

4 月 9 日课后作业

姓名：冯轩

学号：2001210240

域名：www.xfengwyx.ltd

Step1：申请 ssl 证书

名称	修改日期
Apache	2021/4/9 9:32
IIS	2021/4/9 9:32
Nginx	2021/4/9 9:32
Tomcat	2021/4/9 9:32
www.xfengwyx.ltd.csr	2021/4/9 9:32
www.xfengwyx.ltd.key	2021/4/9 9:32
www.xfengwyx.ltd.pem	2021/4/9 9:32

Step2：配置 api，用于接收 code 到后台换取 openid

小程序中使用的是新配置好的 https-API：

```
// 发送 res.code 到后台换取 openId, sessionKey, unionId
wx.request({
  url: 'https://www.xfengwyx.ltd:30030/getopenid',
  data: {
    code: res.code
  },
})
```

运行结果如下，得到了 openid，证明 https-server 配置成功：



服务器端的 log（得到的是小程序端传送过来的 code）：

```
[root@VM-8-17-centos apis]# node app.js
App listening at 30020
02183UFa11VaRA0srvJa19uzLN283UFB
```

运行多次之后，可以发现，code 值每次都不一样，但是 openid 是唯一的，所以可以通过以 openid 作为唯一标识，识别用户身份，绑定用户的数据：

```
[root@VM-8-17-centos apis]# node app.js
App listening at 30020
02183UFa11VaRA0srvJa19uzLN283UFB
091WSbGa1P7tRA0UlUIa1DjYoH3WSbGZ
081Sog0w3bBxbW2aQX0w3jrGhX3Sog0G
041syB000ulvwL1nmZ300tBDpC2syB0A
041ZKmFa1pXDQA00afGa1KIV8g3ZKmFA
```

Step3: 建立数据库

创建小程序使用的 db，并建立第一个 user 表：

#	名字	类型	排序规则	属性	空	默认	注释	额外	操作
<input type="checkbox"/>	1	id	int		否	无		AUTO_INCREMENT	修改 删除 更多
<input type="checkbox"/>	2	openid	varchar(100)	utf8mb4_0900_ai_ci	否	无			修改 删除 更多
<input type="checkbox"/>	3	username	varchar(100)	utf8mb4_0900_ai_ci	否	无			修改 删除 更多
<input type="checkbox"/>	4	name	varchar(100)	utf8mb4_0900_ai_ci	否	无			修改 删除 更多

↑ ☐ 全选 选中项: 浏览 修改 删除 主键 唯一 索引 全文搜索

插入第一条数据：

+ 选项

				id	openid	username	name
<input type="checkbox"/>	编辑	复制	删除	1	o6j7b5Q3qRse0Z0mUZkDuYuG1M4g	admin	admin

↑ ☐ 全选 选中项: 编辑 复制 删除 导出

Step4: 建立数据库连接

下面通过在服务器端建立与 db 的 connection：

```
var connection = mysql.createConnection({
  host: '49.232.162.126',
  port: '3306',
  user: 'root',
  password: '123456',
  database: 'miniprogram'
});
connection.connect();
```

