

学生信息管理系统

项目目标

- 复习数据库相关知识：数据库设计、基本CRUD操作，关联查询操作，JDBC操作
- 复习Servlet处理HTTP请求
- 复习HTTP相关知识：请求方法、响应状态码、数据类型等
- 学习项目开发流程

开发环境与技术栈

- Windows/Mac/Linux
- Maven
- Servlet
- MySQL
- Jackson

项目功能

主要业务：管理班级及学生信息

- 用户注册（可屏蔽不实现）
- 用户登录、会话管理
- 班级管理信息（可不实现）
- 寝室管理（可不实现）
- 学生信息管理（可实现部分）

项目演示

用户登录



用户管理（可不实现）

- 查询用户列表

学生信息管理系统

班级管理 学生管理 用户管理 欢迎您xxx

新增 修改 删除

搜索

<input type="checkbox"/>	用户账号	密码	用户昵称	邮箱	创建时间
<input type="checkbox"/>	abc	123	风一样的男子🏠	123@qq.com	2020-06-15 14:06:20
<input type="checkbox"/>	abc1	123	风一样的男子🏠1	123@qq.com	2020-06-15 14:06:20
<input type="checkbox"/>	abc2	123	风一样的男子🏠2	123@qq.com	2020-06-15 14:06:20
<input type="checkbox"/>	abc3	123	风一样的男子🏠3	123@qq.com	2020-06-15 14:06:20
<input type="checkbox"/>	abc4	123	风一样的男子🏠4	123@qq.com	2020-06-15 14:06:20
<input type="checkbox"/>	abc5	123	风一样的男子🏠5	123@qq.com	2020-06-15 14:06:20

• 新增用户

新增用户

* 用户账号:

密码:

用户昵称:

邮箱:

创建时间:

提交

关闭

• 修改用户

修改用户

* 用户账号:

abc1

密码:

123

用户昵称:

风一样的男子🏠1

邮箱:

123@qq.com

创建时间:

2020-06-09 11:53:21

提交

关闭

• 删除用户

学生信息管理系统

新增 修改 删除

<input type="checkbox"/>	用户账号	密码		
<input type="checkbox"/>	abc	123		
<input checked="" type="checkbox"/>	abc1	123	风一样的男子🏠1	123@qq.com
<input checked="" type="checkbox"/>	abc2	123	风一样的男子🏠2	123@qq.com

删除确认

确认要删除选中数据吗?

确认

取消

班级管理（略，类似用户管理）

学生信息管理

- 学生信息查询

学生信息管理系统

班级管理 学生管理 用户管理 欢迎您xxx

新增 修改 删除

搜索

刷新 打印 导出 更多

<input type="checkbox"/>	姓名	学号	身份证号	班级	毕业年份	专业	邮箱	创建时间
<input type="checkbox"/>	小小的梦想🐱A1	s00001	2222222222222222	幼儿园🧒大班	2020届	计算机科学与技术	123@qq.com	2020-06-15 14:06:20
<input type="checkbox"/>	小小的梦想🐱A2	s00002	2222222222222223	幼儿园🧒大班	2020届	计算机科学与技术	123@qq.com	2020-06-15 14:06:20
<input type="checkbox"/>	小小的梦想🐱A3	s00003	2222222222222224	幼儿园🧒大班	2020届	计算机科学与技术	123@qq.com	2020-06-15 14:06:20
<input type="checkbox"/>	小小的梦想🐱A4	s00004	2222222222222225	幼儿园🧒大班	2020届	计算机科学与技术	123@qq.com	2020-06-15 14:06:20
<input type="checkbox"/>	小小的梦想🐱A5	s00005	2222222222222226	幼儿园🧒大班	2020届	计算机科学与技术	123@qq.com	2020-06-15 14:06:20
<input type="checkbox"/>	小小的梦想🐱B1	s00006	2222222222222227	幼儿园🧒中班	2022届	英语系	123@qq.com	2020-06-15 14:06:20
<input type="checkbox"/>	小小的梦想🐱B2	s00007	2222222222222228	幼儿园🧒中班	2022届	英语系	123@qq.com	2020-06-15 14:06:20

