

2025 年数据库系统原理实践任务书

1 课程介绍

“数据库系统原理实践”是配合“数据库系统原理”课程独立开设的实践课，注重理论与实践相结合。本课程以 MySQL 为例，系统性地设计了一系列的实训任务，基础内容涉及以下几个部分，并可结合实际对 DBMS 原理的掌握情况向内核设计延伸：

- 1) 数据库、表、索引、视图、约束、存储过程、函数、触发器、游标等数据对象的管理与编程；
- 2) 数据查询，数据插入、删除与修改等数据处理相关任务；
- 3) 数据库的安全性控制，完整性控制，恢复机制，并发控制机制等系统内核的实验；
- 4) 数据库的设计与实现；
- 5) 数据库应用系统的开发(JAVA 篇)。

课程依托头歌实践教学平台，实践课程 url 见相关课堂教师发布链接及其邀请码。实验环境为 Linux 操作系统下的 MySQL 8.0.28（主要为 8.028 版本，部分关卡使用 8.022 版本，使用中基本无差别）。在数据库应用开发环节，使用 JAVA 1.8。

在实践进行过程中，大部分时间同学们会面对这样的界面：



同学们可以尽情在实践环境中大展身手，以提升个人能力，当然也要尽最大努力去完成本次实践的所有关卡。如果学习过程中遇到问题，首先学会自己查资料解决，然后求助老师。

本课程相关资料网站：

MYSQL 手册：<https://dev.mysql.com/doc/>

JAVA 手册：<https://docs.oracle.com/javase/8/docs/api/index.html>

课程开放资源：<https://gitee.com/kylin8575543/db2022-spring>

实践中遇到平台故障或其他问题，请及时联系各班老师及助教。

2 实践环节

2.1 任务组成及选题要求

登录头歌上的本课程平台后，将会看到总体任务由一系列实训任务构成，而进入每个实训后将会看到其由若干关卡组成，关卡依据其难易程度和工作量会有不同的分值。依据每个关卡布置的任务，在头歌界面输入相应答题代码后运行头歌的测试功能，系统会自动依据运行结果评判该关卡是否通过，并记录相应分值。

本次课程，若所有关卡全部完成，将会获得头歌平台总分超过 100 分，**然而课程考核并不要求所有关卡全部完成，大部分关卡同学们可依据自身兴趣和能力选择完成或者跳过不做（但对有些实训或者关卡则有具体的必做要求，本任务书接下来将会具体说明），最终程序检查满分只计头哥平台中的 100 分。**

不允许整体跳过的实训（至少要完成该实训中的一个关卡任务）包括：前 6 个实训、第 14 个实训（MySQL-数据库设计与实现）。

此外，实训 15 “数据库应用开发(JAVA 篇)” 则规定最多跳过其中一个关卡。

2.2 实训任务概览

实训 1 数据库、表与完整性约束的定义(Create)

本实训主要考察对数据库对象或约束条件的创建语法的掌握情况，包括创建数据库、表基表的主码约束、外码约束、CHECK 约束 DEFAULT 约束和 UNIQUE 约束，具体任务见任务内各个关卡。

注意头歌平台内每个关卡实践过程中包含了自测运行环境和测评环境，测评环境的重置在头歌平台内可能需要一定的时间。

实训 2 表结构与完整性约束的修改(ALTER)

本实训要求用 Alter 语句对表的定义进行修改（如更换/修改表名、列名、列的类型、列约束、表约束；添加或删除列、约束等）。

同学们需要在每个关卡中在代码文件中完整给出代码。在代码文件中，最好不要使用 use 指定数据库，以免评测出现问题。

实训 3 基于金融应用的数据查询(Select)

本实训采用的是某银行的一个金融场景应用的模拟数据库，测试库中有已有相应测试数据，请依据关卡任务需求完成相应查询动作。数据库中表，表结构以及所有字段的说明如下：

表 1 client(客户表)

字段名称	数据类型	约束	说明
c_id	INTEGER	PRIMARY KEY	客户编号
c_name	VARCHAR(100)	NOT NULL	客户名称
c_mail	CHAR(30)	UNIQUE	客户邮箱

字段名称	数据类型	约束	说明
c_id_card	CHAR(20)	UNIQUE NOT NULL	客户身份证
c_phone	CHAR(20)	UNIQUE NOT NULL	客户手机号
c_password	CHAR(20)	NOT NULL	客户登录密码

