统一约定：

不需要传值的场合，http用get方法，需要传值的话那随便你怎么传。

登陆用post方法。加点聊胜于无的安全性。

至于用ajax提交还是form表单提交这个前端大佬看着办，反正，我喜欢ajax

下面的是否可为空，指的是我们后台对于数据的需求，你们前端根据实际需要看字段能不能为空这个，那就看你们自己咯

登陆：

|  |  |  |  |
| --- | --- | --- | --- |
| **URL** | /login | | |
| **描述** | 登陆 | | |
| **参数名称** | **参数类型** | **参数描述** | **是否可为空** |
| name | String | 登录名称 | 不可 |
| password | String | 登录密码 | 不可 |
| **返回值格式** | { “type” : type , “message”: message } | | |
| **返回值类型** | type : 数字Integer。0代表成功1代表失败，只有这两种。  message : 字符串String。成功时返回成功信息，失败时返回失败信息。 | | |
| **返回值例子** | { “type” : 0 , “message” : “登陆成功”}  { “type” : 1 , “message” : “登陆失败” } | | |
| **备注** | 这个message你要怎么处理就随便你啦~显示给用户也好，自己console.log()出来也好，那我就不管啦，随便你玩~下同 | | |

部门管理：

|  |  |  |  |
| --- | --- | --- | --- |
| **URL** | /department/create | | |
| **描述** | 创建一个部门 | | |
| **参数名称** | **参数类型** | **参数描述** | **是否可为空** |
| name | String | 部门名称 | 不可 |
| parent | Integer | 上级部门（所属部门）id | 可 |
| note | String | 备注 | 可 |
| **返回值格式** | { “type” : type , “message”: message } | | |
| **返回值类型** | type : 数字Integer。0代表成功1代表失败，只有这两种。  message : 字符串String。成功时返回成功信息，失败时返回失败信息。 | | |
| **返回值例子** | { “type” : 0 , “message” : “创建成功”}  { “type” : 1 , “message” : “创建失败”} | | |
| **备注** | 这里的parent指的是其直系上级部门。 | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **URL** | /department/read | | |
| **描述** | 读取全部部门信息 | | |
| **参数名称** | **参数类型** | **参数描述** | **是否可为空** |
| **返回值格式** | [ {  “id” : id ,  “name” : name,  “parent” : parent,  “note” : note }…] | | |
| **返回值类型** | id : 数字Integer。表示部门id。  name : 字符串String。表示部门名称。  parented : 数字Integer。表示上级部门id。  note : 字符串String。备注文字。 | | |
| **返回值例子** | [ {  “id” : 1 ,  “name” : “VR研发部” ,  “parent : 0,  “note” : “两个黄鹂鸣翠柳~你还没有女朋友~”  }…] | | |
| **备注** | 参数名称什么的都没有意思是你请求一下就可以了，不需要发参数。下同。  备注什么的可能为空，你看我是发给你一个字符串null还是索性什么都不发？你要是不嗦发我就默认什么都不发了哈，下同 | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **URL** | /department/update | | |
| **描述** | 修改一个部门 | | |
| **参数名称** | **参数类型** | **参数描述** | **是否可为空** |
| id | Integer | 部门id | 不可 |
| name | String | 部门名称 | 不可 |
| parent | Integer | 上级部门（所属部门）id | 可 |
| note | String | 备注 | 可 |
| **返回值格式** | { “type” : type , “message”: message } | | |
| **返回值类型** | type : 数字Integer。0代表成功1代表失败，只有这两种。  message : 字符串String。成功时返回成功信息，失败时返回失败信息。 | | |
| **返回值例子** | { “type” : 0 , “message” : “修改成功”}  { “type” : 1 , “message” : “修改失败” } | | |
| **备注** | 如果用户没有修改name,root,note的值，请把原本的name,root,note发送过来，当且仅当用户将name,root,note的值清空（或者本来就没有）的情况下再传空值给我，类似的事情都依此办理，下同 | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **URL** | /department/delete | | |
| **描述** | 删除一个部门 | | |
| **参数名称** | **参数类型** | **参数描述** | **是否可为空** |
| id | Integer | 部门id | 不可 |
| **返回值格式** | { “type” : type , “message”: message } | | |
| **返回值类型** | type : 数字Integer。0代表成功1代表失败，只有这两种。  message : 字符串String。成功时返回成功信息，失败时返回失败信息。 | | |
| **返回值例子** | { “type” : 0 , “message” : “删除成功”}  { “type” : 1 , “message” : “删除失败” } | | |
| **备注** |  | | |