- 新增学生信息

新增学生

* 姓名:

学号:

身份证号:

班级:

没有选中任何项

毕业年份:

没有选中任何项

专业:

没有选中任何项

邮箱:

提交

关闭

- 修改学生信息

修改学生

* 姓名:

小小的梦想🐱A2

学号:

s00002

身份证号:

2222222222222223

班级:

幼儿园🧒大班

毕业年份:

2020届

专业:

计算机科学与技术

邮箱:

123@qq.com

创建时间:

2020-06-15 14:06:20

提交

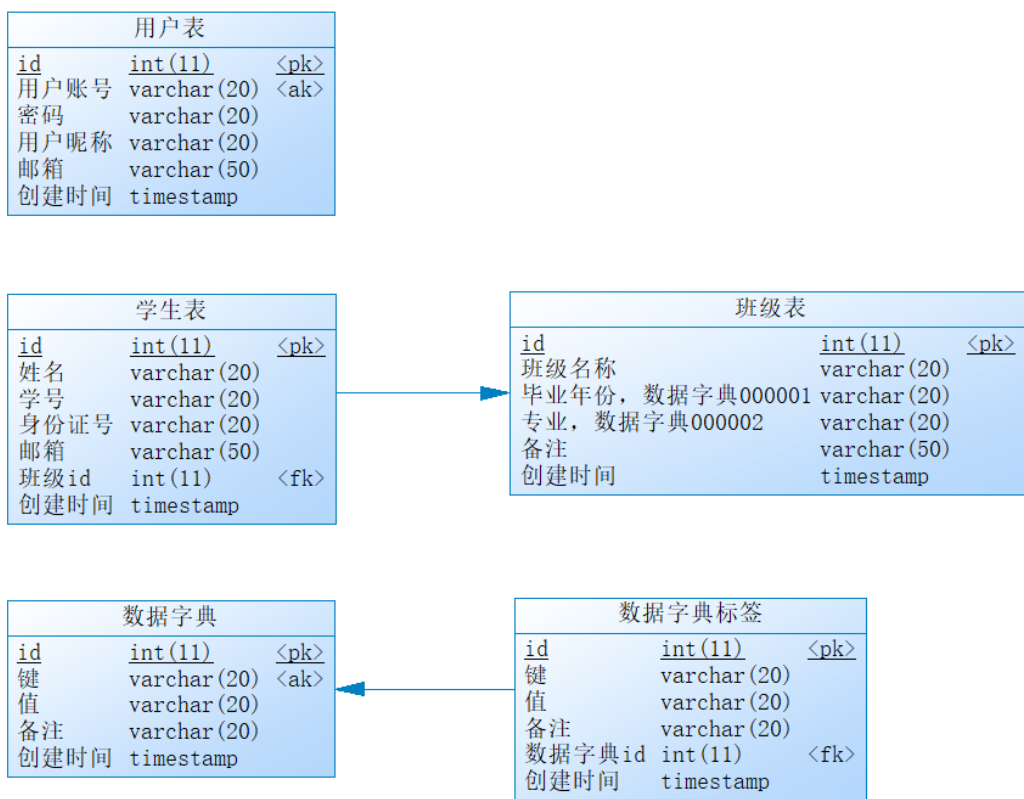
关闭

- 删除学生信息



数据库设计

数据库表关系图



以上关系可见：

2. 班级和学生信息表为一对多关系
3. 数据字典表和数据字典标签表为一对多关系

字典表说明

数据字典表和数据字典标签表主要用在一些通用的下拉菜单选项。

如班级表中的字段：学生是哪一届的，学生专业

- 如果单独设计需要单独一张表，实际存放的数据也不会太多，可以考虑设计在整体的一张表。
- 页面上展示时，这些字段也是下拉选项。
- 一般在设计上考虑为两张表：数据字典表和数据字典标签表（一对多关系）来保存。两张表都是 key、value 的形式，字典表是父节点，字典标签表是子节点，下拉菜单通过父节点的 key 查询出所有关联的子节点，再使用子节点的 key、value 进行下拉菜单选项的初始化

