基本关系运算有：并、差、笛卡儿积、投影、选择(原子运算，其他运算可以由这些运算推导出)

实体完整性：若属性A是基本关系R的主属性，则A不能取空值

分解关系模式目的：消除不合适的函数依赖，解决更新、插入、删除异常等问题

SQL查询的结果是：表

SQL功能：数据定义DDL，数据操纵DML(查询、更新)，数据控制DCL(授权、完整性规则描述)

\*数据库备份：为了维护数据库的一致性(数据库的恢复子系统就是使发生故障的数据库恢复到一致性状态)

概念模式/外模式《》逻辑独立性 概念模式/内模式《》物理独立性

衡量关系分解的两个标准：无损连接性、保持函数依赖

设计E-R图、合并分E-R图∈概念结构设计 E-R图=>关系数据模型∈逻辑结构设计

合并E-R图过程中可能存在的冲突：属性冲突、命名冲突、结构冲突

连接：从两个关系的笛卡儿乘积中选取属性满足一定条件的元组(如果没有公共元素，连接=笛卡尔积)

结构模型三要素：数据结构(用于描述系统静态特征)、数据操作(用于描述系统动态特征)、数据的完整性约束条件

一个关系通常对应一张表，元组是表中一行，字段是表中一列

数据库安全性控制方法：用户标识和鉴别、存取权限控制、使用视图、审计、数据加密

数据恢复是用冗余来恢复

SQL的两种使用方式：自含式、嵌入式

关系模式的问题：

插入异常：应该插入的没有被插入(刚成立的系如果还没有招生就无法存储

系主任信息)

删除异常：不该删除的被删除(某个系的学生全部毕业删除时会丢失系

主任信息)

更新异常：更换系主任必须修改每一个学生信息

数据库：长期存储在计算机内，有组织的、可共享的相关数据的集合

DBMS：位于用户和操作系统之间的一层数据管理软件

等值连接不要求属性相同，自然连接需要(而且要删除重复属性列)

模式是整体逻辑结构，外模式是用户可以看到并可使用的数据库逻辑结构，内模式是数据库整体存储结构

外模式=子模式 概念模式=模式 内模式=存储模式

数据库管理技术的发展：人工管理=>文件系统=>数据库系统

数据模型：表示实体类型和实体间联系的模型

结构数据模型主要有三种：层次模型、网状模型、关系模型

存取权限两要素：数据对象、操作类型

数据库管理系统的功能：数据定义、数据操纵、数据库运行管理、数据组织存储管理、数据库建立与维护功能、数据通信接口功能

数据库系统：计算机硬件为基础的记录保持系统。包括数据库、数据库管理系统、应用系统、**管理员和用户**，有时还包括**计算机硬件**

数据冗余可能导致的问题：浪费存储空间、潜在的数据不一致性

DBA 的职责：设计与定义数据库、监督和控制数据库运行

候选键：在不同的元组中的值是不同的，利用这个（些）值可以唯一地标识关系中地元组

主码（键）：指定正在使用的候选键称主键

超键：能唯一标识元组的属性集，超键包括所有候选键

主属性和非主属性：包含在任一候选键中的属性叫主属性，否则称非主属性

外连接：R、S作自然连接时保留舍弃的元组，新增加的属性值置空

左连接：R的舍弃元组保留

右连接：S的舍弃元组保留

带比较运算的子查询：当确定子查询的返回值是唯一时，可以使用比较运算

符(注意子查询在比较符后) SELECT S#,SN FROM S WHERE SD=

(SELECT SD FROM S WHERE CN=’刘晨’)

INTERSECT 交

UNION 并

MINUS 差

当要插入的元组只包含一部分信息（字段个数<列数） 插入语句需要写出列名

3NF仍然可能存在插入删除异常，BCNF消除插入删除异常(在函数依赖范畴内实现模式彻底分解，达到最高规范化程度).

数据库设计流程：需求分析=>概念结构设计=>逻辑结构设计=>数据库物理设计=>数据库实施=>数据库运维

需求分析结果：数据字典DD、数据流图DFD

概念结构设计：设计E-R图

逻辑结构设计：概念结构转化为数据模型

物理设计：存储结构、存取路径(如何建立索引)

数据库实施：DDL定义数据库结构、加载数据、系统调试

数据库完整性：数据的正确性和相容性

解决数据库并发操作带来的不一致性的方法：封锁

两种基本的锁：共享锁、排他锁

评价物理结构：时间效率、空间效率、维护代价、用户需求

概念模型是一种逻辑抽象，和实际问题相关 与机器独立(硬件、OS、DBMS)

并行数据库：节点扩展

R中所有属性都是主属性，则规范程度至少达到3NF.