设备、课程管理：

|  |  |  |  |
| --- | --- | --- | --- |
| **URL** | /course/create | | |
| **描述** | 创建一个课程 | | |
| **参数名称** | **参数类型** | **参数描述** | **是否可为空** |
| name | String | 设备/课程名称 | 不可 |
| parent | Integer | 上级课程/设备id | 可 |
| image | file | 设备/课程图片 | 可 |
| training | file | 训练文件 | 可 |
| test | file | 考试文件 | 可 |
| movie | file | 视频文件 | 可 |
| note | String | 备注 | 可 |
| **返回值格式** | { “type” : type , “message”: message } | | |
| **返回值类型** | type : 数字Integer。0代表成功1代表失败，只有这两种。  message : 字符串String。成功时返回成功信息，失败时返回失败信息。 | | |
| **返回值例子** | { “type” : 0 , “message” : “创建成功”}  { “type” : 1 , “message” : “创建失败” } | | |
| **备注** | 在设备管理页面，用户创建设备的时候也是请求这个方法，这时请前端只给用户name,image,note三个值的框（当然image和note可以为空），不要传别的。同时在课程管理时，请前端判断一下parent的值，要确保不为空。  关于parent的描述这里跟前面一样，指的是直系上级课程。  关于考试文件，视频文件之类的是否允许为空这种事，以及不要在movie传个txt之类的事情就交给前端来判断了，反正我后台没所谓的。就是这么佛系。类似的事儿都如此办理，下同 | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **URL** | /course/read | | |
| **描述** | 读取全部课程 | | |
| **参数名称** | **参数类型** | **参数描述** | **是否可为空** |
| **返回值格式** | [{  “id” : id ,  “name” : name,  “parent” : parent,  “image” : image,  “training” : training,  “test” : test,  “movie” : movie,  “note” : note } … ] | | |
| **返回值类型** | id : 数字Integer。课程/设备id。  name ：字符串 String。课程/设备名称。  parent ：数字Integer。上级课程/设备id。  image ：字符串String。图片文件url。  training ：字符串String。训练文件url。  test ：字符串String。考试文件url。  movie：字符串String。视频文件url。  note :字符串String。备注文字。 | | |
| **返回值例子** | [ {  “id” : 12306 ,  “name” : “爆破阀拆装”,  “parent” : 10086,  “image” : “/baopofa/baopofachaizhuang.img”,  “training” : “/baopofa/baopofachaizhuangTraining.abc”,  “test” : “/baopofa/baopofachaizhuangTest.abc”,  “movie” : “/baopofa/baopofachaizhuangMovie.abc”,  “note” : “这课真难…”} … ] | | |
| **备注** | 不管是设备管理还是课程管理都是请求这个方法。我都会把这些给你。然后你这边把parent为空的项放在设备管理显示，parent不为空的放在课程管理显示。  或者你要是嫌麻烦那我另外写一个接口，专门给你parent为空的值，然后这个接口就专门给你parent不为空的值。反正我觉得嘛，都可以，你判断一下和我判断一下差别不大。 | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **URL** | /course/update | | |
| **描述** | 修改一个课程 | | |
| **参数名称** | **参数类型** | **参数描述** | **是否可为空** |
| id | Integer | 设备/课程id | 不可 |
| name | String | 设备/课程名称 | 不可 |
| parent | Integer | 上级课程/设备id | 可 |
| image | file | 设备图片 | 可 |
| training | file | 训练文件 | 可 |
| test | file | 考试文件 | 可 |
| movie | file | 视频文件 | 可 |
| note | String | 备注 | 可 |
| **返回值格式** | { “type” : type , “message” : message } | | |
| **返回值类型** | type : 数字Integer。0代表成功1代表失败，只有这两种。  message : 字符串String。成功时返回成功信息，失败时返回失败信息。 | | |
| **返回值例子** | { “type” : 0 , “message” : “修改成功”}  { “type” : 1 , “message” : “修改失败” } | | |
| **备注** |  | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **URL** | /course/delete | | |
| **描述** | 删除一个课程 | | |
| **参数名称** | **参数类型** | **参数描述** | **是否可为空** |
| id | Integer | 设备/课程id | 不可 |
| **返回值格式** | { “type” : type , “message” : message } | | |
| **返回值类型** | type : 数字Integer。0代表成功1代表失败，只有这两种。  message : 字符串String。成功时返回成功信息，失败时返回失败信息。 | | |
| **返回值例子** | { “type” : 0 , “message” : “删除成功”}  { “type” : 1 , “message” : “删除失败” } | | |
| **备注** |  | | |

