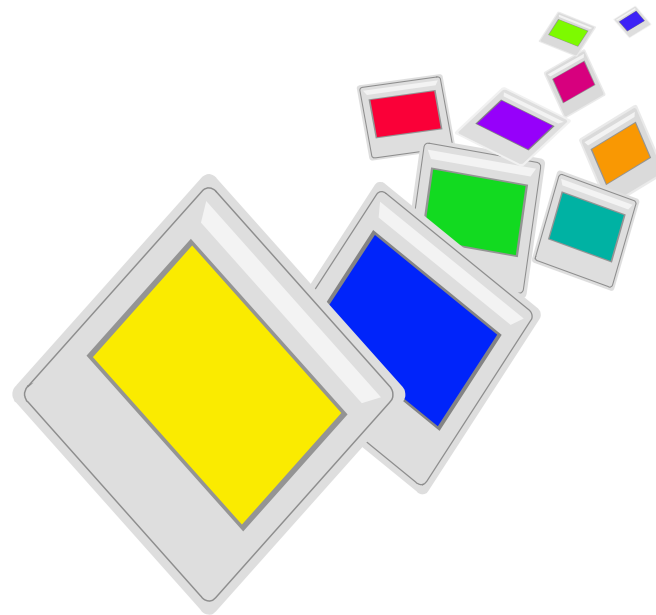


数据库系统及其应用

- 内容提要
- 数据库系统的基本概念
- 结构化查询语言SQL概述
- SQL的数据定义操作
- SQL的数据查询操作
- SQL的数据更新操作
- SQL的视图操作
- 几种新型的数据库系统



基本要求:

- 理解数据库系统的基本概念和基本知识。
- 掌握SQL的数据定义、数据查询、数据更新和视图等操作方法。
- 了解分布式数据库、多媒体数据库、并行数据库、演绎数据库系统、主动数据库和数据仓库的功能、特点和主要实现技术。

数据库系统的定义

数据库系统是在数据库系统中实现对数据进行管理的软件系统，由数据库、数据库管理系统、数据库管理员、数据库应用程序以及用户五部分组成：

- ✓ 数据库（Data Base, DB）

- ◆ 统一管理的相关数据的集合

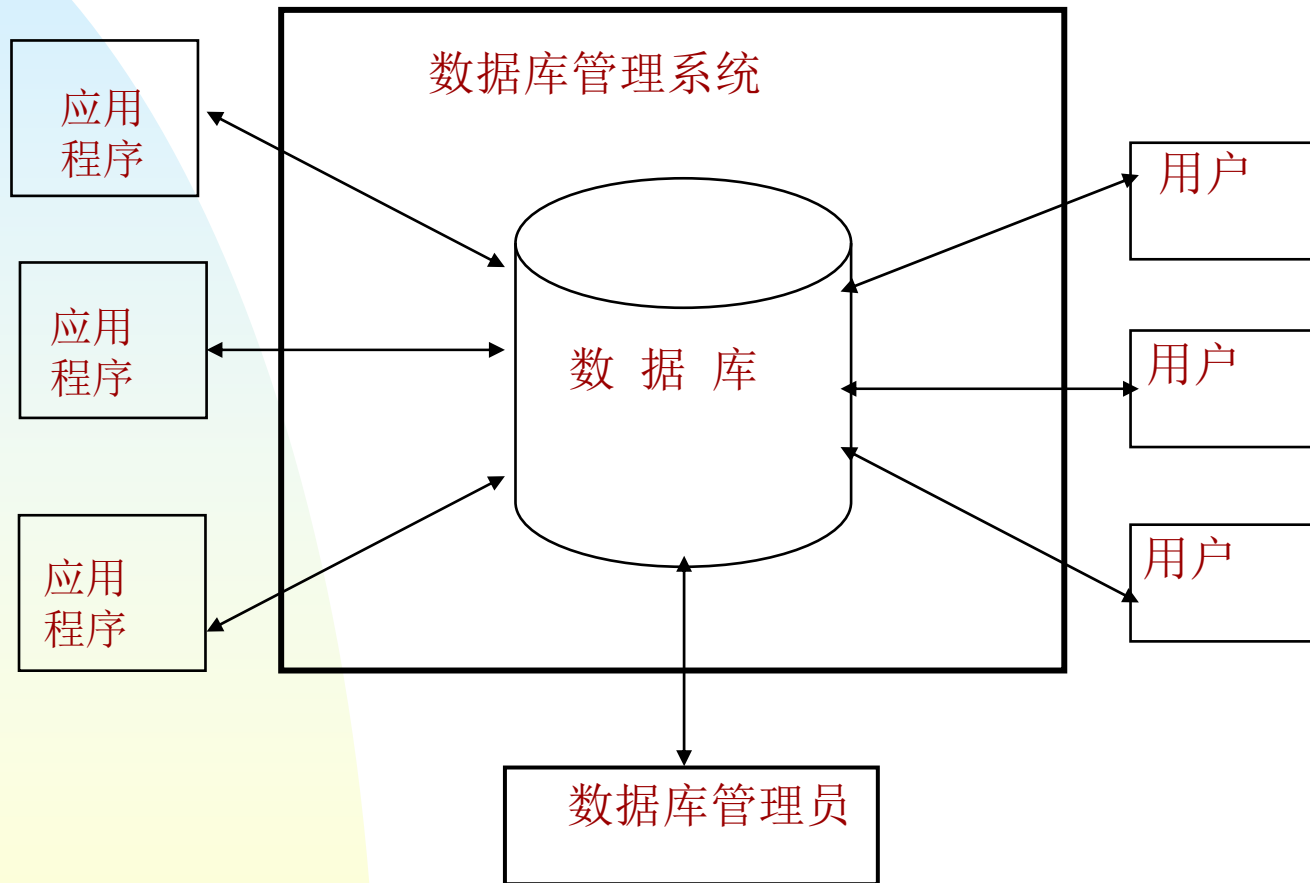
- ✓ 数据库管理系统（Data Base Management System, DBMS）

