**课程预约系统手册**

目录

[1. 系统说明 3](#_Toc6962)

[1.1系统介绍 3](#_Toc380)

[1.2关于云开发 3](#_Toc16173)

[1.3表结构 3](#_Toc26460)

[1.4云函数与云调用 5](#_Toc29247)

[1.5云存储与云数据库 11](#_Toc4534)

[2.后台管理 13](#_Toc19180)

[2.1学科管理 14](#_Toc5156)

[2.1.1学科管理页面 14](#_Toc14463)

[2.1.2新增学科 19](#_Toc25369)

[2.1.3删除学科 22](#_Toc5438)

[2.2教师管理 25](#_Toc15907)

[2.2.1 教师管理页面 26](#_Toc17894)

[2.2.2 删除教师 32](#_Toc9724)

[2.2.3 新增教师 32](#_Toc25482)

[2.3授课管理 36](#_Toc1441)

[3客户端的实现 45](#_Toc29317)

[3.1程序配置 45](#_Toc18446)

[3.2登录 48](#_Toc29128)

[3.3我的预约 52](#_Toc14533)

[3.4课程预约 58](#_Toc3518)

# 系统说明

## 1.1系统介绍

本小程序已通过审核现已上线，微信搜索“**李老师的书店**”查看效果,小程序现为1.0.0版本，可能会出现功能升级的后续版本。

本程序采用微信小程序中的云开发，并不需要自行准备服务器。本程序有后台管理部分，暂时需要指定openid的管理人员才可登录。

## 1.2关于云开发

本程序完成采用的云开发来实现的，所以有必要对云开简单做一个介绍

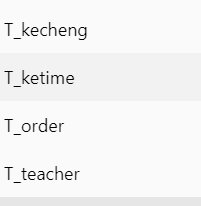
开发者可以使用云开发开发微信小程序、小游戏，无需搭建服务器，即可使用云端能力。

云开发为开发者提供完整的原生云端支持和微信服务支持，弱化后端和运维概念，无需搭建服务器，使用平台提供的 API 进行核心业务开发，即可实现快速上线和迭代，同时这一能力，同开发者已经使用的云服务相互兼容，并不互斥。

| **能力** | **作用** | **说明** |
| --- | --- | --- |
| 云函数 | 无需自建服务器 | 在云端运行的代码，微信私有协议天然鉴权，开发者只需编写自身业务逻辑代码 |
| 数据库 | 无需自建数据库 | 一个既可在小程序前端操作，也能在云函数中读写的 JSON 数据库 |
| 存储 | 无需自建存储和 CDN | 在小程序前端直接上传/下载云端文件，在云开发控制台可视化管理 |
| 云调用 | 原生微信服务集成 | 基于云函数免鉴权使用小程序开放接口的能力，包括服务端调用、获取开放数据等能力 |

## 1.3表结构

在云开发中所有东西都是基于“云”的，而不是自己的服务器写接口，所以这里的表比较特殊，并不是我们熟知的mysql，sqlserver，oracle中的表，而是基础json格式的数据，本程序中主要用到4张json表，如下图所示



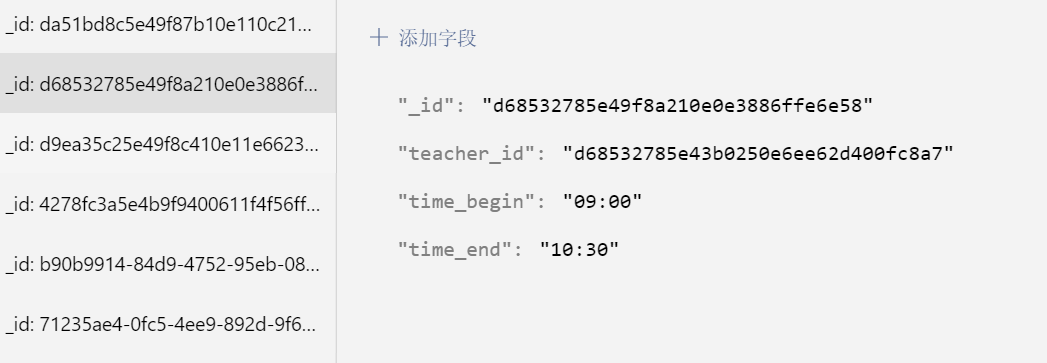
T\_kecheng：主要存储课程的相关信息



T\_teacher 主要存储教师的相关信息



T\_ketime主要存储上课时间段的相关信息



最后是T\_order主要存储 课程预约的相关信息



## 1.4云函数与云调用

云函数是一段运行在云端的代码，无需管理服务器，在开发工具内编写、一键上传部署即可运行后端代码。

小程序内提供了专门用于云函数调用的 API。开发者可以在云函数内使用 [wx-server-sdk](https://developers.weixin.qq.com/miniprogram/dev/wxcloud/guide/functions/wx-server-sdk.html) 提供的 [getWXContext](https://developers.weixin.qq.com/miniprogram/dev/wxcloud/reference-sdk-api/utils/Cloud.getWXContext.html) 方法获取到每次调用的上下文（appid、openid 等），无需维护复杂的鉴权机制，即可获取天然可信任的用户登录态（openid）。

比如我们如下定义一个云函数，命名为 add ，功能是将传入的两个参数 a 和 b 相加：

// index.js 是入口文件，云函数被调用时会执行该文件导出的 main 方法// event 包含了调用端（小程序端）调用该函数时传过来的参数，同时还包含了可以通过 getWXContext 方法获取的用户登录态 `openId` 和小程序 `appId` 信息const cloud = require('wx-server-sdk')

exports.main = (event, context) => {

let { userInfo, a, b} = event

let { OPENID, APPID } = cloud.getWXContext() // 这里获取到的 openId 和 appId 是可信的

let sum = a + b

return {

OPENID,

APPID,

sum

}}

在开发者工具中上传部署云函数后，我们在小程序中可以这么调用：

除了部署云函数进行调用外，我们还支持[云函数本地调试](https://developers.weixin.qq.com/miniprogram/dev/wxcloud/guide/functions/local-debug.html)，可以不用部署云函数即可测试

wx.cloud.callFunction({

// 需调用的云函数名

name: 'add',

// 传给云函数的参数

data: {

a: 12,

b: 19,

},

// 成功回调

complete: console.log})// 当然 promise 方式也是支持的

wx.cloud.callFunction({

name: 'add',

data: {

a: 12,

b: 19

}}).then(console.log)

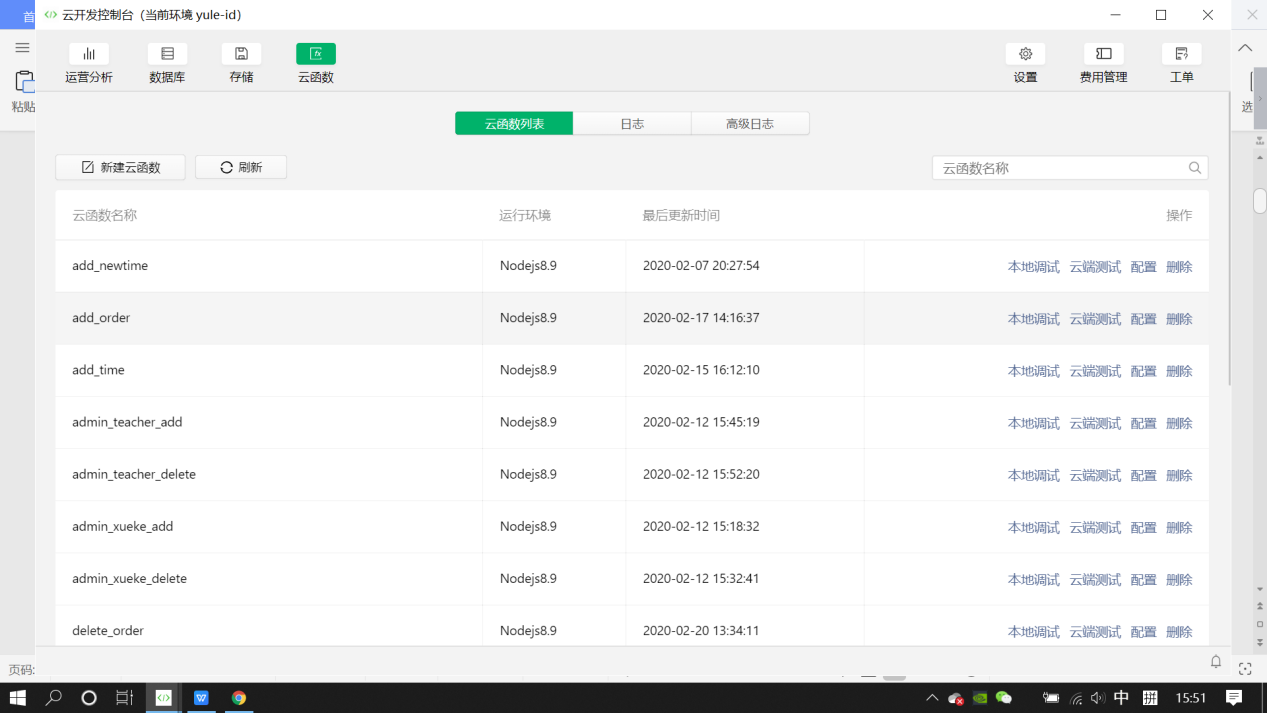
如需在云函数中操作数据库、管理云文件、调用其他云函数等操作，可使用官方提供的 npm 包 [wx-server-sdk](https://developers.weixin.qq.com/miniprogram/dev/wxcloud/reference-server-api/) 进行操作。