模型管理：

|  |  |  |  |
| --- | --- | --- | --- |
| **URL** | /model/create | | |
| **描述** | 创建一个模型 | | |
| **参数名称** | **参数类型** | **参数描述** | **是否可为空** |
| name | String | 模型名称 | 不可 |
| parent | Integer | 所属模型id | 可 |
| model | file | 模型文件（smg） | 可 |
| download | file | 下载文件（CATIA） | 可 |
| note | String | 备注 | 可 |
| **返回值格式** | { “type” : type , “message”: message } | | |
| **返回值类型** | type : 数字Integer。0代表成功1代表失败，只有这两种。  message : 字符串String。成功时返回成功信息，失败时返回失败信息。 | | |
| **返回值例子** | { “type” : 0 , “message” : “创建成功”}  { “type” : 1 , “message” : “创建失败” } | | |
| **备注** |  | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **URL** | / model/read | | |
| **描述** | 读取全部模型 | | |
| **参数名称** | **参数类型** | **参数描述** | **是否可为空** |
| **返回值格式** | { [  “id” : id ,  “name” : name,  “parent” : parent,  “model” : model,  “download” : training,  “note” : note ] … } | | |
| **返回值类型** | id : 数字Integer 。模型id。  name ：字符串 String。课程/设备名称。  parent ：字符串String。所属设备名称。  model ：字符串String。在线预览模型文件url。  download ：字符串String。下载模型文件url。  note :字符串String。备注文字。 | | |
| **返回值例子** | { [  “id” : 12306 ,  “name” : “爆破阀拆装”,  “parent” : “爆破阀”,  “image” : “/baopofa/baopofachaizhuang.img”,  “model” : “/baopofa/baopofachaizhuangModel.abc”,  “download” : “/baopofa/baopofachaizhuangDownload.abc”,  “note” : “这模型真难看…”] … } | | |
| **备注** |  | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **URL** | /model/update | | |
| **描述** | 修改一个模型 | | |
| **参数名称** | **参数类型** | **参数描述** | **是否可为空** |
| id | Integer | 设备/课程 | 不可 |
| name | String | 模型名称 | 不可 |
| parent | String | 所属部分 | 可 |
| model | file | 模型文件（smg） | 可 |
| download | file | 下载文件（CATIA） | 可 |
| note | String | 备注 | 可 |
| **返回值格式** | { “type” : type , “message” : message } | | |
| **返回值类型** | type : 数字Integer。0代表成功1代表失败，只有这两种。  message : 字符串String。成功时返回成功信息，失败时返回失败信息。 | | |
| **返回值例子** | { “type” : 0 , “message” : “修改成功”}  { “type” : 1 , “message” : “修改失败” } | | |
| **备注** |  | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **URL** | /model/delete | | |
| **描述** | 删除一个模型 | | |
| **参数名称** | **参数类型** | **参数描述** | **是否可为空** |
| id | Integer | 设备/课程 | 不可 |
| **返回值格式** | { “type” : type , “message” : message } | | |
| **返回值类型** | type : 数字Integer。0代表成功1代表失败，只有这两种。  message : 字符串String。成功时返回成功信息，失败时返回失败信息。 | | |
| **返回值例子** | { “type” : 0 , “message” : “删除成功”}  { “type” : 1 , “message” : “删除失败” } | | |
| **备注** |  | | |

检修管理 ：

|  |  |  |  |
| --- | --- | --- | --- |
| **URL** | /model/create | | |
| **描述** | 创建一个模型 | | |
| **参数名称** | **参数类型** | **参数描述** | **是否可为空** |
| name | String | 模型名称 | 不可 |
| parent | String | 所属部分 | 可 |
| model | file | 模型文件（smg） | 可 |
| download | file | 下载文件（CATIA） | 可 |
| note | String | 备注 | 可 |
| **返回值格式** | { “type” : type , “message”: message } | | |
| **返回值类型** | type : 数字Integer。0代表成功1代表失败，只有这两种。  message : 字符串String。成功时返回成功信息，失败时返回失败信息。 | | |
| **返回值例子** | { “type” : 0 , “message” : “创建成功”}  { “type” : 1 , “message” : “创建失败” } | | |
| **备注** |  | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **URL** | / model/read | | |
| **描述** | 读取全部模型 | | |
| **参数名称** | **参数类型** | **参数描述** | **是否可为空** |
| **返回值格式** | { [  “id” : id ,  “name” : name,  “parent” : parent,  “model” : model,  “download” : training,  “note” : note ] … } | | |
| **返回值类型** | id : 数字Integer 。模型id。  name ：字符串 String。课程/设备名称。  parent ：字符串String。所属设备名称。  model ：字符串String。在线预览模型文件url。  download ：字符串String。下载模型文件url。  note :字符串String。备注文字。 | | |
| **返回值例子** | { [  “id” : 12306 ,  “name” : “爆破阀拆装”,  “parent” : “爆破阀”,  “image” : “/baopofa/baopofachaizhuang.img”,  “model” : “/baopofa/baopofachaizhuangModel.abc”,  “download” : “/baopofa/baopofachaizhuangDownload.abc”,  “note” : “这模型真难看…”] … } | | |
| **备注** |  | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **URL** | /model/update | | |
| **描述** | 修改一个模型 | | |
| **参数名称** | **参数类型** | **参数描述** | **是否可为空** |
| id | Integer | 设备/课程 | 不可 |
| name | String | 模型名称 | 可 |
| parent | String | 所属部分 | 可 |
| model | file | 模型文件（smg） | 可 |
| download | file | 下载文件（CATIA） | 可 |
| note | String | 备注 | 可 |
| **返回值格式** | { “type” : type , “message” : message } | | |
| **返回值类型** | type : 数字Integer。0代表成功1代表失败，只有这两种。  message : 字符串String。成功时返回成功信息，失败时返回失败信息。 | | |
| **返回值例子** | { “type” : 0 , “message” : “修改成功”}  { “type” : 1 , “message” : “修改失败” } | | |
| **备注** |  | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **URL** | /model/delete | | |
| **描述** | 删除一个模型 | | |
| **参数名称** | **参数类型** | **参数描述** | **是否可为空** |
| id | Integer | 设备/课程 | 不可 |
| **返回值格式** | { “type” : type , “message” : message } | | |
| **返回值类型** | type : 数字Integer。0代表成功1代表失败，只有这两种。  message : 字符串String。成功时返回成功信息，失败时返回失败信息。 | | |
| **返回值例子** | { “type” : 0 , “message” : “删除成功”}  { “type” : 1 , “message” : “删除失败” } | | |
| **备注** |  | | |

