HUAWEI

应用开发入门

华为云 一切皆服务

前言

- 本章为HarmonyOS应用开发的入门课程，主要介绍HarmonyOS应用开发的学习路线，简述包括DevEco Studio集成开发环境、代码结构、ArkTS语言开发入门等知识，为后续课程的展开打下基础，帮助各位开发者更好的上手应用开发实操。

课程目标

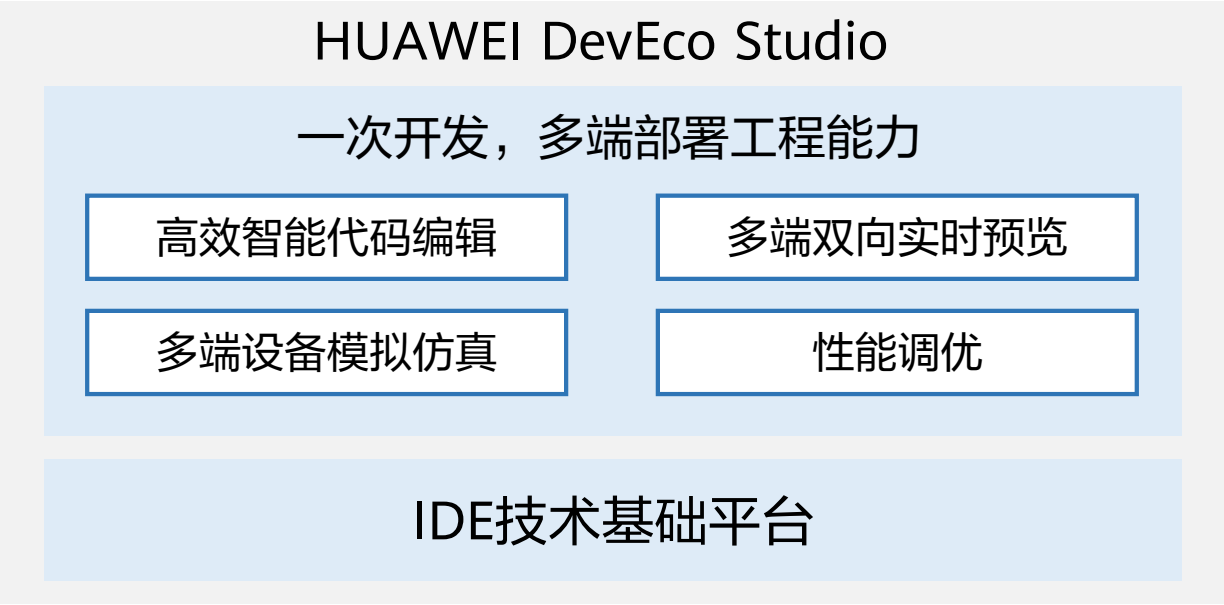
- 学完本课程后，您将能够：
  - 安装DevEco Studio工具，搭建课程中所需要的开发环境
  - 描述DevEco Studio的相关功能
  - 使用ArkTS语言进行应用开发
  - 了解Stage模型应用包结构

目录

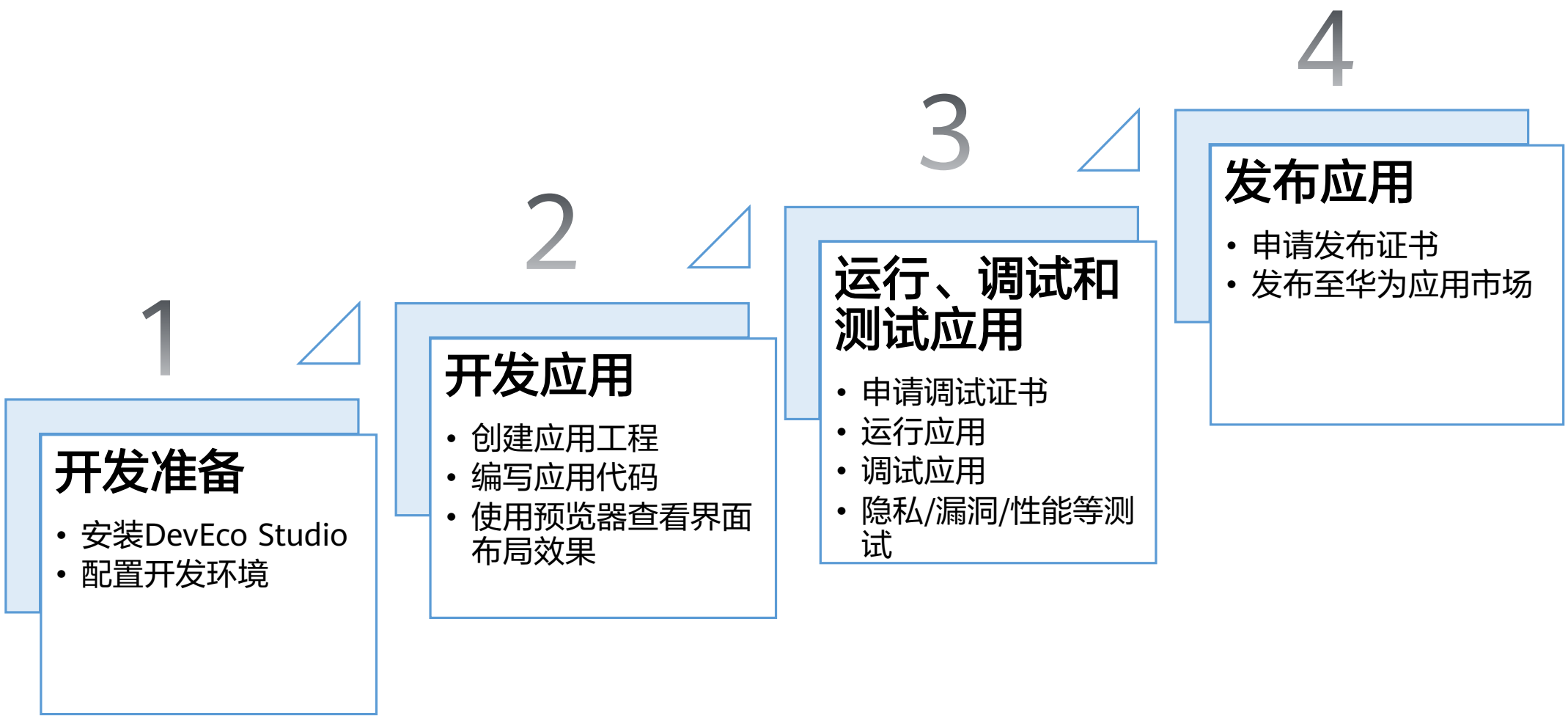
HUAWEI

HUAWEI DevEco Studio简介

- HUAWEI DevEco Studio（以下简称DevEco Studio）是基于InteliJ IDEA Community开源版本打造，为运行在HarmonyOS系统上的应用和服务（以下简称应用/服务）提供一站式的开发平台。作为一款开发工具，除了具有基本的代码开发、编译构建及调测等功能外，DevEco Studio还具有如下特点：



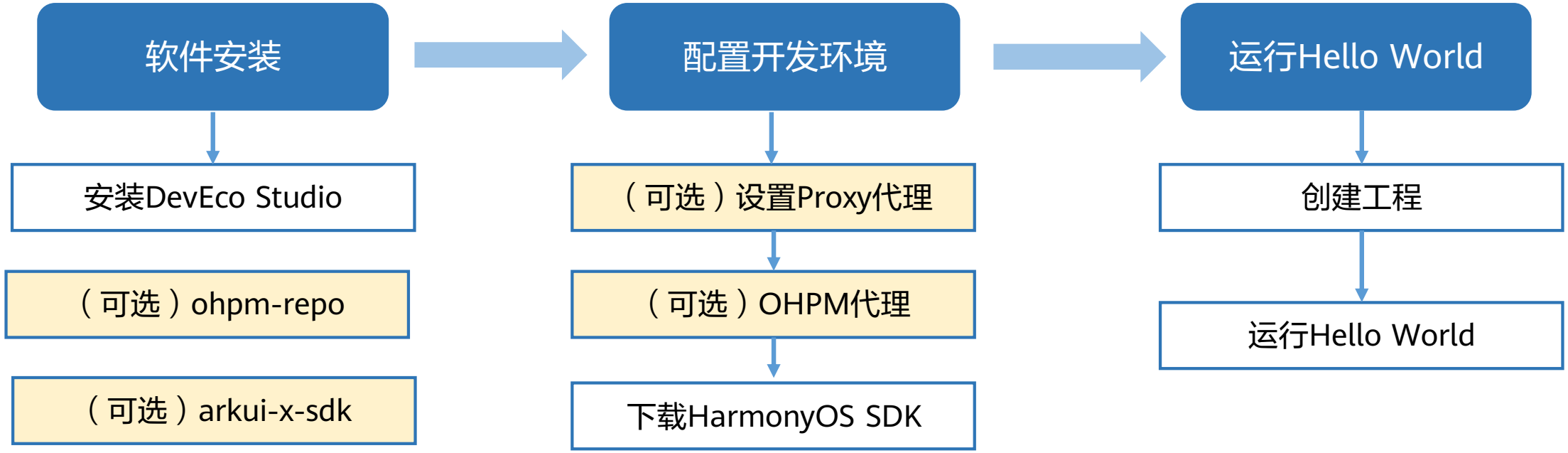
HarmonyOS应用/服务开发流程



HUAWEI

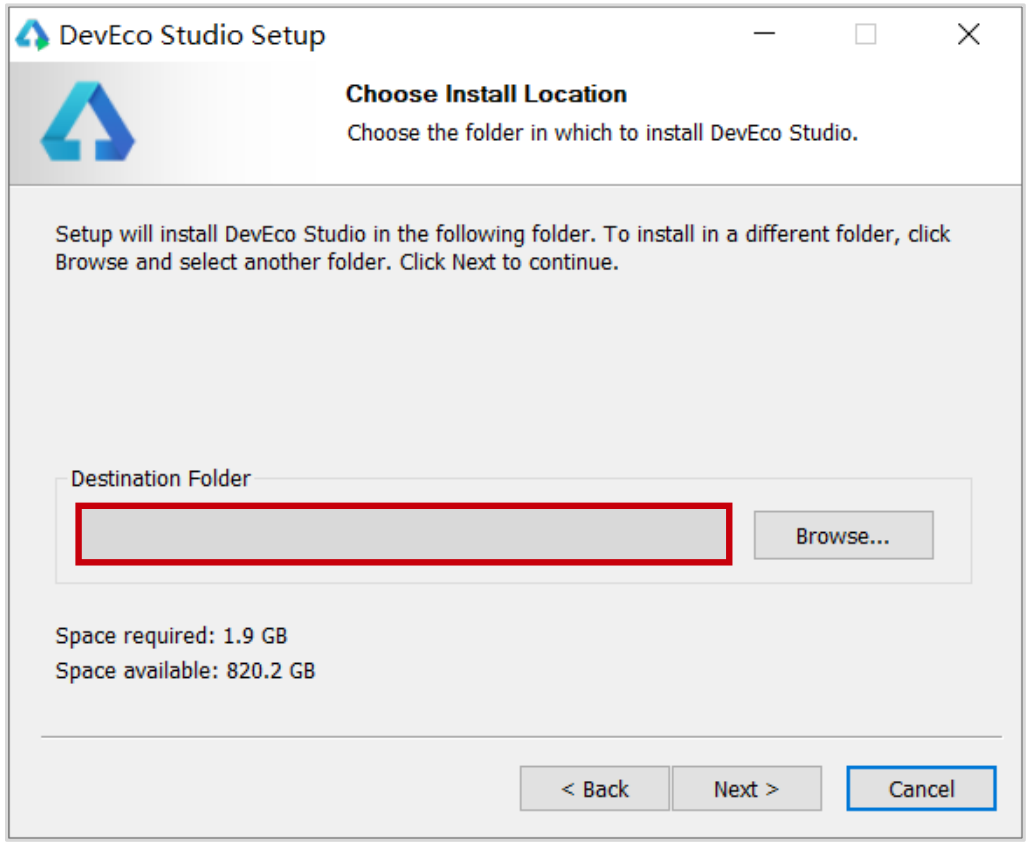
搭建应用开发环境

- 开始开发HarmonyOS应用前，应首先搭建开发环境。HarmonyOS应用开发环境搭建主要包含以下步骤：



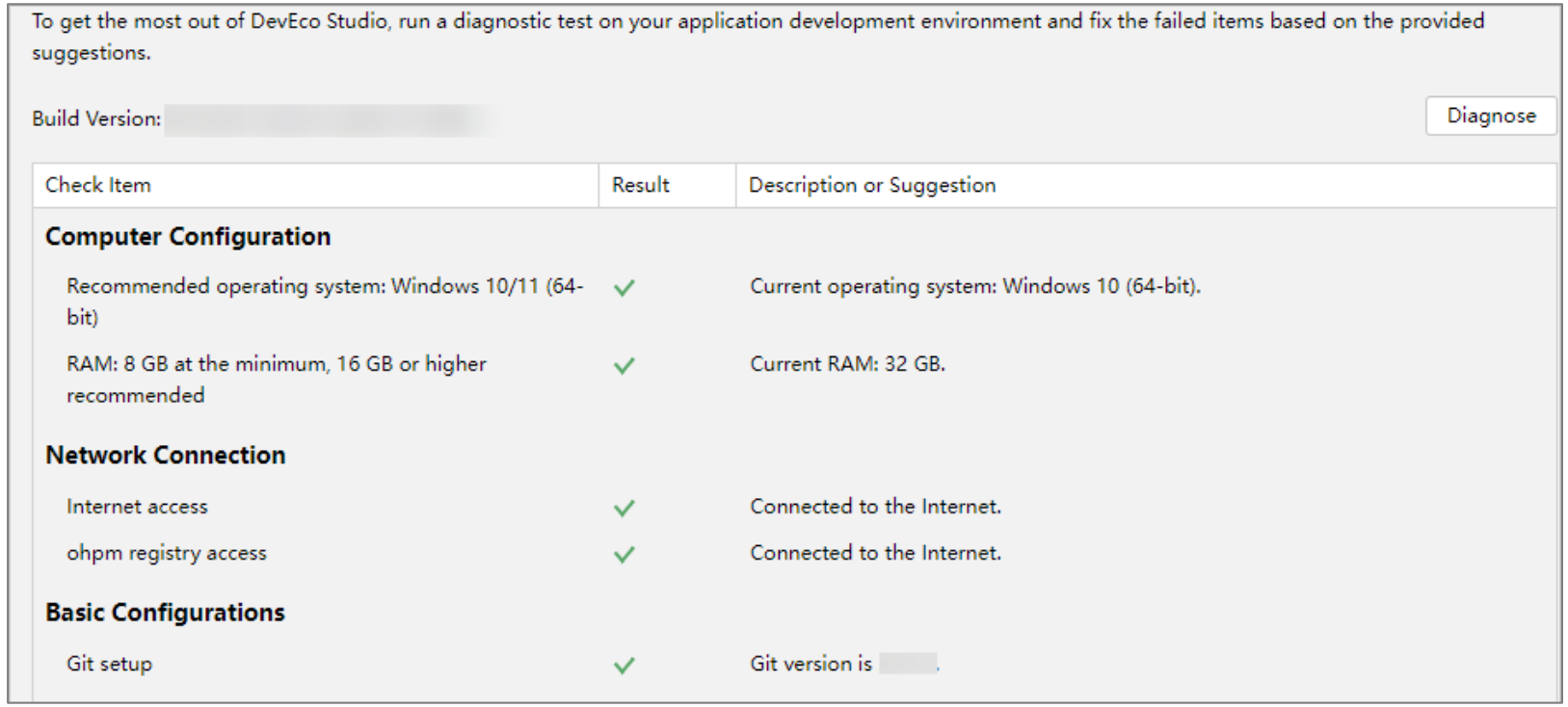
下载与安装软件

- DevEco Studio支持Windows系统和macOS系统进行下载安装；
- 下载时，建议DevEco Studio以及SDK文件**安装路径不要包含中文或空格。**