表 2 bank_card(银行卡)

字段名称	数据类型	约束	说明
b_number	CHAR(30)	PRIMARY KEY	银行卡号
b_type	CHAR(20)	无	银行卡类型(储蓄卡/信用卡)
b_c_id	INTEGER	NOT NULL FOREIGN KEY	所属客户编号,引用自 client 表的 c_id 字段。
b_balance	NUMERIC(10,2)	NOT NULL	余额,信用卡余额系指已透支的金额

说明:银行卡类型只有“储蓄卡”或“信用卡”两种取值。对于 b_balance 列,如果 b_type, b_balance 取值为(“储蓄卡”,10000),表示这张储蓄卡内有 10000 的储蓄余额;而 b_type, b_balance 取值为(“信用卡”,10000),表示这张信用卡已经透支 10000 元。

表 3 finances_product(理财产品表)

字段名称	数据类型	约束	说明
p_name	VARCHAR(100)	NOT NULL	产品名称
p_id	INTEGER	PRIMARY KEY	产品编号
p_description	VARCHAR(4000)	无	产品描述
p_amount	INTEGER	无	购买金额
p_year	INTEGER	无	理财年限

表 4 insurance(保险表)

字段名称	数据类型	约束	说明
i_name	VARCHAR(100)	NOT NULL	保险名称
i_id	INTEGER	PRIMARY KEY	保险编号
i_amount	INTEGER	无	保险金额
i_person	CHAR(20)	无	适用人群
i_year	INTEGER	无	保险年限
i_project	VARCHAR(200)	无	保障项目

表 5 fund(基金表)

字段名称	数据类型	约束	说明
f_name	VARCHAR(100)	NOT NULL	基金名称

字段名称	数据类型	约束	说明
f_id	INTEGER	PRIMARY KEY	基金编号
f_type	CHAR(20)	无	基金类型
f_amount	INTEGER	无	基金金额
risk_level	CHAR(20)	NOT NULL	风险等级
f_manager	INTEGER	NOT NULL	基金管理者

说明：以上 3 张表中的金额都指每购入一份所要花费的金额。

表 6 property(资产表)

字段名称	数据类型	约束	说明
pro_id	INTEGER	PRIMARY KEY	资产编号
pro_c_id	INTEGER	NOT NULL FOREIGN KEY	客户编号
pro_pif_id	INTEGER	NOT NULL	业务约束
pro_type	INTEGER	NOT NULL	商品类型:1 表示理财产品;2 表示保险;3 表示基金
pro_status	CHAR(20)	无	商品状态
pro_quantity	INTEGER	无	商品数量
pro_income	INTEGER	无	商品收益

字段名称	数据类型	约束	说明
pro_purchase_time	DATE	无	购买时间

说明：1.商品状态只有”可用”或”冻结”两种取值。2.商品收益指的是本条资产记录所记录商品的总收益，例如 f_id, f_amount 为(1031, 10000), pro_pif_id, pro_type, pro_quantity, pro_income 为(1031,3,10,27000)表示这条资产记录购入 10 份 1037 号基金，总花费 10*10000=100000 元，总收益为 27000 元。

实训 4 数据查询(Select)-新增

本小节子任务仍然以上述数据库内容为背景，但业务内容与统计、相似性推荐相关。

实训 5 数据查询(Select)-新增 2

本小节子任务在上述数据库内容的基础上，拓展相关内容和表格，包括综合客户表(client)、薪资表(wage)，列出客户的名称、年份、身份证号、全职酬劳总金额（full_t_amount）、兼职酬劳总金额（part_t_amount），查询业务主要涉及酬劳信息及其发放。

实训 6 数据的插入、修改与删除(Insert,Update,Delete)

实训 7 视图

实训 8 存储过程与事务

实训 9 触发器

实训 10 用户自定义函数

实训 11 安全性控制

实训 12 并发控制与事务的隔离级别

背景数据库有表 ticket 记录了航班余票数，其结构如下表所示：

列	类型	说明
flight_no	char(6)	primary key
tickets	int	余票数

有两个涉及该表的并发事务 t1 和 t2，分别定义在 t1.sql 和 t2.sql 代码文件中。平台会让两个事务并发执行，请同学们通过修改代码文件来达到题目预期的并发执行效果。

