## 预览器：Previewer

* 预览特性的典型特征
  + 多端
  + 实时
  + 动态
* 在应用开发过程中，预览器Previewer可以查看应用的UI界面效果，支持Java应用和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，如点击、跳转、滑动等，与应用运行在真机设备上的交互体验一致

## 模拟器

* Simulator
  + Simulator是一款轻量级模拟器，可以调试运行Lite Wearable和Smart Vision设备的应用
  + 可以运行兼容签名与不签名两种类型的HAP
  + 在Simulator中运行应用的步骤如下

1. 运行
2. 选择设备
   * + - Huawei Lite Wearable Simulator
       - Huawei Smart Vision Simulator
3. 查看效果，调试

* Remote Emulator
  + 是一款远程模拟器，可以运行和调试Phone、Tablet、Car、TV和Wearable的应用（但不支持Lite Wearable）
  + 在Remote Emulator上运行应用兼容签名与不签名两种类型的HAP
  + 使用Remote Emulator需要使用已经过实名认证的华为账号登录，并在每次使用时授权
  + Remote Emulator每次使用时长为1小时，到期后会自动释放资源。如果Remote Emulator到期释放后，可以重新申请资源
  + 使用Remote Emulator运行应用的步骤如下

1. 登录授权：启动Device Manager，并使用华为开发者联盟账号进行登录和授权
2. 启动模拟器：在Device Manager设备列表中选择设备类型并运行
3. 编译构建应用程序，并部署运行在远程模拟器上

* 使用远程模拟器调试，可以同时支持签名和不签名两种类型的HAP
  + 如果应用需要签名，则流程与使用真机调试的流程完全一样
  + 如果不签名，调试流程则为

1. 设置调试代码类型
   * + - 调试类型默认情况为Detect Automatically，支持Java、JS、C/C++、JS+Java、Java+C/C++工程的调试。（在JS+Java混合工程中，如果需要单独调试Java代码，这种情况下需要手动修订Debug Type为"Java"）
       - 修改调试类型的方法如下：点击Run>Edit Configurations>Debugger，在HarmanyOS App中，选择相应模块，可以进行调试配置
2. 检查config.json文件属性：
   * + - 在启动Feature模块的调试前，需要检查Feature模块下的config.json文件的abilities数组是否存在"visble"属性
       - 如果不存在，需要手动添加，否则Feature模块的调试无法进入断点，调试完后要进行删除
       - Entry模块的调试不需要做该检查
3. 启动调试
   * + - 在工具栏中，点击Debug
       - 在弹出的界面，选择需要调试的设备
         * 真机设备：一般为可以用USB或IP方式连接的实体设备
         * 远程模拟器：支持Phone、Tablet、Car、TV、Wearable等设备
       - 如果需要设置断点调试，则需要选定要设置断点的有效代码行，在行号（比如：24行）的区域后，单机鼠标左键设置断点（如下图示的红点）。设置断点后，调试能够在正确的断点处中断，并高亮显示该行

## 真机

* 可以用PSP或IP方式连接的实体
* 构建带签名信息的HAP包：带签名信息的HAP包（包括Debug和Release类型），可以在真机设备上运行
* 在构建HAP包前，需要对应用进行签名
  + 方法一（推荐）：通过DevEco Studio自动化签名的方式对应用进行签名
  + 方法二：通过从AppGallery Connect中申请调试证书和Profile文件后，再进行签名
* 使用真机设备进行调试前，需要对HAP进行签名，详细的调试流程如下所示
  + 对应用进行签名
    - 连接真机设备
    - 在AppGallery Connect中创建项目和应用
    - 进入File＞Project Structure＞Project＞Signing Configs界面，点击“Sign In”按钮进行登录
  + 调试设置
    - 设置调试代码类型
    - 检查config.json文件属性
  + 启动调试：连接设备启动调试