诊断开发环境

- 为了您开发应用/服务的良好体验，DevEco Studio提供了开发环境诊断的功能，帮助您识别开发环境是否完备。您可以在欢迎界面单击Help > Diagnose Development Environment进行诊断。

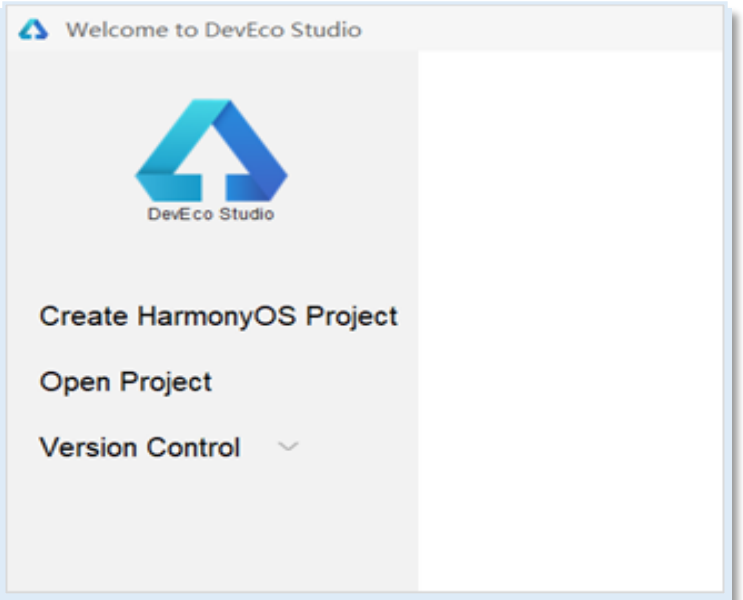


目录

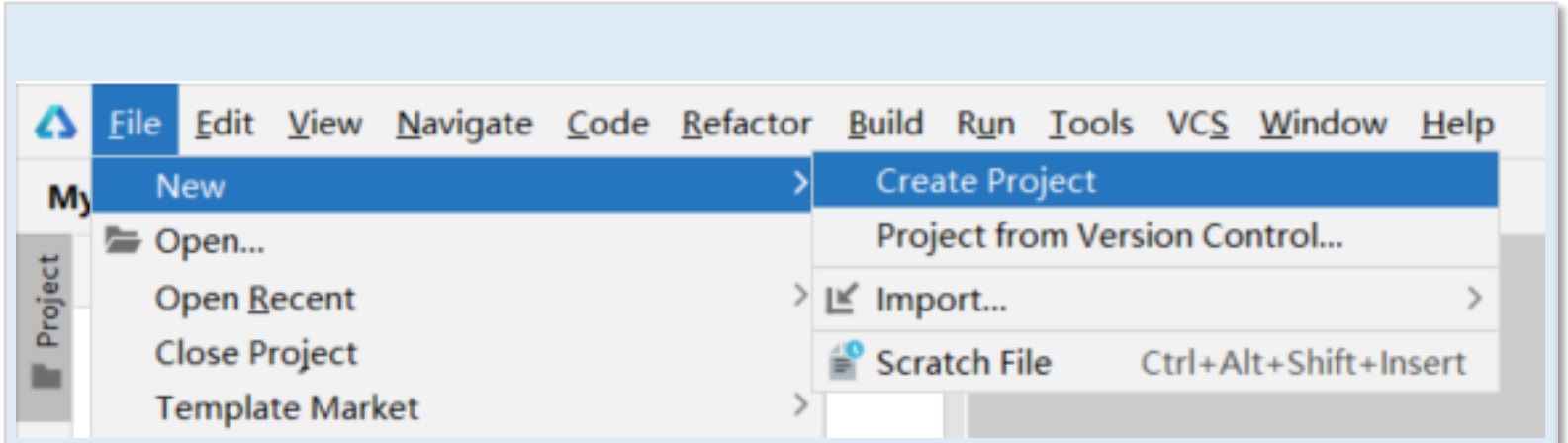
HUAWEI

打开DevEco Studio

- 双击桌面图标打开开发工具，有两种情况：



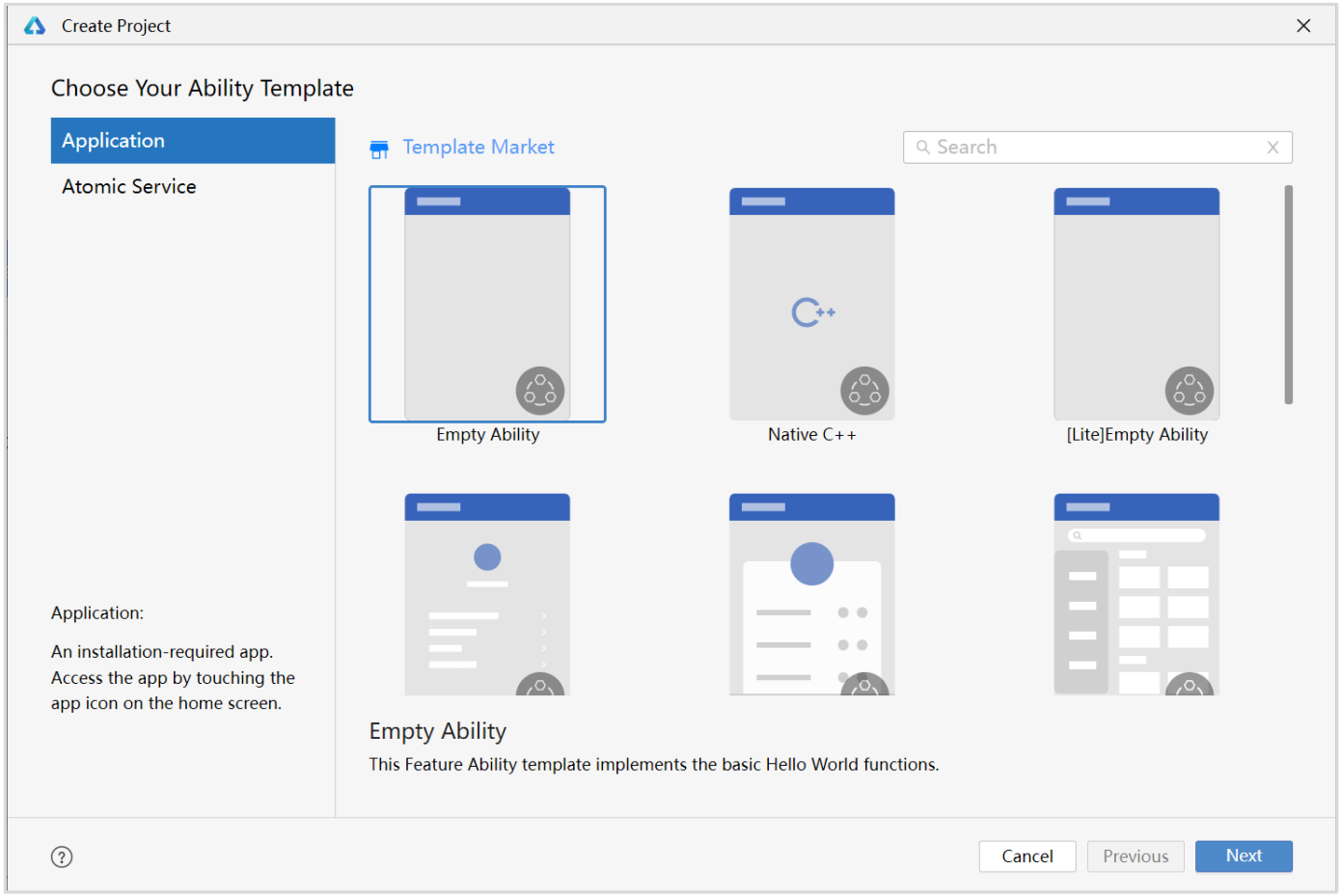
第一次打开，会出现这样的页面向导，点击Create HarmonyOS Project，创建新工程。



如已经建立了HarmonyOS工程，想要创建新的工程，请点击菜单栏选择File>New>Create Project 来创建一个新工程。

选择工程模板

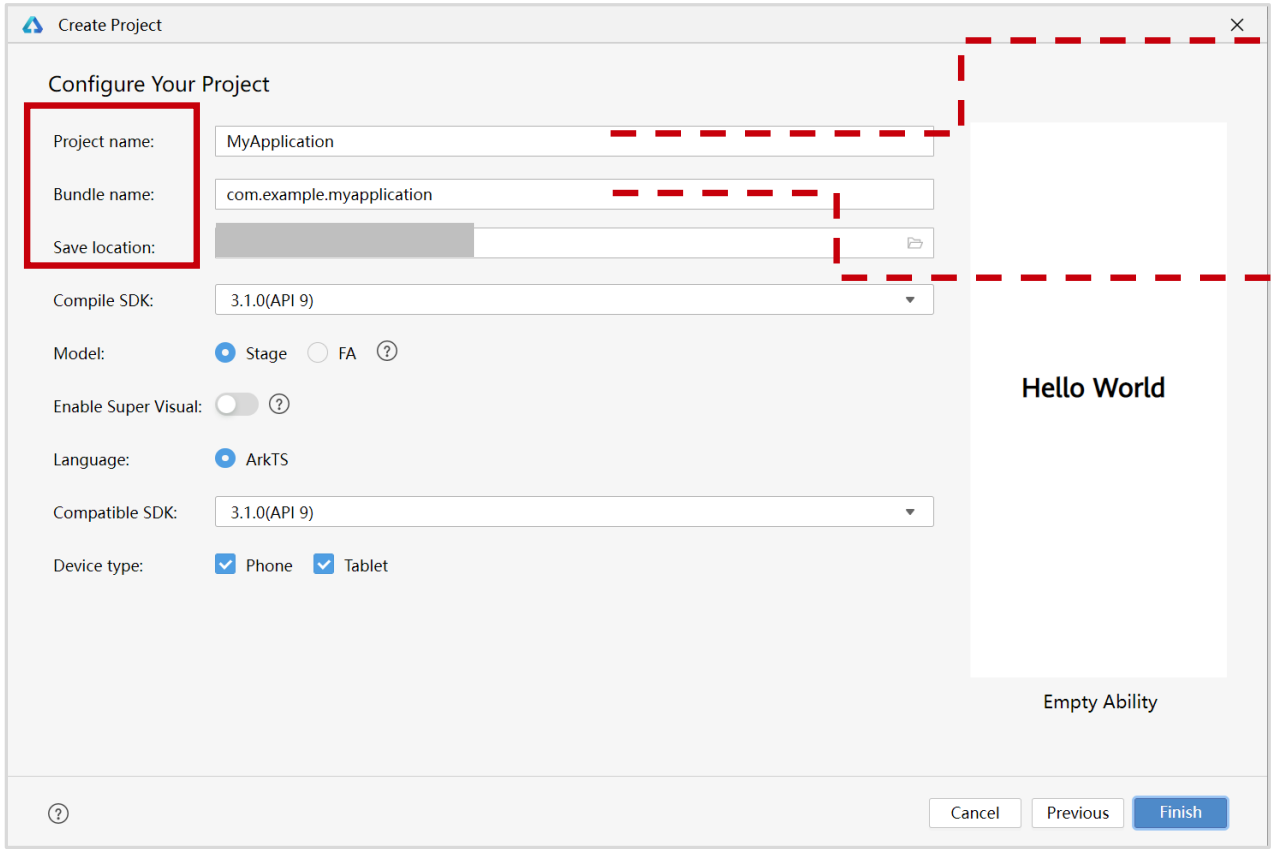
- 选择HarmonyOS模板库，选择模板“Empty Ability”，点击Next进行下一步配置。



HUAWEI

配置工程

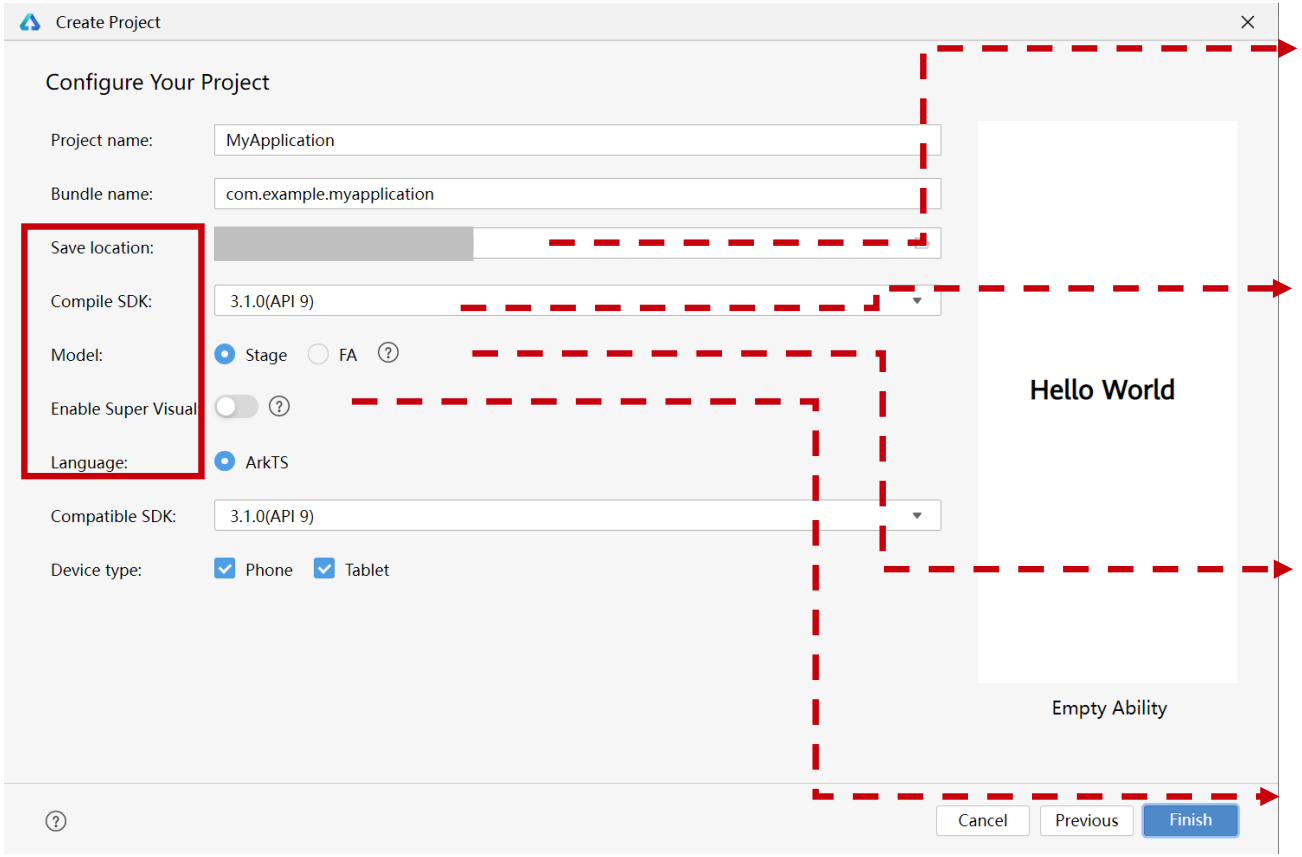
- 进入配置工程界面，选择参数，点击Finish，工具会自动生成示例代码和相关资源，等待工程创建完成。