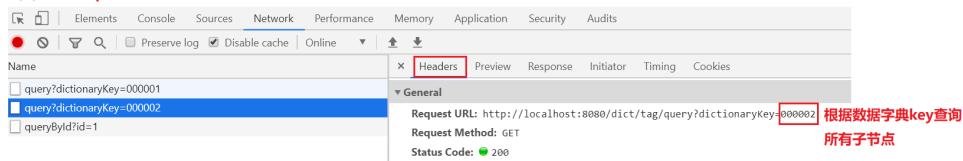
比如一个下拉菜单，页面元素-接口请求/响应-数据库关系如下：

专业: 英语系

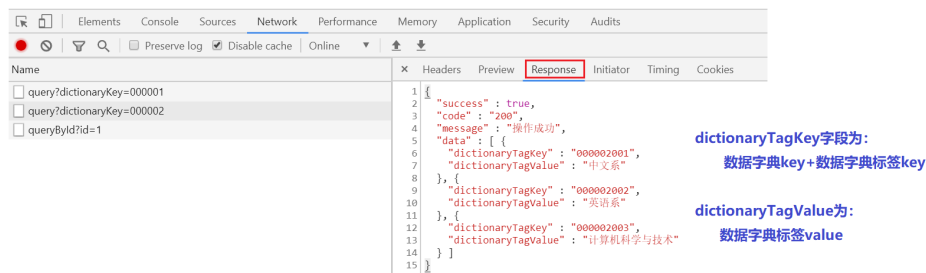
- 中文系
- 英语系
- 计算机科学与技术

下拉菜单的选项都是通过发送http请求，从数据库查询出来的

(1) 发送http请求



(2) 响应内容



(3) 根据http响应的内容，动态的生成下拉菜单选项

```
<div class="form-group col-md-6 align-items-start mt-3 p-0">
  <label for="classes_table_toolbar_update_form_classesMajor" class="col-md-4 p-0 mt-1" style="display: inline; text-align:right">专业:
</label>
  <div class="dropdown bootstrap-select form-control col-md-7">
    <select class="selectpicker form-control col-md-7" id="classes_table_toolbar_update_form_classesMajor" name=
"classes_table_toolbar_update_form_classesMajor" data-container="#classes_table_toolbar_update_modal" tabindex="-98">
      <option value="000002001">中文系</option>
      <option value="000002002">英语系</option>
      <option value="000002003">计算机科学与技术</option>
    </select>
  </div>
</div>
```

父key+子key 子value

一般页面写，但表达的含义和数据字典value一致

创建数据库及表

```
drop database if exists stu_info;
create database stu_info character set utf8mb4;

use stu_info;

drop table if exists user;
create table user(
  id int primary key auto_increment,
  username varchar(20) not null unique comment '用户账号',
  password varchar(20) not null comment '密码',
  nickname varchar(20) comment '用户昵称',
  email varchar(50) comment '邮箱',
  create_time timestamp default NOW() comment '创建时间'
) comment '用户表';

drop table if exists dictionary;
create table dictionary(
  id int primary key auto_increment,
  dictionary_key varchar(20) not null unique comment '键',
  dictionary_value varchar(20) not null comment '值',
  dictionary_desc varchar(20) comment '备注',
  create_time timestamp default NOW() comment '创建时间'
) comment '数据字典';

drop table if exists dictionary_tag;
create table dictionary_tag(
```

```

        id int primary key auto_increment,
        dictionary_tag_key varchar(20) not null comment '键',
        dictionary_tag_value varchar(20) not null comment '值',
        dictionary_tag_desc varchar(20) comment '备注',
        dictionary_id int comment '数据字典id',
        create_time timestamp default NOW() comment '创建时间',
        foreign key (dictionary_id) references dictionary(id)
    ) comment '数据字典标签';

drop table if exists classes;
create table classes(
    id int primary key auto_increment,
    classes_name varchar(20) not null comment '班级名称',
    classes_graduate_year varchar(20) comment '毕业年份, 数据字典000001',
    classes_major varchar(20) comment '专业, 数据字典000002',
    classes_desc varchar(50) comment '备注',
    create_time timestamp default NOW() comment '创建时间'
) comment '班级表';

drop table if exists student;
create table student(
    id int primary key auto_increment,
    student_name varchar(20) not null comment '姓名',
    student_no varchar(20) comment '学号',
    id_card varchar(20) comment '身份证号',
    student_email varchar(50) comment '邮箱',
    classes_id int comment '班级id',
    create_time timestamp default NOW() comment '创建时间',
    foreign key (classes_id) references classes(id)
) comment '学生表';

