



**（3243700）**

Java程序设计

课程设计报告

**题目： 大学课堂点名系统**

**学院： 信息科学与技术学院**

**教师： 何太军**

**姓名： 江胤佐**

**学号： 2018112664**

**班级： 软件2018-01班**

**2019-2020（1）**

**目 录**

分页（如果有目录，则目录与正文分页）

**正文**

主要包括任务内容及要求、分析、设计（概要设计、详细设计、关键算法设计等）、实现（程序结构、各类的设计、实现效果、关键代码段）、部署及使用说明等相关内容。

实验报告主要包括：实验内容指实验要完成的任务及程度；实验过程指实验分析、实验步骤、重要的类或数据结构、算法的设计、界面设计、程序结构、实验中间结果（运行结果截图）等；实验结果（最终结果要截图）包括结果介绍、评价；实验体会主要写收获与存在的问题及解决思路等，要体现出条理，不要胡乱拼凑。如果有其他建议或意见请补充到报告后边。附录附上关键代码。

正文字体：建议宋体五号，注意插图、插表要排版整齐，插图不要太大，以图上的文字与正文差不多大、幅面以能够体现要说明的问题即可（实验报告的规范格式也是考核项）。

提交要求为word文档。

实验报告的详细程度应以参照实验报告步骤及主要说明能再次重现实验为准。

# 一、实验内容

## 1.1 任务要求

设计一个大学课堂点名系统，用于课堂点名、分析、课堂提问记录，汇总点名结果。主要功能有：教学班设置、教学班名单、课程表设置、课堂点名操作、课堂提问。涉及技术有：Java面向对象程序设计、GUI、IO、数据库、多线程、异常、集合...

实验需每周在码云的个人仓库上提交代码。

## 1.2 完成程度

分别使用JavaFX和控制台、MySQL数据库和文件来完成上述任务要求中的功能。

# 二、实验过程

## 2.1 需求分析

### 2.1.1功能需求

程序包括学生端和教师端：学生端需具备查看课表、查看点名记录等功能；教师端需具备查看课表、教学班设置、编辑教学班名单、点名（顺序点名、异常点名、二次点名、随机点名）、提问、编辑学生等功能。详细的功能需求如下图所示。



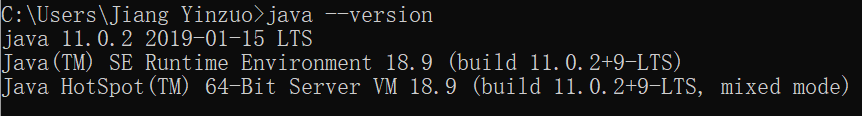
### 2.1.2非功能需求

界面简介美观、交互友好、程序运行流畅。

## 2.2 开发环境与项目依赖

操作系统：Windows 10

Java版本：java 11.0.2 2019-01-15 LTS



MySQL版本：mysql Ver 8.0.16 for Win64 on x86\_64 (MySQL Community Server - GPL)