- **Project name**：工程的名称，可以自定义，由大小写字母、数字和下划线组成。
- **Bundle name**：软件包名称。

配置工程参数 (1)

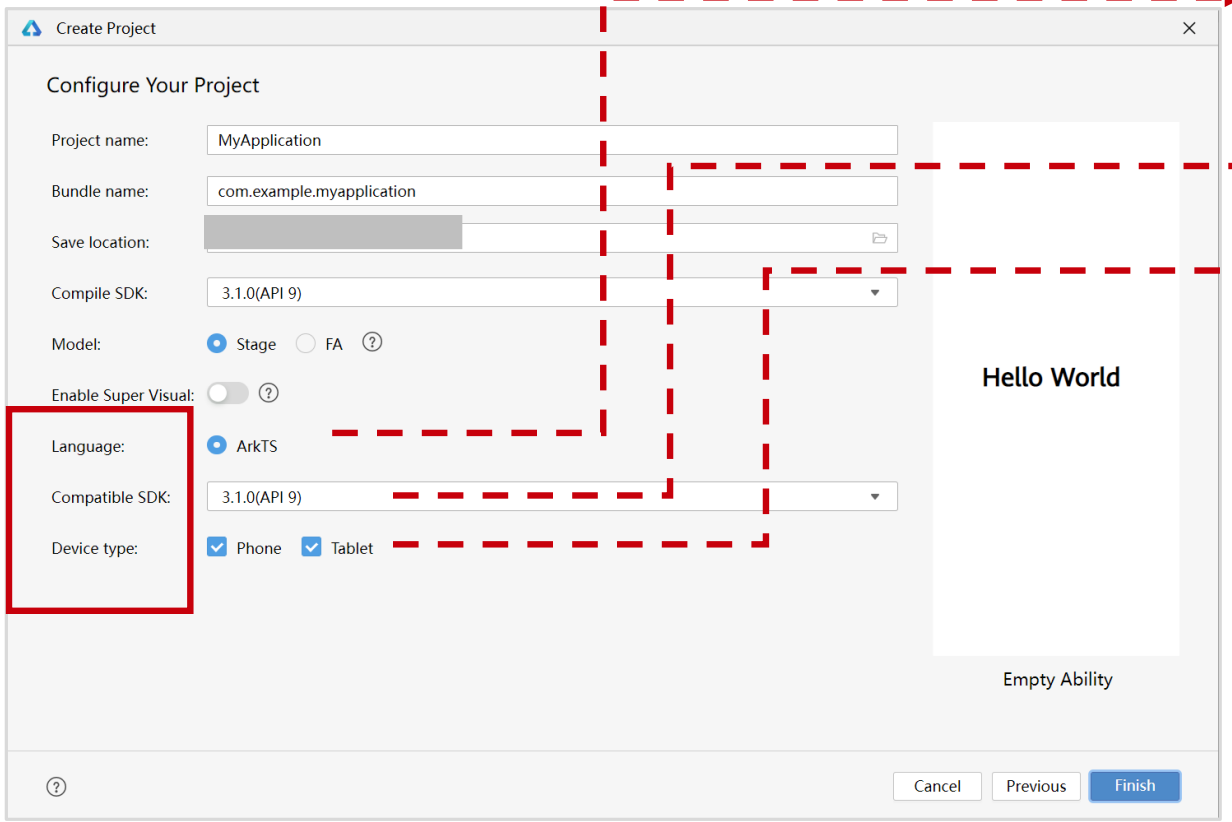
- 进入配置工程界面，选择参数，工具会自动生成示例代码和相关资源，等待工程创建完成。



- **Save location**：工程文件本地存储路径，不能包含中文字符。
- **Compile SDK**：应用/服务的目标API Version，在编译构建时，DevEco Studio会根据指定的Compile API版本进行编译打包。
- **Model**：应用模型，HarmonyOS先后提供了两种应用模型，分为Stage模型与FA模型。API 9支持Stage模型；API Version 4~8只支持FA模型。
- **Enable Super Visual**：支持低代码开发模式。

配置工程参数 (2)

- 进入配置工程界面，选择参数，点击Finish，工具会自动生成示例代码和相关资源，等待工程创建完成。

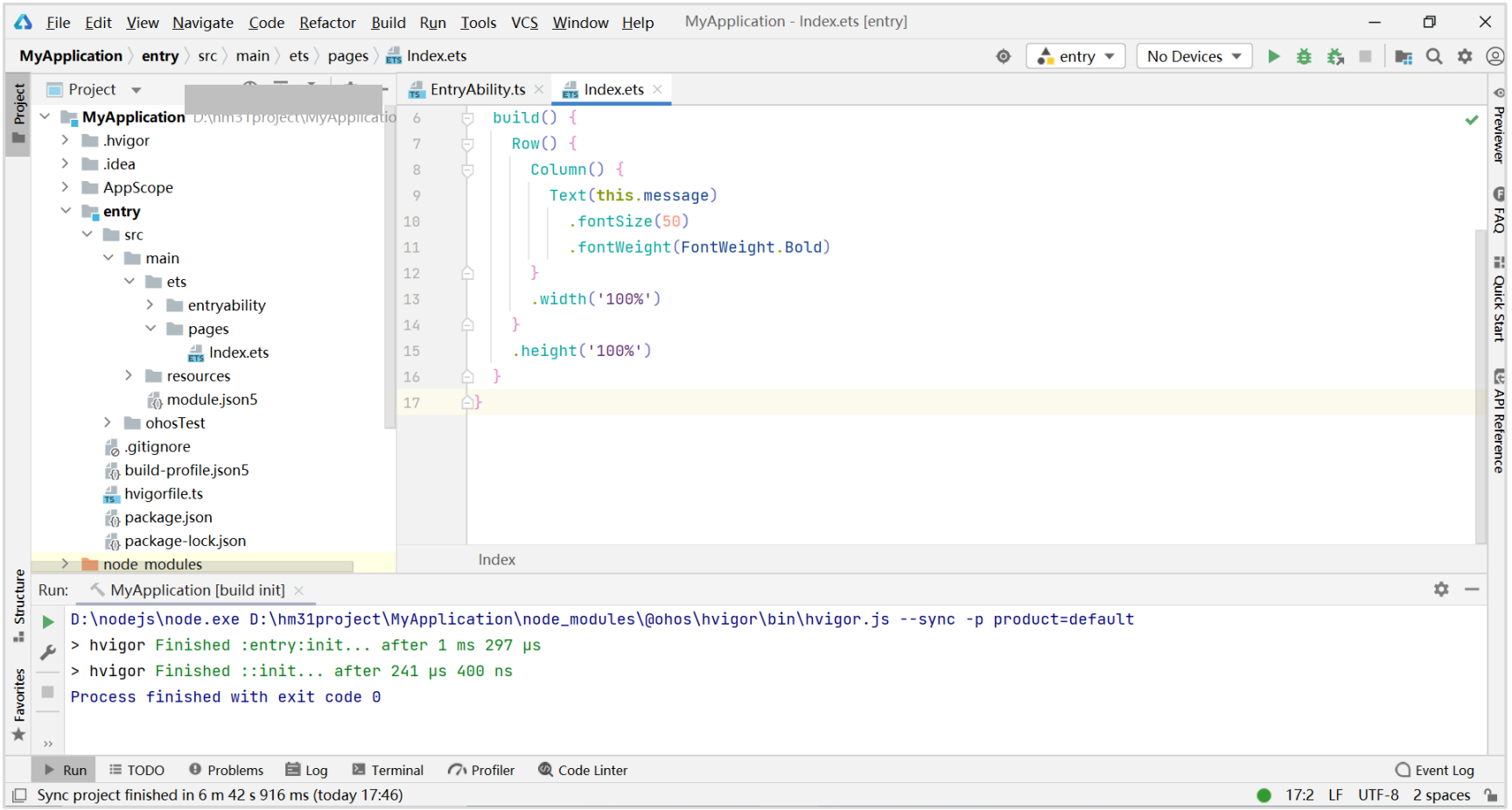


- **Language**：开发语言。（API为9时，FA模型下支持JS语言）
- **Compatible SDK**：兼容的最低API Version。
- **Device type**：该工程模板支持的设备类型，支持多选，默认全部勾选，如果勾选多个设备，表示该应用/服务支持部署在多个设备上。

这里我们选择创建一个API为9、开发语言为ArkTS且为stage模型的传统应用项目工程。

创建完成

- 通过以上步骤，我们就能创建好一个最基础的项目工程。



HUAWEI

目录

HUAWEI

调试与开发辅助工具

- 工欲善其事必先利其器，为了帮助开发者更好的进行应用开发。 DevEco Studio提供了如下调试工具：

01

预览器

预览器，用于查看UI效果

02

远程模拟器

调试运行应用

Phone　Tablet

Car　TV

Wearable

03

本地模拟器

创建运行在本地计算机上

04

远程真机

部署在云端的真机设备资源

HUAWEI

Previewer预览器

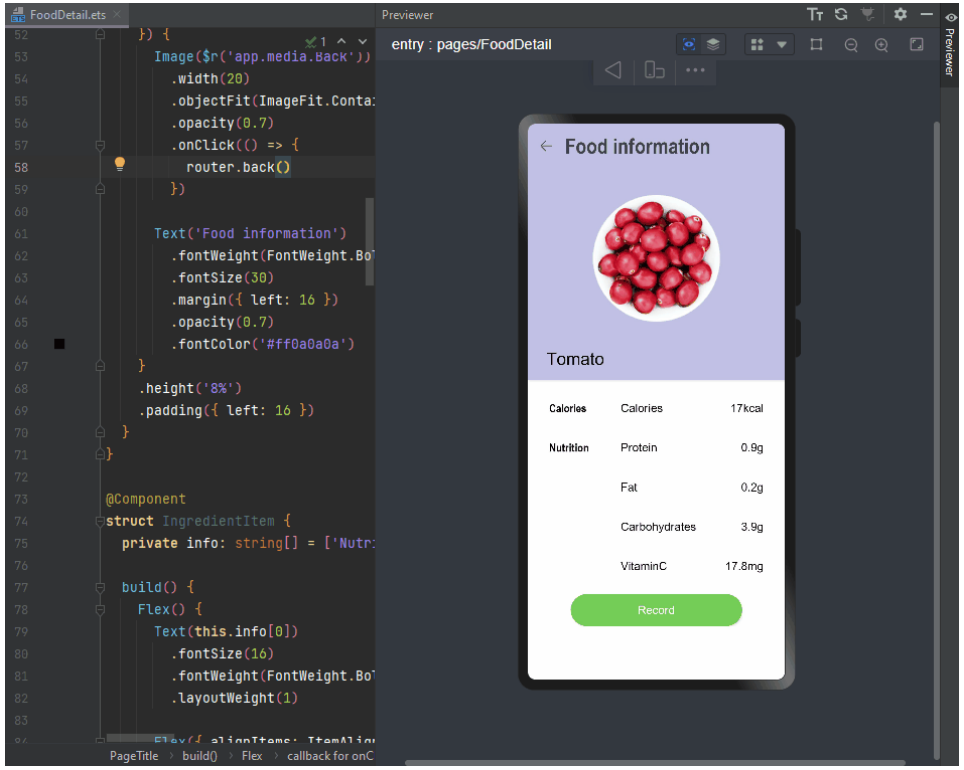
在应用开发过程中，DevEco Studio为开发者提供了预览器的功能，可以查看应用的UI界面效果，支持ArkTS、JS应用的预览。