实训 13 备份+日志：介质故障与数据库恢复

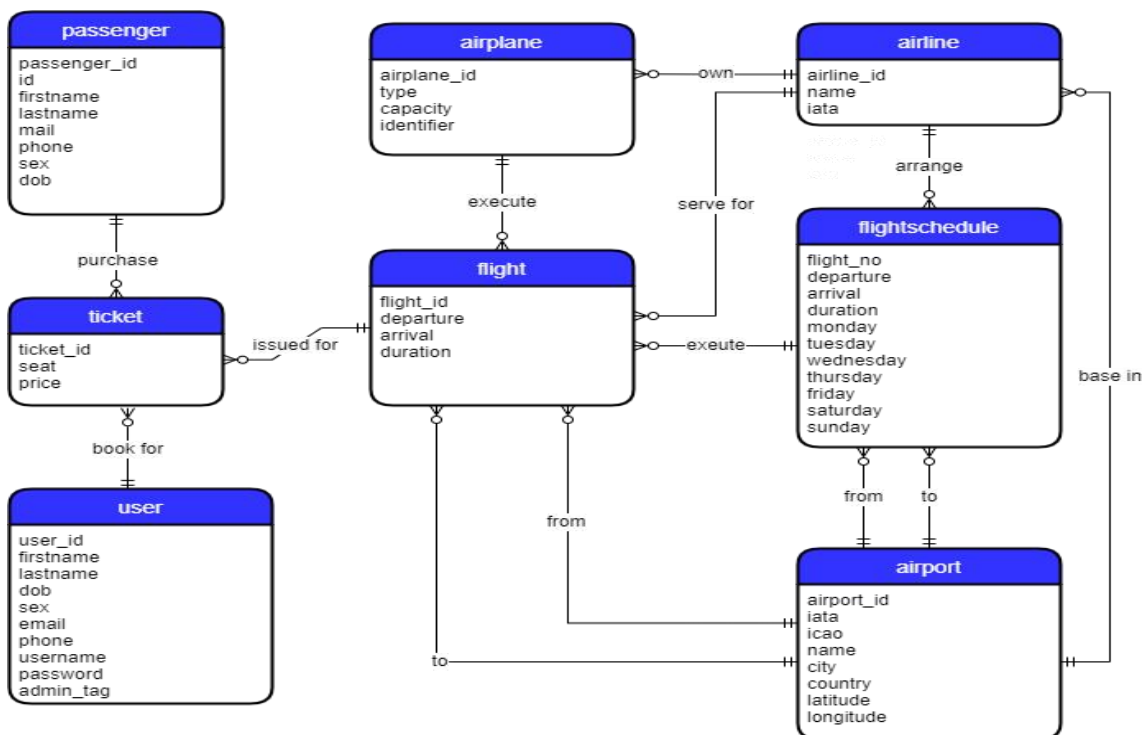
设有居民人口登记数据库 residents,请为该数据库做一次静态的(个人独享服务器)海量逻辑备份，备份文件命名为 residents_bak.sql。然后再用该逻辑备份文件恢复数据库。

注意：请仔细阅读平台上给出的注意部分，以免命令行操作中出现问题。

实训 14 数据库设计与实现

1) 从概念模型到 MySQL 实现

任务要求详见平台编程要求，此处给出 E-R 图方便翻阅。



2) 从需求分析到逻辑模型

设计一个影院管理系统。影院对当前的放映厅和电影进行排片，顾客到来后，可以购买任一排场的电影票，进入对应放映厅观看。系统中有以下实体集：

电影(movie): 属性有标识号(movie_ID)、电影名(title)、类型(type)、时长(runtime)、首映日期(release_date)、导演姓名(director)、主演姓名(starring)。

顾客(customer): 属性有标识号(c_ID)、姓名(name)、手机号(phone)。

放映厅(hall): 属性有标识号(hall_ID)、放映模式(mode)、容纳人数(capacity)、位置(location)。

排场(schedule): 属性有标识号(schedule_ID)、日期(date)、时间(time)、票价(price)、票数(number)。

电影票(ticket): 属性有标识号(ticket_ID)、座位号(seat_num)。

实体间的关系描述如下:

①. 顾客和电影票有一对多的购买关系。每位顾客可以买多张电影票, 每张电影票被一位顾客购买。

②. 电影票和排场有多对一的属于关系。一张电影票只属于一个排场, 一个排场有多张电影票。

③. 排场和电影有一对多的放映关系。每个排场放一部电影, 每部电影可以在多个排场放映。

④. 排场和放映厅有一对多的位于关系。每个排场位于一个放映厅, 每个放映厅可以安排多个排场。

请按照上述要求完成:

1.画出 E-R 图。

2.给出对应的关系模式。

在平台文本编辑器内给出 E-R 图 url 和关系模式。

强烈建议同学们在提交前检查一次自己提交的 url 是否可用, 有些图片链接点击可用是由于在本地上有正确的 cookie。同学们可以将链接发送到手机上, 在手机上尝试是否可以查看图片来确认。

互联网上图片都以图链形式传送, 在 web 中右键点击看到的图片, 菜单中大部

分时间都会有复制图片链接的选项，在本关中需要同学们来提供一个这样的 url 链接，在网络上定位到 E-R 图图片。

以下给出 4 种创建图链的方式，当然还有更多其他方式可以达到目的，同学们可以自己选择其一。

1) git 操作示意

①登录 gitee 或注册一个 gitee 账号。

②如果账号下还没有仓库，按下边操作创建一个可用的仓库，如果已有跳转到 3.



或在右上角



Pull Request, 其中代码提交的次数需本地配置的 git 邮箱是 Gitee 帐号已确

在创建仓库时可以不指定初始化仓库,勾选设置模板中的 Readme 文件即可。

仓库名称 *

test

归属 路径 *

Rainbou / test

仓库介绍 0/100

作为图床使用

☐ 开源 (所有人可见) ?

☒ 私有 (仅仓库成员可见)

☐ 企业内部开源 (仅企业成员可见) ?

☐ 初始化仓库 (设置语言、.gitignore、开源许可证)

☒ 设置模板 (添加 Readme、Issue、Pull Request 模板文件)

☒ Readme 文件 ☐ Issue 模板文件 ? ☐ Pull Request 模板文件 ?

☐ 选择分支模型 (仓库创建后将根据所选模型创建分支)

创建

如果没有勾选设置模板，面对一个原始的仓库，也可以点击这里来达到同样目的。



点击管理来把仓库设置为开源（只需要在第一个设置页下拉即可）。



请选择语言

默认分支

master

是否开源

☒ 开源 (所有人可见) ☐ 私有 (仅仓库成员可见)

仓库公开须知

☒ 我承诺此仓库内容不违反任何国家法律法规

☒ 我承诺此仓库内容不侵犯任何单位及个人的版权和权益

☒ 我承诺本人拥有该仓库所有内容的版权；如仓库内容引用他人作品，我承诺所引用内容均确认为合法内容，不存在版权问题，且对引用内容做了详细标注说明

如违反以上承诺，Gitee 保留追究相关责任的权利。更多信息请访问 [《Gitee.com网站使用条款》](#)

保存

③回到仓库首页，来上传 E-R 图图形文件。

你当前开源项目尚未选择许可证 (LICENSE)，[点此选择并创建开源许可](#)

master 分支 1 标签 0 + Pull Request + Issue 文件 Web IDE 克隆/下载

Rainbou	Initial commit	1cfa9e3	11分钟前	新建文件	1次提交
README.en.md	Initial commit			新建文件夹	11分钟前
README.md	Initial commit			新建子模块	11分钟前
README.md				上传文件	
				搜索文件	