安装完 mysql 模块后，运行 app.js，没有报错，说明数据库连接成功：

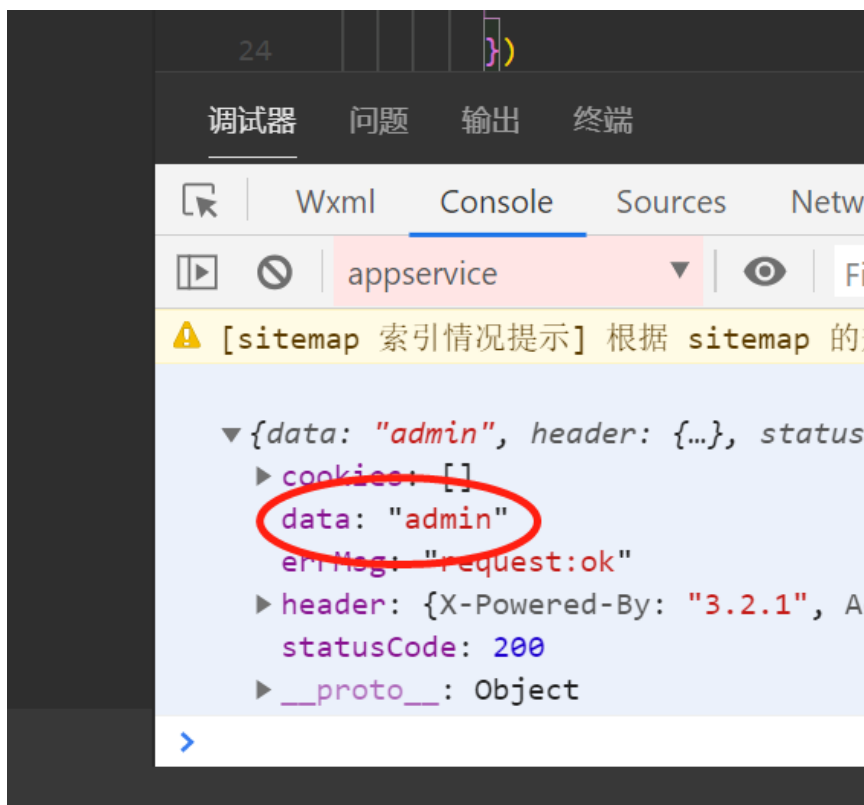
```
[root@VM-8-17-centos apis]# node app.js
App listening at 30020
```

Step5: 服务器返回与 openid 绑定的用户信息，比如上面表中的 name:

数据库查询：

```
var select = "select name from user where openid='" + result.openid + "'";
connection.query(select, (err, result) => {
  if (err) {
    res.send('err');
  }
  if (result == ''){
    res.send("Should sign up !");
  } else {
    console.log(result[0].name);
    res.send(result[0].name);
  }
})
});
```

运行程序，并观察结果，小程序端接收到了 admin，程序成功：



服务器端，也没有问题：

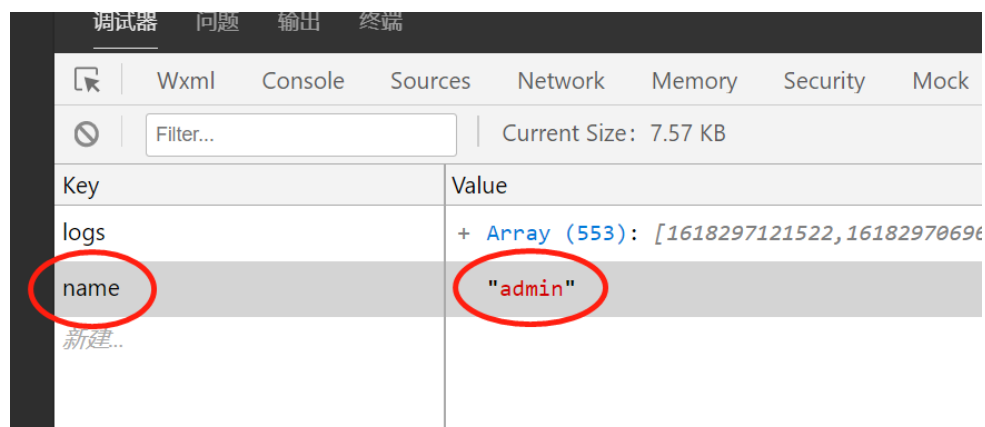
```
[root@VM-8-17-centos apis]# node app.js
App listening at 30020
o6j7b5Q3qRse0Z0mUZkDuYuG1M4g
admin
o6j7b5Q3qRse0Z0mUZkDuYuG1M4g
admin
o6j7b5Q3qRse0Z0mUZkDuYuG1M4g
admin
o6j7b5Q3qRse0Z0mUZkDuYuG1M4g
admin
o6j7b5Q3qRse0Z0mUZkDuYuG1M4g
admin
```

Step6: 在小程序端把接收到的数据库内容存入 Storage

小程序端操作：

```
wx.setStorage({
  key: 'name',
  data: res.data,
})
```

在调试器中查看 Storage，实验成功：



谢谢老师的查阅！