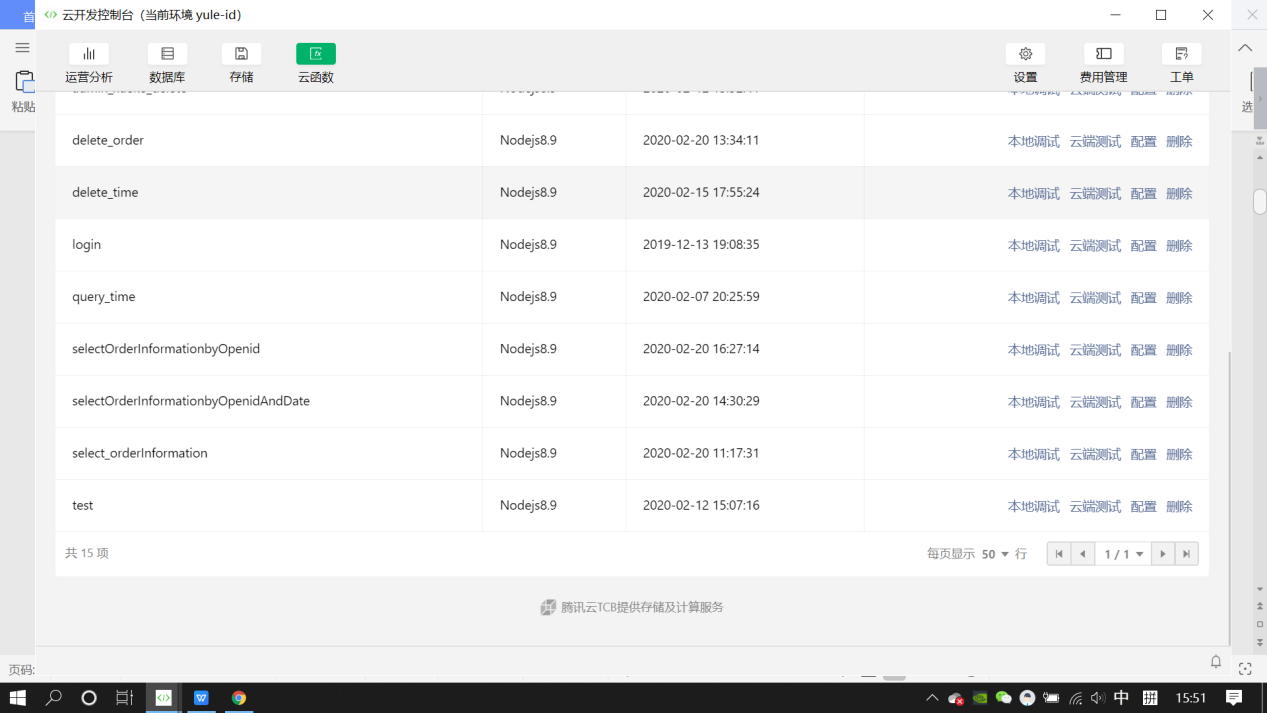
更多的云函数管理和 API，可以参考[云函数指引](https://developers.weixin.qq.com/miniprogram/dev/wxcloud/guide/functions.html)章节。

云调用

云调用是云开发提供的基于云函数使用小程序开放接口的能力，支持在云函数调用服务端开放接口，如发送模板消息、获取小程序码等操作都可以在云函数中完成

因为**多表查询不能在微信小程序端运行，所以只能通过云函数的方法调用，**本程序用到的云函数如下





代码如下

// 云函数入口文件

const cloud = require('wx-server-sdk')

cloud.init()

const db = cloud.database()

const \_ = db.command

// 云函数入口函数

exports.main = async (event, context) => {

const wxContext = cloud.getWXContext() //获取openid

await db.collection("T\_time").where({\_id:\_.neq(0)}).remove() //删除了\_id不等于0的数据，变向删除所有

await db.collection("T\_time").add({

data:{

am\_begin:event.am\_begin,

am\_end: event.am\_end,

pm\_begin: event.pm\_begin,

pm\_end: event.pm\_end,

}

})

}

// 云函数入口文件

const cloud = require('wx-server-sdk')

cloud.init()

const db = cloud.database()

// 云函数入口函数

exports.main = async (event, context) => {

const wxContext = cloud.getWXContext()

return await db.collection("T\_order").add({

data: {

ketime\_id: event.ketime\_id,

order\_date: event.order\_date,

now: event.now,

openid: event.openid,

headimage: event.headimage,

nickname: event.nickname

}

})

}

// 云函数入口文件

const cloud = require('wx-server-sdk')

cloud.init()

const db = cloud.database()

// 云函数入口函数

exports.main = async (event, context) => {

const wxContext = cloud.getWXContext()

return await db.collection("T\_teacher").add({

data: {

teacher\_name: event.teacher\_name,

kecheng\_id: event.kecheng\_id,

kecheng\_name: event.kecheng\_name

}

})

}

// 云函数入口文件

const cloud = require('wx-server-sdk')

cloud.init()

const db = cloud.database()

// 云函数入口函数

exports.main = async (event, context) => {

const wxContext = cloud.getWXContext()

return await db.collection("T\_teacher").doc(event.id).remove()

}

// 云函数入口文件

const cloud = require('wx-server-sdk')

cloud.init()

const db = cloud.database()

// 云函数入口函数

exports.main = async (event, context) => {

const wxContext = cloud.getWXContext()

return await db.collection("T\_kecheng").add({

data:{

kc\_name: event.kc\_name

}

})

}

// 云函数入口文件

const cloud = require('wx-server-sdk')

cloud.init()

const db = cloud.database()

// 云函数入口函数

exports.main = async (event, context) => {

const wxContext = cloud.getWXContext()

return await db.collection("T\_kecheng").doc(event.kc\_id).remove()

}

// 云函数入口文件

const cloud = require('wx-server-sdk')

cloud.init()

const db = cloud.database()

// 云函数入口函数

exports.main = async (event, context) => {

const wxContext = cloud.getWXContext()

return await db.collection("T\_order").doc(event.id).remove()

}

// 云函数入口文件

const cloud = require('wx-server-sdk')

cloud.init()

const db = cloud.database()

// 云函数入口函数

const \_ = db.command

const $ = \_.aggregate

exports.main = async (event, context) => {

const wxContext = cloud.getWXContext()

const date = new Date();

const cur\_year = date.getFullYear();

const cur\_month = date.getMonth() + 1;

const cur\_date = date.getDate();

const now = cur\_year + '-' + cur\_month + '-' + cur\_date;

return await db.collection('T\_order').aggregate().match({

openid: event.openid,

order\_date: \_.gte(now)

}).lookup({

from: 'T\_ketime',

localField: 'ketime\_id',

foreignField: '\_id',

as: 'list',

}).end()

}

// 云函数入口文件

//本函数的功能

//1 T\_ketime 根据教师id过滤

//2 T\_ketime 表左联 T\_order表过滤日期之后的数据

const cloud = require('wx-server-sdk')

cloud.init()

const db = cloud.database()

const \_ = db.command

const $ = \_.aggregate

// 云函数入口函数

exports.main = async(event, context) => {

const wxContext = cloud.getWXContext()

// return await db.collection('T\_ketime').aggregate().match({

// teacher\_id: event.teacher\_id,

// //order\_date: event.order\_date

// }).lookup({

// from: 'T\_order',

// localField: '\_id',

// foreignField: 'ketime\_id',

// as: 'orderlist',

// }).end()

return await db.collection('T\_ketime').aggregate().match({

teacher\_id: event.teacher\_id,

}).lookup({

from: 'T\_order',

let: {

ketime\_id: '$\_id'

},

pipeline: $.pipeline()

.match(

\_.expr($.and([

$.eq(['$ketime\_id', '$$ketime\_id']),

$.eq(['$order\_date', event.order\_date])

]))

)

.done(),

as: 'orderlist',

})

.end()

}

## 1.5云存储与云数据库

数据库

云开发提供了一个 JSON 数据库，顾名思义，数据库中的每条记录都是一个 JSON 格式的对象。一个数据库可以有多个集合（相当于关系型数据中的表），集合可看做一个 JSON 数组，数组中的每个对象就是一条记录，记录的格式是 JSON 对象。

关系型数据库和 JSON 数据库的概念对应关系如下表：

| 关系型 | 文档型 |
| --- | --- |
| 数据库 database | 数据库 database |
| 表 table | 集合 collection |
| 行 row | 记录 record / doc |
| 列 column | 字段 field |

以下是一个示例的集合数据，假设我们有一个 books 集合存放了图书记录，其中有两本书：

[

{

"\_id": "Wzh76lk5\_O\_dt0vO",

"title": "The Catcher in the Rye",

"author": "J. D. Salinger",

"characters": [

"Holden Caulfield",

"Stradlater",

"Mr. Antolini"

],

"publishInfo": {

"year": 1951,

"country": "United States"

}

},

{

"\_id": "Wzia0lk5\_O\_dt0vR",

"\_openid": "ohl4L0Rnhq7vmmbT\_DaNQa4ePaz0",

"title": "The Lady of the Camellias",

"author": "Alexandre Dumas fils",

"characters": [

"Marguerite Gautier",

"Armand Duval",

"Prudence",

"Count de Varville"

],

"publishInfo": {

"year": 1848,

"country": "France"

}

}]

在图书信息中，我们用 title, author 来记录图书标题和作者，用 characters 数组来记录书中的主要人物，用 publishInfo 来记录图书的出版信息。在其中我们可以看到，字段既可以是字符串或数字，还可以是对象或数组，就是一个 JSON 对象。