- ◆ 是对数据库进行管理的软件，是数据库系统的核心

数据库系统的定义(续)

- ✓ 数据库管理员 (Data Base Administrator, DBA)：是对数据库进行规划、设计、协调、维护 and 管理的IT技术人员。
- ✓ 数据库应用程序：是使用数据库语言开发的、能够满足数据处理需求的应用程序。
- ✓ 用户：用户可以通过数据库管理系统直接操纵数据库，或者通过数据库应用程序来操纵数据库

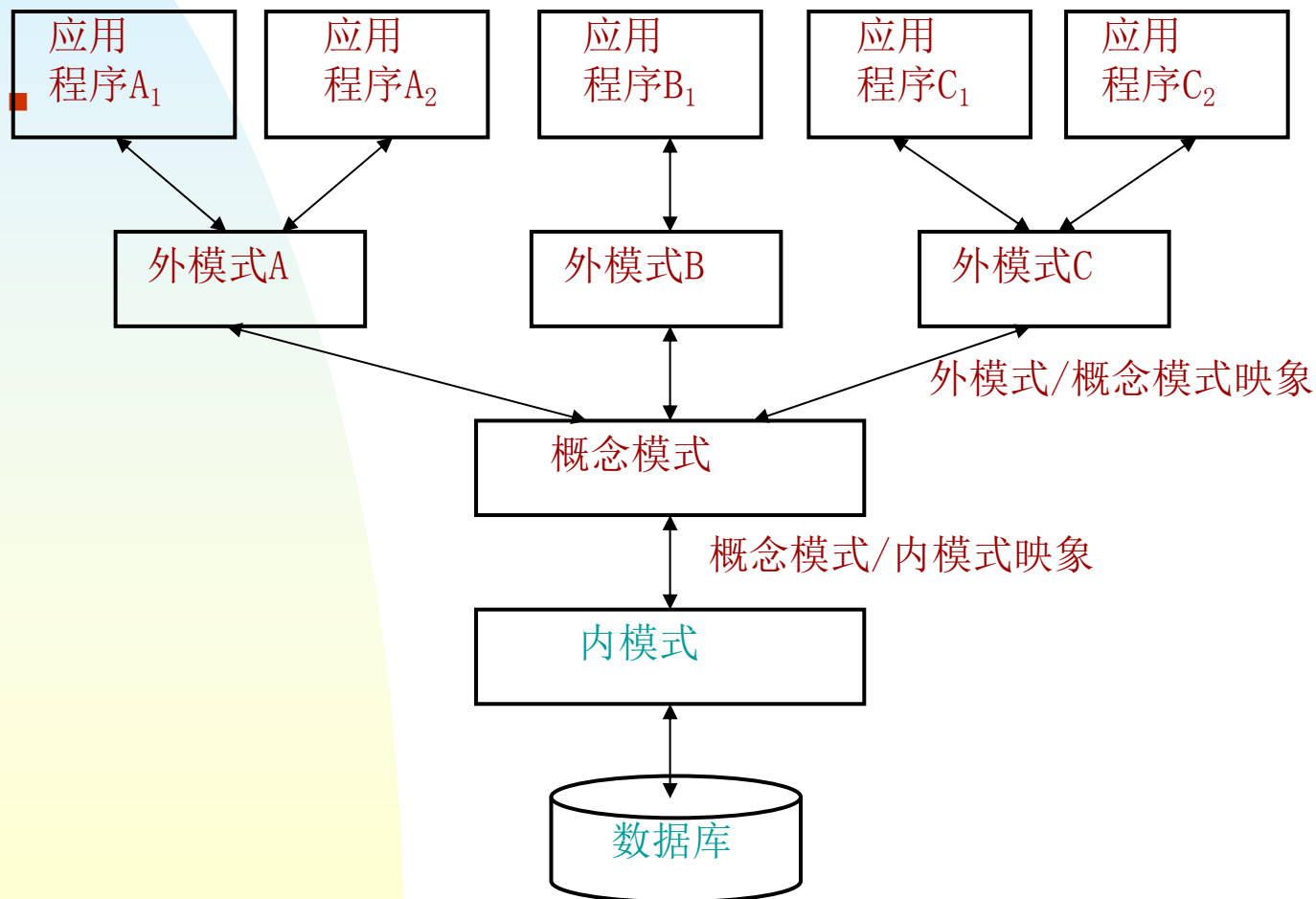
■数据库系统的组成



数据管理技术的发展

- **人工管理阶段：**在20世纪50年代中期以前，数据处理采用批处理方式，没有可直接存取的磁盘，没有专门的数据管理软件。
- **文件系统阶段：**在20世纪50年代后期至60年代后期，磁盘和磁鼓等外存出现，以及操作系统中提供了文件管理功能，数据管理水平也相应提高。**主要特点：**数据独立于程序，并实现了文件的长期保存和按名存取。
- **数据库阶段：**在20世纪70年代，数据库技术进一步发展，克服了文件系统的缺陷。**主要特点：**具有更高的逻辑数据独立性，更强大的控制功能，更有好的用户界面。
- **高级数据库阶段：**在20世纪80年代以来，以分布式数据库和面向对象的数据库技术为代表，应用领域不断扩大。

■数据库系统的体系结构：分为外模式、内模式、概念模式三个层次。概念模式/内模式映象和外模式/概念模式映象由数据库管理系统（DBMS）来实现。



数据库管理系统（DBMS）的功能

DBMS是指在数据库系统中实现对数据进行管理的软件系统，是数据库系统的核心。其基本功能是实现共享数据的有效组织、管理和存取。

并具有以下功能

- 数据库定义功能；
- 数据库操纵功能；
- 数据库事务管理功能；
- 数据库维护功能；
- 格式转换以及网络通信等其他功能。

数据库管理系统（DBMS）的类型

- ❖ **层次型数据库**: 采用层次数据模型，即使用树型结构来表示数据库中的记录及其联系。
