# 1 学校班级课程表

1.1学校班级课表信息

1.1.1 查询数据标识（table）：classArrangement

1.1.2 字段说明：

|  |  |  |
| --- | --- | --- |
| **Field** | **Type** | **Comment** |
| id | varchar(32) | 课表id，逻辑主键 |
| class\_id | int(11) | 班级id |
| gradeid | int(11) | 年级ID |
| classname | varchar(40) | 班级名称 |
| classshort | varchar(10) | 简称 |
| levelorder | int(11) | 序号 |
| gradenum | double | 年级排序字段 |
| serial | int(11) | 班号 |
| course\_id | varchar(32) | 课程id |
| teacher\_id | varchar(32) | 教师id（和teacher对象里面的id对应） |
| table\_id | varchar(10) | 周几第几节，例如：2,3，代表周二第三节 |
| class\_room\_id | varchar(32) | 教室id |
| weekend | varchar(100) | 周次，代表第几周 |
| scrq | timestamp | 生成日期 |
| xn | varchar(10) | 学年 |
| xq | varchar(10) | 学期 |
| week\_start | varchar(10) | 开始日期 |
| day\_time | date | 上课日期 |
| kcmc | varchar(80) | 课程名称 |
| courseshort | varchar(80) | 课程简称 |
| class\_type | varchar(2) | 班级类型，xz代表行政班，fc代表分层班 |
| start\_time | varchar(20) | 课程开始时间 |
| end\_time | varchar(20) | 课程结束时间 |

1.2查询数据

调用接口：学校open服务地址+/api/data/query

|  |  |
| --- | --- |
| **查询** | **值和说明**（open服务地址、clientId及clientSecrect 接口方提供） |
| URL | 学校open服务地址+/api/data/query |
| method | POST |
| 参数 | *clientId*  varchar(32) 分配给第三方客户的id |
| *timestamp* int(13) 时间戳，具体到毫秒，例如（1427961276927），与服务器时间差不超过10分钟 |
| *signature* varchar *签名，验证数据的合法性，*  *格式：*MD5( clientId + clientSecrect +timestamp)，  clientSecrect为分配给第三方的密钥， |
| *table* 查询的数据标识：（基础数据查询标识） |
|  | *data* json字符串，查询条件，格式：  {  groupOp: 'and/or',  orderBy: '排序字段',  rules: [{ field: '字段名', op: 操作符号, data: '查询数据' }…]  }  注意：字段名必须和返回结果里面的字段名保持一致，另外操作符合支持以下操作：  Eq(等于) bw(以…开始) ew(以…结束) cn(包含) |

1.3结果返回

|  |  |
| --- | --- |
| **返回** | 1. 成功返回：json列表   [  {  "id": "001b4424378c4b8c98426ad81864e6e9",  "class\_id": -10021,  "gradeid": 192015,  "classname": "1班",  "classshort": null,  "levelorder": 5,  "gradenum": 3,  "serial": 1,  "course\_id": "dbe35d7106d446c5b63355b92e24d37b",  "teacher\_id": "2191123210000000100",  "table\_id": "5,4",  "class\_room\_id": null,  "weekend": "21",  "scrq": "2017-11-07 11:27:31",  "xn": "2017-2018",  "xq": "第一学期",  "week\_start": "2018-01-15",  "day\_time": "2018-01-19 00:00:00",  "kcmc": "语文",  "courseshort": "语",  "class\_type": "xz",  "start\_time": null,  "end\_time": null  },  {}.....  ]  (2)失败返回错误信息  {  “sucess”:false,  “msg”: “失败信息”  } |

1.4示例

|  |
| --- |
| 示例代码为例：（示例中查询了某学校2017-2018学期，第11周课程表） |
|  |



1.5 附录（MD5）

附录生成MD5值的java代码：

**public** **class** BBt\_md5 {

**public** **static** String getMD5Str(String str) {

MessageDigest messageDigest = **null**;

**try** {

messageDigest = MessageDigest.*getInstance*("MD5");

messageDigest.reset();

messageDigest.update(str.getBytes("UTF-8"));

