

1、小程序开发环境搭建

注册开发者账号 - - - 获取小程序ID - - - 下载最新开发工具 - - - 打开开发工具新建项目

2、小程序项目结构

app.js 全局js项目的入口

定义全局函数，数据，以及全局生命周期函数 **onLaunch**

用户首次打开小程序，触发 onLaunch（全局只触发一次）。

```
globalData: {  // 全局数据存储
  userInfo: null
}
```

在App.js中修改GlobalData的值： `this.globalData.userInfo = userInfo;`

在其它页面获取全局数据：

```
var app = getApp();
```

```
var getAppInfo = app.globalData.openid;
```

修改全局数据： `app.globalData.userInfo = detail.userInfo`

获取全局数据： `app.globalData.userInfo`

app.json 对整个小程序做全局配置（标题、色系）

pages字段：用于描述当前小程序所有页面路径

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

window字段：定义小程序所有页面的顶部背景颜色，文字颜色等

`"backgroundTextStyle": "light",` 全局页面标题背景颜色

`"navigationBarBackgroundColor": "#fff",` 全局页面标题字体背景颜色

"navigationBarTitleText": "小程序开发", 全局页面标题名
"navigationBarTextStyle": "black" 全局页面标题背景颜色

tabBar字段：定义底部导航

```
"list":[  
  {      // 第一个导航  
  
    "pagePath":"pages/index/index",      // 导航的页面  
  
    "text":"首页",      // 导航的标题  
  
    "iconPath":"./assets/tab-bar/home.png",      // 默认导航图标  
  
    "selectedIconPath":"./assets/tab-bar/home-active.png"      // 选中后的导航图标  
  
  },  
  
  {      // 第二个导航  
  
    "pagePath": "pages/logs/logs",  
  
    "text": "我",  
  
    "iconPath": "./assets/tab-bar/user.png",  
  
    "selectedIconPath": "./assets/tab-bar/user-active.png"  
  
  }  
]
```

具体页面的*.json可以修改全局window配置信息

app.wxss

project.config.json

具体页面的js：

Page({.....}) 页面js入口，里面定义了所有的自定义函数、页面的生命周期函数、页面的数据

页面的数据：data:{.....}

```
data: {  
  val:'',      数据字段  
  name: '刘锦荣',  
  todos: ['吃饭', '睡觉', '开课吧学习前端']  
},
```

在当前页面中获取data定义的数据：this.data.字段名

```
// this.data.val      获取数据值  
| todos: [this.data.val]
```

设置页面数据:

```
this.setData({
```

字段名:`detail.value` 获取用户输入的数据

```
})
```

```
this.setData({  
  val: '',      将val设置为空  
  todos: [...this.data.todos, this.data.val]  重设todos的数据  
})
```

页面生命周期函数:

`onLoad(){}.....` 页面加载函数只执行一次

`onShow(){}.....` 页面显示

`onHide(){}.....` 页面隐藏

获取input输入的数据

input标签 `<input value="{{val}}" bindinput='input' placeholder="请输入想做的事"/>`

`value="{{val}}":` input输入框默认数据

`bindinput='input':` 获取input输入框的方法 input的事件绑定

JS文件中获取输入数据的方法:

```
input(e) {  
  val: e.detail.value    //获取用户输入的数据  
}
```

3、小程序核心组件

标签: `<view></view>` 相当于html中的`<div></div>` 标签

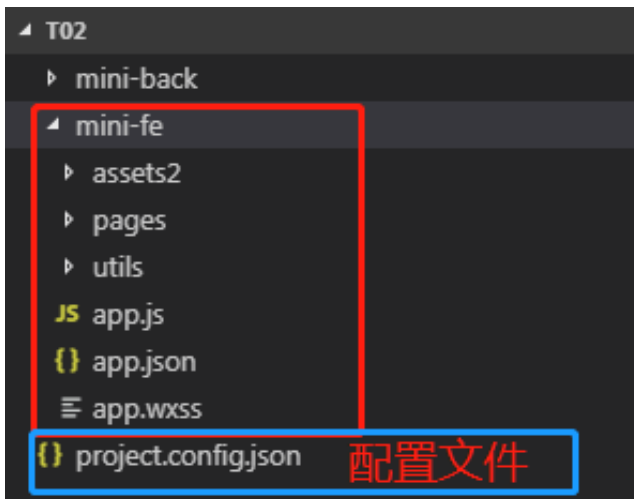
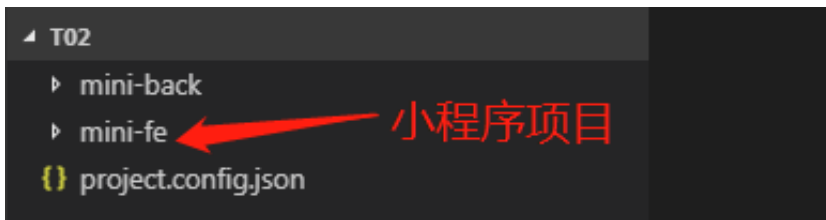
多媒体: `<image>` `<audio>` `<camera>`