```

初始化数据

```

use stu_info;
-- 初始化数据
-- mysql中没有==, 是用=号代替==。为了区分=和==, 赋值时使用:=
set @username:='abc';
set @password:='123';
set @nickname:='风一样的男子🌪';
set @email:='123@qq.com';

set @dictionary_classes_graduate_year='000001';
set @dictionary_classes_major='000002';

set @classes_name:='幼儿园👶';
set @student_name:='小小的梦想🌟';

insert into user(username, nickname, password, email) values (@username,
@nickname, @password, @email);
insert into user(username, nickname, password, email) values (concat(@username,
'1'), concat(@nickname, '1'), @password, @email);
insert into user(username, nickname, password, email) values (concat(@username,
'2'), concat(@nickname, '2'), @password, @email);
insert into user(username, nickname, password, email) values (concat(@username,
'3'), concat(@nickname, '3'), @password, @email);
insert into user(username, nickname, password, email) values (concat(@username,
'4'), concat(@nickname, '4'), @password, @email);

```

```
insert into user(username, nickname, password, email) values (concat(@username, '5'), concat(@nickname, '5'), @password, @email);
```

数据字典: 学生毕业年份

```
insert into dictionary(dictionary_key, dictionary_value, dictionary_desc) values (@dictionary_classes_graduate_year, '毕业年份', '学生毕业的年份');
```

```
insert into dictionary_tag(dictionary_tag_key, dictionary_tag_value, dictionary_id) values ('001', '2020届', 1);
```

```
insert into dictionary_tag(dictionary_tag_key, dictionary_tag_value, dictionary_id) values ('002', '2021届', 1);
```

```
insert into dictionary_tag(dictionary_tag_key, dictionary_tag_value, dictionary_id) values ('003', '2022届', 1);
```

```
insert into dictionary_tag(dictionary_tag_key, dictionary_tag_value, dictionary_id) values ('004', '2023届', 1);
```