- ❖ **网状型数据库**: 采用网状数据模型，即使用有向图（网络）来表示数据库中的记录及其联系。
- ❖ **关系型数据库**: 关系型数据库采用关系数据模型，即使用二维表格的形式来表示数据库中的数据及其联系。由于关系模型比较简单、易于理解且有完备的关系代数作为其理论基础，所以被广泛使用。
- ❖ **面向对象数据库**: 采用面向对象数据模型，是面向对象技术与数据库技术相结合的产物。在面向对象数据库中使用了对象、类、实体、方法和继承等概念，具有类的可扩展性、数据抽象能力、抽象数据类型与方法的封装性、存储主动对象以及自动进行类型检查等特点。

数据库管理系统（DBMS）的构成

- ❖ **应用层：**该层是数据库管理系统与终端用户和应用程序的界面，负责处理各种数据库应用程序对数据库的请求。
- ❖ **语言处理层：**该层由DDL编译器、DML编译器、DCL编译器、查询器等组成，负责完成对数据库语言的各类语句进行词法分析、语法分析和语义分析，生成可执行的代码。此外，还负责进行授权检验、视图转换、完整性检查、查询优化等。
- ❖ **数据存取层：**该层将上层的集合操作转换为对记录的操作，包括扫描、排序、查找、插入、删除、修改等，完成数据的存取、路径的维护以及并发控制等任务。
- ❖ **数据存储层：**该层由文件管理器和缓冲区管理器组成，负责完成数据的页面存储和系统的缓冲区管理等任务。

数据库语言

- ❖ 数据定义语言（DDL）：定义和标识数据库的逻辑结构和物理结构，并给出其唯一的命名
 - 描述各类模式及它们之间的映象。
 - 描述每一个基本数据项的基本特征。
 - 描述安全控制方式和完整性约束条件。
 - 定义数据结构和子结构之间的映象。
- ❖ 数据操纵语言（DML）：用来描述用户对数据库进行的各种操作，包括数据的录入、修改、删除、查询、统计、打印等。
- ❖ 结构化查询语言（Structured Query Language, SQL）：是一种基于关系代数和关系演算的数据操纵语言，目前一般都采用SQL作为共同的数据操纵语言 and 标准接口，是数据库领域的一种主流语言。