支持**实时预览**，只需要将开发的源代码进行保存，就可以通过Previewer实时查看应用运行效果。

支持**动态预览**，在Previewer中预览时，可以操作应用的交互动作。

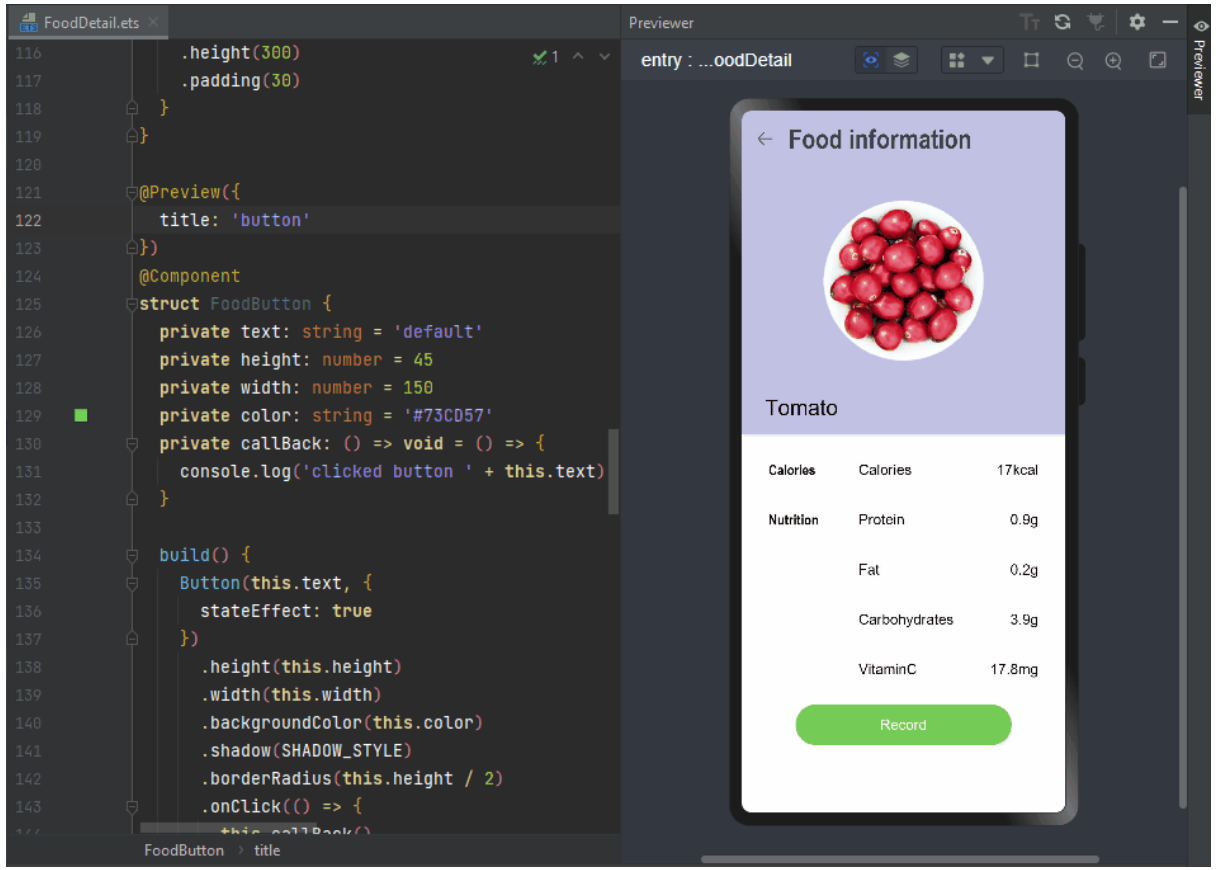
Previewer实时预览

- 实时预览：在开发界面UI代码过程中，对UI组件进行修改后，保存代码预览器就会立即刷新预览结果。如果修改了组件的属性，则会实时（亚秒级）刷新预览结果。实时预览默认开启，如果不需要实时预览，请单击预览器右上角 按钮，关闭实时预览功能。



Previewer动态预览

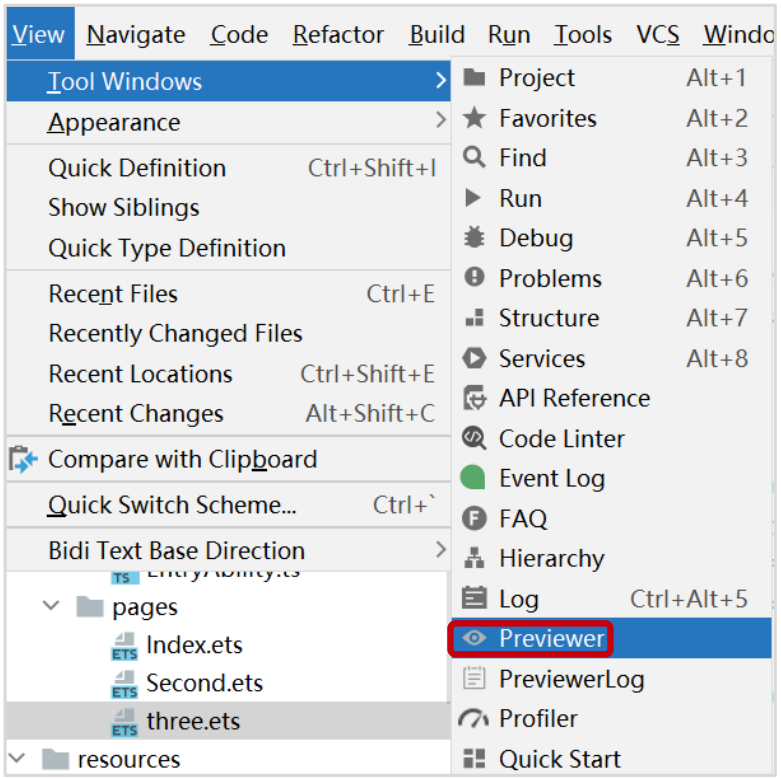
- 在预览器界面，可以在预览器中操作应用/服务的界面交互动作，如单击、跳转、滑动等，与应用/服务运行在真机设备上的界面交互体验一致。



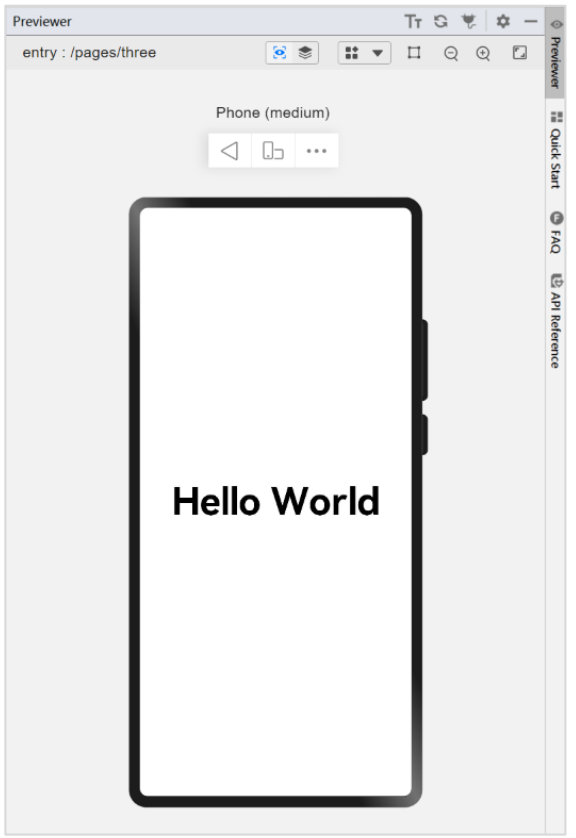
使用ArkTS预览器的方法

- 在创建好的工程目录中，打开任意一个.ets文件，通过如下任意一种方式打开预览器开关。
- 注意，如打开其他文件将无法开启预览器，且会出现报错提示。

1



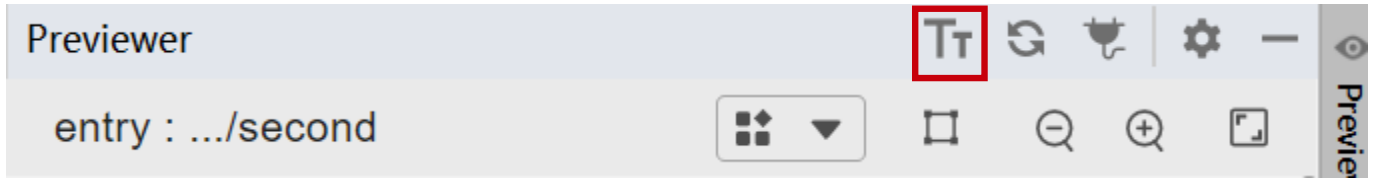
2



HUAWEI

Inspector双向预览

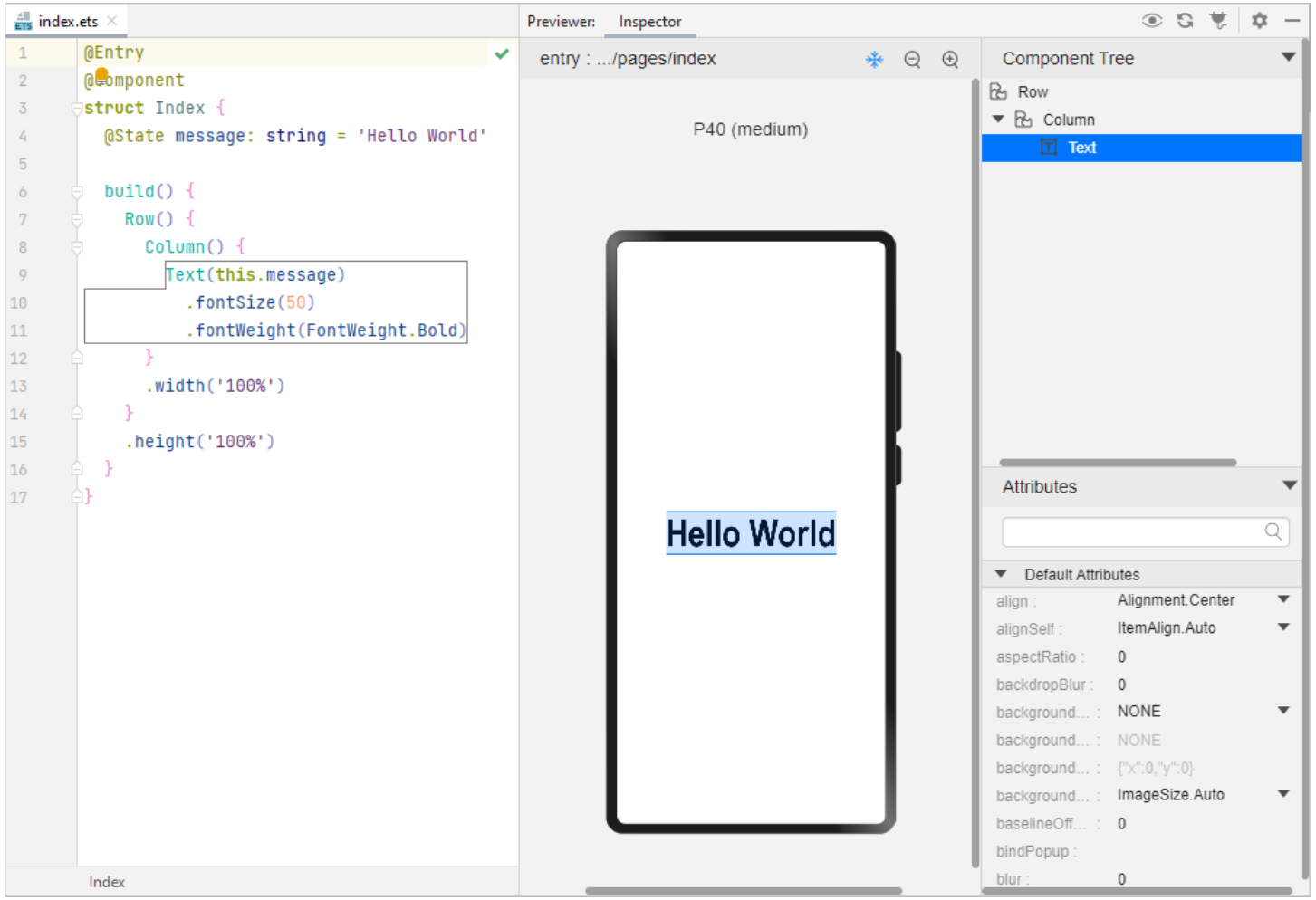
- DevEco Studio提供HarmonyOS应用/服务的UI预览界面与源代码文件间的双向预览功能，支持ets文件、hml文件预览器界面的双向预览；



- 在预览界面还可以通过组件的属性面板修改可修改的属性或样式，在预览界面修改后，预览器会自动同步到代码编辑器中修改源码，并实时的刷新UI界面；
- 如果在代码编辑器中修改源码，也会实时刷新UI界面，并更新组件树信息及组件属性。

开启Inspector双向预览

- 开启双向预览功能后，支持**代码编辑器**、**UI界面**和**Component Tree组件树**三者之间的联动：
  - 选中三者中任何一个，其他两者都会联动显示。例如：选中预览器UI界面中的组件，则组件树上对应的组件将被选中，同时对应的代码块高亮显示。



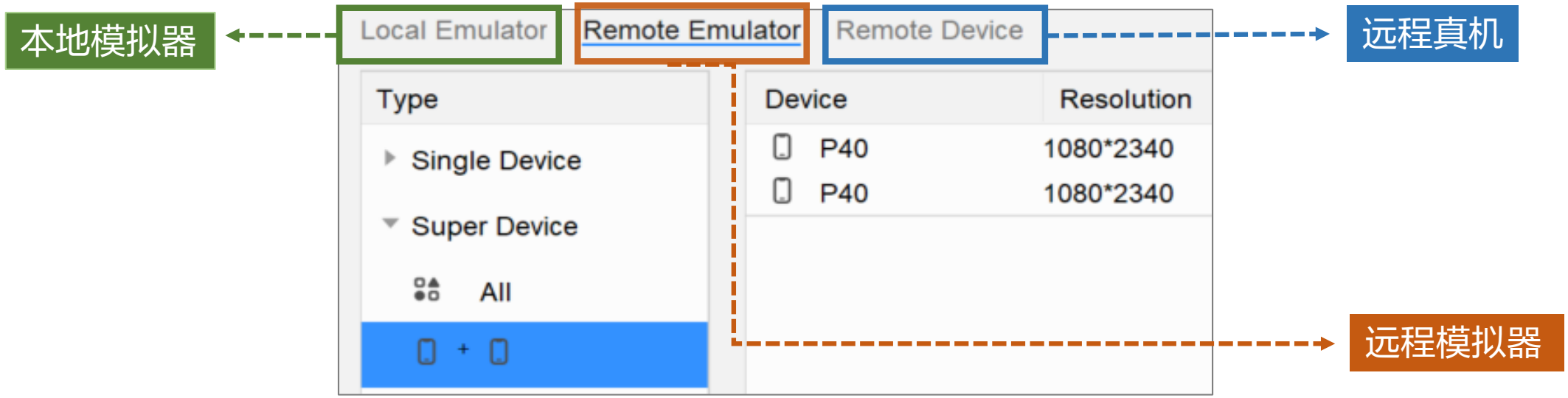
HUAWEI

目录

HUAWEI

远程模拟器： Remote Emulator

- DevEco Studio提供的Remote Emulator是远程模拟器，可以运行和调试Phone、Tablet、Car、TV 和 Wearable设备的应用。在Remote Emulator上运行应用兼容签名与不签名两种类型的HAP。
- Remote Emulator每次使用时长为**2小时**，到期前可申请延长使用时间。如果Remote Emulator到期释放后，可以重新申请资源。



