# 全局约定

## 定义统一返回值格式

1. **返回结果格式**

**ResultResponse**{

String type;//post类型

String description;//处理成功或报错信息

String state;//若处理成功返回success,若处理失败返回error

Object data;//数据

}

3、前台和后台通信采用JSON格式数据，字符编码格式为UTF-8

4、查询时，若不带参数则返回全部数据

# 用户中心

### 1、用户登录

|  |  |
| --- | --- |
| 接口描述 |  |
| 接口URL | **/employment/user/login/query.do** |
| 请求参数 | {queryFormParams:{  **String user\_number;//账号  String user\_password;//密码**  }  } |
| 返回响应 | {type: description:,state:, data:} |

### 2、用户注册

|  |  |
| --- | --- |
| 接口描述 |  |
| 接口URL | **/employment/user/login/post.do** |
| 请求参数 | {postBean:{  **String user\_number;//账号  String user\_password;//密码**  }  } |
| 返回响应 | {type: description:,state:, data:} |

# 学校信息设置

设置学校信息

### 1、查询学校

|  |  |
| --- | --- |
| 接口描述 | 待讨论，需不需要这里查询的时候一直关联到班级全部查询出来返回 |
| 接口URL | **/employment/configCenter/university/query.do** |
| 请求参数 | {queryFormParams:{  String university\_number;//学校编号  String university\_name;//学校名称  }  } |
| 返回响应 | {type: description:,state:, “list”:List<UniversityBean>.toJson:}  Public class UniversityBean{  String university\_id;//学校主键  String university\_number;//学校编号  String university\_name;//学校名字  } |

### 2、查询学校详情

|  |  |
| --- | --- |
| 接口描述 | 如果接口1的描述需要这个接口就没有必要了 |
| 接口URL | **/employment/configCenter/university/queryById.do** |
| 请求参数 | {specifyId:{  String university\_id;//学校主键  }  } |
| 返回响应 | {type: description:,state:, “list”:UniversityBean.toJson:}  Public class UniversityBean{  String university\_id;//学校主键  String university\_number;//学校编号  String university\_name;//学校名字  List<CollegeBean> colleges;//学院集合  }  Public class CollegeBean{  String college\_id;//学院主键  String university\_id;//学校外键  String college\_number;//学院编号  String college\_name;//学院名称    List<MajorBean> majors;//专业集合  }  Public class MajorBean{  String major\_id;//专业主键  String college\_id;//学院外键  String major\_number;//专业编号  String major\_name;//专业名称    List<ClassroomBean> classroomList;//班级集合  }  Public class ClassroomBean{  String classroom\_id;//班级主键  String major\_id;//专业外键  String classroom\_number;//班级编号  }  List集合一层层往下包含，遍历的时候会有些头晕 |

### 3、提交、更新学校

|  |  |
| --- | --- |
| 接口描述 | 一直到班级都要包含在UniversityBean里面提交上来 |
| 接口URL | **/employment/configCenter/university/post.do** |
| 请求参数 | {postBean:{  String university\_id;学校主键  String university\_number;//学校编号  String university\_name;//学校名称  List<CollegeBean> collegeBean;//学院  }  Public class CollegeBean{  String college\_id;//学院主键  String university\_id;//学校外键  String college\_number;//学院编号  String college\_name;//学院名称    List<MajorBean> majors;//专业集合  }  Public class MajorBean{  String major\_id;//专业主键  String college\_id;//学院外键  String major\_number;//专业编号  String major\_name;//专业名称    List<ClassroomBean> classroomList;//班级集合  }  Public class ClassroomBean{  String classroom\_id;//班级主键  String major\_id;//专业外键  String classroom\_number;//班级编号  }  如果这些主键有值的就会更新，没有就会新增  } |
| 返回响应 | {type: description:,state:, data:} |

### 4、删除学校

|  |  |
| --- | --- |
| 接口描述 |  |
| 接口URL | **/employment/configCenter/university/delete.do** |
| 请求参数 | {ids: [id,id,id,......]} |
| 返回响应 | {type: description:,state:, data:}} |

# 学生信息设置

### 1、查询学生

|  |  |
| --- | --- |
| 接口描述 |  |
| 接口URL | **/employment/student/query.do** |
| 请求参数 | {queryFormParams:{  String university\_number;//学校编号  String university\_name;//学校名称  String college\_number;//学院编号  String college\_name;//学院名字  String major\_number;//学院编号  String major\_name;//专业名字  String classroom\_number;//班级编号  }  } |
| 返回响应 | {type: description:,state:, “list”:List<StudentCatchInfoViewBean>.toJson:}  Public class StudentCatchInfoViewBean{  String student\_id;//学生主键  String university\_id;  String college\_id;//学院主键  String major\_id;//专业主键  String classroom\_id;//教室主键  //以下是关联数据  String student\_number;//学号  String student\_name;//学生姓名  String id\_card;//身份证  String native\_place;//籍贯  String graduation\_time;//毕业时间  String sex;//性别  String age;//年龄  String university\_number;//学校编号  String university\_name;//学校名字  String college\_number;//学院编号  String college\_name;//学院名字  String major\_number;//专业编号  String major\_name;//专业名字  String classroom\_number;//班级编号 |