SQL的产生与发展

- ❖ SQL是结构化查询语言（Structure Query Language）的英文缩写，是一种基于关系运算理论的数据库语言。
- ❖ SQL是在1974年由Byce和Chamberlin首先提出的，并在IBM公司研制的关系数据库管理系统SYSTEM-R上实现。
- ❖ 1986年10月美国国家标准局（ANSI）的数据库委员会X3H2将SQL作为关系数据库语言的美国标准。
- ❖ 1987年6月国际标准化组织（ISO）将SQL作为关系数据库语言的国际标准。
- ❖ 1992年ISO对SQL89标准进行了修改和扩充，并颁布了国际标准数据库语言（International Standard Database Language）SQL92。

SQL的特点

- ❖ 功能的一体化
- ❖ 语法结构的统一性
- ❖ 高度的非过程化
- ❖ 语言的简洁性

SQL的功能

- ❖ 数据定义功能
- ❖ 数据操纵功能
- ❖ 数据控制功能
- ❖ 嵌入式功能

SQL的数据定义操作一定义基本表

- 基本表定义语句的一般形式：
- CREATE TABLE <基本表名>
- (<属性名1> <数据类型1>[NOT NULL],
- [<属性名2> <数据类型2>[NOT NULL]] ,
- ...
- [<完整性约束>])
- <完整性约束>主要有3中子句：主键子句（PRIMARY KEY）、检查子句（CHECK）和外键子句（ FOREIGN KEY ）。

SQL提供的基本数据类型

数据类型	说明
CHAR (n)	长度为n的定长字符串
VARCHAR (n)	最大长度为n的变长字符串
最大长度为n的变长字符串	全字长整数
SMALLINT	半字长整数
NUMERIC (p, d)	由p位数字（不包括符号和小数点）组成的定点数，小数点后面有d位数字
REAL	浮点数
DOUBLE PRECISION	双精度浮点数
FLOAT (n)	精度至少为n位数字的浮点数
DATE	日期，形如YYYY-MM-DD（年-月-日）
TIME	时间，形如HH: MM: SS（时: 分: 秒）

定义基本表举例

- 【例6-1】 使用SQL语句建立学生表S、课程表C和学生选课成绩表SC。

- 解：（1）创建基本表S：

- CREATE TABLE S

- （学号 CHAR(4)NOT NULL, 姓名 CHAR(8)NOT NULL,

- 年龄 SMALLINT, 性别 CHAR(2), 系别 CHAR(20),

- PRIMARY KEY (学号))

学号	姓 名	年龄	性别	系 别
20022501	张卫国	21	男	计算机科学与工程系
20022502	李建峰	23	男	计算机科学与工程系
20022601	赵丽	21	女	信息与控制工程系
20022701	钱华	20	男	电气工程系
20022802	王小平	22	男	电气工程系

(a) 学生表 S

定义基本表举例（续）

- 【例6-1】使用SQL语句建立学生表S、课程表C和学生选课成绩表SC。

■解：（2）创建基本表C：

- CREATE TABLE C
- （课程号 CHAR（4）NOT NULL，课程名 CHAR（4）NOT NULL，
- 教师名 CHAR（8），
- PRIMARY KEY （课程号） ）

课程号	课程名	教师名
C001	计算机引论	何东
C002	C语言	张云
C003	数据结构	李峰
C004	操作系统	何东
C005	编译原理	周佳

(b) 课程表 C

定义基本表举例（续）

- （3）创建基本表CS：
- CREATE TABLE SC
- （学号 CHAR（4）NOT NULL，课程号 CHAR（4）NOT NULL，
- 成绩 CHAR（8）SMALLINT，
- PRIMARY KEY（学号，课程号）
- FOREIGN KEY（学号）REFERENCES S（学号），
- FOREIGN KEY（课程号）REFERENCES C（课程号），
- CHECK（成绩 IS NULL）OR（成绩 BETWEEN 0 AND 100））

学号	课程号	成绩
20042501	C001	90
20042502	C001	78
20022602	C003	70
20022505	C003	82
20022501	C005	92
20022502	C005	80

（c）学生选课成绩表 CS

修改基本表

❖ 在基本表中增加列：

- ALTER TABLE <基本表名> ADD <属性名> <数据类型>
- 【例6-2】 在例6-1的学生基本表S中增加一个属性“家庭地址”：
- ALTER TABLE S ADD 家庭地址 VARCHAR (40)