jar包依赖：

Java操作Excel：jxl-2.6.12

Java操作MySQL：mysql-connector-java-8.0.18

JavaFX：openjfx-13.0.1\_windows-x64\_bin-sdk

开发工具：

IDE：Eclipse Jee 2019-09、IntelliJ IDEA 2019.2.2 x64

数据库可视化软件：Navicat Premium 12

FXML可视化编辑：SceneBuilder

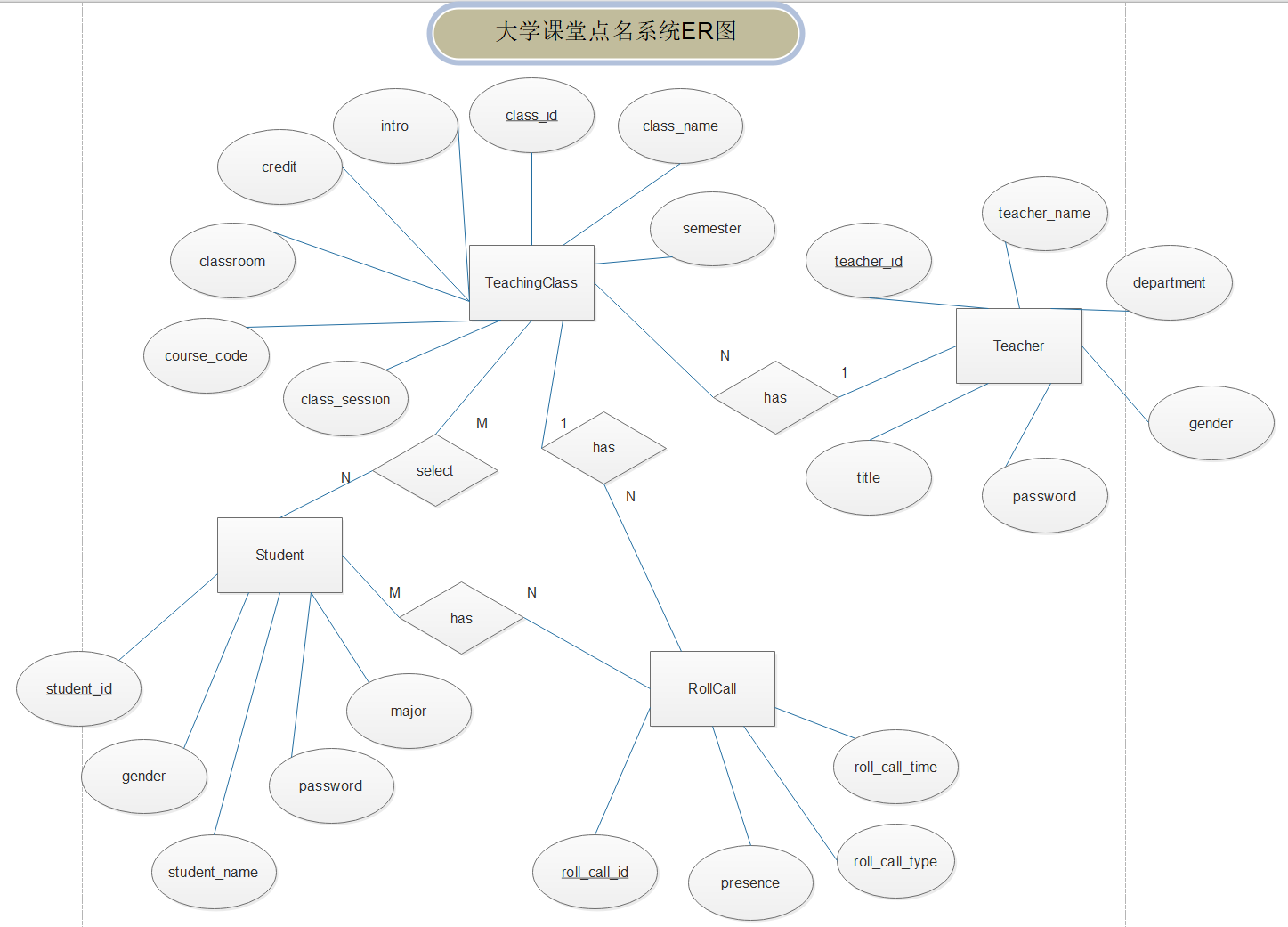
## 2.3 系统概要设计



系统业务流程图如上所示。

## 2.4 数据设计

### 2.4.1 实体关系设计



实体类说明：

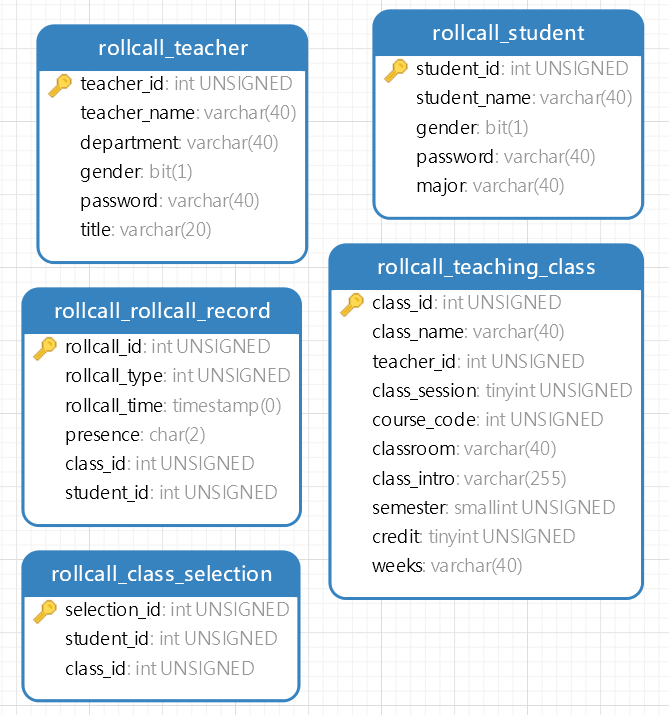
实体包括学生（Student）、点名记录（RollCall）、教师（Teacher）、教学班（TeachingClass）。实体关系及实体属性如上图所示。

### 2.4.2 文件存储设计

通过Java反序列化将实体类存入.txt文件；通过序列化从.txt文件中读取实体类。文件存放在运行程序的当前目录下的files中。包括rollcalls.txt、student.txt、teachers.txt、teachingClasses.txt、classSelection.txt。

### 2.4.3 数据库设计

数据库包含教师表（rollcall\_teacher）、学生表(rollcall\_student)、点名记录表(rollcall\_rollcall\_record)、教学班表（rollcall\_teaching\_class）、选课记录表（rollcall\_class\_selection，同时也是学生—教学班关系表）。数据库模型如下图所示。



各表的DDL如下：

教学班表

CREATE TABLE `rollcall\_teaching\_class` (

`class\_id` int(9) unsigned zerofill NOT NULL AUTO\_INCREMENT COMMENT '教学班id',

`class\_name` varchar(40) NOT NULL COMMENT '课程名称',

`teacher\_id` int(9) unsigned zerofill NOT NULL COMMENT '任课教师id',

`class\_session` tinyint(2) unsigned NOT NULL COMMENT '上课讲次，第一个数代表周几，第二个数代表第几讲',

