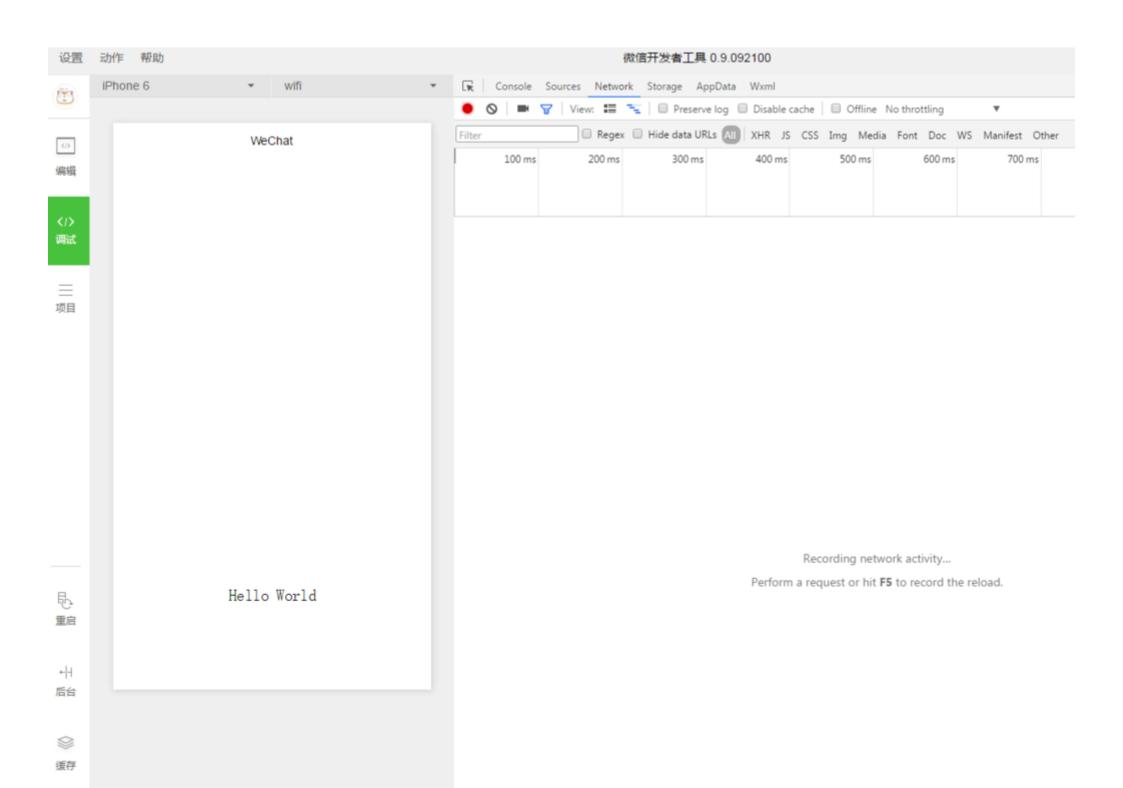
stiller: 微信小程序教程入门篇兄弟篇: 从零开始完整基础流程 ...

- 扫描二维码登录进入微信开发者工具
- (如下图)新建一个项目,勾选在当前目录中创建 quick start 项目

	新建项目				
AppID					
	填写小程序AppID,可在公众平台开发设置页中查看				
项目名称					
本地开发目录		选择			
				取消	添加项目

• (如下图)就会出现一个 HelloWorld 的 APP 实例。

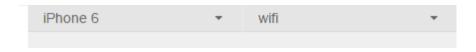


- 编辑菜单就是对于项目文件的编辑和修改,项目目录如下:
- pages
- ▶ **■** utils
- Js app.js
- ^{Jℕ} app.json
- { } app.wxss

• 其中调试菜单可以看到你对于代码的修改,每一次的修改需要点击重启服务

WeChat </>> 编辑 く/〉 调试 三 项目 Hello World し動 --后台

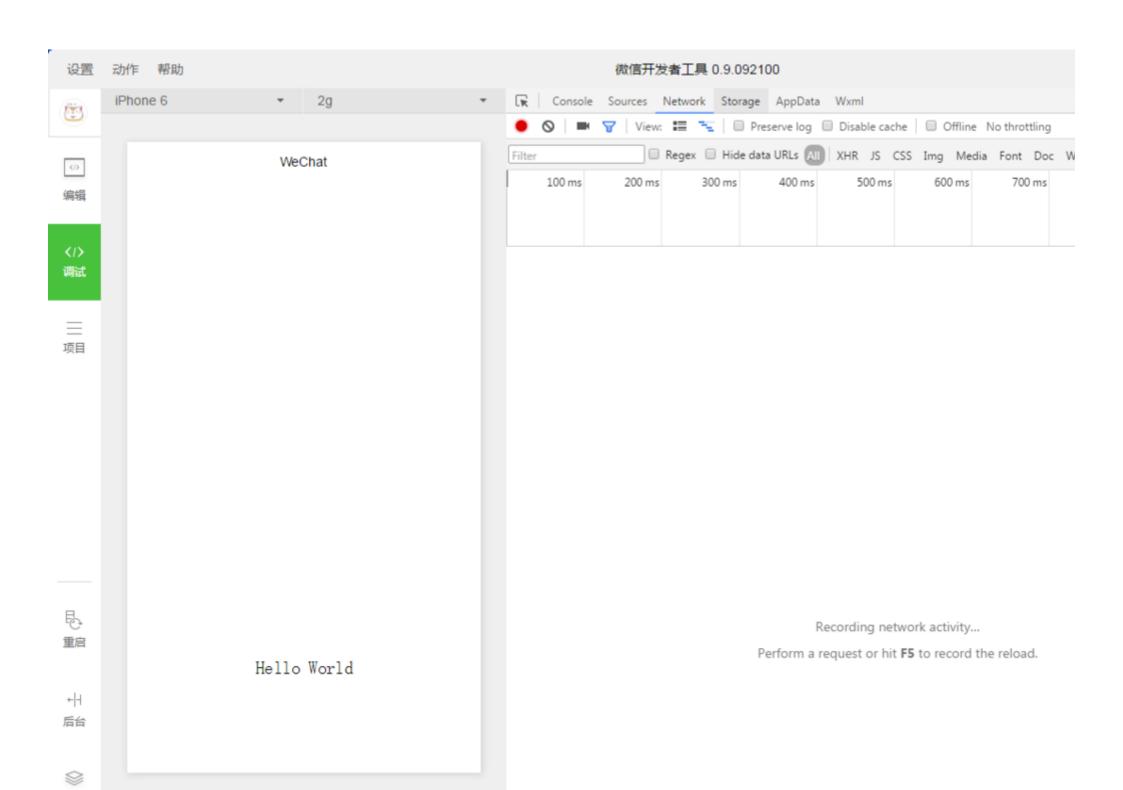
• 同样,在此菜单出可以选择对于不同机型进行不同的适配,默认是 iPhone6 和 wifi 环境下,可以对于不同网络环境和屏幕大小进行 APP 的优化选择。