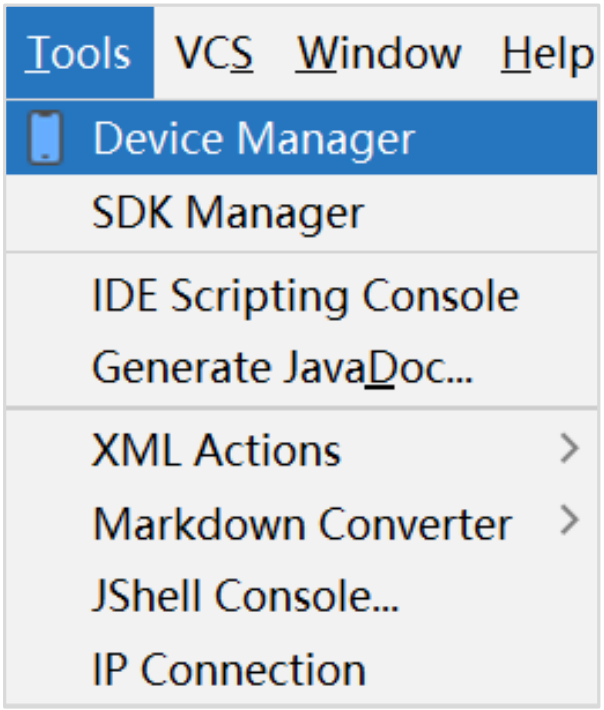
使用Remote Emulator运行应用

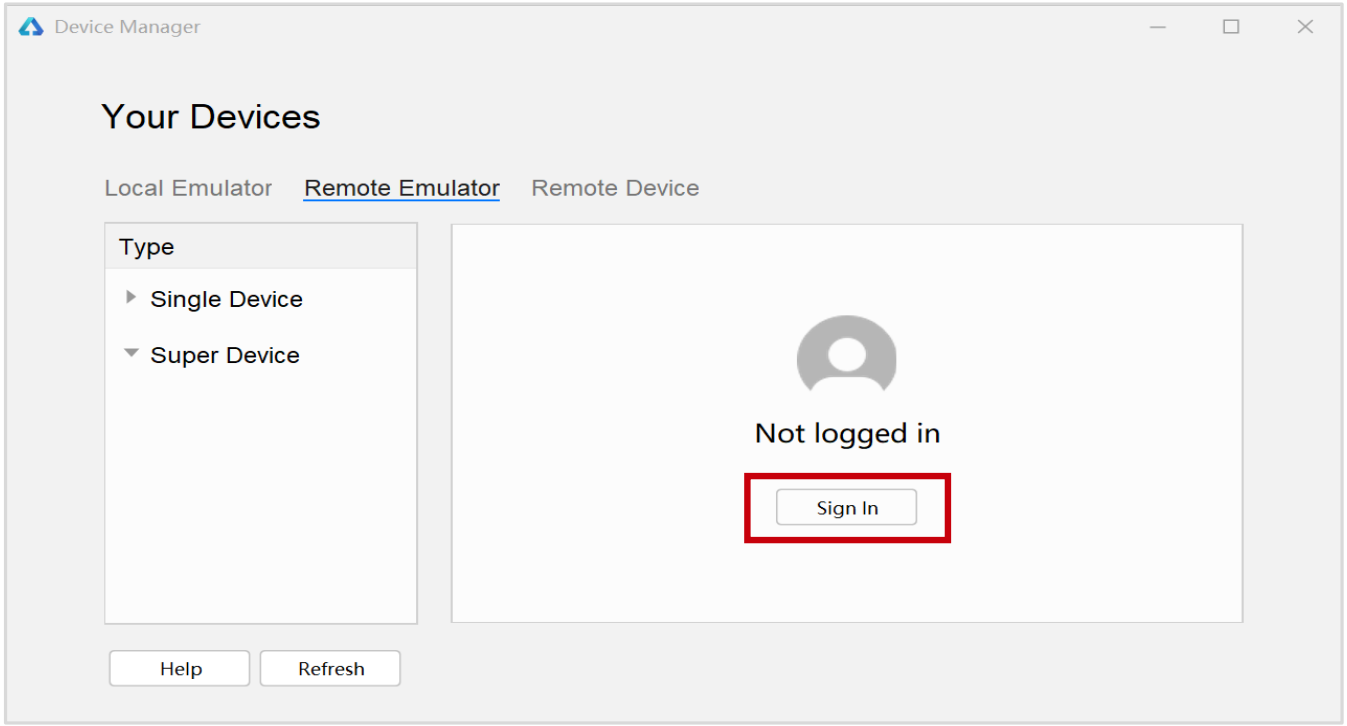
- 使用Remote Emulator运行应用存在以下三个步骤：

01	02	03
登录授权	**启动模拟器**	**运行程序**
• 启动Device Manager，并使用华为开发者联盟帐号进行登录和授权	• 在Device Manager设备列表中选择设备类型并运行	• 编译构建应用程序，并部署运行在远程模拟器上

单设备远程模拟器运行应用/服务

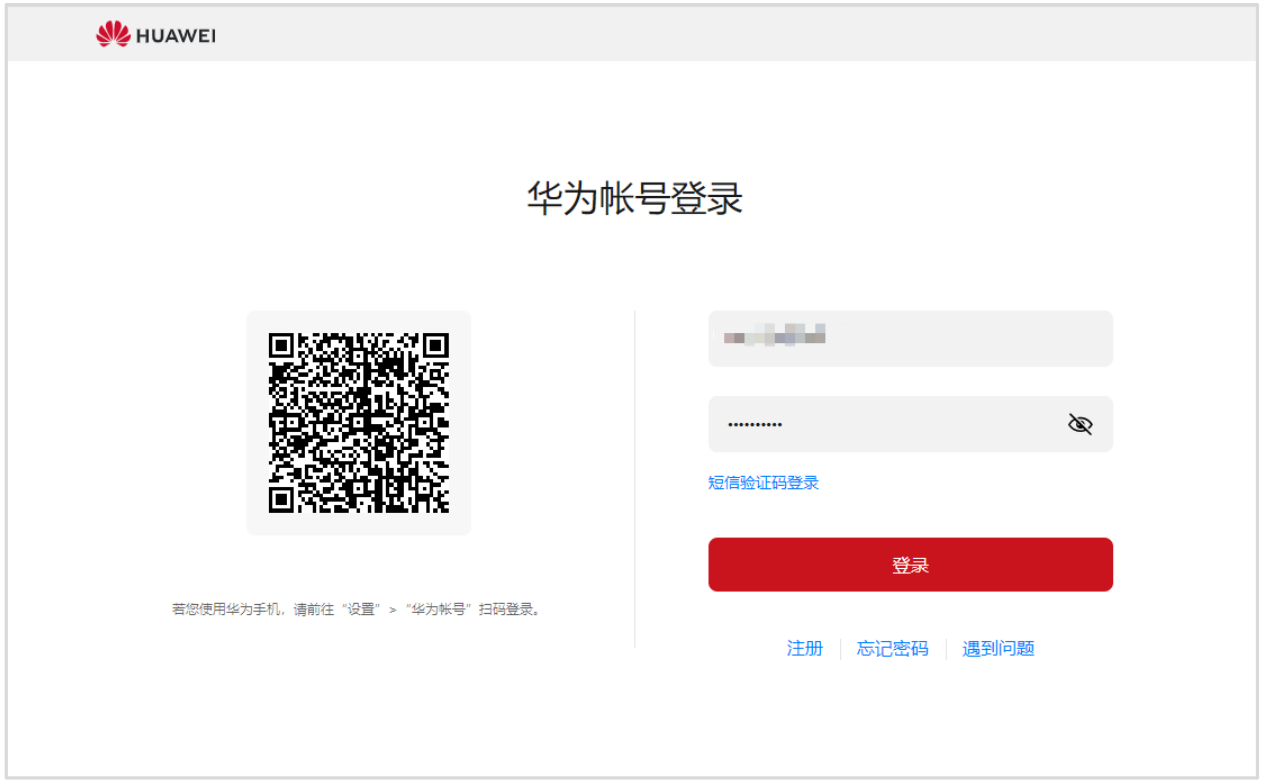
- 在DevEco Studio菜单栏，单击Tools > Device Manager。
- 在Remote Emulator页签中，单击Sign In，在浏览器中弹出华为开发者联盟帐号登录界面。

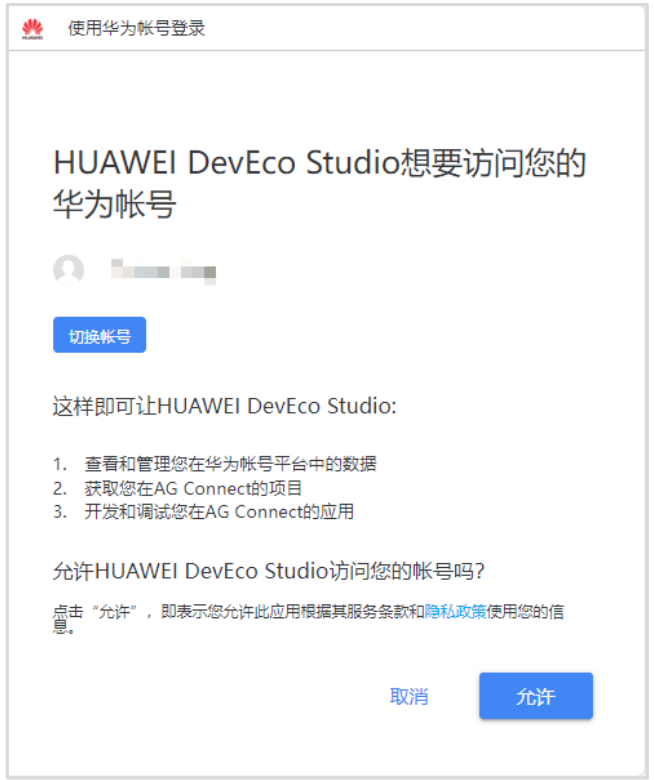




登录华为帐号

- 请输入已实名认证的华为开发者联盟帐号的用户名和密码进行登录。
- 登录后，请单击界面的允许按钮进行授权。

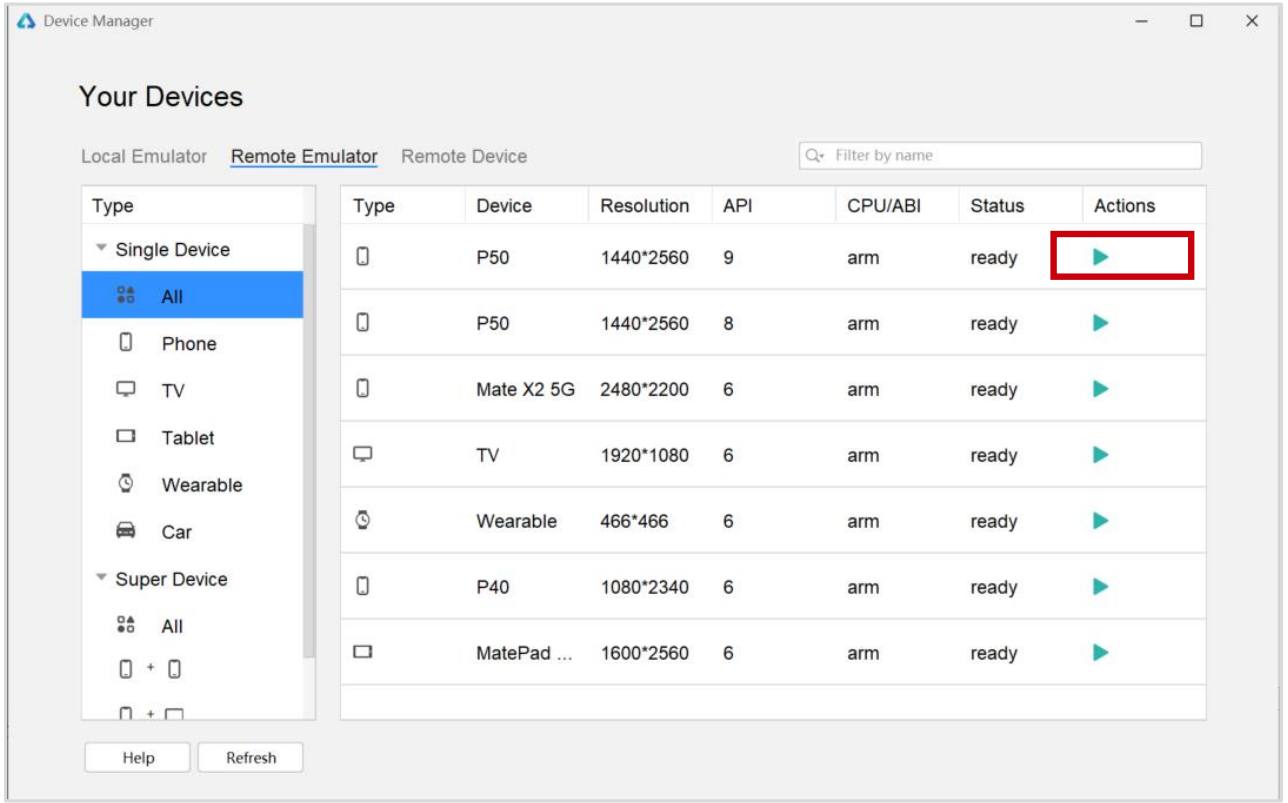




HUAWEI

拉起远程模拟器

- 授权成功页面如下左图，重新切换回Device Manager界面，会出现如下右图界面，在Single Device中选择设备单击 ▶ 拉起远程模拟器（同一时间只能启动一个设备）。



应用/服务运行在单设备远程模拟器

- 单击DevEco Studio的Run > Run“模块名称”或单击 启动按钮。DevEco Studio会启动应用/服务的编译构建，完成后应用/服务即可运行在Remote Emulator上。

HUAWEI

目录

HUAWEI

本地模拟器与远程模拟器的区别

Local Emulator

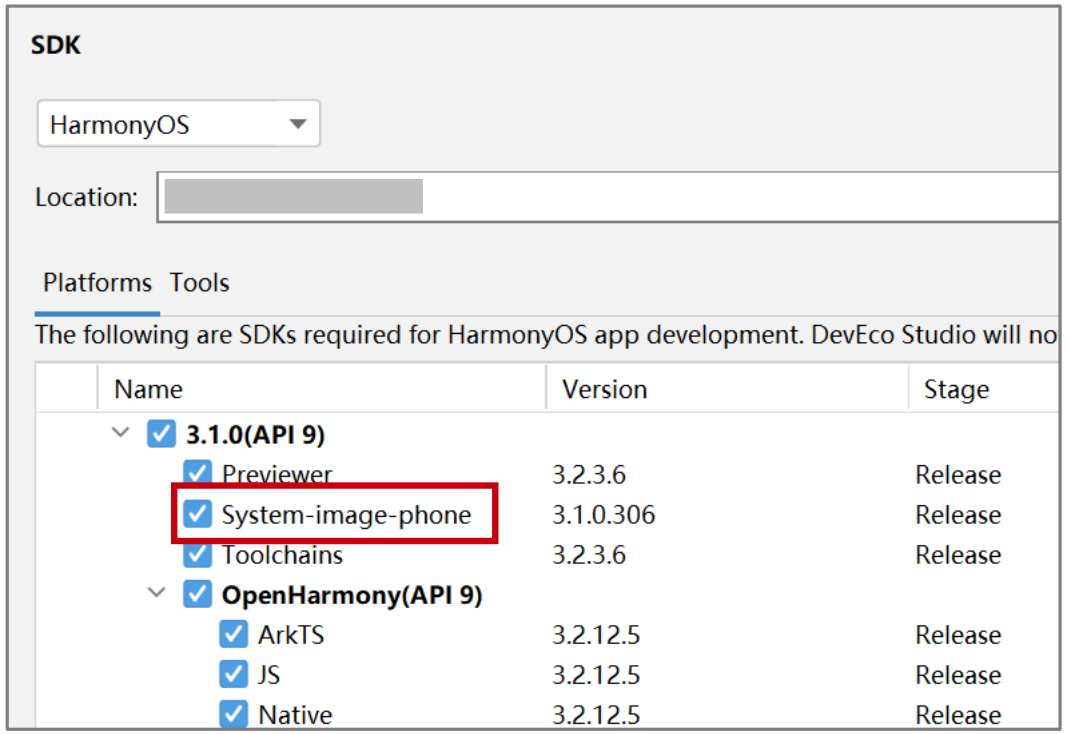
- 创建和运行在本地计算机上
- 不需要登录授权
- 在运行和调试应用/服务时，由于没有网络数据的交换，因此可以保持很好的流畅性和稳定性
- 需要耗费一定的计算机磁盘资源