❖ 在基本表中删除列：

- ALTER TABLE <基本表名> DROP <属性名> <删除方式>
- 这里， <删除方式>可以取CASCADE和 RESTRICT两种方式。
- 【例6-3】 在学生基本表S中删除一个属性“系别”：
- ALTER TABLE S DROP 系别 CASCADE
- 或 ALTER TABLE S DROP 系别 RESTRICT

撤销基本表

❖ 撤销基本表语句:

- DROP TABLE <基本表名> <撤销方式>
- <撤销方式>可以取CASCADE和 RESTRICT两种方式。
- 【例6-4】若要撤销学生基本表S，可用下列语句实现：
- DROP TABLE S CASCADE
- 或 DROP TABLE S RESTRICT

定义索引

❖ 为基本表建立索引的目的是加快对基本表的存取速度。

❖ 定义索引语句：

- CREATE [UNIQUE] INDEX <索引名> ON <基本表名>
- ((<属性名1>[<次序>]][,<属性名2>[<次序>]] ...)
- [PCTFREE={<整数>}]

任选项**UNIQUE**表示每个索引属性值只能对应一个元组；任选项<次序>可以是**ASC**（表示升序排列）或**DESC**（表示降序排列），默认时为升序。

■ 【例6-5】在学生基本表S的列“学号”上按升序建立索引：

- CREATE UNIQUE INDEX SNOINDEX ON S (学号 ASC)

■ 在基本表CS的列“成绩”、“学号”上按降序建立索引：

- CREATE INDEX GRADEINDEX ON S (成绩 DESC 学号 DESC)

撤销索引

❖ 撤销索引语句:

- `DROP INDEX <索引名>`
- 注意：如果撤销某一个基本表，则该基本表上建立的所有索引将被一起撤销。
- 【例6-6】若要撤销学生基本表S上的索引SNOINDEX，则可用下列语句实现：
- `DROP INDEX SNOINDEX`

SQL的数据查询操作

- ❖ 数据查询语句是SQL的核心，是SQL数据操纵功能的重要组成部分。
- ❖ SQL的查询语句使用非常灵活，功能十分强大，它可以实现：
 - 简单查询
 - 连接查询
 - 嵌套查询

SQL的数据查询语句

❖ 数据查询语句的一般形式:

- SELECT <目标属性> FROM <基本表名或视图名>[WHERE <条件表达式>]
- [GROUP BY <属性名1>[HAVING <条件表达式>]]
- [ORDER BY <属性名2>[<次序>]]

❖ 数据查询语句的组成:

- *SELECT*子句
- *FROM*子句
- *WHERE*子句
- *GROUP BY*子句
- *ORDER BY*子句

SQL数据查询语句的含义

- ❖ 根据WHERE子句中的<条件表达式>从由<基本表名或视图名>指定的基本表或视图中找出满足条件的元组，按SELECT子句中的<目标属性>选出元组中的分量，形成查询结果。
- ❖ 如果有ORDER子句，则将结果根据指定的<属性名2>按<次序>所指定的顺序排列（选ASC为升序，选DESC为降序）。
- ❖ 如果有 GROUP子句，则将查询结果按<属性名1>进行分组，每组产生结果表中的一个元组，分组的附加条件用 HAVING 短语中的 <条件表达式> 给出。

SQL数据查询语句中的语法成分

- ❖ SELECT子句中的<目标属性>可以是“*”或<选择列表>。
 - “*”表示查询结果是整个元组；
 - <选择列表>是用逗号分隔的项（属性名、常数或系统内部函数），当指定了<选择列表>时，查询结果将是由<选择列表>所指定的列。
- ❖ WHERE子句中的<条件表达式>实现各种复杂查询，其中可以使用下列运算符：
 - 算术运算符：+、-、*、/
 - 比较运算符：<、<=、=、>=、>、≠
 - 逻辑运算符：AND、OR、NOT
 - 集合运算符：UNION（并）、INTERSECT（交）、EXCEPT（差）
 - 集合成员运算符：IN（属于）、NOT IN（不属于）
 - 谓词：EXISTS（存在量词）、ALL、SOME、UNIQUE
 - 系统内部函数：常用的系统内部函数如表6-2所示。

SQL数据查询语句中常用的系统内部函数

函数名称	一般形式	含 义
平均值	AVG ([DISTINCT] <属性名>)	求列的平均值，有DISTINCT选项时只计算不同值
求和	SUM ([DISTINCT] <属性名>)	求列的和，有DISTINCT选项时只计算不同值
最大值	MAX (<属性名>)	求列的最大值
最小值	MIN (<属性名>)	求列的最小值
计数	COUNT (*) COUNT (DISTINCT) <属性名>)	统计结果表中元组的个数 统计结果表中不同属性名值元组的个数