每条记录都有一个 \_id 字段用以唯一标志一条记录、一个 \_openid 字段用以标志记录的创建者，即小程序的用户。需要特别注意的是，在管理端（控制台和云函数）中创建的不会有 \_openid 字段，因为这是属于管理员创建的记录。开发者可以自定义 \_id，但不可自定义和修改 \_openid 。\_openid 是在文档创建时由系统根据小程序用户默认创建的，开发者可使用其来标识和定位文档。

数据库 API 分为小程序端和服务端两部分，小程序端 API 拥有严格的调用权限控制，开发者可在小程序内直接调用 API 进行非敏感数据的操作。对于有更高安全要求的数据，可在云函数内通过服务端 API 进行操作。云函数的环境是与客户端完全隔离的，在云函数上可以私密且安全的操作数据库。

数据库 API 包含增删改查的能力，使用 API 操作数据库只需三步：获取数据库引用、构造查询/更新条件、发出请求。以下是一个在小程序中查询数据库的发表于美国的图书记录的例子：

// 1. 获取数据库引用const db = wx.cloud.database()// 2. 构造查询语句// collection 方法获取一个集合的引用// where 方法传入一个对象，数据库返回集合中字段等于指定值的 JSON 文档。API 也支持高级的查询条件（比如大于、小于、in 等），具体见文档查看支持列表// get 方法会触发网络请求，往数据库取数据

db.collection('books').where({

publishInfo: {

country: 'United States'

}}).get({

success: function(res) {

// 输出 [{ "title": "The Catcher in the Rye", ... }]

console.log(res)

}})

更多的数据库的 API 的使用和数据库管理，可以参考[数据库指引](https://developers.weixin.qq.com/miniprogram/dev/wxcloud/guide/database.html)章节。

存储

云开发提供了一块存储空间，提供了上传文件到云端、带权限管理的云端下载能力，开发者可以在小程序端和云函数端通过 API 使用云存储功能。

在小程序端可以分别调用 wx.cloud.uploadFile 和 wx.cloud.downloadFile 完成上传和下载云文件操作。下面简单的几行代码，即可实现在小程序内让用户选择一张图片，然后上传到云端管理的功能：

// 让用户选择一张图片

wx.chooseImage({

success: chooseResult => {

// 将图片上传至云存储空间

wx.cloud.uploadFile({

// 指定上传到的云路径

cloudPath: 'my-photo.png',

// 指定要上传的文件的小程序临时文件路径

filePath: chooseResult.tempFilePaths[0],

// 成功回调

success: res => {

console.log('上传成功', res)

},

})

},})

上传完成后可在控制台中看到刚上传的图片。

更多的存储 API 和管理，可以参考[存储指引](https://developers.weixin.qq.com/miniprogram/dev/wxcloud/guide/storage.html)章节。

# 2.后台管理

本小程序有后台管理功能，后台管理主要分3的部分

1：学科管理

2：教师管理

3：授课管理



## 2.1学科管理

学科管理负责管理学科的创建和删除

### 2.1.1学科管理页面

学科管理wxml代码如下

<!--admin/xuekeguanli/xuekeguanli.wxml-->

<!-- <text>admin/xuekeguanli/xuekeguanli.wxml</text> -->

<image class="banner" src="cloud://yule-id.7975-yule-id-1259381293/admin/学科管理.jfif"></image>

<view class="tiao" bindtap="goto\_addxuekepage" style="margin-top:15rpx;">

<view class="left">

<image class="icon" src="cloud://yule-id.7975-yule-id-1259381293/icon/add48.png"></image>

</view>

<view class="center">

<text>创建新学科</text>

</view>

<view class="right">

<image class="icon" src="cloud://yule-id.7975-yule-id-1259381293/icon/right48.png"></image>

</view>

</view>

<view class="container">

<navigator open-type="redirect" class="item" url="/admin/delete\_xueke/delete\_xueke?id={{item.\_id}}&&kc\_name={{item.kc\_name}}" wx:for="{{xueke\_list}}" wx:key="\_id" >

<image src="cloud://yule-id.7975-yule-id-1259381293/icon/学科成绩.png"></image>

{{item.kc\_name}}

</navigator>

</view>

Wxss如下

/\* admin/xuekeguanli/xuekeguanli.wxss \*/

.tiao{

height:120rpx;

width:100vw;

background:white;

border:2rpx solid #F8F8F8;

line-height: 120rpx;

}

.tiao .left{

height:120rpx;

width:180rpx;

float:left;

}

.tiao .center{

height:120rpx;

width:160rpx;

float:left;

}

.tiao .right{

height:120rpx;

width:160rpx;

float:right;

}

.icon{

height:80rpx;

width:80rpx;

margin:20rpx 20rpx 20rpx 50rpx;

}

page {

background: #F8F8F8;

}

.container{

width:700rpx;

margin: 0 auto;

display: flex;

flex-flow: row wrap;

justify-content:space-between;

margin-top:20rpx;

}

.container .item{

height:160rpx;

width:150rpx;

border:10rpx solid #F8F8F8;

background: white;

display: flex;

flex-flow: column nowrap;

justify-content:space-around;

align-items:center;

flex-grow:1;

}

.container .item image{

height:80rpx;

width:80rpx;

}

Js代码如下

// admin/xuekeguanli/xuekeguanli.js

const db = wx.cloud.database();

var app = getApp();

Page({

/\*\*

\* 页面的初始数据

\*/

data: {

xueke\_list:null

},

/\*\*

\* 生命周期函数--监听页面加载

\*/

onLoad: function (options) {

var that = this

//console.log(app.globalData.userInfo.openId);

db.collection('T\_kecheng').where({

// \_openid: app.globalData.userInfo.openId // 填入当前用户 openid

}).get({

success:function(res){

that.setData({

xueke\_list:res.data

})

//console.log(res.data);

}

})

},

/\*\*

\* 生命周期函数--监听页面初次渲染完成

\*/

onReady: function () {

},

/\*\*

\* 生命周期函数--监听页面显示

\*/

onShow: function () {

this.onLoad();

},

/\*\*

\* 生命周期函数--监听页面隐藏

\*/

onHide: function () {

},

/\*\*

\* 生命周期函数--监听页面卸载

\*/

onUnload: function () {

},

/\*\*

\* 页面相关事件处理函数--监听用户下拉动作

\*/

onPullDownRefresh: function () {

},

/\*\*

\* 页面上拉触底事件的处理函数

\*/

onReachBottom: function () {

},

/\*\*

\* 用户点击右上角分享

\*/

onShareAppMessage: function () {

},

goto\_addxuekepage:function(){

// wx.navigateTo({

// url: '/admin/add\_xueke/add\_xueke'

// })

wx.redirectTo({

url: '/admin/add\_xueke/add\_xueke'

})

}

})

效果图如下  


### 2.1.2新增学科

效果如下



代码如下

<!--admin/add\_xueke/add\_xueke.wxml-->

<!-- <text>admin/add\_xueke/add\_xueke.wxml</text> -->

<view class="page-body">

<form bindsubmit="submit">

<view class="page-section">

<view class="weui-cells\_\_title">新增学科</view>

<view class="weui-cells weui-cells\_after-title">

<view class="weui-cell weui-cell\_input">

<input class="weui-input" name="kc\_name" auto-focus placeholder="输入新增的学科名称"/>

</view>

</view>

</view>

<view class="weui-cells\_\_title"></view>

<button type="primary" form-type="submit">确定 </button>

</form>

</view>

/\* admin/add\_xueke/add\_xueke.wxss \*/

@import "/lib/weui.wxss";

// admin/add\_xueke/add\_xueke.js

const db = wx.cloud.database();

Page({

/\*\*

\* 页面的初始数据

\*/

data: {

},

/\*\*

\* 生命周期函数--监听页面加载

\*/

onLoad: function (options) {

},

/\*\*

\* 生命周期函数--监听页面初次渲染完成

\*/

onReady: function () {

},

/\*\*

\* 生命周期函数--监听页面显示

\*/

onShow: function () {

},

/\*\*

\* 生命周期函数--监听页面隐藏

\*/

onHide: function () {

},

/\*\*

\* 生命周期函数--监听页面卸载

\*/

onUnload: function () {

},

/\*\*

\* 页面相关事件处理函数--监听用户下拉动作

\*/

onPullDownRefresh: function () {

},

/\*\*

\* 页面上拉触底事件的处理函数

\*/

onReachBottom: function () {

},

/\*\*

\* 用户点击右上角分享

\*/

onShareAppMessage: function () {

},

submit: function (res) {

////console.log(res.detail.value.kc\_name); //课程名

if (res.detail.value.kc\_name == "" || res.detail.value.kc\_name == null) {

wx.showToast({

title: '警告信息：请输入新增学科名称',

icon: "none",

duration: 2000

})

} else {

wx.cloud.callFunction({

name: 'admin\_xueke\_add', //云函数的名称

data: {

'kc\_name': res.detail.value.kc\_name //传入云函数event中

},

success: res => {

//console.log(res) //res的数据结构如下图

wx.redirectTo({

url: '/admin/xuekeguanli/xuekeguanli',

success: function () {

wx.showToast({

title: '成功',

icon: 'success',

duration: 1200,

})

}

})

},

fail: err => {

console.error('[云函数] [loginInfo] 调用失败', err)

}

})

}

}

})