4、小程序核心API:作用通过js来控制组件,如音乐播放,提示,弹窗等

5、云开发

1、配置云开发

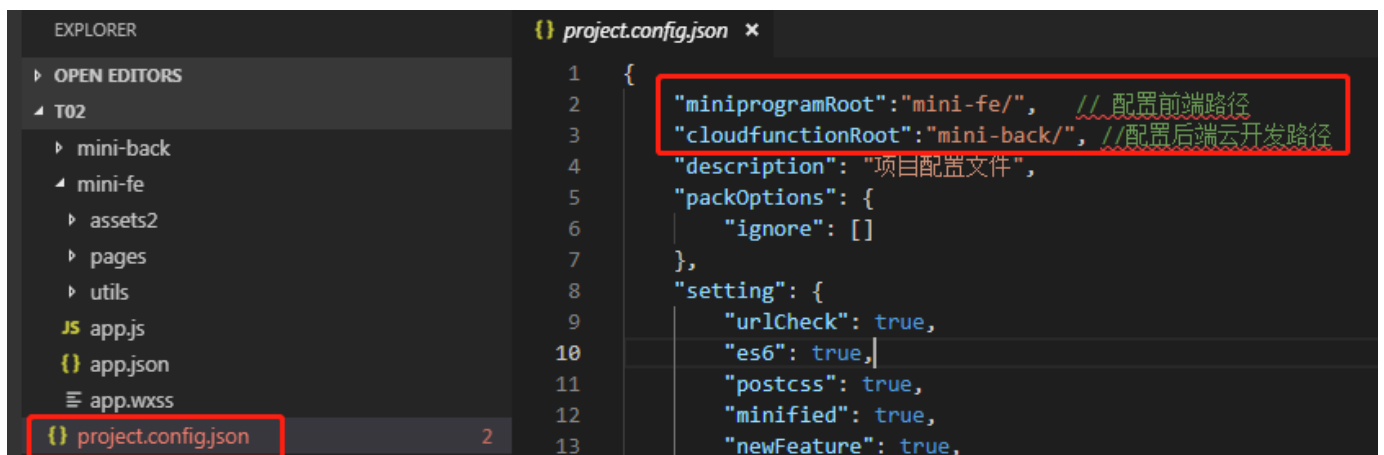
新建前端目录mini-fe和后端目录mini-back。将小程序项目所有文件移入mini-fe目录里（除了project.config.json文件外）



在project.config.json配置文件中配置前后端路径

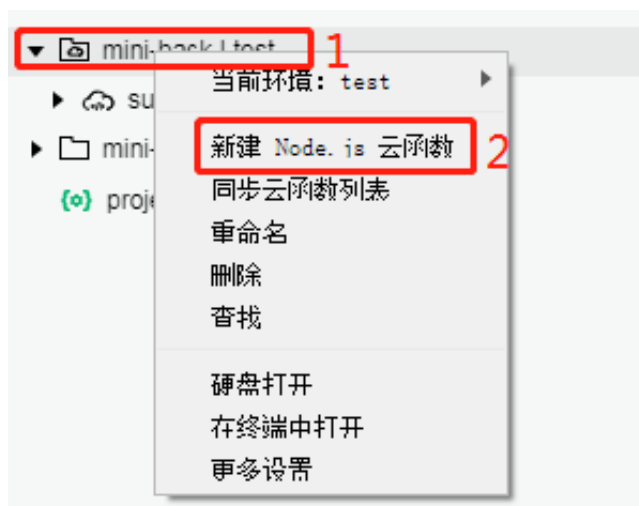
"miniprogramRoot": "mini-fe/", // 配置前端路径

"cloudfunctionRoot": "mini-back/", //配置后端云开发路径



2、新建云函数。右键后端目录 - - - -> 新建云函数 - - - -> 输入函数名会生成js和json文件。js就是用来定义函数的

定义好函数后，对函数文件执行上传并部署所有文件



3、使用云函数

使用云函数前的配置：

app.json里需要配置



```
"cloud": true,
```

app.js里需要配置



```
wx.cloud.init({
  traceUser: true //这个用户追踪可以不用
})
```

使用：在方法中调用云函数

```
index.js  x
32  addToDo() {
33      // 调用云函数
34      wx.cloud.callFunction({
35          name: 'sum',
36          data: {
37              a: 5,
38              b: 5
39          },
40          success: res => {
41              console.log(res.result)
42              wx.showToast({
43                  title: '调用成功'
44              })
45              this.setData({
46                  val: res.result.sum
47              })
48          }
49      })
50  }
```

```
wx.cloud.callFunction({ //调用云函数
  name: 'sum',           //云函数的名称
  data: {                //云函数要传入的数据
    a: 5,
    b: 5
  },
  success: res => {      //云函数返回的结果
    console.log(res.result)
  }
})
```