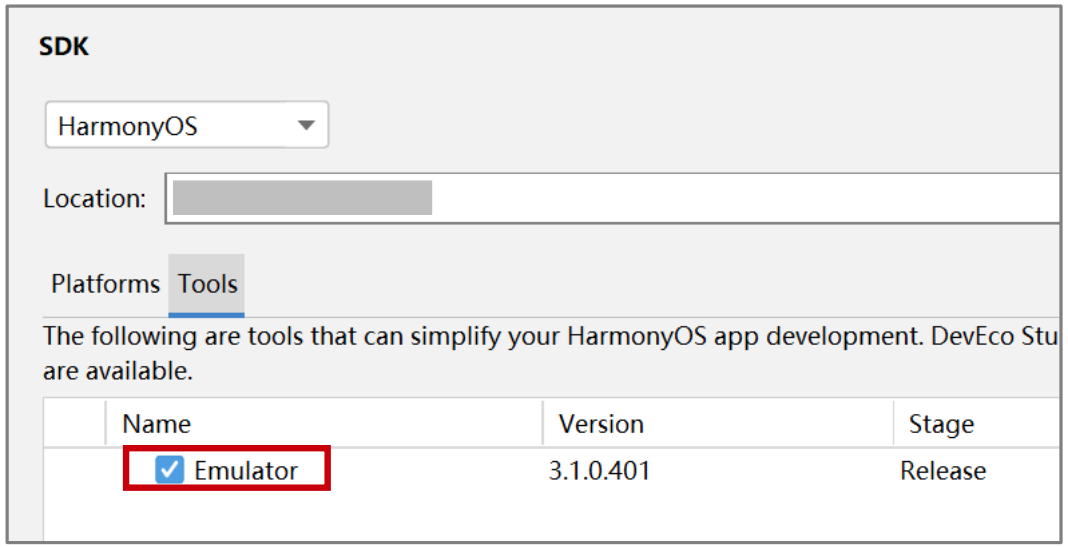
Remote Emulator

- 创建和运行在远程服务器上
- 需要登录授权
- 在运行和调试应用/服务时，需要网络数据的交换
- 不需要耗费计算机磁盘资源

资源准备

- 单击File > Settings > SDK（macOS系统为DevEco Studio > Preferences > SDK），下拉框选择HarmonyOS，勾选并下载Platforms下的System-image-phone和Tools下的Emulator资源。





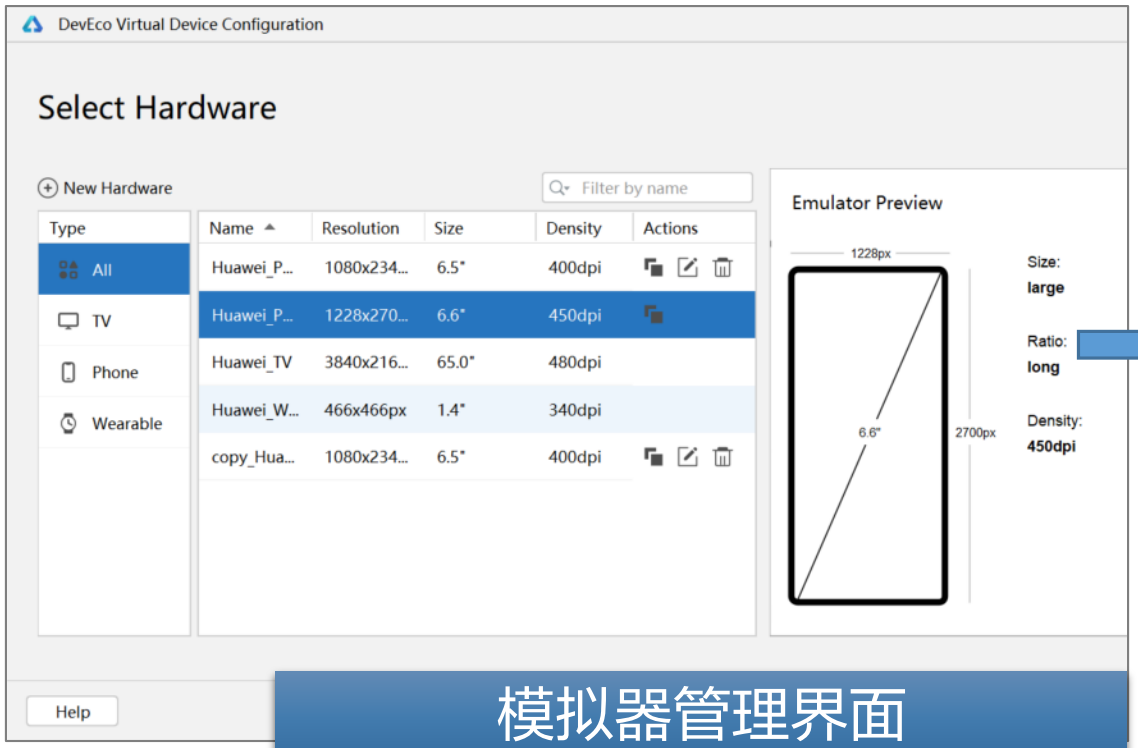
HUAWEI

创建本地模拟器

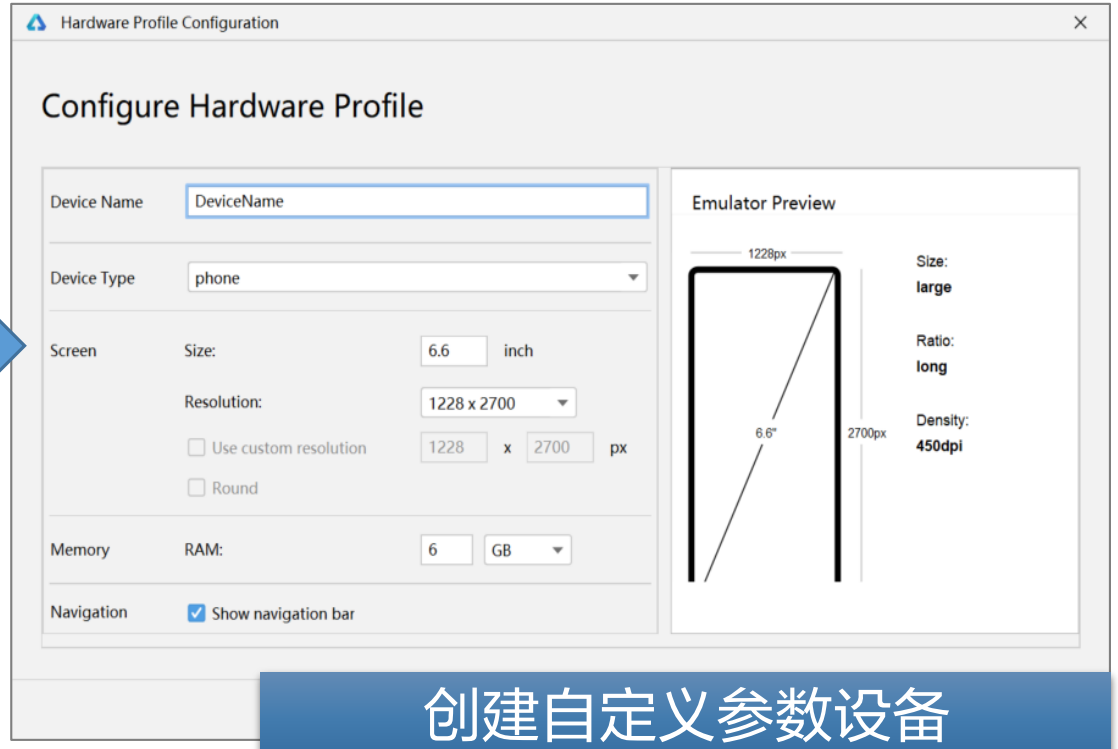
- 单击菜单栏的Tools > Device Manager，在Local Emulator页签，单击设置本地模拟器的存储路径Local Emulator Location。单击右下角的New Emulator按钮，创建一个本地模拟器。

自定义设备参数

- 在创建模拟器界面，可以选择一个默认的设备；同时也可以单击New Hardware或默认设备后的克隆图标，添加一个新设备，以便自定义设备的相关参数，如尺寸、分辨率、内存等参数。



模拟器管理界面



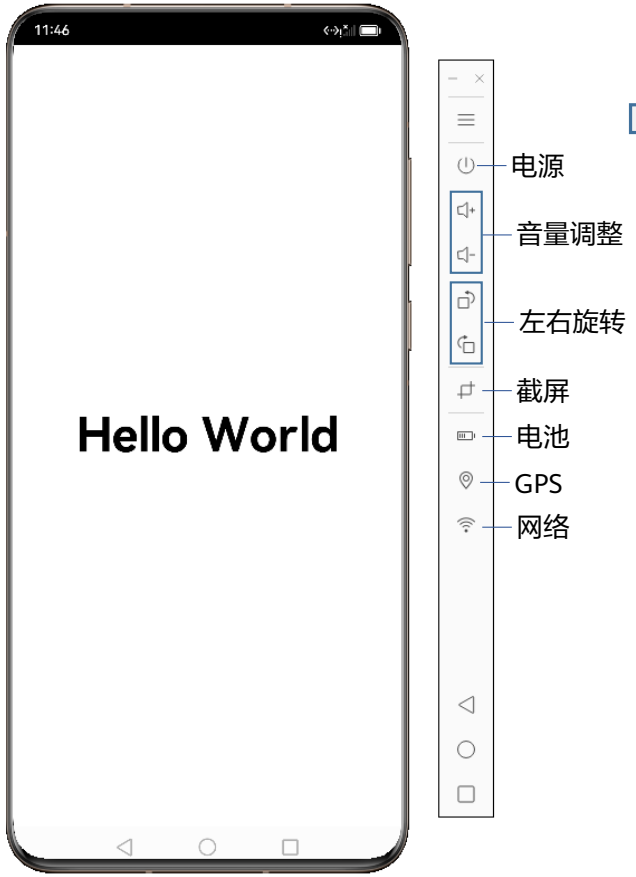
创建自定义参数设备

下载镜像文件

- 设置完设备参数，单击Next，可以看到模拟器的镜像信息，如API、Version、CPU/ABI等信息。下载完镜像文件单击Next，核实确定需要创建的模拟器信息后，单击Finish创建本地模拟器。

应用/服务运行在本地模拟器

- 在设备管理器页面，单击 启动模拟器。



除了可以像真机一样直接在手机屏幕内操作，本地模拟器右侧的工具栏提供了操作便捷、丰富和数据注入能力，包括：调整音量、电池模拟、GPS模拟、网络模拟、虚拟传感器模拟等。

HUAWEI

目录

HUAWEI

为什么需要远程真机?

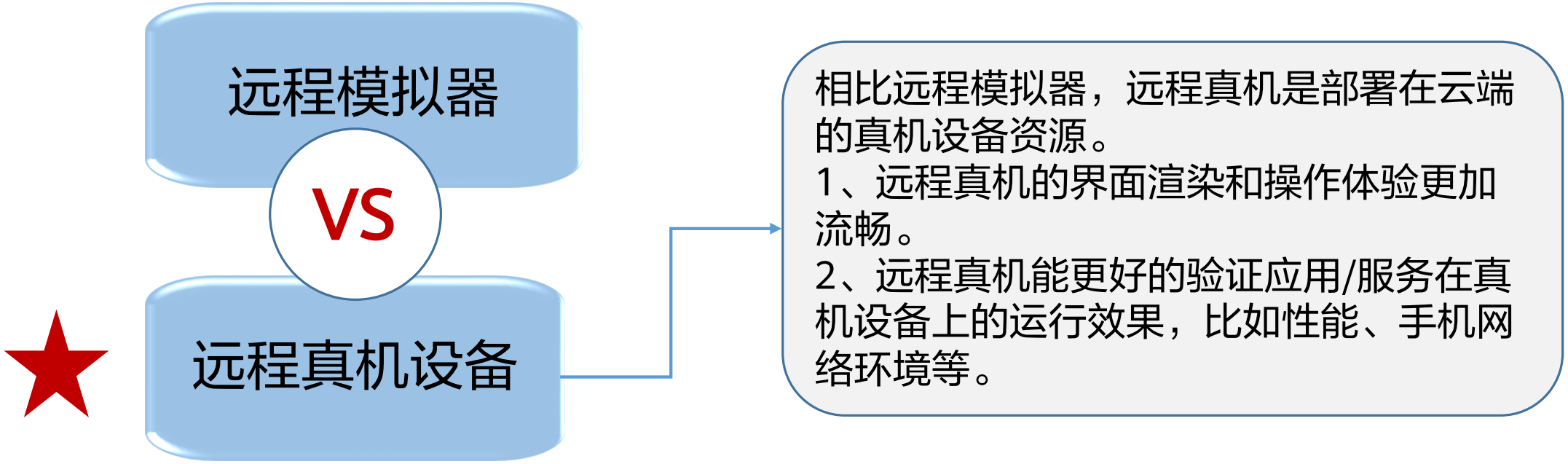
虽然，模拟器很好用。
但是我还是想使用真机进行应用调试。

但专门买个测试机成本又太高了！

如果开发者没有真机设备资源，则不能很方便的调试和验证HarmonyOS应用。
为方便开发者，DevEco Studio提供了**Remote Device**远程真机设备资源供开发者使用，减少开发成本。

