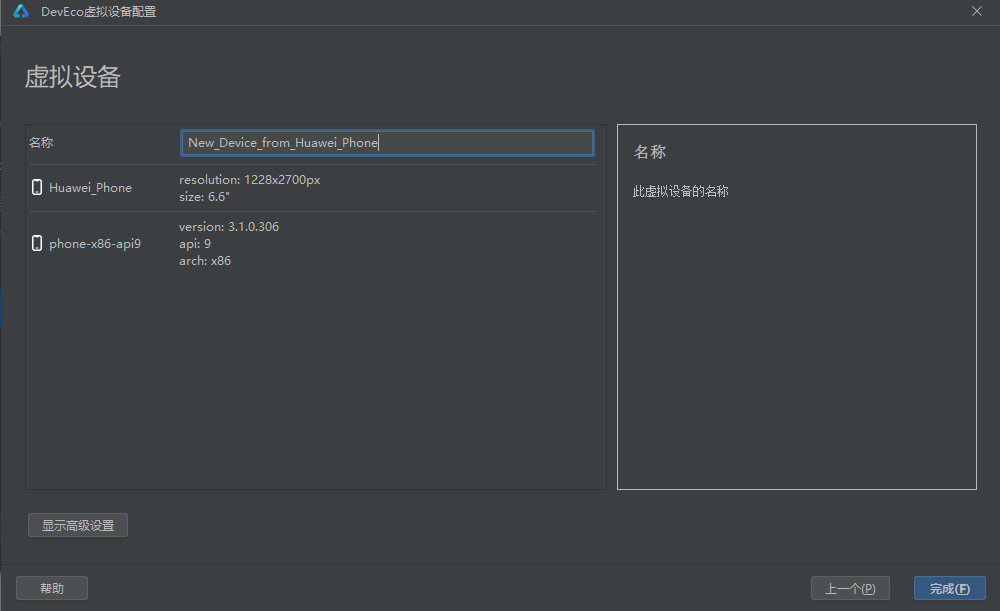


下载失败，这种情况是磁盘空间不足导致得，最少得10G的空间吧



完成





模拟器虚拟机创建完成

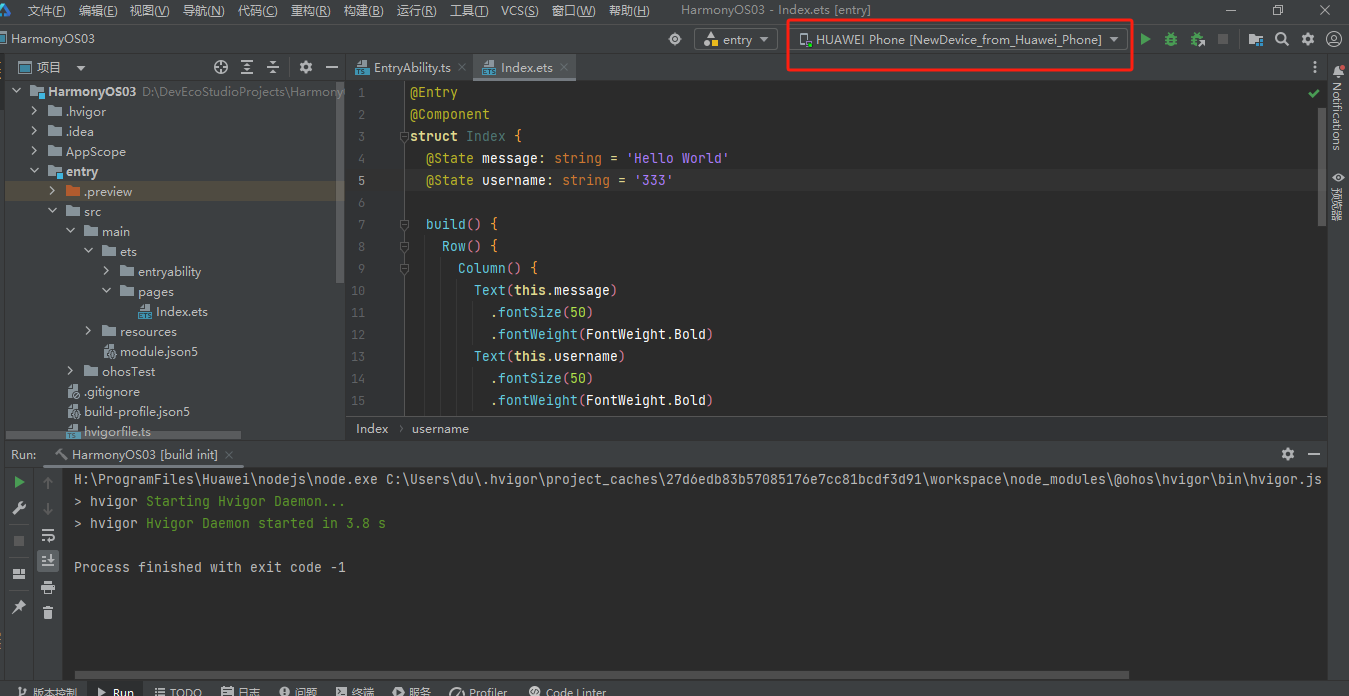


启动虚拟机



开启调试模式

虚拟模拟器启动以后，如果在设备中找不到我们启动得设备，则需要开启调试模式



开启步骤如下：

设置—系统个更新—开发人员选项 到开发人员选项后往下拉，有USB调试，把USB调试开关打开



# ArkTS详解

**官方文档：**[**https://developer.harmonyos.com/cn/develop/arkts/**](https://developer.harmonyos.com/cn/develop/arkts/)