test

- 介绍
- 参与开发

④点击图形文件查看，并在图片上右键选择复制链接，接下来就可以把 url 粘贴到平台上进行提交。



例如: https://gitee.com/rainbou_1/test/raw/master/2.png

2) 路过图床

①<https://imgtu.com/> 点击链接进入图床站。

②上传图片。

注: 支持上传 jpg、png、gif 格式图片。

③下划到图片链接区域, 点击复制 url 链接, 在平台提交。

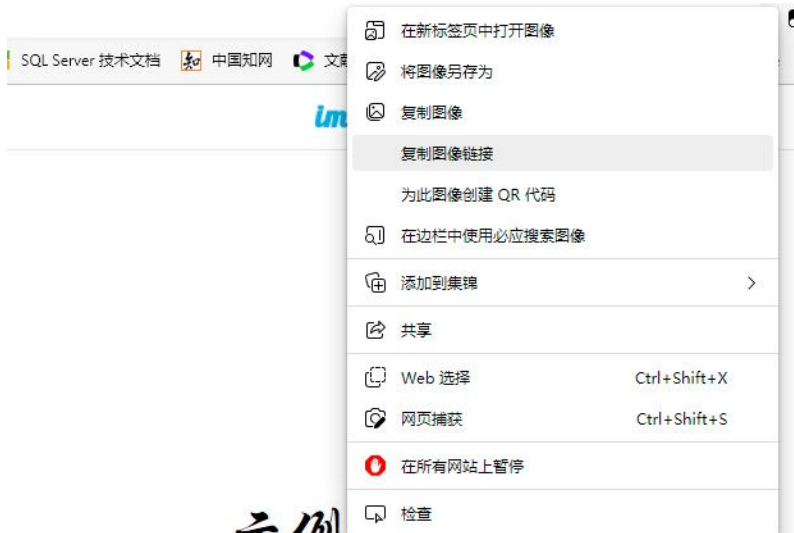
完整图片链接

图片链接	<input type="text" value="https://imgtu.com/i/qsLNDg"/>
网页贴图代码(HTML)	<input "="" type="text" value='
论坛贴图代码(BBCode)	<input type="text" value="[url=https://imgtu.com/i/qsLNDg][img]https://s1.ax1x.com/2C"/>
Markdown代码	<input type="text" value="[![qsLNDg.jpg](https://s1.ax1x.com/2022/03/29/qsLNDg.jpg)"/>
图片URL链接	<input type="text" value="https://s1.ax1x.com/2022/03/29/qsLNDg.jpg"/> <input type="button" value="COPY"/>

例如: <https://s1.ax1x.com/2022/03/29/qsLNDg.jpg>

3) imbbb 图床

- ①<https://imbbb.com/> 点击链接进入图床站.
- ②上传图片, 上传成功后点击图片进入查看大图界面。
- ③图片上右键选择复制链接, 接下来就可以把 url 粘贴到平台上进行提交。



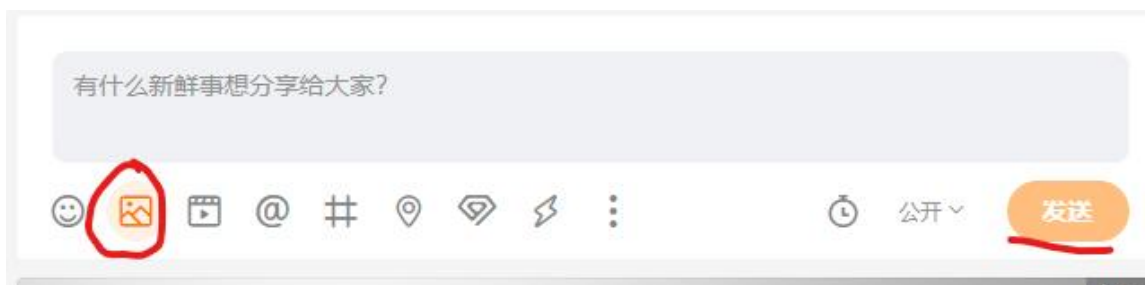
示例图

例如: <https://i.ibb.co/KjvsVC2/2.jpg>

4) 新浪微博

①登录微博 <https://weibo.com/>

②创建一条图片微博。



③点击发出的微博查看大图，并右键复制链接，粘贴到平台上提交。



例如: <https://wx4.sinaimg.cn/mw2000/0077zLsRly1h0qiguzgwqj305k02s3ye.jpg>

3) 建模工具的使用

下载资料包中的 rbac.mwb，利用 MySQL Workbench 建模模块中的 Forward engineering 功能，自动导出 SQL 脚本，把脚本粘贴到代码文件中。

实训 15 数据库应用开发(JAVA 篇)（最多跳过一题）

1) JDBC 体系结构和简单的查询

正确使用 JDBC，查询金融应用场景数据库 finance 的 client 表(客户表)中邮箱不为空的客户信息，列出客户姓名，邮箱和电话。

输出格式要求：

姓名	邮箱	电话
----	----	----

夏雅惠	<u>57433144004@qq.com</u>	18962433158
-----	---------------------------	-------------

钟庭玮	<u>59434199077@163.com</u>	18110434192
-----	----------------------------	-------------

其中，标题以及字段值之间用制表符隔开。第 1 列和第 2 列间用一个制表符，第 2 列和第 3 列间，标题用 4 个制表符，字段值用两个制表符隔开。

2) 用户登录

编程体验客户登录功能.程序先后提示客户输用户名和密码:

请输入用户名:

请输入密码:

客户的邮箱(c_mail)充当用户名,而不是编号(c_id).通常邮箱更容易记住.