简单查询举例

❖ **简单查询：**在单个基本表上的查询，不涉及多个基本表的连接查询和嵌套查询等复杂的情况。

■ 【例6-7】在基本表S中查询全体学生的详细信息：

■ `SELECT * FROM S`

■ 该查询语句等价于：

■ `SELECT 学号, 姓名, 年龄, 性别, 系别 FROM S`

简单查询举例

■ 【例6-8】 在基本表S中查询计算机科学与工程系和电气工程系学生的学号、姓名和年龄：

■ SELECT 学号, 姓名, 年龄 FROM S
■ WHERE 系别 = '计算机科学与工程系' OR 系名 = '电气工程系'

■ 该查询语句等价于：

■ SELECT 学号, 姓名, 年龄 FROM S
■ WHERE 系别 IN ('计算机科学与工程系', '电气工程系')

简单查询举例

- 【例6-9】 在基本表S中查询年龄在18~20岁学生的姓名和性别:

- ```
SELECT 姓名, 性别 FROM S
```
- ```
WHERE 年龄 >= 18 AND 年龄 <= 20
```

- 该查询语句等价于:

- ```
SELECT 姓名, 性别 FROM S
```
- ```
WHERE 年龄 BETWEEN 18 AND 20
```

简单查询举例

- 【例6-10】 在基本表S中查询所有男学生的学号、姓名和出生年份：

- ```
SELECT 学号, 姓名, 2004-年龄 FROM S
```
- ```
WHERE 性别 = '男'
```

- 【例6-11】 在基本表S中查询计算机科学与工程系女学生的总数和平均年龄，可用以下查询语句实现：

- ```
SELECT COUNT (*), AVG (年龄) FROM S
```
- ```
WHERE 系别='计算机科学与工程系'AND 性别='女'
```


简单查询举例

- 【例6-12】 在基本表SC中查询有课程号为C001的课程成绩学生的学号、和成绩，查询结果按成绩降序排列：

- SELECT 学号, 成绩 FROM SC
- WHERE 课程号 = 'C001'
- ORDER BY DESC

多表查询举例

- ❖ **多表查询：**查询涉及两个或两个以上基本表，又称为连接查询。
- **【例6-13】**在基本表S、C和SC中查询选修课程号为C001学生的学号和姓名。
- `SELECT S.学号, 姓名 FROM S, SC`
- `WHERE S.学号=SC.学号 AND 课程号= 'C001'`

学号	姓 名	年龄	性别	系 别
20022501	张卫国	21	男	计算机科学与工程系
20022502	李建峰	23	男	计算机科学与工程系
20022601	赵丽	21	女	信息与控制工程系
20022701	钱华	20	男	电气工程系
20022702	王小平	22	男	电气工程系

(a) 学生表 S

课程号	课程名	教师名
C001	计算机引论	何东
C002	C 语言	张云
C003	数据结构	李峰
C004	操作系统	何东
C005	编译原理	周佳

(b) 课程表 C

学号	课程号	成绩
20042501	C001	90
20042502	C001	78
20022602	C003	70
20022505	C003	82
20022501	C005	92
20022502	C005	80

(c) 学生选课成绩表 SC

图 6-3 关系 S、C 和 SC

多表查询举例

- SELECT S.学号, 姓名 FROM S, SC
- WHERE S.学号=SC.学号 AND 课程号='C001'

学号	姓名	年龄	性别	系别
20022501	张卫国	21	男	计算机科学与工程系
20022502	李建峰	23	男	计算机科学与工程系
20022601	赵丽	21	女	信息与控制工程系
20022701	钱华	20	男	电气工程系
20022702	王小平	22	男	电气工程系

(a) 学生表 S

课程号	课程名	教师名
C001	计算机引论	何东
C002	C语言	张云
C003	数据结构	李峰
C004	操作系统	何东
C005	编译原理	周佳

(b) 课程表 C

学号	课程号	成绩
20042501	C001	90
20042502	C001	78
20022602	C003	70
20022505	C003	82
20022501	C005	92
20022502	C005	80

(c) 学生选课成绩表 SC

图 6-3 关系 S、C 和 SC

❖ 通过条件“S.学号=SC.学号”实现基本表S和SC的连接，称该条件为连接条件或连接谓词。

❖ “S.学号”表示基本表S中的属性名“学号”，“SC.学号”表示基本表SC中的属性名“学号”，以区分不同基本表中相同的属性名，对于仅出现在一个基本表中的属性名可以不加前缀。