检修规划管理 ：

|  |  |  |  |
| --- | --- | --- | --- |
| **URL** | /overhaul/create | | |
| **描述** | 创建一个规划 | | |
| **参数名称** | **参数类型** | **参数描述** | **是否可为空** |
| name | String | 规划名称 | 不可 |
| parent | String | 所属部分 | 不可 |
| file | file | 检修规划文件 | 可 |
| note | String | 备注 | 可 |
| **返回值格式** | { “type” : type , “message”: message } | | |
| **返回值类型** | type : 数字Integer。0代表成功1代表失败，只有这两种。  message : 字符串String。成功时返回成功信息，失败时返回失败信息。 | | |
| **返回值例子** | { “type” : 0 , “message” : “创建成功”}  { “type” : 1 , “message” : “创建失败” } | | |
| **备注** |  | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **URL** | /overhaul/read | | |
| **描述** | 读取全部规划 | | |
| **参数名称** | **参数类型** | **参数描述** | **是否可为空** |
| **返回值格式** | [ {  “id” : id ,  “name” : name,  “parent” : parent,  “file” : file,  “note” : note } … ] | | |
| **返回值类型** | id : 数字Integer 。模型id。  name ：字符串 String。规划名称。  parent ：字符串String。所属设备名称。  file ：字符串String。检修规划文件url。  note :字符串String。备注文字。 | | |
| **返回值例子** | [ {  “id” : 12306 ,  “name” : “随便起个名”,  “parent” : “爆破阀”,  “image” : “/baopofa/baopofachaizhuang.img”,  “file” : “/baopofa/baopofachaizhuangFile.abc”,  “note” : “那周报咋那么复杂…”} … ] | | |
| **备注** |  | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **URL** | /overhaul/update | | |
| **描述** | 修改一个规划 | | |
| **参数名称** | **参数类型** | **参数描述** | **是否可为空** |
| id | Integer | 设备/课程 | 不可 |
| name | String | 模型名称 | 不可 |
| parent | String | 所属部分 | 不可 |
| file | file | 检修规划文件 | 可 |
| note | String | 备注 | 可 |
| **返回值格式** | { “type” : type , “message” : message } | | |
| **返回值类型** | type : 数字Integer。0代表成功1代表失败，只有这两种。  message : 字符串String。成功时返回成功信息，失败时返回失败信息。 | | |
| **返回值例子** | { “type” : 0 , “message” : “修改成功”}  { “type” : 1 , “message” : “修改失败” } | | |
| **备注** |  | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **URL** | /overhaul/delete | | |
| **描述** | 删除一个规划 | | |
| **参数名称** | **参数类型** | **参数描述** | **是否可为空** |
| id | Integer | 规划id | 不可 |
| **返回值格式** | { “type” : type , “message” : message } | | |
| **返回值类型** | type : 数字Integer。0代表成功1代表失败，只有这两种。  message : 字符串String。成功时返回成功信息，失败时返回失败信息。 | | |
| **返回值例子** | { “type” : 0 , “message” : “删除成功”}  { “type” : 1 , “message” : “删除失败” } | | |
| **备注** |  | | |