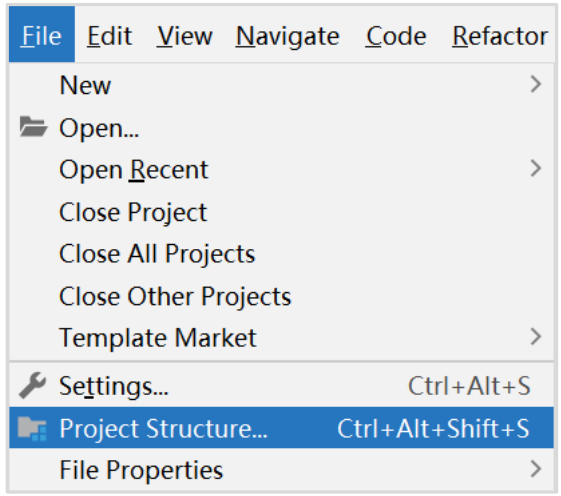
远程真机介绍

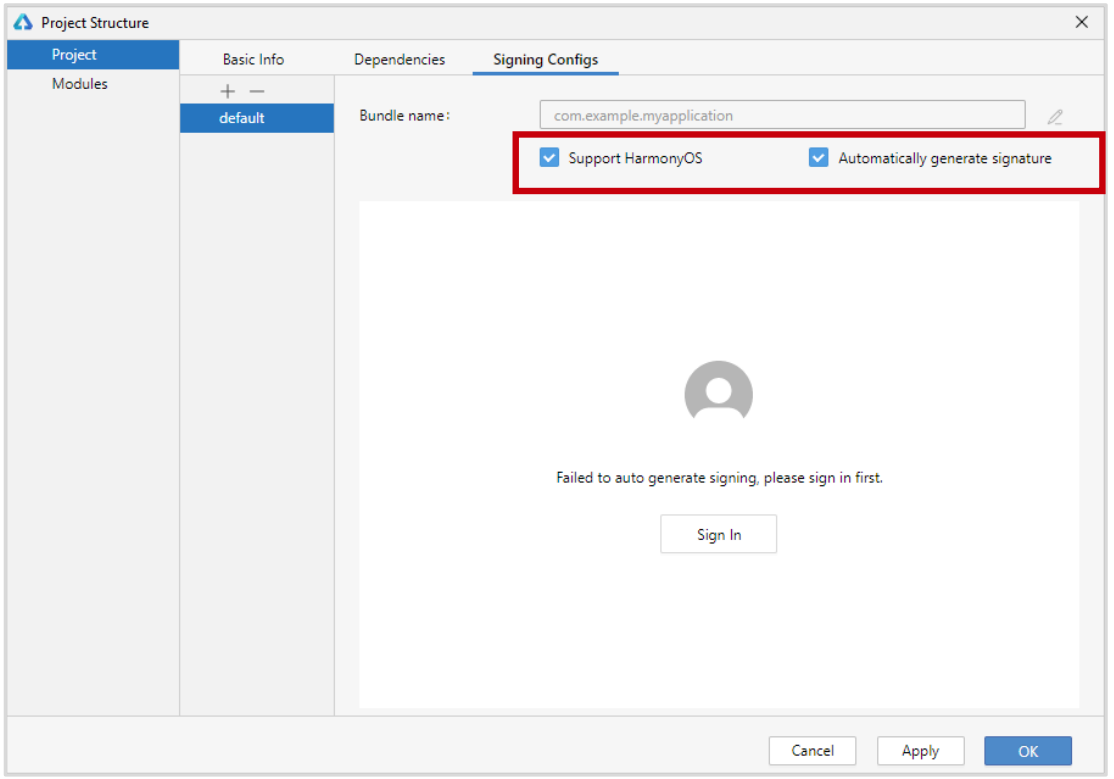
- 目前，远程真机支持Phone和Wearable设备，开发者使用远程真机调试和运行应用时，**同本地物理真机设备一样，需要对应用/服务进行签名才能运行**。



为应用/服务进行签名（自动化签名方案）

- 进入File > Project Structure > Project > Signing Configs界面，勾选“Automatically generate signature”，即可完成签名。如果未登录，请先单击Sign In进行登录，然后自动完成签名。

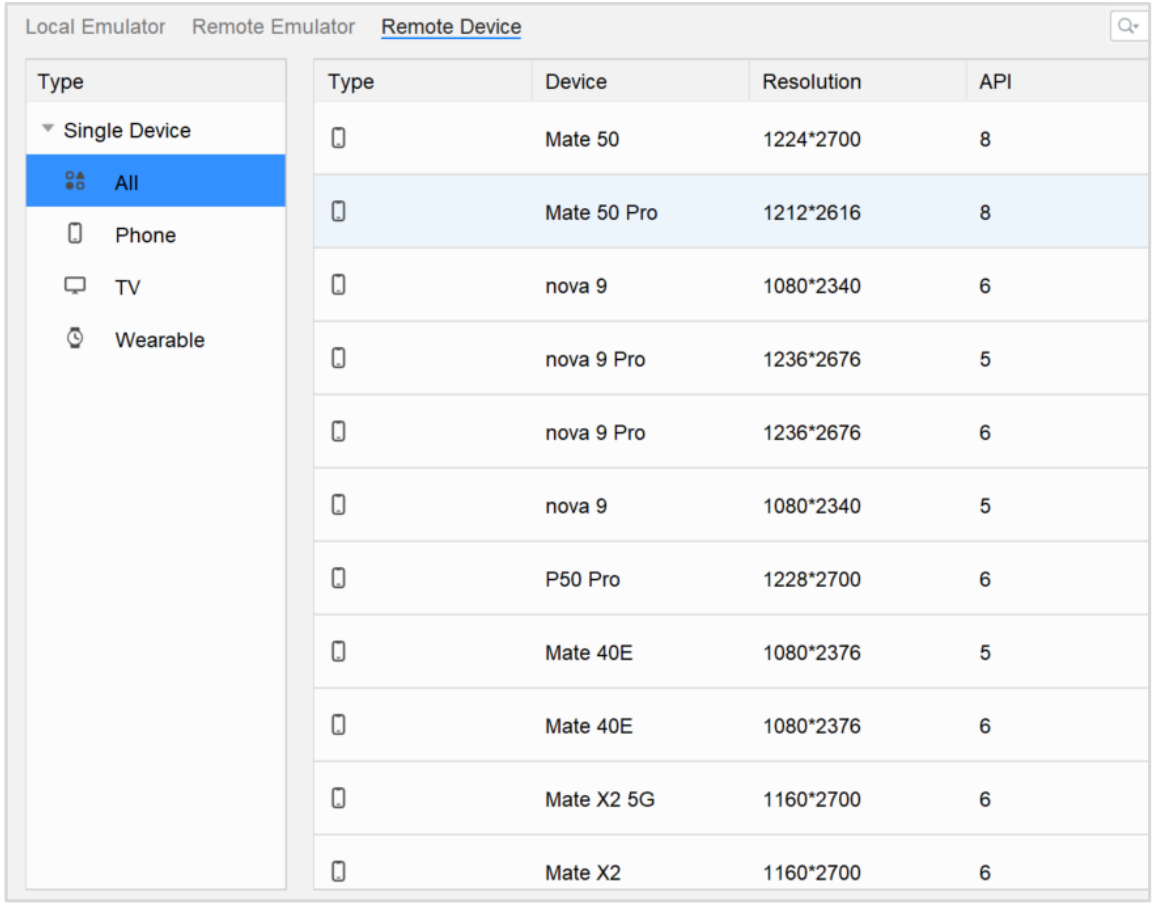




使用远程真机运行应用

运程真机需要对应用进行签名才能运行，除此需要注意的是，远程真机每次释放后重新申请，服务端分配的设备都不一样。因此，每次重新申请远程真机后，都需要对应用/服务重新进行签名。

操作步骤与使用运程模拟器类似，需要有实名认证成功的华为开发者账号。



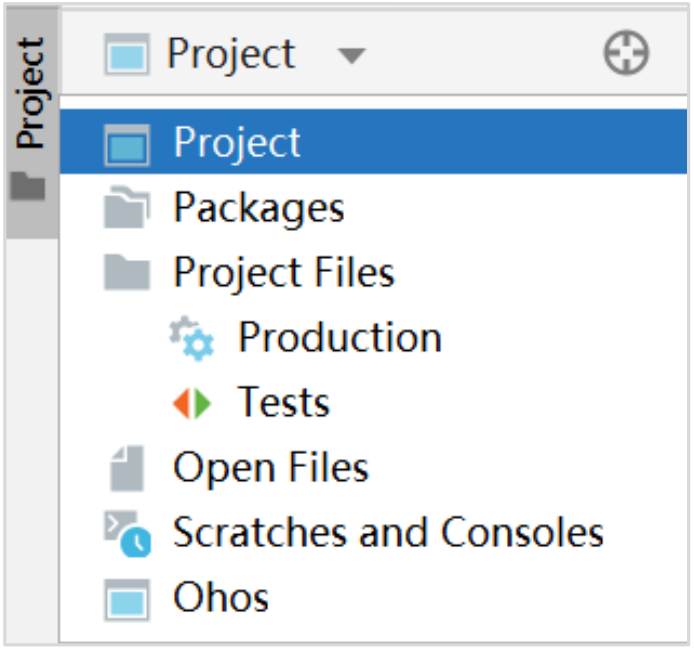
HUAWEI

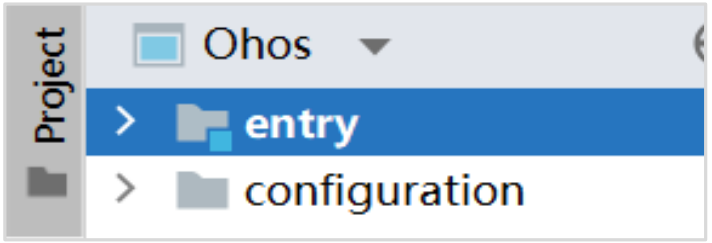
目录

HUAWEI

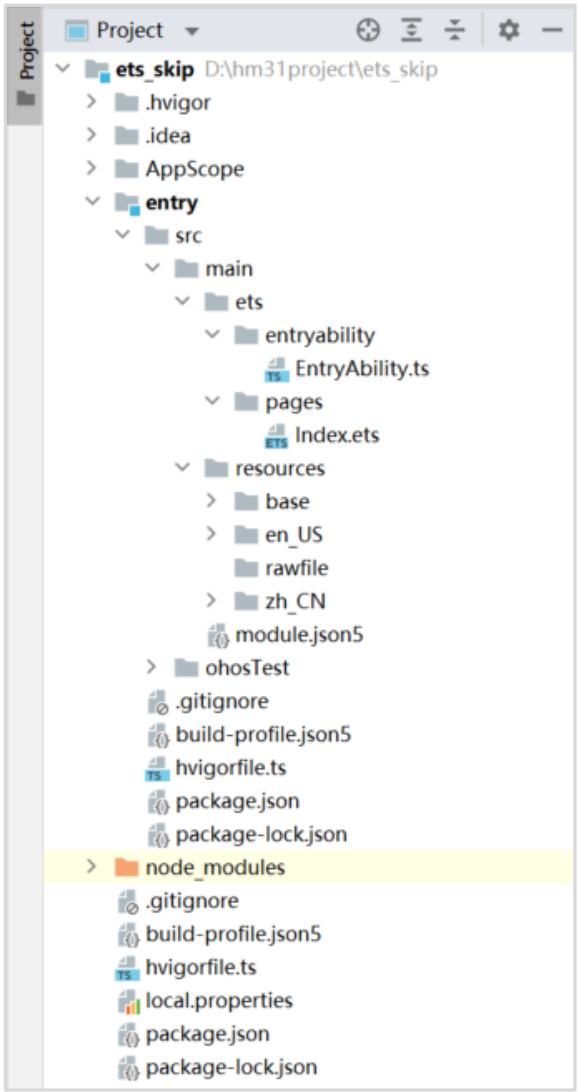
HarmonyOS工程目录结构视图

- DevEco Studio工程目录结构提供工程视图，包视图以及HarmonyOS视图。
- **Project视图：**
  - 以Project为根目录列出；
  - 常用的视图模式，且为工程创建或打开后的默认展示；
  - 可以看到工程内所有文件。
- **Packages视图：**
  - 以包为单位列出所有的源代码以及资源。
- **HarmonyOS视图：**
  - 以HAP包为单位列出。





ArkTS工程目录结构



- **entry：主模块**，HarmonyOS工程模块，编译构建生成一个HAP包。
  - **src > main > ets**：用于存放ets源码。
  - **src > main > ets > EntryAbility**：应用/服务的入口。
  - **src > main > ets > pages**：MainAbility包含的页面。
  - **src > main > resources**：用于存放应用/服务所用到的资源文件，如图形、多媒体、字符串、布局文件等。
  - **src > main > resources > profile>main_pages.json**：用于存放应用中页面的路由。
  - **src > main > module.json5**：模块配置文件。主要包含HAP包的配置信息、应用/服务在具体设备上的配置信息以及应用/服务的全局配置信息。
  - **entry >build-profile.json5**：当前的模块信息、编译信息配置项，包括buildOption、targets配置等。
  - **entry >hvigorfile.ts**：模块级编译构建任务脚本，开发者可以自定义相关任务和代码实现。
- **build-profile.json5**：应用级配置信息，包括签名、产品配置等。
- **hvigorfile.ts**：应用级编译构建任务脚本。

构建第一个页面 (1)

- **使用文本组件**

工程同步完成后，在“Project”窗口，点击“entry > src > main > ets > pages”，打开“Index.ets”文件，可以看到页面由Text组件组成，修改文字，“Index.ets”文件的示例如下：

```
@Entry
@Component
struct Index {
  @State message: string = '第一个页面'
  build() {
    Row() {
      Column() {
        Text(this.message)
          .fontSize(50)
          .fontWeight(FontWeight.Bold)
      }
      .width('100%')
    }
    .height('100%')
  }
}
```

构建第一个页面 (2)