`course\_code` int(6) unsigned zerofill NOT NULL COMMENT '课程代码',

`classroom` varchar(40) NOT NULL DEFAULT '' COMMENT '教室',

`class\_intro` varchar(255) NOT NULL DEFAULT '' COMMENT '课程简介',

`semester` smallint(5) unsigned zerofill NOT NULL COMMENT '上课学期',

`credit` tinyint(3) unsigned NOT NULL COMMENT '学分',

`weeks` varchar(40) NOT NULL DEFAULT '' COMMENT '周次',

PRIMARY KEY (`class\_id`)

) ENGINE=InnoDB AUTO\_INCREMENT=670 DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4\_0900\_ai\_ci COMMENT='教学班表';

点名记录表

CREATE TABLE `rollcall\_rollcall\_record` (

`rollcall\_id` int(9) unsigned zerofill NOT NULL AUTO\_INCREMENT COMMENT '点名id',

`rollcall\_type` int(4) unsigned NOT NULL COMMENT '点名类型 1: 点名; 2: 提问',

`rollcall\_time` timestamp NOT NULL DEFAULT CURRENT\_TIMESTAMP COMMENT '点名时间',

`presence` char(2) NOT NULL COMMENT '点名情况',

`class\_id` int(9) unsigned NOT NULL COMMENT '教学班id',

`student\_id` int(9) unsigned zerofill NOT NULL COMMENT '学号',

PRIMARY KEY (`rollcall\_id`)

) ENGINE=InnoDB AUTO\_INCREMENT=541 DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4\_0900\_ai\_ci COMMENT='点名表';