数据字典: 学生专业

```
insert into dictionary(dictionary_key, dictionary_value, dictionary_desc) values (@dictionary_classes_major, '专业', '学生的专业');
```

```
insert into dictionary_tag(dictionary_tag_key, dictionary_tag_value, dictionary_id) values ('001', '中文系', 2);
```

```
insert into dictionary_tag(dictionary_tag_key, dictionary_tag_value, dictionary_id) values ('002', '英语系', 2);
```

```
insert into dictionary_tag(dictionary_tag_key, dictionary_tag_value, dictionary_id) values ('003', '计算机科学与技术', 2);
```

```
insert into classes(classes_name, classes_graduate_year, classes_major, classes_desc) values (concat(@classes_name, '大班'), '000001001', '000002003', '已经4岁, 即将成为大人的大班同学, 不再是3岁小孩子了呀');
```

```
insert into classes(classes_name, classes_graduate_year, classes_major, classes_desc) values (concat(@classes_name, '中班'), '000001002', '000002003', '虽然还是3岁的小孩子, 但也开始思考人生了呢');
```

```
insert into classes(classes_name, classes_graduate_year, classes_major, classes_desc) values (concat(@classes_name, '小班'), '000001003', '000002003', '挂着鼻涕的跟屁虫');
```

```
insert into student(student_name, student_no, id_card, student_email, classes_id) values (concat(@student_name, 'A1'), 's00001', '2222222222222222', @email, 1);
```

```
insert into student(student_name, student_no, id_card, student_email, classes_id) values (concat(@student_name, 'A2'), 's00002', '2222222222222223', @email, 1);
```

```
insert into student(student_name, student_no, id_card, student_email, classes_id) values (concat(@student_name, 'A3'), 's00003', '2222222222222224', @email, 1);
```

```
insert into student(student_name, student_no, id_card, student_email, classes_id) values (concat(@student_name, 'A4'), 's00004', '2222222222222225', @email, 1);
```

```
insert into student(student_name, student_no, id_card, student_email, classes_id) values (concat(@student_name, 'A5'), 's00005', '2222222222222226', @email, 1);
```

```
insert into student(student_name, student_no, id_card, student_email, classes_id) values (concat(@student_name, 'B1'), 's00006', '2222222222222227', @email, 2);
```

```
insert into student(student_name, student_no, id_card, student_email, classes_id) values (concat(@student_name, 'B2'), 's00007', '2222222222222228', @email, 2);
```

```
insert into student(student_name, student_no, id_card, student_email,
classes_id) values (concat(@student_name, 'B3'), 's00008', '2222222222222229',
@email, 2);
insert into student(student_name, student_no, id_card, student_email,
classes_id) values (concat(@student_name, 'B4'), 's00009', '2222222222222230',
@email, 2);
insert into student(student_name, student_no, id_card, student_email,
classes_id) values (concat(@student_name, 'B5'), 's00010', '2222222222222231',
@email, 2);
```

前后端接口

要实现功能，需要先明确前后端约定好的接口，我们主要实现图书借阅信息管理，所以只实现该业务相关接口。

需要说明的是，接口的定义一般是前后端约定好的，所以也和前端代码息息相关，前端需要什么数据，需要什么格式的数据，也会在接口中体现。

查询专业字典

请求

```
GET dict/tag/query?dictionaryKey=000002
```

响应

```
{
  "success" : true,
  "code" : "200",
  "message" : "操作成功",
  "data" : [ {
    "dictionaryTagKey" : "000002001",
    "dictionaryTagValue" : "中文系"
  }, {
    "dictionaryTagKey" : "000002002",
    "dictionaryTagValue" : "英语系"
  }, {
    "dictionaryTagKey" : "000002003",
    "dictionaryTagValue" : "计算机科学与技术"
  } ]
}
```

查询毕业年份字典

请求

```
GET dict/tag/query?dictionaryKey=000001
```

响应

```
{
  "success" : true,
  "code" : "200",
  "message" : "操作成功",
```



```
"data" : [ {
  "dictionaryTagKey" : "000001001",
  "dictionaryTagValue" : "2020届"
}, {
  "dictionaryTagKey" : "000001002",
  "dictionaryTagValue" : "2021届"
}, {
  "dictionaryTagKey" : "000001003",
  "dictionaryTagValue" : "2022届"
}, {
  "dictionaryTagKey" : "000001004",
  "dictionaryTagValue" : "2023届"
} ]
}
```

查询学生信息

请求

```
GET student/query?
searchText=&sortOrder=asc&pageSize=7&pageNumber=1&_=1594089188797
```