<https://developer.harmonyos.com/cn/docs/documentation/doc-guides-V3/arkts-get-started-0000001504769321-V3>

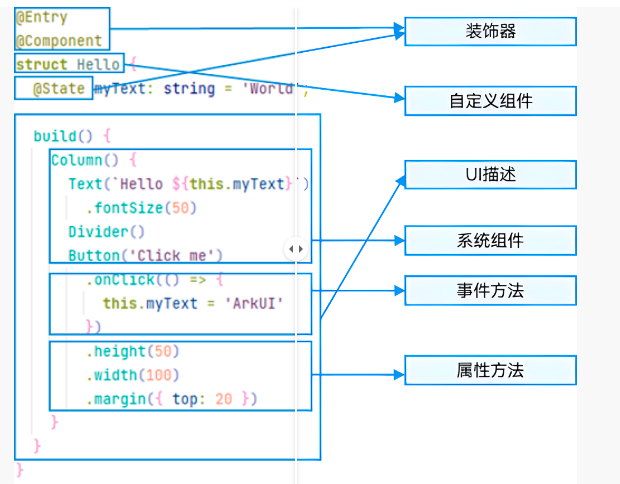
ArkTS是HarmonyOS优选的主力应用开发语言。ArkTS围绕应用开发在[TypeScript](https://www.typescriptlang.org/)（简称TS）生态基础上做了进一步扩展，继承了TS的所有特性，是TS的超集。因此，在学习ArkTS语言之前，建议开发者具备TS语言开发能力。

