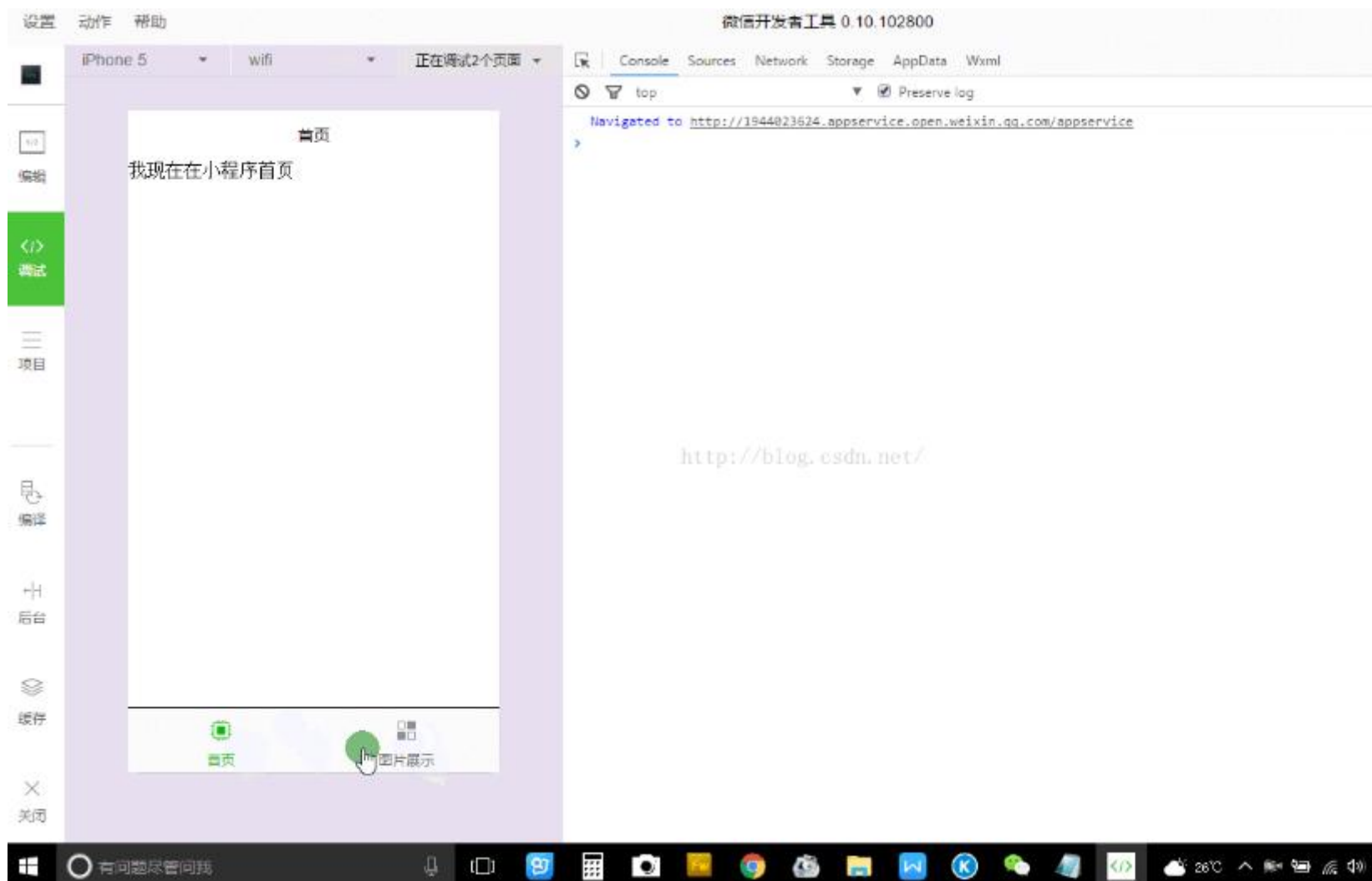


tabBar 的使用小技巧（附数据绑定一个对象到数组中，然后取值的一个注意点） ...