考试管理：

题库管理：

|  |  |  |  |
| --- | --- | --- | --- |
| **URL** | /questions/create | | |
| **描述** | 新增一道试题 | | |
| **参数名称** | **参数类型** | **参数描述** | **是否可为空** |
| type | Integer | 题目类型id | 不可 |
| course | String | 所属课程名称 | 不可 |
| body | String | 题干内容 | 不可 |
| answer | String | 选项内容 | 不可 |
| right | Integer | 正确答案id | 不可 |
| image | file | 题目图像 | 可 |
| **返回值格式** | { “type” : type , “message” : message } | | |
| **返回值类型** | type : 数字Integer。0代表成功1代表失败，只有这两种。  message : 字符串String。成功时返回成功信息，失败时返回失败信息。 | | |
| **返回值例子** | { “type” : 0 , “message” : “删除成功”}  { “type” : 1 , “message” : “删除失败” } | | |
| **备注** | 所有的选项，不管他有三个五个十个还是100个选项，参数名都用answer，然后正确答案id，简单打个比方，a就是0,b就是1,c就是2,以此类推。第n个选项的id就是n-1。  answer不能为空的意思是你至少要有一个选项，不过一共设立几个选项那就随意了，大于0 就行。  另外题目类型id..就老规矩吧，单选题0，判断题1，多选题2，实操题3.如果有别的类型的题那就456往下排。 | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **URL** | /questions/readByName | | |
| **描述** | 根据课程名读取相应试题 | | |
| **参数名称** | **参数类型** | **参数描述** | **是否可为空** |
| name | String | 课程名称 | 不可 |
| **返回值格式** | [ {  “id” : id ,  “type” : type,  “course” : course,  “body” : body,  “answerA”: answerA,  “answerB”: answerB,  “answerC”: answerC,  …  “right1”: right1,  “right2”: right2,  “right3” : right3,  …  “image”: image  } … ] | | |
| **返回值类型** | id : 数字Integer，试题id。  type ：数字Integer，题目类型id。  course ：字符串String，所属课程名称。  body ：字符串String，题干内容。  answer : 字符串String，选项内容。  right ： 数字Integer，正确答案id。  image : 字符串String，图像url地址。 | | |
| **返回值例子** | [ {  “id” : 12345,  “type” : 2,  “course” : “爆破阀拆装”,  “body” : “刘奎帅不帅？”,  “answerA”: “不帅”,  “answerB”: “特别丑”,  “answerC”: “丑到掉渣”,  “answerD” : “以上选项全部错误。”,  “right1”: 0 ,  “right2” :1 ,  “right3”: 2,  “image”: image} … ] | | |
| **备注** | answer按照顺序依次是abcd…如果你需要我给他们起不同的名字的话那你再跟我说，不过我记得java script是可以一次性批量取若干同名属性的。  right 同理。  如果他们没有多选题的话，我就给你一个right1。 | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **URL** | /questions/update | | |
| **描述** | 修改一道试题 | | |
| **参数名称** | **参数类型** | **参数描述** | **是否可为空** |
| id | Integer | 试题id | 不可 |
| type | Integer | 题目类型id | 不可 |
| course | String | 所属课程名称 | 不可 |
| body | String | 题干内容 | 不可 |
| answer | String | 选项内容 | 不可 |
| right | Integer | 正确答案id | 不可 |
| image | file | 题目图像 | 可 |
| **返回值格式** | { “type” : type , “message” : message } | | |
| **返回值类型** | type : 数字Integer。0代表成功1代表失败，只有这两种。  message : 字符串String。成功时返回成功信息，失败时返回失败信息。 | | |
| **返回值例子** | { “type” : 0 , “message” : “修改成功”}  { “type” : 1 , “message” : “修改失败” } | | |
| **备注** |  | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **URL** | /questions/delete | | |
| **描述** | 删除一道试题 | | |
| **参数名称** | **参数类型** | **参数描述** | **是否可为空** |
| id | Integer | 试题id | 不可 |
| **返回值格式** | { “type” : type , “message” : message } | | |
| **返回值类型** | type : 数字Integer。0代表成功1代表失败，只有这两种。  message : 字符串String。成功时返回成功信息，失败时返回失败信息。 | | |
| **返回值例子** | { “type” : 0 , “message” : “删除成功”}  { “type” : 1 , “message” : “删除失败” } | | |
| **备注** |  | | |

