

Reported by Wanen Gao

课程总结

Completion Time: 2021.12.30

By Wanen Gao

目录

一、课程概论 1

二、课程要点 1

 2.1 数据库原理概述 1

 2.2 数据模型 1

 2.3 E-R 模型 1

 2.4 范式分析 2

 2.5 SQL 语言 2

 2.6 常用的数据库对象及使用 3

 2.7 查询优化 3

 2.8 事务处理 3

 2.9 并发控制 4

三、课程收获 4

四、课程感想 5

一、课程概论

本课程系统的讲述了数据库的基本原理，主要分为三大部分。第一部分为设计篇，主要介绍数据库原理概述、数据模型和关系模型在数据库中的体现、关系理论原理及其表达、ER 模型和数据库设计、范式分析、SQL 语言及其在系统的应用、视图、存储过程、触发器等。第二部分为实现部分，主要介绍数据库的存储原理、索引和查询、事务处理等。第三部分为高级数据库技术，主要介绍数据仓库与数据挖掘、新型数据库简介和结合面向对象语言的数据库编程等。

二、课程要点

2.1 数据库原理概述

- (1) 数据库的基本概念
- (2) 数据库系统的特点
- (3) 数据模型概述
- (4) 关系模型和数据库系统的结构
- (5) 数据库系统的组成

2.2 数据模型

- (1) 数据模型的基本概念和种类
- (2) 关系模式与关系型数据库
- (3) 关系代数及其操作
- (4) 关系的完整性

2.3 E-R 模型

- (1) 概念模型的概念和表示方法
- (2) 数据结构设计的一般方法和步骤

- (3) ER 模型的设计
- (4) ER 模型向关系模型的转换
- (5) 对实际模型绘制 ER 图

2.4 范式分析

- (1) 数据依赖的概念和类型
- (2) 函数依赖

平凡函数依赖与非平凡的函数依赖的基本概念

完全函数依赖与部分函数依赖的基本概念

传递函数依赖的基本概念

- (3) 范式的概念和种类
- (4) 范式的判定
- (5) 范式的分解

2.5 SQL 语言

- (1) SQL 简介
- (2) 数据查询语言

投影、选择、连接的使用；

空值的概念与其使用；

限定查询的用法；

聚集函数的使用；

子查询语句的使用；

特殊的查询语句；

(3) 数据操纵语言

插入、删除、修改的使用

(4) 数据定义语言

建表语句、建库语句、建模式语句的使用

(5) 数据控制语言

授权语句和撤权语句的使用方法

2.6 常用的数据库对象及使用

(1) 视图的概念与用法

(2) 存储过程的概念与常见操作方法

(3) 触发器的概念以及使用

(4) 游标的概念以及使用

(5) 常见的扩展的 SQL 语言

2.7 查询优化

(1) 索引的概念与使用;

(2) 查询优化的概念与实现方法;

(3) 代数优化的概念与实现

(4) 物理优化的概念与实现

2.8 事务处理

(1) 事务的基本概念与 ACID 特性

(2) 数据库恢复概述

(3) 故障种类

(4) 数据转储与日志

- (5) 常见的恢复策略
- (6) 具有检查点的恢复技术
- (7) 数据库镜像

2.9 并发控制

- (1) 并发控制概述
- (2) 封锁机制与封锁协议
- (3) 并发调度的可串行性
- (4) 两端锁协议
- (5) 封锁粒度

三. 课程收获

通过这学期的数据库的学习，总结来说有以下收获：

一是对数据库基础知识点的掌握

二是完成了从前端到后端的全站开发工作，体会了网站开发中的各项工作流程，学习了各阶段中用到的大量知识

三是提升了自己的实践动手能力，体会到了开发知识的庞杂，实践中体会到了以后在工作中，势必是从需求出发进行碎片式学习的学习方法。尽管没有系统的知识，有一定的基础，再加上碎片式的学习，在如今模块化程度高，框架成熟的背景下，就可以进行相当一部分的工作。这种体会在全站开发，面对众多未知的问题时十分深刻，同样，在以后面对新领域的棘手问题时，同样应该保持学习心态，面对需求学习，事半功倍。

四. 课程感想

课程中间有十分多的感想和体悟，数据库带给我的不是仅仅几个有关数据库的知识点，这在多年后也许会遗忘，而不会遗忘的是数据库这门课带给我的面对一个问题，系统地解决的一种思路。从大量数据的存储不方便，到各种数据管理系统，中间经历的是大胆创新，小心求证的探索之路。一条如何快速准确的增删改查数据之路，走得相当不易。制造工具来解决问题、提升效率才是面对一个问题的最佳思路。人与自然界各种生物的区别，在于善于利用工具，人类文明之路的先行者们总是致力于制造工具，提高效率。

我在数据库课程设计中，一开始并没有认真进行需求分析以及数据库的设计，而是急于去编码实现界面等前台的东西。结果编到第二个界面的时候，发现数据库设计不合理，只好重新回去修改。我的解决方法是：详细进行需求分析，争取做到对整个系统了然于胸；详细进行系统功能划分，分到最小模块为止，然后再一个一个模块添加代码即可，当然，要考虑模块与模块之间的耦合程度以及模块与模块之间的信息共享。

在未来，我将通过自己在数据库课程中学到的知识去进行更多的实践，创造出更多有价值，有意义的内容。