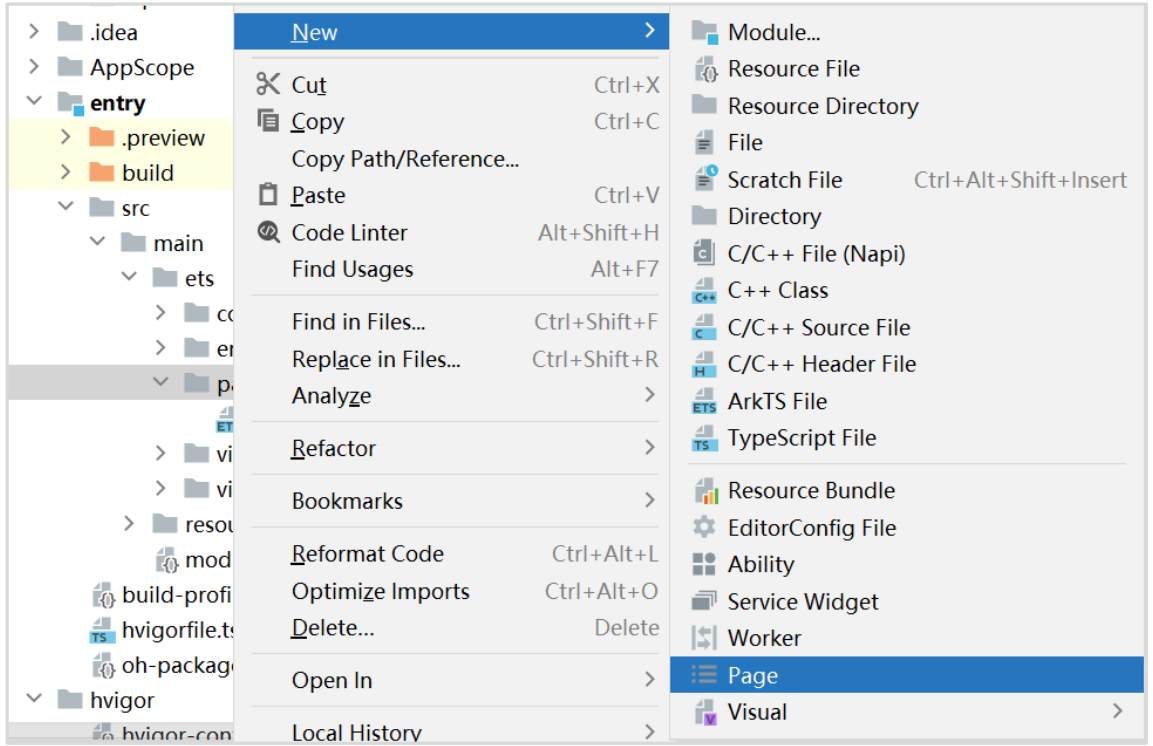
- **添加按钮**

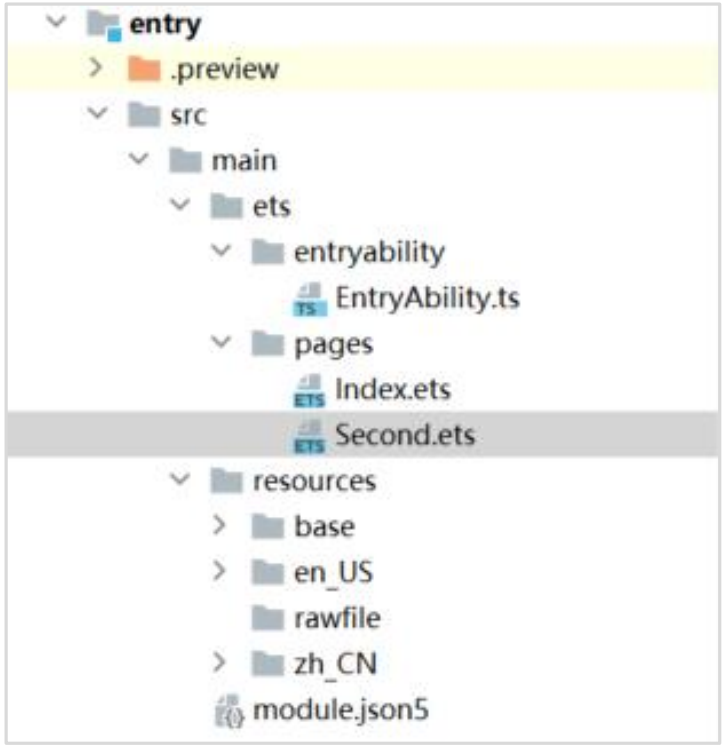
在此页面基础上，我们添加一个Button组件，作为按钮响应用户点击，从而实现跳转到另一个页面。“Index.ets”文件的示例如右：


```
@Entry
@Component
struct Index {
  @State message: string = '第一个页面'
  build() {
    Row() {
      Column() {
        Text(this.message)
          .fontSize(50)
          .fontWeight(FontWeight.Bold)
        // 添加按钮，以响应用户点击
        Button() {
          Text('跳转')
            .fontSize(30)
            .fontWeight(FontWeight.Bold)
        }
        .type(ButtonType.Capsule)
        .margin({
          top: 20
        })
        .backgroundColor('#0D9FFB')
        .width('40%')
        .height('5%')
      }
      .width('100%')
    }
    .height('100%')}}
```

构建第二个页面 (1)

- 新建第二个页面文件。在“Project”窗口，打开“entry > src > main > ets > pages”，右键点击“pages”文件夹，选择“New > Page”，命名为“Second”，点击“Finish”。可以看到文件目录结构如下：





HUAWEI

构建第二个页面 (2)

- 查看第二个页面的路由。在“Project”窗口，打开“entry > src > main > resources > base > profile”，由于使用的创建方式为Page，因此在main_pages.json文件中的“src”下系统会自动生成第二个页面的路由“pages/Second”。示例如下：

```
{
  "src": [
    "pages/Index",
    “pages/Second”  // 系统添加第二个页面的路由
  ]
}
```

构建第二个页面 (3)

- 添加文本及按钮

参照第一个页面，在第二个页面添加Text组件、Button组件等，并设置其样式。“Second.ets”文件的示例如下：


```
@Entry
@Component
struct Second {
  @State message: string = '第二个页面'
  build() {
    Row() {
      Column() {
        Text(this.message)
          .fontSize(50)
          .fontWeight(FontWeight.Bold)
        Button('返回')
        .type(ButtonType.Capsule)
        .margin({
          top: 20
        })
        .backgroundColor('#0D9FFB')
        .width('40%')
        .height('5%')
      }
      .width('100%')
    }
    .height('100%')
  }}
```


HUAWEI

实现页面间跳转（1）

- 页面间的导航同样可以通过页面路由router来实现。页面路由router根据页面url找到目标页面，从而实现跳转。使用页面路由请导入router模块。
- 第一个页面跳转到第二个页面
  - 在第一个页面中，跳转按钮绑定onclick方法，点击按钮时跳转到第二页。“Index.ets”添加如下代码：

```
// Index.ets
// @ohos.router模块功能从API version 8开始支持，请使用对应匹配的SDK
import router from '@ohos.router';
```

```
...
.backgroundColor('#0D9FFB')
        .width('40%')
        .height('5%')
        // 跳转按钮绑定onClick事件，点击时跳转到第二页
        .onClick(() => {
          router.pushUrl({ url: 'pages/Second' })
        }...
```

实现页面间跳转 (2)

- 第二个页面返回到第一个页面。在第二个页面中，返回按钮绑定back方法，点击按钮时返回到第一页。“Second.ets”示例如下

```
Button() {
        Text('返回')
          .fontSize(25)
          .fontWeight(FontWeight.Bold)
      }
      .type(ButtonType.Capsule)
      .margin({
        top: 20
      })
      .backgroundColor('#0D9FFB')
      .width('40%')
      .height('5%')
      // 返回按钮绑定onClick事件，点击按钮时返回到第一页
      .onClick(() => {
        router.back()
      })
```

目录

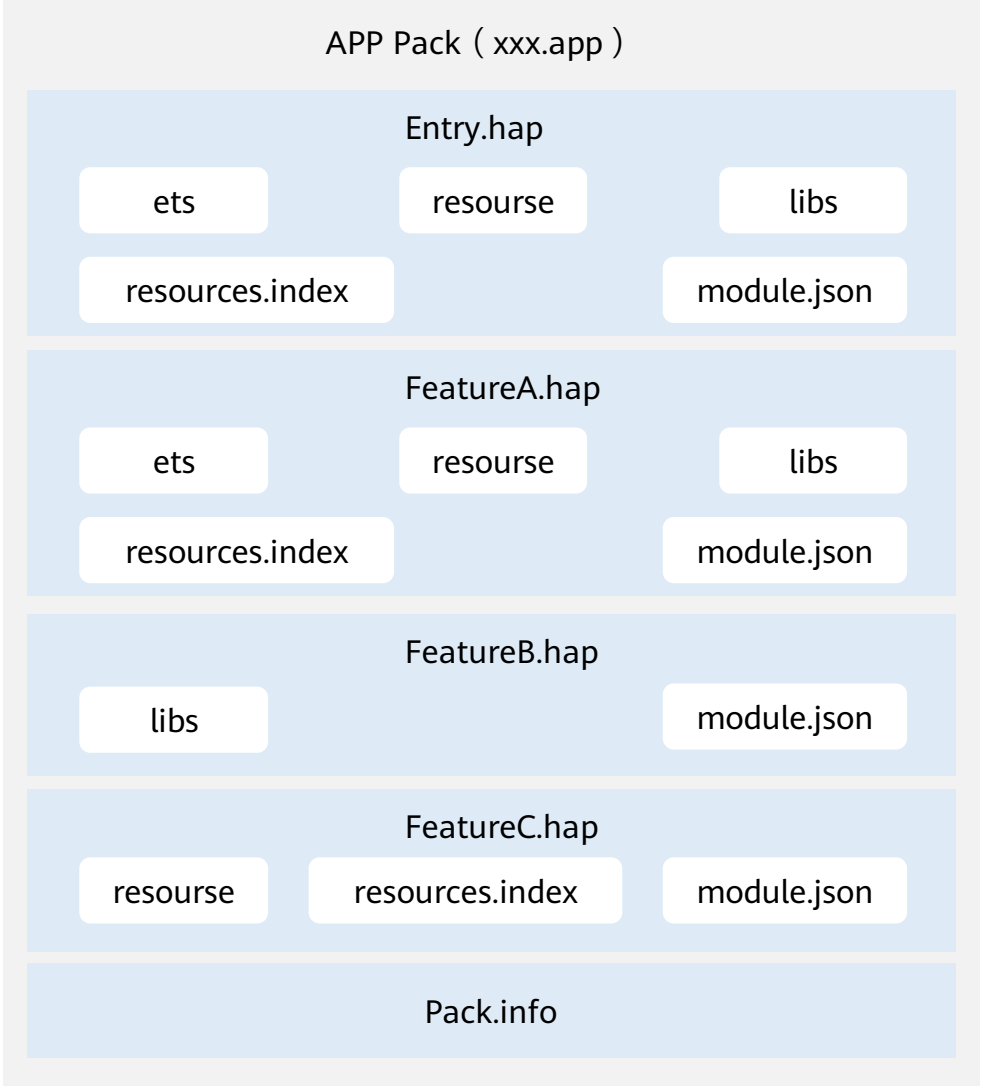
HUAWEI

HarmonyOS APP工程结构

- 在进行HarmonyOS应用开发前，应该掌握HarmonyOS应用的逻辑结构。
- HarmonyOS应用发布形态为**APP Pack**（Application Package，简称APP），它是由一个或多个**HAP**（HarmonyOS Ability Package）包以及描述APP Pack属性的**pack.info**文件组成。
- HAP是**Ability**的部署包，HarmonyOS应用代码围绕Ability组件展开，它是由一个或多个Ability组成。

Stage模型应用程序包结构

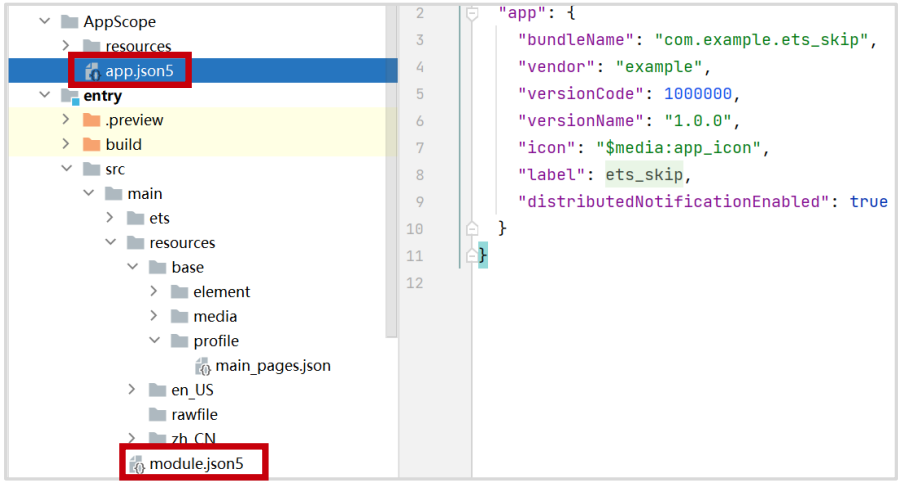
- HAP是HarmonyOS应用安装的基本单位，包含了编译后的代码、资源、三方库及配置文件。
- HAP可分为Entry和Feature两种类型。
- **Entry**：是应用的主模块，在module.json5中的type属性配置为entry类型。
- **Feature**：是应用的动态特性模块，在module.json5中的type属性配置为feature类型。一个应用程序包可以包含一个或多个Feature类型的HAP，也可以不包含。



HUAWEI

Stage模型应用包结构配置文件说明

- 开发stage模型下的应用程序时，需要在module.json5和app.json5配置文件中对应用的包结构进行声明。文件内容主要涵盖以下两个部分：
  - App：应用的全局配置信息，包含应用的包名、生产厂商、版本号等基本信息。
    - **同一个应用的不同HAP包的app配置必须保持一致**
  - Module：HAP包的配置信息，包含每个Ability必须定义的基本属性（如包名、类名、类型以及Ability提供的能力）等。该标签下的配置只对当前HAP包生效。



pack.info

- 描述应用软件包中每个HAP的属性，由IDE编译生成，应用市场根据该文件进行拆包和HAP的分类存储。HAP的具体属性包括：
  - delivery-with-install: 表示该HAP是否支持随应用安装。“true”表示支持随应用安装；“false”表示不支持随应用安装。
  - name：HAP文件名。
  - module-type：模块类型，**entry或feature**。
  - device-type：表示支持该HAP运行的设备类型。

本章小结

- 主要通过HarmonyOS应用开发的基本流程逐步展开。从开发前准备开始，介绍了DevEco Studio集成开发环境、以及其调试运行工具的使用；再以使用ArkTS语言进行开发的角度讲述工程目录结构等基础知识。

思考题

1. （多选题）以下哪几项属于Previewer预览器支持的功能？（ ）

A. 能够查看使用ArkTS语言开发的应用UI效果

B. 动态预览

C. 观看动态图片

D. 播放语音

思考题

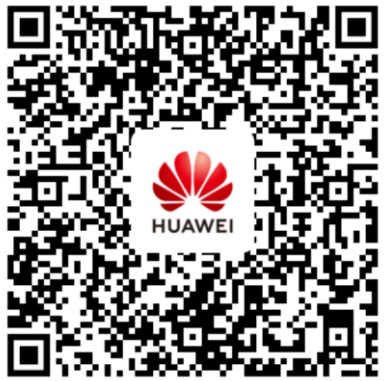
2. （判断题）使用远程真机运行应用时，不需要对应用进行签名。（ ）

A. 正确

B. 错误

学习推荐

- 官方学习网站
  - HarmonyOS官网：https://developer.harmonyos.com/
  - HarmonyOS应用开发文档： https://developer.huawei.com/consumer/cn/
  - OpenHarmony官网：https://edu.huaweicloud.com/
  - 华为开发者论坛： https://developer.huawei.com/consumer/cn/forum/



华为云开发者学堂

HUAWEI

感谢