教师表

CREATE TABLE `rollcall\_teacher` (

`teacher\_id` int(9) unsigned NOT NULL COMMENT '教师工号',

`teacher\_name` varchar(40) NOT NULL COMMENT '教师姓名',

`department` varchar(40) NOT NULL COMMENT '教师所属部门',

`gender` bit(1) NOT NULL COMMENT '1: 男; 0: 女',

`password` varchar(40) NOT NULL COMMENT '密码',

`title` varchar(20) NOT NULL COMMENT '职称',

PRIMARY KEY (`teacher\_id`)

) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4\_0900\_ai\_ci COMMENT='教师表';

学生表

CREATE TABLE `rollcall\_student` (

`student\_id` int(9) unsigned NOT NULL COMMENT '学号',

`student\_name` varchar(40) NOT NULL COMMENT '学生姓名',

`gender` bit(1) NOT NULL COMMENT '1: 男; 0: 女',

`password` varchar(40) CHARACTER SET utf8mb4 COLLATE utf8mb4\_0900\_ai\_ci NOT NULL DEFAULT '123456' COMMENT '密码',

`major` varchar(40) NOT NULL COMMENT '专业',

PRIMARY KEY (`student\_id`)

) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4\_0900\_ai\_ci COMMENT='学生表';

选课记录表（学生—教学班关系表）

CREATE TABLE `rollcall\_class\_selection` (

`selection\_id` int(9) unsigned zerofill NOT NULL AUTO\_INCREMENT COMMENT '选课操作id',

`student\_id` int(9) unsigned NOT NULL COMMENT '学号',

`class\_id` int(9) unsigned zerofill NOT NULL COMMENT '课程id',

PRIMARY KEY (`selection\_id`)

) ENGINE=InnoDB AUTO\_INCREMENT=19 DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4\_0900\_ai\_ci COMMENT='选课表（学生-教学班关系表）';

## 2.5 系统架构设计

### 2.5.1 软件架构设计



系统采用分层架构，总体可分为DAO层、Service层和UI层。

持久化存储辅助类FileHelper和MySQLHelper分别对Java文件IO、JDBC进行了封装，实现了实体类（Entity）和数据传输对象（DTO）与持久化存储（文件、数据库表的字段）之间的映射。

DAO层借助持久化存储的辅助类分别对文件和MySQL数据库进行持久化操作。DAO层定义了一系列接口，由文件实现类和MySQL实现类分别操作文件和数据库，对Service层接口一致，体现了面向接口编程的思想。

Service层是对DAO层进一步的封装，向UI层提供业务逻辑方法接口。Service层通过工厂方法和反射来读取配置文件，并动态创建DAO的实现类。

UI层包括控制台UI和JavaFX UI。

控制台UI借助控制台IO辅助类来完成用户交互。UI容器类负责创建UI对象，并加载界面。

JavaFX UI采用MVC模式。Controller实现用户交互的业务逻辑，调用Service层的持久化操作接口，并借助FXML加载工具类和舞台跳转管理类来加载FXML、CSS、图片等静态资源文件。