响应

```
{
  "success" : true,
  "code" : "200",
  "message" : "操作成功",
  "total" : 10,
  "data" : [ {
    "id" : 1,
    "createTime" : "2020-06-15 14:06:20",
    "studentName" : "小小的梦想\uD83D\uDC37A1",
    "studentNo" : "s00001",
    "idCard" : "2222222222222222",
    "studentEmail" : "123@qq.com",
    "classesId" : 1,
    "classes" : {
      "id" : 1,
      "createTime" : "2020-06-15 14:06:20",
      "className" : "幼儿园\uD83D\uDE02大班",
      "classesGraduateYear" : "000001001",
      "classesMajor" : "000002003",
      "classesDesc" : "已经4岁，即将成为大人的大班同学，不再是3岁小孩子了呀"
    }
  }, {
    "id" : 2,
    "createTime" : "2020-06-15 14:06:20",
    "studentName" : "小小的梦想\uD83D\uDC37A2",
    "studentNo" : "s00002",
    "idCard" : "2222222222222223",
    "studentEmail" : "123@qq.com",
    "classesId" : 1,
    "classes" : {
      "id" : 1,
      "createTime" : "2020-06-15 14:06:20",
```

```
    "className" : "幼儿园\uD83D\uDE02大班",
    "classesGraduateYear" : "000001001",
    "classesMajor" : "000002003",
    "classesDesc" : "已经4岁，即将成为大人的大班同学，不再是3岁小孩子了呀"
  }
}, {
  "id" : 3,
  "createTime" : "2020-06-15 14:06:20",
  "studentName" : "小小的梦想\uD83D\uDC37A3",
  "studentNo" : "s00003",
  "idCard" : "2222222222222224",
  "studentEmail" : "123@qq.com",
  "classesId" : 1,
  "classes" : {
    "id" : 1,
    "createTime" : "2020-06-15 14:06:20",
    "className" : "幼儿园\uD83D\uDE02大班",
    "classesGraduateYear" : "000001001",
    "classesMajor" : "000002003",
    "classesDesc" : "已经4岁，即将成为大人的大班同学，不再是3岁小孩子了呀"
  }
}, {
  "id" : 4,
  "createTime" : "2020-06-15 14:06:20",
  "studentName" : "小小的梦想\uD83D\uDC37A4",
  "studentNo" : "s00004",
  "idCard" : "2222222222222225",
  "studentEmail" : "123@qq.com",
  "classesId" : 1,
  "classes" : {
    "id" : 1,
    "createTime" : "2020-06-15 14:06:20",
    "className" : "幼儿园\uD83D\uDE02大班",
    "classesGraduateYear" : "000001001",
    "classesMajor" : "000002003",
    "classesDesc" : "已经4岁，即将成为大人的大班同学，不再是3岁小孩子了呀"
  }
}, {
  "id" : 5,
  "createTime" : "2020-06-15 14:06:20",
  "studentName" : "小小的梦想\uD83D\uDC37A5",
  "studentNo" : "s00005",
  "idCard" : "2222222222222226",
  "studentEmail" : "123@qq.com",
  "classesId" : 1,
  "classes" : {
    "id" : 1,
    "createTime" : "2020-06-15 14:06:20",
    "className" : "幼儿园\uD83D\uDE02大班",
    "classesGraduateYear" : "000001001",
    "classesMajor" : "000002003",
    "classesDesc" : "已经4岁，即将成为大人的大班同学，不再是3岁小孩子了呀"
  }
}, {
  "id" : 6,
  "createTime" : "2020-06-15 14:06:20",
  "studentName" : "小小的梦想\uD83D\uDC37B1",
  "studentNo" : "s00006",
```

```

    "idCard" : "2222222222222227",
    "studentEmail" : "123@qq.com",
    "classesId" : 2,
    "classes" : {
      "id" : 2,
      "createTime" : "2020-06-15 14:06:20",
      "className" : "幼儿园\uD83D\uDE02中班",
      "classesGraduateYear" : "000001003",
      "classesMajor" : "000002002",
      "classesDesc" : "虽然还是3岁的小孩子，但也开始思考人生了呢"
    }
  }, {
    "id" : 7,
    "createTime" : "2020-06-15 14:06:20",
    "studentName" : "小小的梦想\uD83D\uDC37B2",
    "studentNo" : "s00007",
    "idCard" : "2222222222222228",
    "studentEmail" : "123@qq.com",
    "classesId" : 2,
    "classes" : {
      "id" : 2,
      "createTime" : "2020-06-15 14:06:20",
      "className" : "幼儿园\uD83D\uDE02中班",
      "classesGraduateYear" : "000001003",
      "classesMajor" : "000002002",
      "classesDesc" : "虽然还是3岁的小孩子，但也开始思考人生了呢"
    }
  }
]
}