试卷管理：

|  |  |  |  |
| --- | --- | --- | --- |
| **URL** | /paper/create | | |
| **描述** | 新增一套试卷 | | |
| **参数名称** | **参数类型** | **参数描述** | **是否可为空** |
| name | String | 试卷名称 | 不可 |
| machine | String | 所属设备名称 | 不可 |
| course | String | 所属课程名称 | 不可 |
| question | Integer | 所属题目id | 不可 |
| **返回值格式** | { “type” : type , “message” : message } | | |
| **返回值类型** | type : 数字Integer。0代表成功1代表失败，只有这两种。  message : 字符串String。成功时返回成功信息，失败时返回失败信息。 | | |
| **返回值例子** | { “type” : 0 , “message” : “删除成功”}  { “type” : 1 , “message” : “删除失败” } | | |
| **备注** | question和前面的answer等一样，不再赘述  另外教员页面有个模拟试卷的页面。  方法跟这边一样。唯一不同在于，他们模拟卷和试卷是不同的，你创建模拟卷的时候，在发送course的时候在前面加一个“模拟：”吧  打个比方，就原本course是“爆破阀拆装”，现在就是“模拟：爆破阀拆装” | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **URL** | /paper/read | | |
| **描述** | 读取全部试卷 | | |
| **参数名称** | **参数类型** | **参数描述** | **是否可为空** |
| **返回值格式** | [ {  “id” : id ,  “name” : name,  “machine” : machine,  “course” : course,  “questionBody”: question1body,  “questionId” : questione1Id,  “questionBody”: question2body,  “questionId”: question2Id,  …  } … ] | | |
| **返回值类型** | id : 数字Integer，试卷id。  name : 字符串String，试卷名称。  machine ：字符串id，设备名称。  course ：字符串String，所属课程名称。  questionBody: 字符串String，试卷内题目的题干。  questioneId : 数字Integer，试卷内题目id。 | | |
| **返回值例子** | [ {  “id” : 54321,  “name” : “VR研发部面试题目”,  “machine” : “电脑”,  “course” : “Java”,  “questionBody” : “Java这个名字是怎么来的？”,  “questionId” : 007  …  } … ] | | |
| **备注** | 事实上根据.Net版本现有功能来看，我这边其实是不需要给你questionId这个属性。他们根本就不能更改试卷内的题目，不能对已有试卷增删题目。然而我还是觉得给你一个id比较好，未来客户可能会要求我们加上这些功能，到时候好改。 | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **URL** | /paper/tests | | |
| **描述** | 学员身份查看考试试卷 | | |
| **参数名称** | **参数类型** | **参数描述** | **是否可为空** |
| **返回值格式** | [{  “machine” : machine,  “course” : course,  “name” : name,  “sendTime” : sendTime,  “submitTime” : submitTime  }…] | | |
| **返回值类型** | machine ： 字符串String，设备名称。  course ：字符串String，  name ：字符串String，  sendTime ：日期Date，  submitTime ：日期Date， | | |
| **返回值例子** | {  “paperName” : “JAVA面试卷”,  “userName” : “刘奎”,  “operationScore” :20,  “total” : 59,  “sendDate” : “2018-4-24”,  “submitDate” : “2018-4-24”,  “checkDate” : “2018-4-24”  } | | |
| **备注** | 我忘了java script里面有没有date这个数据类型了。如果没有的话你就当字符串解析就行。 | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **URL** | /paper/score | | |
| **描述** | 查看考试结果 | | |
| **参数名称** | **参数类型** | **参数描述** | **是否可为空** |
| id | Integer | 试卷id | 不可 |
| **返回值格式** | {  “paperName” : paperName,  “userName” : username,  “operationScore” : operationScore,  “total” : total,  “sendDate” : sendDate,  “submitDate” : submitDate,  “checkDate” : chechDate  } | | |
| **返回值类型** | paperName ：字符串String，试卷名称  userName ：字符串String，学员名称  operationScore ： 数字Integer，实操题得分。  total ：数字Integer，总得分。  sendDate ：日期Date，发卷时间  submitDate : 日期Date，交卷时间  checkDate ：日期Data，评卷时间 | | |
| **返回值例子** | {  “paperName” : “JAVA面试卷”,  “userName” : “刘奎”,  “operationScore” :20,  “total” : 59,  “sendDate” : “2018-4-24”,  “submitDate” : “2018-4-24”,  “checkDate” : “2018-4-24”  } | | |