索引一般用在where条件中的属性上用于提高检索速度

为了提高检索成绩的速度，索引应该在(学号、课程号)上：因为查成绩是

select grade from sc where s#=’’ and c#=’’

where条件中涉及的属性是 s# c#

网状模型效率高性能好、关系模型性能低

层次模型(树)、网状模型(有向图/网)、关系模型(表)

关系运算、优化、设计关系模式、关系理论

数据库设计核心：逻辑设计和物理设计

要求检索“所有”可以考虑 除运算 或者SQL中没有(x)p，故须转换为¬(x(¬p))

alter table s add 列名 数据类型 约束条件；modify 列名 数据类型；drop 约束条件(这里的drop不是删除表，特定指删除约束条件)

drop table s

create (unique)(cluster) index 索引名 on S(列名)

drop index S.索引名

select where 条件部分 any任一 all所有

带exists的子查询：子查询的查询条件依赖于父查询 相关子查询

SQL中没有(x)p，故须转换为¬(x(¬p))

不更新视图的情况：

由两个以上基本表导出的视图

视图的字段来自集函数

视图中含有GROUP BY子句

视图中含有DISTINCT语句

视图定义有嵌套查询，且内层查询涉及到导出本视图的基本表(相关子查询)

不允许更新的视图上定义的视图

视图的优点：简化操作、多角度看待数据、逻辑独立性、安全性

DCL：grant 权限 on Table S to 用户 权限可以为 select update update某一列 等等

revoke 权限 on Table S from 用户

嵌入式SQL DBMS的两种处理方式：

预编译、修改和扩充宿主语言以处理SQL

ESQL： EXEC SQL <SQL语句> SQL语句包括：可执行(DDL、DML、DCL)、说明性

ESQL和主语言通信：SQL通信区(共享内存)、主变量(单个)、游标(集合)·

select top 1：查询结果只显示首条记录

数据依赖：函数依赖、多值依赖、连接依赖

衡量关系分解的两个标准：无损连接、保持函数依赖(二者独立，即具有无损连接性不一定保持函数依赖，保持函数依赖也不一定具有无损连接性)

若要求关系模式分解具有无损连接性，则分解一定可达到4NF

若要求关系模式分解保持函数依赖，则分解一定可达3NF，但不一定达到BCNF

若要求关系模式分解既具有无损连接性，又保持函数依赖，则分解一定可达到3NF，但不一定达到BCNF

范式：数据依赖满足某种条件级别的关系模式的集合

DBA(系统管理员)为用户建立用户名、口令

CREATE USER <username> IDENTIFIED BY <password>

数据库完整性约束：

对象：列级(列取值类型、范围)、元组级(各字段之间关系)、关系级(各元组之间关系)

状态：静态、动态

静态关系约束常见有四种：实体完整性约束、参照完整性约束、函数依赖约束、统计约束

触发器：

create trigger 触发器名：

before op of 列名 on 表名

for each row222

when(:new.列名=’值’)

if(:new.列名 满足条件) then

op;

end if

end

事务是并发控制的单位，是用户定义的一组操作序列

特性：原子性、一致性、隔离性、持久性

可串行性是衡量并发事务正确性的唯一标准.指多个事务并发时是正确的，当且仅当结果和某种次序串行执行它们时的结果相同.

数据库复制：

对等复制——各个点平等，互相复制

主从复制——主数据库复制到从数据库

级联复制——主数据库复制到从数据库，再从从数据库复制到其他数据库

select语句中 如果一个表重复做笛卡尔积多次，需要进行编号，并且where条件中直接使用对应编号

例如：select s1.S# from S s1,S s2 where s1.SD=s2.SD and s2.SN=’’

带exists的相关子查询例子：

SELECT SN FROM S WHERE EXISTS

(SELECT \* FROM SC WHERE S# = S.S# AND C# = ‘1’)

直接写关系模式 可以按照E-R图转化关系模式的思路

一个实体一个关系模式

实体之间的联系

数据库的封锁协议——事务隔离级别

事务隔离级别0对应一级封锁协议

SQL语句表达完整性：

实体完整性：constraint pk\_1 primary key (列名)

参照完整性：constraint fk\_1 foreign key (列名) reference 表名 (列名) on delete|update cascade|restricted|set null

自定义完整性：constraint chk1 check （完整性条件）可以表示列级(直接跟在字段后面)或元组级(在所有字段定义结束后定义) 关系级自定义完整性需要用触发器

设计时v.s.运行时 同样的控件表现出不同的功能 delphi大多数属性值在设计时可以修改

数据连接控件：ADO(实验中使用ADOConnection、ADOQuery)

数据访问控件：Data Access(实验中使用Data Source)

数据控制控件：Data Control(实验中使用DBGrid)