### 2、提交、编辑学生

|  |  |
| --- | --- |
| 接口描述 |  |
| 接口URL | **/employment/student/post.do** |
| 请求参数 | {postBean:{  String student\_id;//学生主键  String student\_number;//学号  String student\_name;//学生姓名  String id\_card;//身份证  String native\_place;//籍贯  String university\_id;//学校主键  String college\_id;//学院主键  String major\_id;//专业主键  String classroom\_id;//教室主键  String company\_id;  Date graduation\_time;//毕业时间  String sex;//性别  String age;//年龄  }  如果这些主键有值的就会更新，没有就会新增  } |
| 返回响应 | {type: description:,state:, data:} |

### 3、查询学校

|  |  |
| --- | --- |
| 接口描述 | 给学生设置信息对应的学校，学院，专业，班级 |
| 接口URL | **/employment/configCenter/university/query.do** |
| 请求参数 | {queryFormParams:{  可以传空参数，这样就会查找所有  }  } |
| 返回响应 | {type: description:,state:, “list”:List<UniversityBean>.toJson:}  Public class UniversityBean{  String university\_id;//学校主键  String university\_number;//学校编号  String university\_name;//学校名字  List<CollegeBean> colleges;//学院集合  }  Public class CollegeBean{  String college\_id;//学院主键  String university\_id;//学校外键  String college\_number;//学院编号  String college\_name;//学院名称    List<MajorBean> majors;//专业集合  }  Public class MajorBean{  String major\_id;//专业主键  String college\_id;//学院外键  String major\_number;//专业编号  String major\_name;//专业名称    List<ClassroomBean> classroomList;//班级集合  }  Public class ClassroomBean{  String classroom\_id;//班级主键  String major\_id;//专业外键  String classroom\_number;//班级编号  }  } |

### 4、删除学生

|  |  |
| --- | --- |
| 接口描述 |  |
| 接口URL | **/employment/student/delete.do** |
| 请求参数 | {ids: [id,id,id,......]} |
| 返回响应 | {type: description:,state:, data:}} |

# 公司信息设置

### 1、查询公司

|  |  |
| --- | --- |
| 接口描述 |  |
| 接口URL | **/employment/company/query.do** |
| 请求参数 | {queryFormParams:{  String company\_name;//公司名称  String comapny\_address;//公司地址  } |
| 返回响应 | {type: description:,state:, “list”:List<CompanyBean>.toJson:}  Public class CompanyBean{  String company\_id;//公司主键  String company\_name;//公司名称  String comapny\_address;//公司地址  String manager;//公司负责人  } |

### 2、提交、编辑公司

|  |  |
| --- | --- |
| 接口描述 |  |
| 接口URL | **/employment/company/post.do** |
| 请求参数 | {postBean:{  String company\_id;//公司主键  String company\_name;//公司名称  String comapny\_address;//公司地址  String manager;//公司负责人  }  如果这些主键有值的就会更新，没有就会新增 |
| 返回响应 | {type: description:,state:, data:} |

### 3、删除公司

|  |  |
| --- | --- |
| 接口描述 |  |
| 接口URL | **/employment/company/delete.do** |
| 请求参数 | {ids: [id,id,id,......]} |
| 返回响应 | {type: description:,state:, data:}} |

# 公司信息设置

### 1、查询公司

|  |  |
| --- | --- |
| 接口描述 |  |
| 接口URL | **/employment/company/query.do** |
| 请求参数 | {queryFormParams:{  String company\_name;//公司名称  String comapny\_address;//公司地址  } |
| 返回响应 | {type: description:,state:, “list”:List<CompanyBean>.toJson:}  Public class CompanyBean{  String company\_id;//公司主键  String company\_name;//公司名称  String comapny\_address;//公司地址  String manager;//公司负责人  } |

### 2、提交、编辑公司

|  |  |
| --- | --- |
| 接口描述 |  |
| 接口URL | **/employment/company/post.do** |
| 请求参数 | {postBean:{  String company\_id;//公司主键  String company\_name;//公司名称  String comapny\_address;//公司地址  String manager;//公司负责人  }  如果这些主键有值的就会更新，没有就会新增 |
| 返回响应 | {type: description:,state:, data:} |

### 3、删除公司

|  |  |
| --- | --- |
| 接口描述 |  |
| 接口URL | **/employment/company/delete.do** |
| 请求参数 | {ids: [id,id,id,......]} |
| 返回响应 | {type: description:,state:, data:}} |

# 统计结果

### 1、学校统计

|  |  |
| --- | --- |
| 接口描述 |  |
| 接口URL | **/employment/result/university/query.do** |
| 请求参数 | {queryFormParams:{  String university\_number;//学校编号  String university\_name;//学校名称  } |
| 返回响应 | {type: description:,state:, “list”:List<ResultBean>.toJson:}  Public class ResultBean{  university\_id;//公司主键  university\_number;//公司编号  university\_name;//公司名称  university\_student\_amount;//学生总人数  job\_amount;//就业总人数  job\_proportion;//就业百分比，己经计算好返回  } |

后面的接口已经写完了但是显示方式及参数还需要讨论

### 2、学院统计

### 专业统计

### 4、班级统计