• 项目菜单,可以用来打开或者新建新的项目,并且对于已经开发好的项目可以上传处理。



• 新建一个项目,打开一个HelloWorld



• 看一下项目目录的文件。

- → pages
 → index
 - JS index.js
 - <> index.wxml
 - { } index.wxss
 - → logs
 - Js logs.js
 - Jℕ logs.json
 - <> logs.wxml
 - { } logs.wxss
 - main
 - Js main.js
 - \leftrightarrow main.wxml
 - { } main.wxss
- →

 mathrid

 mat
 - JS util.js
- JS app.js
- ^{Jℕ} app.json
- { } app.wxss

• 这个项目目录中, app. js 是小应用的基本入口,包含了对于登录接口的调用,获得用户数据

```
//app.js
App({
  onLaunch: function () {
    //调用API从本地缓存中获取数据
   var logs = wx.getStorageSync('logs') || []
   logs.unshift(Date.now())
   wx.setStorageSync('logs', logs)
  getUserInfo:function(cb){
   var that = this;
   if(this.globalData.userInfo){
     typeof cb == "function" && cb(this.globalData.userInfo)
   }else{
     //调用登录接口
     wx.login({
        success: function () {
         wx.getUserInfo({
           success: function (res) {
             that.globalData.userInfo = res.userInfo;
             typeof cb == "function" && cb(that.globalData.userInfo)
         })
     });
  globalData:{
    userInfo:null
})
```

• app. json 这个文件是对于这个应用外观的一些配置项,包括字体和导航菜单等样式的配置以及对于路由页面的配置

```
1
       "pages":
 2
         "pages/index/index",
 3
         "pages/logs/logs"
 4
       "window":{
         "backgroundTextStyle":"light",
 7
         "navigationBarBackgroundColor": "#fff",
 8
         "navigationBarTitleText": "WeChat",
 9
         "navigationBarTextStyle": "black"
10
11
12
13
```

在说一下对于一个完整的页面而言,其实小程序就是对于 node 和 react 的打包实现所以基本的一个页面也是采用 HTML+CSS 得实现方式,那么对于一个页面而言,比如说 index 页面部分,. wxml ,. xss 就是包装版的 HTML(XML)或是 CSS。

```
/**index.wxss**/
     .userinfo {
 2
      display: flex;
 3
      flex-direction: column;
 4
       align-items: center;
 6
 7
     .userinfo-avatar {
 8
      width: 128rpx;
9
      height: 128rpx;
10
      margin: 20rpx;
11
       border-radius: 50%;
12
13
14
     .userinfo-nickname {
15
       color: #aaa;
16
17
18
     .usermotto {
19
      margin-top: 200px;
20
21
```

- 那么尝试些一个小的应用
- 先在 app. json 中进行配置,新增一个新的页面,就直接丢到 index 下

```
{
    "pages":[
        "pages/index/index",
        "pages/logs/logs",
        "pages/index/hello"
],
```

新建一个 hello. wxml 和一个 hello. js ,因为不做特别的用处,所以直接将 index. js 的代码 copy 过来就好了

```
//index.js
1
    //获取应用实例
2
    var app = getApp()
    Page({
4
      data: {
5
        motto: 'Hello',
6
7
        userInfo: {}
8
      },
      //事件处理函数
9
      bindViewTap: function() {
10
        wx.navigateTo({
1
12
          url: '../logs/logs'
L3
        })
4
      },
      onLoad: function () {
15
        console.log('onLoad')
16
        var that = this
17
        //调用应用实例的方法获取全局数据
18
19
        app.getUserInfo(function(userInfo){
          //更新数据
20
          that.setData({
21
22
            userInfo:userInfo
23
          that.update()
24
25
        })
26
      }
27
    })
28
```

同样操作 hello. wxml 为了区别它和 index 的不同,给它加上一个<image>标签,随便丢张图片

Paste_Image.png

• 再在配置文件 app. json 中加上一个 list 作为一个导航链接

```
"tabBar":{
    "color":"#eff",
    "selectedColor":"write",
    "backgroundColor":"#eef",
    "list": [{
        "pagePath": "pages/index/index",
        "text": "首页"
    }, {
        "pagePath": "pages/index/hello",
        "text": "hello"
    }]
}
```

切换到调试界面点击重启,观看效果



小程序



Hello