❖ 该查询语句在执行时，先要对S和SC做笛卡尔积运算，然后再根据条件“课程号='C001'”选择出符合要求的元组和列。

多表查询举例

- 【例6-14】在基本表S、C和SC中查询选修课程名为“操作系统”学生的学号和姓名。
- SELECT S.学号, 姓名 FROM S, C, SC
- WHERE S.学号=SC.学号 AND C.课程号=SC.课程号
- AND 课程名=‘操作系统’
- ❖ 条件“S.学号=SC.学号 AND C.课程号=SC.课程号”实现基本表S、C和SC的连接
- ❖ 执行笛卡尔积运算后再根据条件“课程名=‘操作系统’”选择出符合要求的元组和列。

学号	姓 名	年龄	性别	系 别
20022501	张卫国	21	男	计算机科学与工程系
20022502	李建峰	23	男	计算机科学与工程系
20022601	赵丽	21	女	信息与控制工程系
20022701	钱华	20	男	电气工程系
20022702	王小平	22	男	电气工程系

(a) 学生表 S

课程号	课程名	教师名
C001	计算机引论	何东
C002	C 语言	张云
C003	数据结构	李峰
C004	操作系统	何东
C005	编译原理	周佳

学号	课程号	成绩
20042501	C001	90
20042502	C001	78
20022602	C003	70
20022505	C003	82
20022501	C005	92
20022502	C005	80

多表查询举例

- 【例6-15】 在基本表S、C和SC中查询选修课程号为C001且成绩为80分及80分以上学生的学号、姓名和成绩。
- SELECT S.学号, 姓名, 成绩 FROM S, SC
- WHERE S.学号=SC.学号 AND SC.课程号= 'C001' AND 成绩 >= 80

学号	姓 名	年 龄	性 别	系 别
20022501	张卫国	21	男	计算机科学与工程系
20022502	李建峰	23	男	计算机科学与工程系
20022601	赵丽	21	女	信息与控制工程系
20022701	钱华	20	男	电气工程系
20022702	王小平	22	男	电气工程系

(a) 学生表 S

课程号	课程名	教师名
C001	计算机引论	何东
C002	C 语言	张云
C003	数据结构	李峰
C004	操作系统	何东
C005	编译原理	周佳

(b) 课程表 C

学号	课程号	成绩
20042501	C001	90
20042502	C001	78
20022602	C003	70
20022505	C003	82
20022501	C005	92
20022502	C005	80

(c) 学生选课成绩表 SC

图 6-3 关系 S、C 和 SC

嵌套查询举例

❖ **嵌套查询**：在查询语句WHERE后面的条件表达式中出现另一个查询,该查询又称为子查询，SQL允许多层嵌套。

- 【例6-16】在基本表S、C和SC中查询选修课程号为C001学生的学号和姓名。（同【例6-13】）

- SELECT 学号, 姓名 FROM S
- WHERE S.学号 IN
- (SELECT 学号 FROM SC WHERE 课程号 = 'C001')

学号	姓名	年龄	性别	系别
20022501	张卫国	21	男	计算机科学与工程系
20022502	李建峰	23	男	计算机科学与工程系
20022601	赵丽	21	女	信息与控制工程系
20022701	钱华	20	男	电气工程系
20022702	王小平	22	男	电气工程系

(a) 学生表 S

课程号	课程名	教师名
C001	计算机引论	何东
C002	C语言	张云
C003	数据结构	李峰
C004	操作系统	何东
C005	编译原理	周佳

学号	课程号	成绩
20042501	C001	90
20042502	C001	78
20022602	C003	70
20022505	C003	82
20022501	C005	92
20022502	C005	80

嵌套查询举例

- 【例6-17】 在基本表S、C和SC中查询选修课程名为操作系统学生的学号和姓名：（同【例6-14】）

- SELECT 学号, 姓名 FROM S
- WHERE 学号 IN
- (SELECT 学号 FROM SC WHERE 课程号 IN
- (SELECT 课程号 FROM C WHERE 课程名 = '操作系统'))

学号	姓 名	年龄	性别	系 别
20022501	张卫国	21	男	计算机科学与工程系
20022502	李建峰	23	男	计算机科学与工程系
20022601	赵丽	21	女	信息与控制工程系
20022701	钱华	20	男	电气工程系
20022702	王小平	22	男	电气工程系

(a) 学生表 S

课程号	课程名	教师名
C001	计算机引论	何东
C002	C 语言	张云
C003	数据结构	李峰
C004	操作系统	何东
C005	编译原理	周佳

(b) 课程表 C

学号	课程号	成绩
20042501	C001	90
20042502	C001	78
20022602	C003	70
20022505	C003	82
20022501	C005	92
20022502	C005	80

(c) 学生选课成绩表 CS

SQL的数据更新操作

- ❖ 数据插入
- ❖ 数据删除
- ❖ 数据修改

数据插入语句

数据插入语句:

❖形式1:

- INSERT INTO <基本表名>[(<属性名1>[, <属性名2>] ...)]
- VALUER (<常量1>[, <常量2>] ...)