} **catch** (NoSuchAlgorithmException e) {

System.*out*.println("NoSuchAlgorithmException caught!");

} **catch** (UnsupportedEncodingException e) {

e.printStackTrace();

}

**byte**[] byteArray = messageDigest.digest();

StringBuffer md5StrBuff = **new** StringBuffer();

**for** (**int** i = 0; i < byteArray.length; i++) {

**if** (Integer.*toHexString*(0xFF & byteArray[i]).length() == 1)

md5StrBuff.append("0").append(

Integer.*toHexString*(0xFF & byteArray[i]));

**else**

md5StrBuff.append(Integer.*toHexString*(0xFF & byteArray[i]));

}

**return** md5StrBuff.toString();

}

}

1.6 校平台open服务基地址获取

发送http请求：http://xxx/bbt/bbt/getSchoolAdress.do?sid=学校id

返回成功示例：

{

"code": "success",

"domain": "aqsyqzx.aqedu.cn"

}

返回失败示例：

{

"code": "fail",

"msg": "查询为空"

}

或

{

"code": "fail",

"msg": "学校id不能为空"

}

# 2 教师信息与教师授课信息

* 1. 请求步骤

1. 发送数据生成请求

以http请求: [http://XXX/bbt/bbt/index.do?sid=学校id](http://XXXXXXXXXXXXX/bbt/bbt/index.do?sid=学校id)

1. 第一步生成数据请求之后半小时（数据整合查询较慢，为保证数据生成），获取该学校的教师及授课信息

教师信息：<http://xxx/bbt/resources/schoolExcel/>学校id/学校id\_teacher.xlsx



授课信息：<http://xxx/bbt/resources/schoolExcel/学校id/学校id_shouke.xlsx>



或者获取json数据<http://xxx/bbt/bbt/getSchoolMsgJson.do?sid>=学校id

返回格式如下：

{

"code": "success",

"list\_shouke": [

{

"ID": "xxx",

"姓名": "孟玉良",

"学段": "高中",

"授课班级": "2014-(3)班",

"授课课程": "其他",

"生日": ""

},

{

"ID": "xxxx",

"姓名": "叶曙光",

"学段": "高中",

"授课班级": "2014-(1)班",

"授课课程": "生物",

"生日": ""

}

],

"list\_teacher": [

{

"Id": "xxx",

"出生年月": "",

"备注": "",

"姓名": "宣青",

"学历": "",

"性别": "女",

"教师职称": "",

"注册时间": "",

"移动电话": "xxx"

},

{

"Id": "2134000017000493068",

"出生年月": "",

"备注": "",

"姓名": "叶莉莉",

"学历": "",

"性别": "女",

"注册时间": "",

"移动电话": "xxxx"

}

]

}

# 3部署校平台open服务和教师及授课开放接口（文档提供方需部署，三方忽略此步骤）

3.1部署open服务

需要部署校open服务，部署之后访问：校open地址+/ opendata /api/data/query，出现效果如下图所示，则open部署成功



校open的地址需要提供给第三方厂商。

3.2 配置三方接口权限

在数据库中配置第三方厂商拉取数据权限信息，需要配置如下表数据：

data\_sync\_client表



如上图需要在这张表中加入一条记录，id代表课程表请求字段中的clientId，access\_key课程表请求字段中的clientSecrect，name代表哪家公司对接；

client\_op\_permission表



如上图需要在这张表中加入一条记录，id和data\_sync\_client表中的id保持一致，op\_query字段代表第三方厂商需要同步的数据对象。每个订阅对象之前以英文半角逗号隔开，每个订阅对象含义可以参考接口里面的基础数据，里面罗列了目前能同步的对象及每个对象可以提供的字段。

# 4注意

1.1班级课程表

1.请求需要的关键字段如：open服务地址、clientId及clientSecrect 联系我方提供

2.请求字段data为非必填，默认查询学校所有班级所有学周课程表

* 1. 教师信息与授课信息

学校id联系我方提供。