| **备注** | 我忘了java script里面有没有date这个数据类型了。如果没有的话你就当字符串解析就行。 | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **URL** | /paper/update | | |
| **描述** | 修改一张试卷 | | |
| **参数名称** | **参数类型** | **参数描述** | **是否可为空** |
| id | Integer | 试题id | 不可 |
| name | String | 试卷名称 | 不可 |
| machine | String | 所属设备名称 | 不可 |
| course | String | 所属课程名称 | 不可 |
| question | Integer | 所属题目id | 不可 |
| **返回值格式** | { “type” : type , “message” : message } | | |
| **返回值类型** | type : 数字Integer。0代表成功1代表失败，只有这两种。  message : 字符串String。成功时返回成功信息，失败时返回失败信息。 | | |
| **返回值例子** | { “type” : 0 , “message” : “修改成功”}  { “type” : 1 , “message” : “修改失败” } | | |
| **备注** | .Net版本目前是不能修改试卷内的题目的，就不能在已有试卷里增加或者减少题目。反正彭总他也没打算把这个项目做得多好，他是要借这个辣鸡项目来彰显我们java版多好。我想着我们就按照.Net他们那个样子做。他们现在只能修改试卷名字。不过我接口还是都给你预留着，我估计以后客户可能要提意见叫我们改。 | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **URL** | /paper/sendOnePaper | | |
| **描述** | 为多名学员分配一张试卷 | | |
| **参数名称** | **参数类型** | **参数描述** | **是否可为空** |
| id | Integer | 用户id | 不可 |
| **返回值格式** | { “type” : type , “message” : message } | | |
| **返回值类型** | type : 数字Integer。0代表成功1代表失败，只有这两种。  message : 字符串String。成功时返回成功信息，失败时返回失败信息。 | | |
| **返回值例子** | { “type” : 0 , “message” : “删除成功”}  { “type” : 1 , “message” : “删除失败” } | | |
| **备注** | 这个是教师页面分配试卷的方法。.Net版本是可以一次性给多个学员分配试卷的。所以这个Id也可以一次性发多个。反正都叫id就行了。 | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **URL** | /paper/sendOneUser | | |
| **描述** | 为一名学员分配多张试卷 | | |
| **参数名称** | **参数类型** | **参数描述** | **是否可为空** |
| id | Integer | 试卷id | 不可 |
| **返回值格式** | { “type” : type , “message” : message } | | |
| **返回值类型** | type : 数字Integer。0代表成功1代表失败，只有这两种。  message : 字符串String。成功时返回成功信息，失败时返回失败信息。 | | |
| **返回值例子** | { “type” : 0 , “message” : “删除成功”}  { “type” : 1 , “message” : “删除失败” } | | |
| **备注** | 我忘了.Net版本有没有这个方法了，反正是跟上面的对应的。他有这个功能你就加上，没有就算了。 | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **URL** | /paper/answer | | |
| **描述** | 学员答卷 | | |
| **参数名称** | **参数类型** | **参数描述** | **是否可为空** |
| id | Integer | 试卷id | 不可 |
| answer | Integer | 每道题的选项ID | 不可 |
| **返回值格式** | { “type” : type , “message” : message } | | |
| **返回值类型** | type : 数字Integer。0代表成功1代表失败，只有这两种。  message : 字符串String。成功时返回成功信息，失败时返回失败信息。 | | |
| **返回值例子** | { “type” : 0 , “message” : “删除成功”}  { “type” : 1 , “message” : “删除失败” } | | |
| **备注** | 学员的答卷方法。  answer指的是每道题的选项  a就是0，b就是1，c就是2，以此类推。  每张卷子会有多道题，每次发来的answer也会有多个，反正你这边按顺序，都叫answer发过来就行了。 | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **URL** | /paper/delete | | |
| **描述** | 删除一张试卷 | | |
| **参数名称** | **参数类型** | **参数描述** | **是否可为空** |
| id | Integer | 试题id | 不可 |
| **返回值格式** | { “type” : type , “message” : message } | | |
| **返回值类型** | type : 数字Integer。0代表成功1代表失败，只有这两种。  message : 字符串String。成功时返回成功信息，失败时返回失败信息。 | | |
| **返回值例子** | { “type” : 0 , “message” : “删除成功”}  { “type” : 1 , “message” : “删除失败” } | | |
| **备注** |  | | |