### 2.1.3删除学科

负责学科的删除



代码如下

<!--admin/delete\_xueke/delete\_xueke.wxml-->

<!-- <text>admin/delete\_xueke/delete\_xueke.wxml</text> -->

<view class="page-body">

<view class="page-section">

<view class="weui-cells\_\_title">{{kc\_name}}</view>

<view class="weui-cells weui-cells\_after-title">

</view>

</view>

<view class="weui-cells\_\_title"></view>

<button type="primary" bindtap="delete\_xueke">删除 </button>

</view>

/\* admin/delete\_xueke/delete\_xueke.wxss \*/

@import "/lib/weui.wxss";

// admin/delete\_xueke/delete\_xueke.js

const db = wx.cloud.database();

Page({

/\*\*

\* 页面的初始数据

\*/

data: {

id: "",

kc\_name: ""

},

/\*\*

\* 生命周期函数--监听页面加载

\*/

onLoad: function (options) {

//console.log(options.kc\_name); //要删除的学科的名称

this.setData({

id: options.id,

kc\_name: options.kc\_name

})

},

/\*\*

\* 生命周期函数--监听页面初次渲染完成

\*/

onReady: function () {

},

/\*\*

\* 生命周期函数--监听页面显示

\*/

onShow: function () {

},

/\*\*

\* 生命周期函数--监听页面隐藏

\*/

onHide: function () {

},

/\*\*

\* 生命周期函数--监听页面卸载

\*/

onUnload: function () {

},

/\*\*

\* 页面相关事件处理函数--监听用户下拉动作

\*/

onPullDownRefresh: function () {

},

/\*\*

\* 页面上拉触底事件的处理函数

\*/

onReachBottom: function () {

},

/\*\*

\* 用户点击右上角分享

\*/

onShareAppMessage: function () {

},

delete\_xueke: function () {

var that = this;

wx.showModal({

title: '提示',

content: '你确定删除' + that.data.kc\_name + "吗？",

success(res) {

if (res.confirm) {

wx.cloud.callFunction({

name: 'admin\_xueke\_delete', //云函数的名称

data: {

'kc\_id': that.data.id //传入云函数event中

},

success: res => {

//console.log(res) //res的数据结构如下图

wx.redirectTo({

url: '/admin/xuekeguanli/xuekeguanli',

success: function () {

wx.showToast({

title: '成功',

icon: 'success',

duration: 1200,

})

}

})

},

fail: err => {

console.error('[云函数] [loginInfo] 调用失败', err)

}

})

} else if (res.cancel) { //用户点击取消

}

}

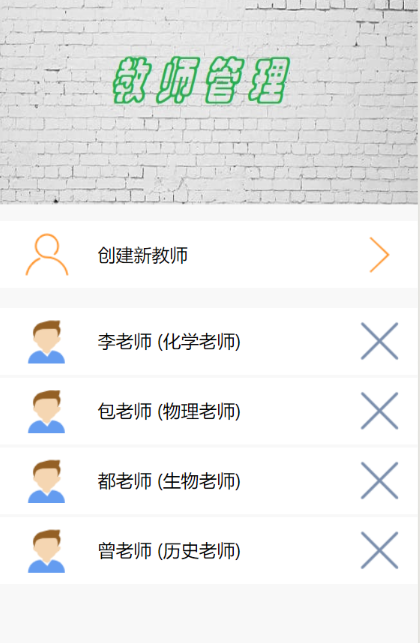
})

}

})

## 2.2教师管理

教师管理负责教师信息的基本维护，包括新增教师和删除教师以及教师信息查看，实现效果如下



### 2.2.1 教师管理页面

主要代码如下

<!--admin/jiaoshiguanli/jiaoshiguanli.wxml-->

<!-- <text>admin/jiaoshiguanli/jiaoshiguanli.wxml</text> -->

<image class="banner" src="cloud://yule-id.7975-yule-id-1259381293/admin/教师管理.jfif"></image>

<view class="tiao" bindtap="goto\_addteacher" style="margin-top:15rpx;margin-bottom:30rpx;">

<view class="left">

<image class="icon" src="cloud://yule-id.7975-yule-id-1259381293/icon/teacher48.png"></image>

</view>

<view class="center">

<text>创建新教师</text>

</view>

<view class="right">

<image class="icon" src="cloud://yule-id.7975-yule-id-1259381293/icon/right48.png"></image>

</view>

</view>

<!-- <view class="container">

<navigator open-type="redirect" class="item" url="/admin/delete\_xueke/delete\_xueke?id={{item.\_id}}&&kc\_name={{item.kc\_name}}" wx:for="{{xueke\_list}}" wx:key="\_id" >

<image src="cloud://yule-id.7975-yule-id-1259381293/icon/学科成绩.png"></image>

{{item.kc\_name}}

</navigator>

</view> -->

<view class="tiao" wx:for="{{teacher\_list}}" wx:key="\_id" >

<view class="left">

<image class="icon" src="cloud://yule-id.7975-yule-id-1259381293/icon/teacher\_color.png"></image>

</view>

<view class="center">

<text>{{item.teacher\_name}} ({{item.kecheng\_name}}老师)</text>

</view>

<view class="right" bindtap="delete\_teacher" data-id="{{item.\_id}}" data-teacher="{{item.teacher\_name}}">

<image class="icon" src="cloud://yule-id.7975-yule-id-1259381293/icon/delete1.png"></image>

</view>

</view>

/\* admin/jiaoshiguanli/jiaoshiguanli.wxss \*/

/\* admin/xuekeguanli/xuekeguanli.wxss \*/

.tiao{

height:120rpx;

width:100vw;

background:white;

border:2rpx solid #F8F8F8;

line-height: 120rpx;

}

.tiao .left{

height:120rpx;

width:180rpx;

float:left;

}

.tiao .center{

height:120rpx;

width:400rpx;

float:left;

}

.tiao .right{

height:120rpx;

width:160rpx;

float:right;

}

.icon{

height:80rpx;

width:80rpx;

margin:20rpx 20rpx 20rpx 50rpx;

}

page {

background: #F8F8F8;

}

.container{

width:700rpx;

margin: 0 auto;

display: flex;

flex-flow: row wrap;

justify-content:space-between;

margin-top:20rpx;

}

.container .item{

height:160rpx;

width:150rpx;

border:10rpx solid #F8F8F8;

background: white;

display: flex;

flex-flow: column nowrap;

justify-content:space-around;

align-items:center;

flex-grow:1;

}

/\* .container .item image{

height:80rpx;

width:80rpx;

} \*/

// admin/jiaoshiguanli/jiaoshiguanli.js

const db = wx.cloud.database();

Page({

/\*\*

\* 页面的初始数据

\*/

data: {

teacher\_list:null

},

/\*\*

\* 生命周期函数--监听页面加载

\*/

onLoad: function (options) {

var that = this;

db.collection("T\_teacher").where({}).get({

success:function(res){

//console.log(res.data);

that.setData({

teacher\_list:res.data

})

}

})

////////////////////////////////////////////////////////

// wx.cloud.callFunction({

// name: 'get\_teacher',

// data: {

// a: 12,

// b: 19

// }

// }).then(console.log)

// db.collection('T\_teacher').aggregate().lookup({

// from:'T\_kecheng',

// localField:'kecheng\_id',

// foreignField: '\_id',

// as:'teacherList'

// }).end().then(res => console.log(res)).catch(err => console.error(err))

// db.collection('orders').aggregate()

// .lookup({

// from: 'T\_kecheng',

// localField: 'kecheng\_id',

// foreignField: '\_id',

// as: 'teacherList',

// })

// .end()

// .then(res => console.log(res))

// .catch(err => console.error(err))

// varres = await db.collection('T\_teacher')

// .aggregate()

// .lookup({

// from: 'T\_kecheng',

// localField: 'kecheng\_id',

// foreignField: '\_id',

// as: 'teacherList',

// })

// .end();

},

/\*\*

\* 生命周期函数--监听页面初次渲染完成

\*/

onReady: function () {

},

/\*\*

\* 生命周期函数--监听页面显示

\*/

onShow: function () {

this.onLoad();

},

/\*\*

\* 生命周期函数--监听页面隐藏

\*/

onHide: function () {

},

/\*\*

\* 生命周期函数--监听页面卸载

\*/

onUnload: function () {

},

/\*\*

\* 页面相关事件处理函数--监听用户下拉动作

\*/

onPullDownRefresh: function () {

},

/\*\*

\* 页面上拉触底事件的处理函数

\*/

onReachBottom: function () {

},

/\*\*

\* 用户点击右上角分享

\*/

onShareAppMessage: function () {

},

goto\_addteacher:function(){

wx.redirectTo({

url: '/admin/add\_teacher/add\_teacher'

})

},

delete\_teacher:function(res){

//console.log(res.currentTarget.dataset.id);

//console.log(res.currentTarget.dataset.teacher);

var id = res.currentTarget.dataset.id;

var teacher = res.currentTarget.dataset.teacher;

var that = this;

wx.showModal({

title: '提示',

content: '你确定删除' + teacher + "吗？",

success(res) {

if (res.confirm) { //用户点击确定

wx.cloud.callFunction({

name: 'admin\_teacher\_delete', //云函数的名称

data: {

id:id

},

success: res => {

that.onLoad();

},

fail: err => {

console.error('[云函数] [loginInfo] 调用失败', err)

}

})

} else if (res.cancel) { //用户点击取消

}

}

})

}

})

