Lua 热修复Debug

我们可以使用 Lua debugger 工具来调试 lua 代码

使用步骤

- 下载 ZeroBraneStudio 并运行 ZeroBraneStudio
- 导入 lua 代码: 点击 ZeroBraneStudio 菜单栏上的



图标,选择要导入的 lua 代码目录

- 先启动 debug 服务器: 点击 Project->Start Debugger Server.
- 添加调试扩展

```
1 // 添加调试库
2 [WPFix addExtensionDebug];
3 // 启动
4 [WPFix start];
5 // 执行测试脚本
6 [WPFix runLuaFile:@"test.lua"];
```

- 添加 lua 代码 require('mobdebug').start('YOUR_MAC_IP_ADDRESS') 到 lua 文件的第一行.如果使用的是模拟器, 'YOUR_MAC_IP_ADDRESS' 可以留空。
- 启动 当 require('mobdebug').start() 调用时, ZeroBraneStudio 就会被激活,此时可以添加断点,也可以输入调试命令。

调试功能

remote console: 可以输入要打印的变量

stack window: 打开 view->Stack Window,可以查看 lua 栈

调试截图

```
Project Outline
                                                                test.lua
                                                                                     wpfix.print("【LUA】ViewController 打印原生入参 thingName", thingName)
      [ /Users/karosli/Documents/Work/TanChiShe/Cod
                                                                                          设置/获取
         CustomTableViewController.lua
                                                                                     self.aa = "hh"
                                                                 74
         test.lua
                                                                 75
                                                                                     wpfix.print("【LUA】ViewController 获取 lua 属性 aa", self.aa)
                                                                76
77
78
                                                                                     -- 动态添加 设置/获取 原生 属性
self:setSex_("男")
                                                                 79
80
                                                                81 🕖
                                                                                     wpfix.print(" [LUA] ViewController 获取 动态 原生 属性 sex", self:sex())
                                                                82
                                                                                        - 设置/获取 原生 私有 变量
                                                                                     医性f:setfvar_withInteger_("_aInteger", 666)
wpfix.print(" [LUA] ViewController 获取 原生 私有变量 _aInteger", <u>self</u>:getIvarInteger_("_aInteger"))
                                                                 83
                                                                84
                                                                85
                                                                                         调用实例方法
                                                                                     wpfix.print(" [LUA] ViewController 获取原生调用结果 getHello() ", self:getHello()) wpfix.print(" [LUA] ViewController 获取原生调用结果 getViewSize()", tostring(self:getViewSize())) -- 调用当前类的静态方法
                                                                 86
                                                                87
                                                                 88
                                                                 29
                                                                                     ViewController:printHello()
                              Markers
                                             Remote console
  Welcome to the interactive Lua interpreter. Enter Lua code and press Enter to run it. Use Shift-Enter for multiline code.

Use 'clear' to clear the shell output and the history. Use 'reset' to clear the environment.

Prepend '=' to show complex values on multiple lines. Prepend '!' to force local execution.
"做你"
>> self:age()
≫wpfix
  appVersion = "1.0", device = {screenHeight = 812, screenScale = 3, screenWidth = 375} --[[table: 0x2822c9a00]], exit = function() --[[..skipped..]] end --[[function: 0x102740164]], os
= {geOS = function() --[[..skipped..]] end --[[function: 0x102740010]], systemVersion = "14.8.1"} --[[table: 0x2822c99c0]], print = function() --[[..skipped..]] end --[[function: 0x102740164]], systemVersion = "14.8.1"} --[[table: 0x2822c99c0]], print = function() --[[..skipped..]] end --[[function: 0x1027402c3]], setConfig = function() --[[..skipped..]] end --[[function: 0x1027606a8]], version = "1.0.0"} --[[table: 0x2822c9980]]
```