当前，ArkTS在TS的基础上主要扩展了如下能力：

* [基本语法](https://developer.harmonyos.com/cn/docs/documentation/doc-guides-V3/arkts-basic-syntax-overview-0000001531611153-V3)：ArkTS定义了声明式UI描述、自定义组件和动态扩展UI元素的能力，再配合ArkUI开发框架中的系统组件及其相关的事件方法、属性方法等共同构成了UI开发的主体。
* [状态管理](https://developer.harmonyos.com/cn/docs/documentation/doc-guides-V3/arkts-state-management-overview-0000001524537145-V3)：ArkTS提供了多维度的状态管理机制。在UI开发框架中，与UI相关联的数据可以在组件内使用，也可以在不同组件层级间传递，比如父子组件之间、爷孙组件之间，还可以在应用全局范围内传递或跨设备传递。另外，从数据的传递形式来看，可分为只读的单向传递和可变更的双向传递。开发者可以灵活的利用这些能力来实现数据和UI的联动。
* [渲染控制](https://developer.harmonyos.com/cn/docs/documentation/doc-guides-V3/arkts-rendering-control-overview-0000001543911149-V3)：ArkTS提供了渲染控制的能力。条件渲染可根据应用的不同状态，渲染对应状态下的UI内容。循环渲染可从数据源中迭代获取数据，并在每次迭代过程中创建相应的组件。数据懒加载从数据源中按需迭代数据，并在每次迭代过程中创建相应的组件。

未来，ArkTS会结合应用开发/运行的需求持续演进，逐步提供并行和并发能力增强、系统类型增强、分布式开发范式等更多特性。

## ArkTS基本组成



* 装饰器： 用于装饰类、结构、方法以及变量，并赋予其特殊的含义。如上述示例中@Entry、@Component和@State都是装饰器，[@Component](https://developer.harmonyos.com/cn/docs/documentation/doc-guides-V3/arkts-create-custom-components-0000001473537046-V3#section1430055924816)表示自定义组件，[@Entry](https://developer.harmonyos.com/cn/docs/documentation/doc-guides-V3/arkts-create-custom-components-0000001473537046-V3#section1430055924816)表示该自定义组件为入口组件，[@State](https://developer.harmonyos.com/cn/docs/documentation/doc-guides-V3/arkts-state-0000001474017162-V3)表示组件中的状态变量，状态变量变化会触发UI刷新。
* [UI描述](https://developer.harmonyos.com/cn/docs/documentation/doc-guides-V3/arkts-declarative-ui-description-0000001524416537-V3)：以声明式的方式来描述UI的结构，例如build()方法中的代码块。
* [自定义组件](https://developer.harmonyos.com/cn/docs/documentation/doc-guides-V3/arkts-create-custom-components-0000001473537046-V3)：可复用的UI单元，可组合其他组件，如上述被@Component装饰的struct Hello。
* 系统组件：ArkUI框架中默认内置的基础和容器组件，可直接被开发者调用，比如示例中的Column、Text、Divider、Button。
* 属性方法：组件可以通过链式调用配置多项属性，如fontSize()、width()、height()、backgroundColor()等。
* 事件方法：组件可以通过链式调用设置多个事件的响应逻辑，如跟随在Button后面的onClick()。
* 系统组件、属性方法、事件方法具体使用可参考[基于ArkTS的声明式开发范式](https://developer.harmonyos.com/cn/docs/documentation/doc-references-V3/ts-components-summary-0000001478181369-V3)。

除此之外，ArkTS扩展了多种语法范式来使开发更加便捷：

* [@Builder](https://developer.harmonyos.com/cn/docs/documentation/doc-guides-V3/arkts-builder-0000001524176981-V3)/[@BuilderParam](https://developer.harmonyos.com/cn/docs/documentation/doc-guides-V3/arkts-builderparam-0000001524416541-V3)：特殊的封装UI描述的方法，细粒度的封装和复用UI描述。
* [@Extend](https://developer.harmonyos.com/cn/docs/documentation/doc-guides-V3/arkts-extend-0000001473696678-V3)/[@Style](https://developer.harmonyos.com/cn/docs/documentation/doc-guides-V3/arkts-style-0000001473856690-V3)：扩展内置组件和封装属性样式，更灵活地组合内置组件。
* [stateStyles](https://developer.harmonyos.com/cn/docs/documentation/doc-guides-V3/arkts-statestyles-0000001482592098-V3)：多态样式，可以依据组件的内部状态的不同，设置不同样式。