### 2.5.2 Java包规划

pers

└─jiangyinzuo

└─rollcall

├─common

│ CustomException.java // 自定义异常处理类

│

├─config

│ Config.java // 全局配置类

│

├─dao // DAO层接口

│ │ RollCallDao.java

│ │ StudentDao.java

│ │ TeacherDao.java

│ │ TeachingClassDao.java

│ │

│ ├─fileimpl // DAO层文件实现类

│ │ RollCallDaoFileImpl.java

│ │ StudentDaoFileImpl.java

│ │ TeacherDaoFileImpl.java

│ │ TeachingClassDaoFileImpl.java

│ │

│ └─mysqlimpl // DAO层数据库实现类

│ RollCallDaoMysqlImpl.java

│ StudentDaoMysqlImpl.java

│ TeacherDaoMysqlImpl.java

│ TeachingClassDaoMysqlImpl.java

│

├─domain // 领域对象模型

│ ├─dto // 数据传输对象，在实体类无法满足需求时使用

│ │ ClassSelectionRecordDTO.java

│ │ StudentRollCallResultDTO.java

│ │

│ ├─entity // 实体类

│ │ RollCall.java

│ │ Student.java

│ │ Teacher.java

│ │ TeachingClass.java

│ │

│ ├─mapper // 实体类或DTO映射到数据库表字段的注解

│ │ FieldMapper.java

│ │ TableMapper.java

│ │

│ └─repository // 同实体类密切相关的持久化操作

│ RollCallRepo.java

│ StudentRepo.java

│ TeachingClassRepo.java

│

├─factory // 创建DAO对象的工厂

│ DaoFactory.java

│

├─helper // 持久化存储和控制台IO的辅助类

│ ConsoleIoHelper.java

│ FileHelper.java

│ MySqlHelper.java

│

├─service // Service层接口

│ │ LoginService.java

│ │ RollCallService.java

│ │ StudentService.java

│ │ TeachingClassService.java

│ │

│ └─impl // Service层实现类

│ LoginServiceImpl.java

│ RollCallServiceImpl.java

│ StudentServiceImpl.java

│ TeachingClassServiceImpl.java

│

├─ui // UI层

│ ├─common

│ │ RollCallManager.java // 控制台UI和JavaFX UI

// 共同的点名管理类

│ │

│ ├─console // 控制台UI

│ │ │ AbstractUi.java // UI的抽象基类

│ │ │ LoginUi.java

│ │ │ Main.java

│ │ │ UiContainer.java // 创建UI的容器类

│ │ │

│ │ ├─student // 学生界面UI

│ │ │ StudentMainUi.java

│ │ │

│ │ └─teacher // 教师界面UI

│ │ RollCallUi.java

│ │ TeacherMainUi.java

│ │ TeachingClassDetailUi.java

│ │

│ ├─javafx // JavaFX UI

│ │ │ Main.java // JavaFX应用的入口类

│ │ │

│ │ ├─common

│ │ │ CustomAlertBoard.java // 自定义消息弹框

│ │ │

│ │ ├─controller //作为场景加载的 JavaFX 控制器

│ │ │ │ AddTeachingClassController.java

│ │ │ │ EditClassController.java

│ │ │ │ FxController.java

│ │ │ │ LoginController.java

│ │ │ │ RollCallBoardController.java

│ │ │ │ RollCallDetailController.java

│ │ │ │ RollCallRecordController.java

│ │ │ │ RollCallStatisticController.java

│ │ │ │ ScheduleController.java

│ │ │ │ StudentListController.java

│ │ │ │ StudentMainBoardController.java

│ │ │ │ StudentManagementController.java

│ │ │ │ TeacherMainBoardController.java

│ │ │ │

│ │ │ ├─components // 作为面板加载的JavaFX 控制器

│ │ │ │ RollCallCmpController.java

│ │ │ │ RollCallRecordCmpController.java

│ │ │ │ RollCallStatisticCmpController.java

│ │ │ │ StudentCmpController.java

│ │ │ │ TeachingClassCmpController.java

│ │ │ │

│ │ │ └─proxy // 控制器访问代理类

│ │ │ ControllerProxy.java

│ │ │

│ │ ├─router // 舞台跳转管理类

│ │ │ StageManager.java

│ │ │

│ │ └─utils // FXML面板加载工具类

│ │ FxmlCmpLoaderUtil.java

│ │

│ └─state // 全局状态管理类，存储用户信息、当前教学班等

│ SelectedTeachingClassState.java

│ UserState.java

│

└─util

DeepCopy.java // 对象深拷贝工具类

## 2.6 系统详细设计与实现

### 2.6.1 持久化存储辅助类实现

#### 2.6.1.1 文件读写辅助类实现

pers.jiangyinzuo.rollcall.helper.FileHelper辅助类是对Java文件IO的简单封装，主要包括读和写操作。