### 2.2.2 删除教师



教师的删除点击教师循环列表右侧的小叉即可完成代码已在2.2.1中

### 2.2.3 新增教师

新增教师部分主要完成教师的新增操作效果如下图





代码如下

<!--admin/add\_teacher/add\_teacher.wxml-->

<!-- <text>admin/add\_teacher/add\_teacher.wxml</text> -->

<view class="page-body">

<form bindsubmit="submit">

<view class="page-section">

<view class="weui-cells\_\_title">新增教师（最多5个字）</view>

<view class="weui-cells weui-cells\_after-title">

<view class="weui-cell weui-cell\_input">

<input maxlength="5" class="weui-input" name="teacher\_name" auto-focus placeholder="输入新增的教师姓名"/>

</view>

</view>

</view>

<view class="page-section">

<view class="weui-cells\_\_title">选择教师教授学科</view>

<view class="weui-cells weui-cells\_after-title">

<view class="weui-cell weui-cell\_input"><picker></picker>

<!-- <input class="weui-input" name="teacher\_name" auto-focus placeholder="输入新增的教师姓名"/> -->

<picker class="weui-input" name="xueke" bindchange="picker\_change" range-key="kc\_name" mode="selector" range="{{list}}">当前选择学科:{{xueke}}</picker>

</view>

</view>

</view>

<button type="primary" form-type="submit">确定 </button>

</form>

</view>

/\* admin/add\_teacher/add\_teacher.wxss \*/

@import "/lib/weui.wxss";

// admin/add\_teacher/add\_teacher.js

const db = wx.cloud.database();

Page({

/\*\*

\* 页面的初始数据

\*/

data: {

list:[], //picker选择学科的数组

xueke:""

},

/\*\*

\* 生命周期函数--监听页面加载

\*/

onLoad: function (options) {

var that = this;

db.collection("T\_kecheng").where({

}).get({

success:function(res){

// console.log(res.data);

that.setData({

list: res.data

})

}

})

},

/\*\*

\* 生命周期函数--监听页面初次渲染完成

\*/

onReady: function () {

},

/\*\*

\* 生命周期函数--监听页面显示

\*/

onShow: function () {

},

/\*\*

\* 生命周期函数--监听页面隐藏

\*/

onHide: function () {

},

/\*\*

\* 生命周期函数--监听页面卸载

\*/

onUnload: function () {

},

/\*\*

\* 页面相关事件处理函数--监听用户下拉动作

\*/

onPullDownRefresh: function () {

},

/\*\*

\* 页面上拉触底事件的处理函数

\*/

onReachBottom: function () {

},

/\*\*

\* 用户点击右上角分享

\*/

onShareAppMessage: function () {

},

picker\_change:function(res){

//console.log(res.detail.value); //下标

this.setData({

xueke: this.data.list[res.detail.value].kc\_name

})

},

submit:function(res){

var that = this;

var teacher = res.detail.value.teacher\_name;

var index = res.detail.value.xueke;

if (index == "" || index == null || teacher == "" || teacher == null){

wx.showToast({

title: '警告信息：请输入教师姓名和选择学科',

icon:"none",

duration:2000

})

}else{

var xueke\_id = that.data.list[index].\_id;

wx.cloud.callFunction({

name: 'admin\_teacher\_add', //云函数的名称

data: {

teacher\_name: teacher,

kecheng\_id: xueke\_id,

kecheng\_name: that.data.list[index].kc\_name

},

success: res => {

wx.redirectTo({

url: '/admin/jiaoshiguanli/jiaoshiguanli',

success: function () {

wx.showToast({

title: '成功',

icon: 'success',

duration: 1200,

})

}

})

},

fail: err => {

console.error('[云函数] [loginInfo] 调用失败', err)

}

})

}

}

})

## 2.3授课管理

在教师信息与课程信息维护完成之后才可使用授课管理模块，此模块主要功能是根据教师所教授的课程分配上课时间，效果图如下



主要代码如下

<!--admin/time\_guanli\_new/time\_guanli\_new.wxml-->

<!-- <text>admin/time\_guanli\_new/time\_guanli\_new.wxml</text> -->

<image class="banner" src="cloud://yule-id.7975-yule-id-1259381293/admin/banner1.jpg"></image>

<view class="page-body">

<block wx:if="{{time\_list==null}}">

<view style="color:red;text-align:center;">请选择教师</view>

</block>

<block wx:elif="{{time\_list.length>0}}">

<view class="tiao" wx:for="{{time\_list}}" wx:key="\_id">

<view class="left">

<image class="icon" src="cloud://yule-id.7975-yule-id-1259381293/icon/time48.png"></image>

</view>

<view class="center">

<text>课程时长: {{item.time\_begin}}---{{item.time\_end}}</text>

</view>

<view class="right" bindtap="delete\_time" data-id="{{item.\_id}}">

<image class="icon" src="cloud://yule-id.7975-yule-id-1259381293/icon/delete1.png"></image>

</view>

</view>

</block>

<block wx:else>

<view style="color:red;text-align:center;">当前老师没有课程安排</view>

</block>

<form bindsubmit="submit">

<view class="page-section">

<view class="weui-cells\_\_title">授课教师</view>

<view class="weui-cells weui-cells\_after-title">

<view class="weui-cell weui-cell\_input">

<picker name="teacher\_id" value="{{teacher\_list[teacher\_index].\_id}}" class="weui-input" range="{{teacher\_list}}" range-key="teacher\_name" bindchange="teacher\_list\_change">教师姓名:{{teacher\_list[teacher\_index].teacher\_name}}</picker>

</view>

</view>

</view>

<view class="time\_kuang {{showView?'show':'hide'}}">

<view class="page-section">

<view class="weui-cells\_\_title">开始时间:</view>

<view class="weui-cells weui-cells\_after-title">

<view class="weui-cell weui-cell\_input">

<picker name="begin\_time" class="weui-input" value="{{time\_begin}}" name="time\_begin" bindchange="time\_begin\_change" mode="time">开始:{{time\_begin}}</picker>

</view>

</view>

</view>

<view class="page-section">

<view class="weui-cells\_\_title">结束时间:</view>

<view class="weui-cells weui-cells\_after-title">

<view class="weui-cell weui-cell\_input">

<picker name="end\_time" class="weui-input" value="{{time\_end}}" name="time\_end" bindchange="time\_end\_change" mode="time">结束:{{time\_end}}</picker>

</view>

</view>

</view>

<view class="page-section">

<button size="mini" form-type="submit" style="margin-top:20rpx;">添加</button>

<button size="mini" bindtap="reset" style="margin-top:20rpx;">取消</button>

</view>

</view>

<button style="margin-top:20rpx;" class="{{showButton?'show':'hide'}}" type="primary" bindtap="add" disabled="{{teacher\_index!=''?false:true}}">添加课程</button>

</form>

</view>

Wxss如下

/\* admin/time\_guanli\_new/time\_guanli\_new.wxss \*/

@import "/lib/weui.wxss";

.show{

display:block;

}

.hide{

display:none;

}

.time\_kuang{

border:0rpx solid gray;

width:700rpx;

margin:0 auto;

margin-top:20rpx;

border-radius:10rpx;

}

.tiao{

height:120rpx;

width:100vw;

background:white;

border:2rpx solid #F8F8F8;

line-height: 120rpx;

}

.tiao .left{

height:120rpx;

width:180rpx;

float:left;

}

.tiao .center{

height:120rpx;

width:400rpx;

float:left;

border:0px solid red;

color:green;

}

.tiao .right{

height:120rpx;

width:160rpx;

float:right;

}

.icon{

height:80rpx;

width:80rpx;

margin:20rpx 20rpx 20rpx 50rpx;

}

Js部分如下

// admin/time\_guanli\_new/time\_guanli\_new.js

const db = wx.cloud.database();

Page({

/\*\*

\* 页面的初始数据

\*/

data: {

teacher\_list: null,

showButton: true,

showView: false,

teacher\_index: "",

time\_begin: "",

time\_end: "",

time\_list: null

},

reset: function () {

this.setData({

time\_begin: "",

time\_end: "",

showButton: true,

showView: false,

})

},

/\*\*

\* 生命周期函数--监听页面加载

\*/

onLoad: function (options) {

var that = this;

db.collection("T\_teacher").get({

success: function (res) {

//console.log(res.data);

that.setData({

teacher\_list: res.data

})

}

});

// db.collection("T\_ketime").aggregate().sort({

// time\_begin:1 //1代表升序

// }).end().then(res=>that.setData({

// time\_list:res.list

// }));

},

/\*\*

\* 生命周期函数--监听页面初次渲染完成

\*/

onReady: function () {

},

/\*\*

\* 生命周期函数--监听页面显示

\*/

onShow: function () {

},

/\*\*

\* 生命周期函数--监听页面隐藏

\*/

onHide: function () {

},

/\*\*

\* 生命周期函数--监听页面卸载

\*/

onUnload: function () {

},

/\*\*

\* 页面相关事件处理函数--监听用户下拉动作

\*/

onPullDownRefresh: function () {

},

/\*\*

\* 页面上拉触底事件的处理函数

\*/

onReachBottom: function () {

},

/\*\*

\* 用户点击右上角分享

\*/

onShareAppMessage: function () {

},

teacher\_list\_change: function (res) {

//console.log(res.detail.value); //teacher\_list 下标

},

add: function () {

this.setData({

showView: true,

showButton: false

})

},

teacher\_list\_change: function (res) {

//console.log(res.detail.value); 下标

var that = this;

this.setData({

teacher\_index: res.detail.value

});

var teacher\_id = this.data.teacher\_list[res.detail.value].\_id;

//console.log(teacher\_id);

db.collection("T\_ketime").aggregate().match({

teacher\_id: teacher\_id

}).sort({

time\_begin: 1 //1代表升序

}).end({

success: function (res) {

//console.log(res.list);

that.setData({

time\_list: res.list

})

