|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 2024-2025学年第一学期《数据库系统原理》大作业个人工作汇报表 | | | | | |
| 学号 | 102201307 | 姓名 | 邱聪祝 | 组长学号 | 102201307 |
| 组长姓名 | 邱聪祝 |
| 个  人  工  作  汇  报  个  人  工  作  汇  报 | 本次的数据库设计大作业可以分为两个部分，一个是去了解国内外数据库技术的现状和发展趋势，并以此分析大作业系统的选型；另一个是针对大作业系统的需求，进行数据库相关部分的设计工作。在第一个部分中我广泛学习认识了与数据库相关的多种技术，并且能搞清楚不同技术之间的基本异同与优劣；在第二个部分中，我亲身体验了数据库设计流程中的需求分析、概念结构设计、逻辑结构设计、数据库模式的创建等核心部分，掌握了不少设计过程中的要领，以及需要注意的事项。通过完成本次的大作业，我不仅拓展了课外的数据库技术理论知识，也获得了在具体需求下的动手实操经验，进一步提升了自己的数据库设计能力与数据库知识素养。  首先，关于数据库技术的现状和发展，我了解到在过去，像MySQL、Oracle这样的传统关系型数据库应用得最为广泛，在技术成熟得同时还有着强大的数据一致性和事务管理等能力。但是近年来，他们却面临着性能瓶颈，尤其是在高并发、大数据量的应用场景下。因此，许多新型数据库技术得到了广泛应用，比如分布式数据库、NoSQL数据库、内存数据库等。分布式数据库技术如Sharding、Replication等，可以帮助解决传统数据库在扩展性、性能和可用性方面的问题；而NoSQL数据库由于其高扩展性和灵活性，在大数据处理、实时分析等场景中也得到了广泛的应用。之后通过更加深入的学习，我进一步了解了数据库设计中如何结合这些技术来处理多表关联、性能优化等问题，并掌握了如何根据项目需求选择适合的技术栈，对本系统使用关系型数据库的合理性做出了一定的分析。  其次，是针对需求中的实践任务平台进行具体的数据库设计工作。由于我是单人完成，所以负责的工作包含了设计流程中的各个部分。在对需求说明进行阅读分析之后，应该能够大致确定系统所需要具备的功能都有哪些。但是我在这里却遇到了作业完成中的第一个重点问题，那就是部分需求描述的并不是非常明确，对于很多细节的处理方式也非常模糊。比如实践任务平台是一个课程还是多个课程？如果是多个课程那么每个课程的任务都是固定相同的那两个吗？这些描述不明确的地方，以及部分有悖常理的地方导致难以确定数据库中的实体和属性。可能涉及到的情况众多，迟迟无法取得进展。经过课下向老师进行详细的询问之后，我明白了需求说明上的内容可以理解为一个最低的要求，在此基础上可以自己进行合理的补充与扩展。  所以，在明确了目标后，我对需求进行了一系列的扩充设计。比如假定课程可以有很多个；并且每个课程中的任务不仅仅局限于那两个任务，而是可以由教师自己设定；甚至对于测试类型的任务还可以进行题目的选取。这样的设定虽然设计起来更为复杂，但是却更贴合实际，使得课程与任务的多元性得到提高，课程网站也能具有长期的使用价值。利用任务模式表+完成记录表的形式，来替代普通的任务表1、任务表2。允许了不同的任务进行统一且规范化的存储，而不用局限于满足需求中的两项任务来设定两张固定的任务表。具备更强的灵活性。  在概念数据模型的设计中，我梳理清楚了每个不同实体所具有的属性，以及其实体之间相互的联系，并进行了E-R图的逐步绘制。为了使图片整齐直观，我省去了各实体的属性，并在后文加以补充说明。之后需要进行逻辑结构的设计，这时就遇到了第二个重点问题，部分多对多关系的表示出现错误。比如任务和题目这两个实体，起初我想在任务表的某个属性中，存储所包含题目的集合。但是在后来经过了在数据库理论课上，对于关系数据库理论中规范化相关内容的学习，我意识到了这样的设计违反了第一范式：每一个变量必须是不可分的数据项的原则，需要进行修改。为了解决这个问题，我复习了课本上逻辑结构设计章节中，E-R图向关系模型转换部分的内容，得知了一个多对多的联系，可以转换为另外一个关系模式，专门用来存储相连的两个实体间的对应关系。可以看作像是中介表这样的存在，即SC表、CT表、TQ表三个。但在实际处理的过程中，还发现了某些多对多关系其实并不需要进行额外的转换，可以间接的通过外键和已有的转换来实现，这种关系相对来说应该没那么重要，或许是可以省略的。  在逻辑结构设置清晰之后，需要对数据库对象的创建语句进行编写。由于事先完成了在头歌平台上对应MySQL数据库的实践任务，所以编写过程中遇到的问题并不是很大，只是对于部分触发器和存储过程的创建格式仍不够熟练。经过这部分的练习，也让我在此扎实了SQL语句的使用基础，增强了记忆。由于像视图、索引、触发器、存储过程这类对象的一般需要根据实际情况来增添。在投入使用前进行预测性的编写往往无法面面俱到，所以在这部分我选取了一些典型的使用场景或情况进行编写，比如为一些明显需要连接操作的表来使用多对多关系的对象创建视图；为一些常用的查询关键字，比如学生名字、任务ID这些，创建索引，加快查找的速度；为一些可能会破坏数据完整性和一致性的操作，比如删除一个任务这类，创建触发器来进行维护。通过仔细寻找各种细枝末节处可能的优化设计，让我充分了解到数据库实施过程中所需要考虑到的方方面面。  本次数据库设计的最后一项任务是考虑安全性、完整性以及性能提升的有关措施。这部分内容可以通过结合课本中的基础知识，以及网络上的资料来进行分析。首先需要搞清楚的是我们当前系统的程度，从需求中可知，初衷是设计给本校的师生进行使用，课程和学生数量并不会很多。这说明系统的体量较小、数据规模较小，对于安全性而言，身份验证、用户授权、加密存储、加密传输是非常有必要的，安全审计和监控可以进行考虑，但对于像自主存取控制、强制存取控制这样高级别的保护措施，就不一定是具有性价比的选择。同理，对于完整性而言，三种约束条件与触发器是根本，而事务控制和数据恢复则视情况而定。性能提升中的索引优化、反范式设置、缓存机制也应该都有立竿见影的效果，而分布式架构则面临收益降低的风险，需要发展为更大的系统才更加适用。通过这部分措施的考虑，我对课本中对应概念的记忆得到了巩固，结合网络上的知识也使我的理解变得更加的深入。  总而言之，本次实验不仅让我掌握了数据库设计的流程与优化的核心概念，还让我在实际操作中锻炼了E-R图的绘制与SQL语句的编写能力。通过解决系统中存在的一个个实际问题，我对数据库设计的方法也有了一个全面的认识，并积累了在数据库开发领域的经验。这些经验将对我未来的项目开发和学习提供非常宝贵的参考。 | | | | |