一：tabBar 的使用小技巧



原理：在 app.json 中配置 tabBar 属性

3、关键代码

```
1 "tabBar": {  
2     "color": "#666666",  
3     "selectedColor": "#06bd04",  
4     "list": [{  
5         "pagePath": "index",  
6         "text": "首页",  
7         "iconPath": "images/index.png",  
8         "selectedIconPath": "images/indexHL.png"  
9     }, {  
10        "pagePath": "picDisplay",  
11        "text": "图片展示",  
12        "iconPath": "images/picDisplay.png",  
13        "selectedIconPath": "images/picDisplayHL.png"  
14    }]  
15 }
```

来自 CODE 的代码片

snippet_file_0.txt

4、操作方法

- 新建一个项目，打开 app.json 文件，将关键代码复制到"window":{}后面，注意 window 的大括号前加逗号，如下图

```
app.json x
1 {
2   "pages": [
3     "index",
4     "picDisplay"
5   ],
6   "window": {
7     "navigationBarBackgroundColor": "#ffffff",
8     "navigationBarTextStyle": "black",
9     "navigationBarTitleText": "首页",
10    "backgroundColor": "#eeeeee",
11    "backgroundTextStyle": "light",
12    "enablePullDownRefresh": false
13  },
14  "tabBar": {
15    "color": "#666666",
16    "selectedColor": "#06bd04",
17    "list": [{
18      "pagePath": "index",
19      "text": "首页",
20      "iconPath": "images/index.png",
21      "selectedIconPath": "images/indexHL.png"
22    }, {
23      "pagePath": "picDisplay",
24      "text": "图片展示",
25      "iconPath": "images/picDisplay.png",
26      "selectedIconPath": "images/picDisplayHL.png"
27    }]
28  }
29
30 }
```

- 配置 tabBar 属性值

```
1 "tabBar": {  
2     //设置 tabBar 文字默认颜色  
3     "color": "#666666",  
4     //设置 tabBar 文字被选中是的颜色  
5     "selectedColor": "#06bd04",  
6     //tab 列表，数组类型，改数组内至少要有两个但不大于 5 个的 tab 对象  
7     "list": [{  
8         //设置 tab 跳转页面链接  
9         "pagePath": "index",  
10        //设置 tab 上的文字  
11        "text": "首页",  
12        //设置 tab 上的默认图标  
13        "iconPath": "images/index.png",  
14        //设置 tab 被选中时的图标  
15        "selectedIconPath": "images/indexHL.png"  
16    }],  
}
```

```
17     "pagePath": "picDisplay",  
18     "text": "图片展示",  
19     "iconPath": "images/picDisplay.png",  
20     "selectedIconPath": "images/picDisplayHL.png"  
21  }}  
22 }
```

二：数据绑定一个对象到数组中，然后取值的一个注意点（分享者：guonei001）

通过 wx.request 获取到服务器数据之后将数据(一个 json 对象)绑定至 post 数组，然后页面中获取数据的时候可以直接通过（数组名.属性）直接取值。数组中有多个 json 对象的话，则需要 wx:for 遍历取值。

单个 json 对象数据绑定

```

Page({
  data: {
    post: [],
    comment: []
  },
  onLoad: function(o) {
    console.log("id=" + o.id);
    var that = this;
    wx.request({
      url: 'https://www.v2ex.com/api/topics/show.json?id=' + o.id,
      //data: {"id": "520"},
      method: 'GET',
      success: function(res) {
        console.log(res);
        that.setData({
          post: res.data[0]
        });
      }
    })
    wx.request({
      url: 'https://www.v2ex.com/api/replies/show.json?topic_id=' + o.id,
      method: 'GET',
      success: function(res) {
        console.log(res);
        that.setData({
          comment: res.data
        });
      }
    })
  },
})

```

<http://blog.csdn.net/>

页面取值：

```
<view><text class="f14">作者</text></view>
<view><text class="f16">{{post.title}}</text></view>
</view class="justify-content-between">
```