}

})

},

delete\_time: function (res) {

//console.log(res.currentTarget.dataset.id);

var id = res.currentTarget.dataset.id;

var that = this;

wx.showModal({

title: '提示',

content: '确定删除吗？',

success(res) {

if (res.confirm) {

wx.cloud.callFunction({

name: 'delete\_time', //云函数的名称

data: {

id: id

},

success: res => {

//////////////(删除之后重新做一次查询，相当于刷新)重新做一个time\_list的查询并赋值//////////

var teacher\_id = that.data.teacher\_list[that.data.teacher\_index].\_id;

//console.log(teacher\_id);

db.collection("T\_ketime").aggregate().match({

teacher\_id: teacher\_id

}).sort({

time\_begin: 1 //1代表升序

}).end({

success: function (res) {

//console.log(res.list);

that.setData({

time\_list: res.list

})

}

})

///////////////////////////////////////////////////////////////////////////////////

},

fail: err => {

console.error('[云函数] [loginInfo] 调用失败', err)

}

})

} else if (res.cancel) { //用户点击取消

}

}

})

},

time\_end\_change: function (res) {

console.log(res.detail.value);

this.setData({

time\_end: res.detail.value

})

},

time\_begin\_change: function (res) {

console.log(res.detail.value);

this.setData({

time\_begin: res.detail.value

})

},

submit: function (res) {

var that = this;

// console.log(res.detail.value.teacher\_id);

// console.log(res.detail.value.time\_begin);

// console.log(res.detail.value.time\_end);

if (res.detail.value.time\_end <= res.detail.value.time\_begin) {

wx.showToast({

title: '结束时间不能小于开始时间',

icon: 'none',

duration: 1500

})

} else {

wx.cloud.callFunction({

name: 'add\_time', //云函数的名称

data: {

teacher\_id: res.detail.value.teacher\_id,

time\_begin: res.detail.value.time\_begin,

time\_end: res.detail.value.time\_end

},

success: res => {

wx.showToast({

title: '成功',

icon: 'success',

duration: 1200,

success: function () {

that.setData({

time\_begin: "",

time\_end: "",

showButton: true,

showView: false,

});

//////////////////////////////////////////////////////////////////////

var teacher\_id = that.data.teacher\_list[that.data.teacher\_index].\_id;

//console.log(teacher\_id);

db.collection("T\_ketime").aggregate().match({

teacher\_id: teacher\_id

}).sort({

time\_begin: 1 //1代表升序

}).end({

success: function (res) {

//console.log(res.list);

that.setData({

time\_list: res.list

})

}

})

////////////////////////////////////////////////////////////////////////

}

});

// wx.redirectTo({

// url: '/admin/jiaoshiguanli/jiaoshiguanli',

// success: function () {

// wx.showToast({

// title: '成功',

// icon: 'success',

// duration: 1200,

// })

// }

// })

},

fail: err => {

console.error('[云函数] [loginInfo] 调用失败', err)

}

})

that.setData({

showView: false,

showButton: true

})

}

}

})

# 3客户端的实现

## 3.1程序配置

我们在后台将基础信息文虎好之后，就可以使用客户端了，整个客户端执行效果如下图



整个程序的配置在app.Json中如下所示

{

"pages": [

"new\_pages/login/login",

"admin/main/main",

"admin/time\_guanli\_new/time\_guanli\_new",

"new\_pages/mine/mine",

"new\_pages/shipin/shipin",

"admin/time\_guanli/time\_guanli",

"admin/jiaoshiguanli/jiaoshiguanli",

"admin/add\_teacher/add\_teacher",

"pages/index/index",

"admin/xuekeguanli/xuekeguanli",

"admin/add\_xueke/add\_xueke",

"admin/delete\_xueke/delete\_xueke",

"pages/userConsole/userConsole",

"pages/storageConsole/storageConsole",

"pages/databaseGuide/databaseGuide",

"pages/addFunction/addFunction",

"pages/deployFunctions/deployFunctions",

"pages/chooseLib/chooseLib",

"pages/openapi/openapi",

"pages/login/login",

"new\_pages/main/main",

"new\_pages/tiku/tiku"

],

"window": {

"backgroundColor": "#F6F6F6",

"backgroundTextStyle": "light",

"navigationBarBackgroundColor": "#33CC66",

"navigationBarTitleText": "课程预约系统",

"navigationBarTextStyle": "white"

},

"sitemapLocation": "sitemap.json",

"tabBar": {

"borderStyle": "white",

"backgroundColor": "#33CC66",

"selectedColor": "#00FFFF",

"color": "#FFFFFF",

"list": [

{

"pagePath": "new\_pages/main/main",

"iconPath": "tab/book\_normal.png",

"selectedIconPath": "tab/book\_select.png",

"text": "预约课程"

},

{

"pagePath": "new\_pages/tiku/tiku",

"iconPath": "tab/order\_normal.png",

"selectedIconPath": "tab/order\_select.png",

"text": "精选题库"

},

{

"pagePath": "new\_pages/shipin/shipin",

"iconPath": "tab/store\_normal.png",

"selectedIconPath": "tab/store\_select.png",

"text": "视频资源"

},

{

"pagePath": "new\_pages/mine/mine",

"iconPath": "tab/person\_normal.png",

"selectedIconPath": "tab/person\_select.png",

"text": "我的预约"

}

]

}

}

全局的一些数据如下

//app.js

App({

onLaunch: function () {

if (!wx.cloud) {

console.error('请使用 2.2.3 或以上的基础库以使用云能力')

} else {

wx.cloud.init({

traceUser: true,

})

}

this.globalData = {

userInfo:{

headImage:null,

nickName:null,

openId:null

}

}

}

})

## 3.2登录

首先是登录功能，登录这里采用的是授权自动登录

<!--new\_pages/login/login.wxml-->

<image src="cloud://yule-id.7975-yule-id-1259381293/login.png"></image>

<!-- <button class='bt\_go' open-type="getUserInfo" bindgetuserinfo="getUserInfo" hidden='{{isTrue}}'>立即体验</button> -->

<button class="button" open-type="getUserInfo" bindgetuserinfo="getUserInfo" >点击按钮开始使用</button>

<button class="admin" open-type="getUserInfo" bindgetuserinfo="goto\_admin">管理员入口</button>

/\* new\_pages/login/login.wxss \*/

image{

width:100%;

height:100vh;

}

.button{

background:#33CC66;

color:#fff;

position: absolute;

bottom: 48vh;

width:50vw;

left:25vw;

border-radius: 20rpx;

border:2rpx solid white;

}

.admin{

position: absolute;

z-index: 200;

color:white;

background:#33CC66;

top:1vh;

left:1vw;

border-radius: 20rpx;

border:2rpx solid white;

}

// new\_pages/login/login.js

var app = getApp();

Page({

/\*\*

\* 页面的初始数据

\*/

data: {

},

/\*\*

\* 生命周期函数--监听页面加载

\*/

onLoad: function (options) {

},

/\*\*

\* 生命周期函数--监听页面初次渲染完成

\*/

onReady: function () {

},

/\*\*

\* 生命周期函数--监听页面显示

\*/

onShow: function () {

},

/\*\*

\* 生命周期函数--监听页面隐藏

\*/

onHide: function () {

},

/\*\*

\* 生命周期函数--监听页面卸载

\*/

onUnload: function () {

},

/\*\*

\* 页面相关事件处理函数--监听用户下拉动作

\*/

onPullDownRefresh: function () {

},

/\*\*

\* 页面上拉触底事件的处理函数

\*/

onReachBottom: function () {

},

/\*\*

\* 用户点击右上角分享

\*/

onShareAppMessage: function () {

},

getUserInfo:function(e){

wx.cloud.callFunction({

// 需调用的云函数名

name: 'login',

// 传给云函数的参数

// 成功回调

complete: function(res){

wx.hideNavigationBarLoading();

app.globalData.userInfo.headImage = e.detail.userInfo.avatarUrl;

app.globalData.userInfo.nickName = e.detail.userInfo.nickName;

//console.log(res.result.openid);

app.globalData.userInfo.openId = res.result.openid;

wx.switchTab({

url: '/new\_pages/mine/mine',

success: function (res) {

wx.showToast({

title: '登陆成功',

icon: 'success',

duration: 1000

})

}

})

}

})

},

goto\_admin:function(e){

wx.cloud.callFunction({

// 需调用的云函数名

name: 'login',

// 传给云函数的参数

// 成功回调

complete: function (res) {

wx.hideNavigationBarLoading();

if (res.result.openid == "oWtwk0TwpB1-m2YVt5YAPHz0Qdao" || res.result.openid =="oWtwk0e6SbObM-B2a4PJuzEEPjfM")

{

app.globalData.userInfo.headImage = e.detail.userInfo.avatarUrl;

app.globalData.userInfo.nickName = e.detail.userInfo.nickName;

// console.log(res.result.openid);

app.globalData.userInfo.openId = res.result.openid;

wx.redirectTo({

url: '/admin/main/main',

})

}else{

wx.showToast({

title: '对不起，您不是管理员',

icon: 'none',

duration: 1500

})

}

}

})

}

})

## 3.3我的预约



<!--new\_pages/mine/mine.wxml-->

<!-- <text>new\_pages/mine/mine.wxml</text> -->

<view class="top">

<image class="head" src="cloud://yule-id.7975-yule-id-1259381293/mine/mine{{suiji}}.jpg"></image>