```

查询班级（数据字典）

使用在下拉菜单中，要求按照数据字典响应格式返回

请求

```
GET classes/queryAsDict
```

响应

```

{
  "success" : true,
  "code" : "200",
  "message" : "操作成功",
  "data" : [ {
    "dictionaryTagKey" : "1",
    "dictionaryTagValue" : "幼儿园\uD83D\uDE02大班",
    "classesGraduateYear" : "000001001",
    "classesMajor" : "000002003"
  }, {
    "dictionaryTagKey" : "2",
    "dictionaryTagValue" : "幼儿园\uD83D\uDE02中班",
    "classesGraduateYear" : "000001003",
    "classesMajor" : "000002002"
  }, {
    "dictionaryTagKey" : "3",

```

```
"dictionaryTagValue" : "幼儿园\uD83D\uDE02小班",
"classesGraduateYear" : "000001003",
"classesMajor" : "000002003"
} ]
}
```

新增学生信息

请求

```
POST student/add
Content-Type: application/json

{"studentName":"阿sir","studentNo":"1111","idCard":"32432423","classesId":"2","studentEmail":""}
```

响应

```
{
  "success" : true,
  "code" : "200",
  "message" : "操作成功"
}
```

删除学生信息

请求

```
GET student/delete?ids=2&ids=3
```

响应

```
{
  "success" : true,
  "code" : "200",
  "message" : "操作成功"
}
```

查询学生详情

请求

```
GET student/queryById?id=4
```

响应

```
{
  "success" : true,
  "code" : "200",
  "message" : "操作成功",
  "data" : {
    "id" : 4,
    "createTime" : "2020-06-15 14:06:20",
  }
}
```

```
"studentName" : "小小的梦想\uD83D\uDC37A4",
"studentNo" : "s00004",
"idCard" : "2222222222222225",
"studentEmail" : "123@qq.com",
"classesId" : 1,
"classes" : {
  "id" : 1,
  "createTime" : "2020-06-15 14:06:20",
  "className" : "幼儿园\uD83D\uDE02大班",
  "classesGraduateYear" : "000001001",
  "classesMajor" : "000002003",
  "classesDesc" : "已经4岁，即将成为大人的大班同学，不再是3岁小孩子了呀"
}
}
```

修改学生信息

请求

```
POST student/update
Content-Type: application/json

{"id":"4","studentName":"小小的梦想🐱A4","studentNo":"s00004","idCard":"2222222222222225","classesId":"3","studentEmail":"123@qq.com"}
```

响应

```
{
  "success" : true,
  "code" : "200",
  "message" : "操作成功"
}
```

开发环境准备

Maven环境配置

1. 下载文件

下载上传的settings.xml文件和本地仓库压缩包M2Repository.zip，确保存放在本地某个路径中，之后别移动了。并解压压缩包，解压后是这样的：

2. 打开配置面板

3. IDEA配置Maven

IDEA其他配置

参考课件《15. IDEA配置.pdf》，要求项目及IDEA配置部分内容全部完成。

提前安装lombok插件：

准备前端资源

代码设计

设计数据库实体类

设计http请求基类

主要针对前端表格中，可以输入文本搜索，并根据页码显示列表数据

```
import Lombok.Getter;  
import Lombok.Setter;  
  
@Getter  
@Setter  
public class BaseEntity {
```

```
private Integer pageNumber;

private Integer pageSize;

private String searchText;

private String sortOrder;
}
```

设计统一响应类

```
import com.github.pagehelper.Page;
import lombok.Getter;
import lombok.Setter;
import lombok.ToString;

import java.io.PrintWriter;
import java.io.StringWriter;

@Getter
@Setter
@ToString
public class ResponseResult {

    private boolean success;

    private String code;

    private String message;

    private Long total;

    private Object data;

    private String stackTrace;
}
```

设计工具类

可扩展部分

功能上其他面板中的功能都可以实现

session及权限校验可以考虑使用过滤器实现