

物理独立性

逻辑独立性

视图：不会加快查询效率，里面没有真实数据

创建视图

超码、候选码、主码、外码

完整性约束：

- 1.实体(主体)完整性
- 2.参照完整性
- 3.用户自定义完整性

索引：

加快检索速度

一张表可以有多个索引

创建索引的语句：

最多有一个索引是主索引或者主索引

关系代数：

- 1、6个基本运算（选择，并，笛卡尔积，连接，）
- 2、表达式
- 3.关系代数表达式和sql

sql：

sql：结构化查询语言

1.dml

2.ddl

E-R：实体关系图

sql的语句：

看实验

1.join

2.聚合函数，尤其是其对于Null的处理，count *的处理

3.权限：对象

4.删除

5.增加

6.嵌套语句 (exist, in)

数据库的设计：

实体联系模型

几个阶段：

1.需求分析

2.概念设计阶段：关系模型，关系模式

3.

1.E-R图（常用的元素符号要知道）

2.数据表

数量关系：

一对一，一对多，多对多

规范化

3个范式

1.函数依赖

2.函数依赖的闭包：属性值的闭包（判断函数依赖是否成立）

3.分解：无损分解（完全按照规则），判断是否无损分解（两个属性集的交集是其中一张表的Key）

4.依赖保持：判断是否依赖保持。（?）

范式解决的问题：

数据冗余

3个异常：

插入异常，删除异常，更新异常

事务管理和并发控制

事务管理：

操作单元：

4个属性：

原子性、一致性、独立性、持久性

回滚操作表面上是解决原子性，其实是一致性

冲突可串行化：

并发控制：

二阶段锁协议

1.上升阶段：加锁，升级

2.下降阶段：解锁，降级

死锁：不同的事物互相持有别人需要的锁