<image class="headimage" src="{{userInfo.headImage}}"></image>

<view class="name"></view>

<view class="out" bindtap="out">退出登录</view>

</view>

<block wx:if="{{list==null}}">

<view class="notice">您还没有预约任何课程</view>

</block>

<block wx:else>

<view class="notice">只显示未来7天的预约课程情况</view>

<block wx:for="{{list}}" wx:key="nothing">

<view class="tiao" data-teacherid="{{item.list[0].teacher\_id}}" bindtap="xiangqing">

<view class="left">

<image class="icon" src="cloud://yule-id.7975-yule-id-1259381293/icon/teacher48.png"></image>

</view>

<view class="center">

<text>{{item.order\_date}} {{item.list[0].time\_begin}}-{{item.list[0].time\_end}}</text>

</view>

<view class="right">

<image class="icon" src="cloud://yule-id.7975-yule-id-1259381293/icon/right48.png"></image>

</view>

</view>

</block>

</block>

/\* new\_pages/mine/mine.wxss \*/

.top{

border:0px solid black;

height:37vh;

width:100vw;

position:relative;

margin-bottom:30rpx;

}

.head{

width:100vw;

height:30vh;

position: absolute;

}

.headimage{

position: absolute;

left:100rpx;

top:23vh;

height:14vh;

width:14vh;

border-radius: 7vh;

border:5rpx solid white;

}

.name{

width:400rpx;

border:0px solid red;

position: absolute;

left:300rpx;

z-index: 100;

color:white;

font-weight:bolder;

font-size: 45rpx;

top:25vh;

}

.out{

background: lightgray;

position: absolute;

left:300rpx;

color:white;

font-weight:bolder;

font-size: 30rpx;

top:31vh;

width:20vw;

border-radius: 15rpx;

text-align: center;

}

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*这是一条华丽的分给线\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

/\* admin/main/main.wxss \*/

page {

background: #F8F8F8;

}

.tiao{

height:120rpx;

width:100vw;

background:white;

border:2rpx solid #F8F8F8;

line-height: 120rpx;

}

.tiao .left{

height:120rpx;

width:180rpx;

float:left;

}

.tiao .center{

height:120rpx;

float:left;

font-size:30rpx;

}

.tiao .right{

height:120rpx;

width:160rpx;

float:right;

}

.icon{

height:80rpx;

width:80rpx;

margin:20rpx 20rpx 20rpx 50rpx;

}

.notice{

text-align: center;

color:lightgreen;

font-size: 30rpx;

margin:10rpx;

}

// new\_pages/mine/mine.js

var app = getApp();

var db = wx.cloud.database();

Page({

/\*\*

\* 页面的初始数据

\*/

data: {

suiji:2,

userInfo:null,

list:null,

teacher:""

},

/\*\*

\* 生命周期函数--监听页面加载

\*/

onLoad: function (options) {

var that = this;

that.setData({

userInfo: app.globalData.userInfo

})

},

/\*\*

\* 生命周期函数--监听页面初次渲染完成

\*/

onReady: function () {

},

xiangqing:function(res){

//console.log(res.currentTarget.dataset.teacherid);

var teacherid = res.currentTarget.dataset.teacherid;

db.collection("T\_teacher").where({

\_id: teacherid

}).get().then(res => {

//console.log(res.data[0].teacher\_name)

wx.showModal({

title: '课程信息',

content: res.data[0].kecheng\_name + " " +res.data[0].teacher\_name,

showCancel: false

})

})

},

/\*\*

\* 生命周期函数--监听页面显示

\*/

onShow: function () {

var that = this;

this.setData({

suiji: Math.floor(Math.random() \* 2) + 1

})

wx.cloud.callFunction({

name: 'selectOrderInformationbyOpenid', //云函数的名称

data: {

openid: that.data.userInfo.openId

},

success: res => {

//console.log(res.result.list)

that.setData({

list: res.result.list

})

},

fail: err => {

console.error('[云函数] [loginInfo] 调用失败', err)

}

})

},

/\*\*

\* 生命周期函数--监听页面隐藏

\*/

onHide: function () {

},

/\*\*

\* 生命周期函数--监听页面卸载

\*/

onUnload: function () {

},

/\*\*

\* 页面相关事件处理函数--监听用户下拉动作

\*/

onPullDownRefresh: function () {

},

/\*\*

\* 页面上拉触底事件的处理函数

\*/

onReachBottom: function () {

},

/\*\*

\* 用户点击右上角分享

\*/

onShareAppMessage: function () {

},

out:function(){ //退出登录返回登录页

app.globalData.userInfo.headImage = null;

app.globalData.userInfo.nickName = null;

app.globalData.userInfo.openId = null;

wx.redirectTo({

url: '/new\_pages/login/login',

success: function(res) {},

fail: function(res) {

console.log(res);

},

complete: function(res) {},

})

}

})

## 3.4课程预约

课程预约模块完成课程预约功能，首先选择教师，根据教师去查询教师的课程，然后实现逻辑



代码如下

<!--miniprogram/new\_pages/main/main.wxml-->

<!-- <text>miniprogram/new\_pages/main/main.wxml</text> -->

<!-- {{userInfo.openId}}

{{userInfo.headImage}}

{{userInfo.nickName}} -->

<view class="page-body">

<view class="page-section">

<view class="weui-cells\_\_title">授课教师</view>

<view class="weui-cells weui-cells\_after-title">

<view class="weui-cell weui-cell\_input">

<picker name="teacher\_id" value="{{teacher\_list[teacher\_index].\_id}}" class="weui-input" range="{{teacher\_list}}" range-key="teacher\_name" bindchange="teacher\_list\_change">教师姓名:{{teacher\_list[teacher\_index].teacher\_name}}</picker>

</view>

</view>

</view>

<!---------------------------------- 下面是插件 --------------------------------------------->

<!---------------------------------- 下面是插件 --------------------------------------------->

<!---------------------------------- 下面是插件 --------------------------------------------->

<view wx:if="{{teacher\_index!=''}}" class='contain' style="display:{{blockOrNone}}">

<scroll-view class="scroll-view\_H" scroll-x>

<view class='list' style='width:{{ width }}rpx'>

<view bindtap="select" wx:for="{{ calendar }}" wx:for-item="item" wx:for-index="index" data-index="{{ index }}" class='listItem {{index==currentIndex? "current":""}}' wx:key='nothing' data-date="{{ item.date}}">

<text class='name'>{{ item.week }}</text>

<text class='date'>{{ item.date }}</text>

</view>

</view>

</scroll-view>

<view wx:if="{{timeArr.length!=0}}" class='time'>

<view wx:for="{{ timeArr }}" wx:for-item="timeItem" wx:for-index="timeIndex" class='listItem {{ currentTime==timeIndex? "current":"" }}' wx:key='nothing'>

<text>{{ timeItem.time\_begin }} - {{timeItem.time\_end}}</text>

<block wx:if="{{timeItem.orderlist.length==0}}">

<text data-Tindex="{{ timeIndex }}" data-ketimeid="{{ timeItem.\_id}}" bindtap='selectTime' style="color:green;">点击预约</text>

</block>

<block wx:else>

<block wx:if="{{userinfo.openId == timeItem.orderlist[0].openid}}">

<text style="color:red;">我的预约</text>

<text data-orderid="{{timeItem.orderlist[0].\_id}}" bindtap="cancel">取消</text>

</block>

<block wx:else>

<text style="color:red;">已被({{timeItem.orderlist[0].nickname}})预约</text>

</block>

</block>

</view>

</view>

<view wx:else style="color:red;text-align:center;margin-top:50rpx;">该教师暂时没有安排课程</view>

<!-- <view class='operate'>

<button class='del'>删除</button>

<button class='save'>保存</button>

</view> -->

</view>

<!---------------------------------- 上面是插件 --------------------------------------------->

<!---------------------------------- 上面是插件 --------------------------------------------->

<!---------------------------------- 上面是插件 --------------------------------------------->

<view wx:else>

<view style="color:red;text-align:center;margin-top:50rpx;">请先选择教师</view>

</view>

</view>

/\* miniprogram/new\_pages/main/main.wxss \*/

@import "/lib/weui.wxss";

/\* pages/orderTime/index.wxss \*/

.contain{

margin-top:30rpx;

}

scroll-view{

height: 128rpx;

width: 100%;

}

scroll-view .list{

display: flex;

flex-wrap: nowrap;

justify-content: flex-start;

width: 1302rpx;

}

scroll-view .listItem{

text-align: center;

width: 186rpx;

height: 128rpx;

background-color: #f1f2f6;

padding-top: 30rpx;

box-sizing: border-box;

/\* float: left; \*/

display: inline-block;

}

scroll-view .listItem text{

display: block;

}

scroll-view .listItem .name{

font-size: 30rpx;

}

scroll-view .listItem .date{

font-size: 30rpx;

}

scroll-view .current{

background-color: #76aef8;

}

scroll-view .current text{

color: #fff;

}

.time{

width: 95%;

display: flex;

flex-wrap: wrap;

justify-content: flex-start;

margin: 0 auto;

margin-top: 30rpx;

}

.time .listItem{

width: 100%;

height: 100rpx;

text-align: center;

box-sizing: border-box;

background-color: #fff;

outline: 1px solid #b9c1c8;

display: flex;

flex-wrap:nowrap;

justify-content:space-around;

}

.time .listItem text{

display: block;

font-size: 30rpx;

height: 100rpx;

line-height: 100rpx;

}

.time .current{

outline: 1px solid #76aef8;

}

.time .current text{

color: #76aef8;

}

.operate button{

width:300rpx;

height: 88rpx;

background-color: #fff;