角色管理：

角色管理页面实际分为三个不同的部分，不过我懒得挨个针对着写接口了，给你个通用点的，到时候你看着用吧。

|  |  |  |  |
| --- | --- | --- | --- |
| **URL** | /user/create | | |
| **描述** | 新增一个账户 | | |
| **参数名称** | **参数类型** | **参数描述** | **是否可为空** |
| number | String | 工号 | 不可 |
| loginName | String | 登录名称 | 不可 |
| userName | String | 用户名称 | 不可 |
| password | String | 登录密码 | 不可 |
| character | Integer | 所属角色id | 可 |
| department | String | 所属部门名字 | 可 |
| sex | Integer | 性别id | 可 |
| tel | String | 电话号码 | 可 |
| lock | Integer | 是否锁定 | 可 |
| email | String | 电子邮箱 | 可 |
| **返回值格式** | { “type” : type , “message” : message } | | |
| **返回值类型** | type : 数字Integer。0代表成功1代表失败，只有这两种。  message : 字符串String。成功时返回成功信息，失败时返回失败信息。 | | |
| **返回值例子** | { “type” : 0 , “message” : “删除成功”}  { “type” : 1 , “message” : “删除失败” } | | |
| **备注** | 目前他们创建账号的时候是没有邮箱的。反正跟前面一样，我先把功能开放给你，反正你不给用户就行了。  角色id：管理员0，教员1，学员2  性别id : 男1，女0  锁定状态id：1为锁定，0为未锁定。  就这么约定着吧。 | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **URL** | /user/read | | |
| **描述** | 读取全部账户 | | |
| **参数名称** | **参数类型** | **参数描述** | **是否可为空** |
| **返回值格式** | [ {  “id” : id,  “number” : number,  “loginName” : loginName,  “userName” : userName,  “password” : password,  “character” : character,  “department” : department,  “sex” : sex,  “tel” : tel,  “lock” : lock,  “email” : email  } … ] | | |
| **返回值类型** | id : 数字Integer，试题id。  number : 字符串String，工号  loginName : 字符串String，登录名称  userName : 字符串String，用户名称  password : 字符串String，登录密码  character : 数字Integer，角色id  department : 字符串String，部门名称  sex : 数字Integer，性别id  tel : 字符串String，电话号码  lock : 数字Integer，锁定id  email : 字符串String，电子邮箱 | | |
| **返回值例子** | [ {  “id” : 666,  “number” : “No.666”,  “loginName” : “white”,  “userName” : “卢本伟”,  “password” : “fivefiveopen”,  “character” : 2,  “department” : ”电子竞技事业部” ,  “sex” : 1,  “tel” : 18686868686,  “lock” : 1,  “email” : 55open@dounaiTV.com  } … ] | | |
| **备注** | 他们好像还有根据部门什么的来筛选的功能，反正我就全给你了哈，分页筛选之类的就交给前端大佬了。 | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **URL** | | /user/update | | |
| **描述** | | 修改一个用户 | | |
| **参数名称** | | **参数类型** | **参数描述** | **是否可为空** |
| id | | Integer | 试题id | 不可 |
| number | | String | 工号 | 不可 |
| loginName | | String | 登录名称 | 不可 |
| userName | | String | 用户名称 | 不可 |
| password | | String | 登录密码 | 不可 |
| character | | Integer | 所属角色id | 可 |
| department | | String | 所属部门名字 | 可 |
| sex | | Integer | 性别id | 可 |
| tel | | Integer | 电话号码 | 可 |
| lock | | Integer | 是否锁定 | 可 |
| email | | String | 电子邮箱 | 可 |
| **返回值格式** | { “type” : type , “message” : message } | | | |
| **返回值类型** | type : 数字Integer。0代表成功1代表失败，只有这两种。  message : 字符串String。成功时返回成功信息，失败时返回失败信息。 | | | |
| **返回值例子** | { “type” : 0 , “message” : “修改成功”}  { “type” : 1 , “message” : “修改失败” } | | | |
| **备注** | .Net版本目前不支持这么多修改，反正你看着给用户吧，我还是先都给你。另外在所属部门和所属角色那两栏里面的修改，以及教师学员页面自己修改性别邮箱那个，也可以用这里的方法。反正依旧是，用户没改的部分你就把原本的数据给我，别他没改你就传个空就行。 | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **URL** | /user/delete | | |
| **描述** | 删除一张试卷 | | |
| **参数名称** | **参数类型** | **参数描述** | **是否可为空** |
| id | Integer | 用户id | 不可 |
| **返回值格式** | { “type” : type , “message” : message } | | |
| **返回值类型** | type : 数字Integer。0代表成功1代表失败，只有这两种。  message : 字符串String。成功时返回成功信息，失败时返回失败信息。 | | |
| **返回值例子** | { “type” : 0 , “message” : “删除成功”}  { “type” : 1 , “message” : “删除失败” } | | |
| **备注** |  | | |

PE管理：

|  |  |  |  |
| --- | --- | --- | --- |
| **URL** | /pe/read | | |
| **描述** | 读取pe信息 | | |
| **参数名称** | **参数类型** | **参数描述** | **是否可为空** |
| **返回值格式** | {“id” : id, “document” : document} | | |
| **返回值类型** | id : 数字Integer，试题id。  document : 字符串String，安装文档url。 | | |
| **返回值例子** | {“id” : 720, “document” : ”pe/document.txt” } | | |
| **备注** |  | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **URL** | /pe/update | | |
| **描述** | 修改pe信息 | | |
| **参数名称** | **参数类型** | **参数描述** | **是否可为空** |
| id | Integer | 试题id | 不可 |
| document | file | 插件安装文档 | 不可 |
| **返回值格式** | { “type” : type , “message” : message } | | |
| **返回值类型** | type : 数字Integer。0代表成功1代表失败，只有这两种。  message : 字符串String。成功时返回成功信息，失败时返回失败信息。 | | |
| **返回值例子** | { “type” : 0 , “message” : “修改成功”}  { “type” : 1 , “message” : “修改失败” } | | |
| **备注** |  | | |

分页管理：

|  |  |  |  |
| --- | --- | --- | --- |
| **URL** | /page/read | | |
| **描述** | 读取分页信息 | | |
| **参数名称** | **参数类型** | **参数描述** | **是否可为空** |
| **返回值格式** | {“number” : number} | | |
| **返回值类型** | number ： 数字Integer，每页条数，默认10条/页 | | |
| **返回值例子** | {“number” : 10 } | | |
| **备注** |  | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **URL** | /page/update | | |
| **描述** | 修改分页信息 | | |
| **参数名称** | **参数类型** | **参数描述** | **是否可为空** |
| number | Integer | 每页条数 | 不可 |
| **返回值格式** | { “type” : type , “message” : message } | | |
| **返回值类型** | type : 数字Integer。0代表成功1代表失败，只有这两种。  message : 字符串String。成功时返回成功信息，失败时返回失败信息。 | | |
| **返回值例子** | { “type” : 0 , “message” : “修改成功”}  { “type” : 1 , “message” : “修改失败” } | | |
| **备注** |  | | |

清理临时文件：

我实在不晓得这玩意是干嘛玩的= =如果是清理什么cookie之类的，那就前端大佬自己搞定吧~

软件下载：

都是写死的。前端大佬直接把url写死得了。不用跟后台交互了。