根据客户的输入,输出以下两类信息之一:

登录成功。

用户名或密码错误！

3) 添加新客户

补充代码，实现向 `client` 表插入客户信息的方法，返回插入的行数。

其他要求详见平台。

4) 银行卡销户

补充代码，实现向银行卡销号的方法，只要客户编号和银行卡号匹配，即从 `bank_card` 表中删除该银行卡，返回被删除的行数。

5) 客户修改密码

补充代码，实现修改密码的方法。客户修改密码通常需要确认客户身份，即客户需提供用户名(以邮箱为用户名)和密码，方法返回一个整数：1 - 密码修改成功；2 - 用户不存在；3 - 密码不正确；-1 - 程序异常(如没能连接到数据库等)。

6) 事务与转账操作

补充代码，实现一个银行卡转账的方法，方法返回 `boolean` 值，`true` 表示转账成功，`false` 表示转账失败，并不需要细分或解释失败的原因。

下列任一情形都不可转账(转账失败的原因)：

转出或转入帐号不存在

转出账号是信用卡

转出帐号余额不足

7) 把稀疏表格转为键值对存储

一个表有很多列，但只有少数列存储有值，这样的表称为稀疏表。举个例子，大学里有几千名学生和几百门课程，为记录学生的成绩，设计这样一张表：每门课

程占一列，每个学生占一行。但是，绝大多数学生只会选修这几百门课程中的一小部分，因此，整个表只会稀疏地存储了一些数据，这将产生大量的闲置空间。

一种更有效的方法是以(学号, 课程, 成绩)这样的三元组形式存储。

设有高考成绩登记表 `entrance_exam`,其结构如下：

列名	类型	说明
sno	int	学号，主码
chinese	int	语文
math	int	数学
English	int	英语
physics	int	物理
chemistry	int	化学
biology	int	生物
history	int	历史
geography	int	地理
politics	int	政治

转存表 `sc` 结构如下：

列名	类型	说明
sno	int	学号
col_name	varchar(50)	列名
col_value	varchar(50)	列值

补充代码，完成转存任务。`sc` 表初始为空表，程序依前述规则将 `entrance_exam`

表的值转写到 sc 表。对每一行，请从左至右依次考察每一列，转存非空列。

实训 16 存储管理(Storage Manager)

数据在磁盘文件中是按照页面（Page）形式组织的。为避免直接访问磁盘数据页面而造成高昂的 I/O 开销，存储子系统在内存中创建缓冲池（Buffer Pool）来缓存部分磁盘数据页面。缓冲池维护固定数量的内存页面，每个内存页面称为“帧”（Frame），一般情况下，每一帧的大小与磁盘数据页面的大小保持一致。受制于内存容量，缓冲池只能缓存部分数据页面。因此，缓冲池管理的目标，就是在受限缓冲池大小的前提下，设计合适的内外存页面调度策略，尽可能将经常访问的磁盘数据页面维护在缓冲池中，从而减少磁盘 I/O 开销。本实验涉及缓冲池管理的重要内容，学生需要实现数据库存储系统中的缓冲池管理器，即 BufferPoolManager 类。它的数据结构包括 Page、DiskManager、Replacer 类的对象等。

数据库表中的一行数据，称为元组（Tuple）或者记录（Record），每条记录由多个字段（Field）组成。DBMS 存储系统负责将其解释成属性类型和值。记录可以按行存储，亦可按列存储，本实验采用行存储记录。

本实验有四个子任务：Disk Manager、Page Replacer、Buffer Pool Manager 和 Record Manger。

3 撰写课程实践报告

格式参见实践报告模板。

4 成绩占比说明

实践课综合成绩

=实践课程头歌成绩（占 70%）+检查时现场陈述与回答问题（占 15%）

+实践报告（占 15%）。