}

.operate .del{

color: #e54449;

}

.operate button::after{

outline: 2px solid #e54449;

}

.operate{

display: flex;

flex-wrap: nowrap;

justify-content: space-around;

}

.operate button:nth-child(2):after{

outline: 2px solid #e54449;

}

.operate .save{

color: #fff;

background-color: #e54449;

}

.operate{

width: 100%;

padding: 30rpx 0;

background-color: #fff;

position: fixed;

bottom: 0;

}

const db = wx.cloud.database();

var app = getApp();

Page({

/\*\*

\* 页面的初始数据

\*/

data: {

userinfo: null,

teacher\_index: "",

teacher\_list: null,

////////////////////////////下面是预约模块需要的data/////////////////////////////////

calendar: [],

width: 0,

currentIndex: 0,

currentTime: 0,

timeArr: null

// timeArr: [

// { "time": "8:00-10:00", "status": "约满" },

// { "time": "8:00-10:00", "status": "约满" },

// { "time": "8:00-10:00", "status": "约满" },

// { "time": "8:00-10:00", "status": "约满" },

// { "time": "8:00-10:00", "status": "约满" },

// { "time": "8:00-10:00", "status": "约满" },

// { "time": "8:00-10:00", "status": "约满" },

// { "time": "8:00-10:00", "status": "约满" },

// { "time": "8:00-10:00", "status": "约满" },

// { "time": "8:00-10:00", "status": "约满" },

// { "time": "8:00-10:00", "status": "约满" },

// { "time": "8:00-10:00", "status": "约满" },

// { "time": "8:00-22:00", "status": "约满" }

// ]

},

/\*\*

\* 生命周期函数--监听页面加载

\*/

teacher\_list\_change: function(res) {

var that = this;

that.setData({

teacher\_index: res.detail.value

});

var teacher\_id = that.data.teacher\_list[res.detail.value].\_id; //这是教师的id

var order\_date = that.data.calendar[that.data.currentIndex].date; //这是今天的日期

//console.log(order\_date);

wx.cloud.callFunction({

name: 'select\_orderInformation', //云函数的名称

data: {

teacher\_id: teacher\_id,

order\_date: order\_date

},

success: res => {

// console.log(res.result.list);

// console.log(res);

that.setData({

timeArr: res.result.list

})

},

fail: err => {

console.error('[云函数] [loginInfo] 调用失败', err)

}

})

},

onLoad: function(options) {

//console.log(app.globalData.userInfo);

var that = this;

that.setData({

userinfo:app.globalData.userInfo

})

db.collection("T\_teacher").get({

success: function(res) {

//console.log(res.data);

that.setData({

teacher\_list: res.data

})

}

});

////////////////////////////////////下面是预约的相关功能////////////////////////////////////

////////////////////////////////////下面是预约的相关功能////////////////////////////////////

////////////////////////////////////下面是预约的相关功能////////////////////////////////////

function getThisMonthDays(year, month) {

return new Date(year, month, 0).getDate();

}

// 计算每月第一天是星期几

function getFirstDayOfWeek(year, month) {

return new Date(Date.UTC(year, month - 1, 1)).getDay();

}

const date = new Date();

const cur\_year = date.getFullYear();

const cur\_month = date.getMonth() + 1;

const cur\_date = date.getDate();

const weeks\_ch = ['日', '一', '二', '三', '四', '五', '六'];

//利用构造函数创建对象

function calendar(date, week) {

this.date = cur\_year + '-' + cur\_month + '-' + date;

if (date == cur\_date) {

this.week = "今天";

} else if (date == cur\_date + 1) {

this.week = "明天";

} else {

this.week = '星期' + week;

}

}

//当前月份的天数

var monthLength = getThisMonthDays(cur\_year, cur\_month)

//当前月份的第一天是星期几

var week = getFirstDayOfWeek(cur\_year, cur\_month)

var x = week;

for (var i = 1; i <= monthLength; i++) {

//当循环完一周后，初始化再次循环

if (x > 6) {

x = 0;

}

//利用构造函数创建对象

that.data.calendar[i] = new calendar(i, [weeks\_ch[x]][0])

x++;

}

//限制要渲染的日历数据天数为7天以内（用户体验）

var flag = that.data.calendar.splice(cur\_date, that.data.calendar.length - cur\_date <= 7 ? that.data.calendar.length : 7)

that.setData({

calendar: flag

})

//设置scroll-view的子容器的宽度

that.setData({

width: 186 \* parseInt(that.data.calendar.length - cur\_date <= 7 ? that.data.calendar.length : 7)

})

////////////////////////////////上面是预约的相关功能////////////////////////////////////

////////////////////////////////上面是预约的相关功能////////////////////////////////////

////////////////////////////////上面是预约的相关功能////////////////////////////////////

},

/\*\*

\* 生命周期函数--监听页面初次渲染完成

\*/

onReady: function() {},

/\*\*

\* 生命周期函数--监听页面显示

\*/

onShow: function() {},

/\*\*

\* 生命周期函数--监听页面隐藏

\*/

onHide: function() {},

/\*\*

\* 生命周期函数--监听页面卸载

\*/

onUnload: function() {},

/\*\*

\* 页面相关事件处理函数--监听用户下拉动作

\*/

onPullDownRefresh: function() {},

/\*\*

\* 页面上拉触底事件的处理函数

\*/

onReachBottom: function() {},

/\*\*

\* 用户点击右上角分享

\*/

onShareAppMessage: function() {},

select: function(event) { //点击了日期执行的方法

var that = this;

//为上半部分的点击事件

this.setData({

currentIndex: event.currentTarget.dataset.index

})

//console.log(event.currentTarget.dataset.date)

var order\_date = event.currentTarget.dataset.date;

var teacher\_id = that.data.teacher\_list[that.data.teacher\_index].\_id; //这是教师的id

wx.cloud.callFunction({

name: 'select\_orderInformation', //云函数的名称

data: {

teacher\_id: teacher\_id,

order\_date: order\_date

},

success: res => {

// console.log(res.result.list);

// console.log(res);

that.setData({

timeArr: res.result.list

})

},

fail: err => {

console.error('[云函数] [loginInfo] 调用失败', err)

}

})

},

selectTime: function(event) {

var that = this;

//为下半部分的点击事件

// this.setData({

// currentTime: event.currentTarget.dataset.tindex

// })

//console.log(event.currentTarget.dataset.ketimeid)//T\_ketime表的id,T\_ketime里面有教师id和上课时间

var ketime\_id = event.currentTarget.dataset.ketimeid;

var order\_date = that.data.calendar[that.data.currentIndex].date;

var user = app.globalData.userInfo;

var d = new Date();

var day = d.getDate();

var month = d.getMonth() + 1;

var year = d.getFullYear();

var hour = d.getHours();

var minute = d.getMinutes();

var second = d.getSeconds();

var user = app.globalData.userInfo;

var now = year + "-" + month + "-" + day + " " + hour + ":" + minute + ":" + second;

var ketime\_id = event.currentTarget.dataset.ketimeid;

var order\_date = that.data.calendar[that.data.currentIndex].date;

var openid = user.openId;

var headimage = user.headImage;

var nickname = user.nickName;

// console.log(openid);

// console.log(headImage);

// console.log(nickName);

wx.cloud.callFunction({

name: 'add\_order', //云函数的名称

data: {

ketime\_id: ketime\_id,

order\_date: order\_date,

now: now,

openid: openid,

headimage: headimage,

nickname: nickname

},

success: res => {

//////////////////////////////////下面是点击预约之后重新刷新一下预约列表//////////////////

var teacher\_id = that.data.teacher\_list[that.data.teacher\_index].\_id; //这是教师的id

var order\_date = that.data.calendar[that.data.currentIndex].date;

wx.cloud.callFunction({

name: 'select\_orderInformation', //云函数的名称

data: {

teacher\_id: teacher\_id,

order\_date: order\_date

},

success: res => {

// console.log(res.result.list);

// console.log(res);

that.setData({

timeArr: res.result.list

})

},

fail: err => {

console.error('[云函数] [loginInfo] 调用失败', err)

}

})

//////////////////////////////////上面是点击预约之后重新刷新一下预约列表//////////////////

wx.showToast({

title: '预约成功',

icon: 'success',

duration: 1200,

})

},

fail: err => {

console.error('[云函数] [loginInfo] 调用失败', err)

}

})

},

cancel:function(res){

var that = this;

//console.log(res.currentTarget.dataset.orderid);

var orderid = res.currentTarget.dataset.orderid;

wx.showModal({

title: '提示',

content: '确定取消当前预约课程吗？',

success(res) {

if (res.confirm) {

wx.cloud.callFunction({

name: 'delete\_order', //云函数的名称

data: {

id: orderid,

},

success: res => {

var teacher\_id = that.data.teacher\_list[that.data.teacher\_index].\_id; //这是教师的id

var order\_date = that.data.calendar[that.data.currentIndex].date;

wx.cloud.callFunction({

name: 'select\_orderInformation', //云函数的名称

data: {

teacher\_id: teacher\_id,

order\_date: order\_date

},

success: res => {

// console.log(res.result.list);

// console.log(res);

that.setData({

timeArr: res.result.list

})

wx.showToast({

title: '已取消',

icon: 'success',

duration: 1200,

})

},

fail: err => {

console.error('[云函数] [loginInfo] 调用失败', err)

}

})

},

fail: err => {

console.error('[云函数] [loginInfo] 调用失败', err)

}

})

} else if (res.cancel) {

//点了取消不做任何操作

}

}

})

}

})