(该形式一次可以插入一个元组, 在<属性名>表缺省时, VALUER后面的常量表应给出该元组每一列的值。)

❖形式2:

- INSERT INTO <基本表名> [(<属性名1>[, <属性名2>] ...)]
- <SELECT语句>

(该形式一次可将SELECT查询得到的一组值插入到指定的基本表中。)

SQL的数据插入操作举例

- 【例6-18】在学生基本表S中插入一名新学生的元组，其学号、姓名、年龄、性别和所在系别分别为20022503、周冰、22、男和计算机科学与工程系。
- INSERT INTO S
- VALUES ('20022503', '周冰', 22, '男', '计算机科学与工程系')

SQL的数据插入操作举例

- 【例6-19】 创建一个存放各个系学生平均年龄的基本表DEPTAGE（系别，平均年龄），然后把基本表S中每一个系学生的平均年龄存入基本表DEPTAGE。
- （1）创建基本表DEPTAGE：
 - CREATE TABLE DEPTAGE
 - （系别 CHAR（20）NOT NULL，平均成绩 SMALLINT，
 - PRIMARY KEY（系别））
- （2）将查询结果存入基本表DEPTAGE：
 - INSERT INTO DEPTAGE（系别，平均年龄）
 - SELECT 系别，AVG（年龄）FROM S
 - GROUP BY 系别

学号	姓 名	年 龄	性 别	系 别
20022501	张卫国	21	男	计算机科学与工程系
20022502	李建峰	23	男	计算机科学与工程系
20022601	赵丽	21	女	信息与控制工程系
20022701	钱华	20	男	电气工程系
20022702	王小平	22	男	电气工程系

(a) 学生表 S

SQL的数据删除操作

❖ 数据删除语句:

- DELETE FROM <基本表名>[WHERE <条件表达式>]

❖ 注意:

- 该语句从指定的基本表中删除满足条件的那些元组，**当没有WHERE子句时表示删除该基本表中的全部元组，但该基本表的定义仍然保存在数据字典中。**
 - **删除操作是删除元组，而不是删除元组中的某些属性值。**
 - **删除操作只能从一个基本表中删除元组，执行删除操作有可能产生破坏数据一致性的情况，若要从多个基本表中删除元组，则需使用多个DELETE语句。**
 - **执行“DELETE FROM <基本表名>”操作将使指定的基本表为空表，使用时要格外慎重。**
- 【例6-20】在学生基本表S中删除学号为20022501学生的记录:
 - DELETE FROM S WHERE 学号 = '20022502'

SQL的数据修改操作

❖ 数据修改语句:

- UPDATE <基本表> SET <属性名1>=<表达式1>[, <属性名2>=<表达式2> , ...][WHERE <条件表达式>]

❖ 含义：修改指定基本表中满足WHERE条件表达式的元组中的指定属性值，SET子句中的表达式给出了修改后的值。

SQL的数据修改操作举例

- 【例6-21】 在学生基本表S中把学号为20022501学生的系别改为‘电气工程系’：

- UPDATE S
- SET 系别 = ‘电气工程系’
- WHERE 学号 = ‘20022501’
- 【例6-22】 在学生基本表S中把所有学生的年龄加1：
- UPDATE S SET 年龄 = 年龄 + 1

SQL的视图操作

- ❖ 视图：从多个基本表或其他视图构造出来的表。定义视图时系统只是将其定义保存起来，并不存储视图所对应的数据，当用户需要使用该视图时才取出实际数据。
- ❖ 视图操作：
 - 定义
 - 查询
 - 更新
 - 撤销

定义视图

❖ 定义视图语句的一般形式:

- CREATE VIEW <视图名> [(<属性名1> [, <属性名2>] ...)]
- AS <SELECT 语句>
- [WITH CHECK OPTION]

❖ 注意:

- 该语句执行的结果是将有关视图的定义存储到数据字典中，只有当对视图进行操作时才根据定义从基本表中形成实际数据供用户使用。
- 在“CREATE VIEW”语句的 SELECT 语句中不能含有操作符UNION和ORDER BY子句。

定义视图举例

- 【例6-23】对于基本表S、C和SC创建一个视图CSS，它给出计算机科学与工程系学生的基本情况

- CREATE VIEW CSS
- AS SELECT 学号, 姓名, 性别, 年龄
- FROM S WHERE 系别='计算机科学与工程系'

学号	姓 名	年 龄	性 别	系 别
20022501	张卫国	21	男	计算机科学与工程系
20022502	李建峰	23	男	计算机科学与工程系
20022601	赵丽	21	女	信息与控制工程系
20022701	钱华	20	男	电气工程系
20022702	王小平	22	男	电气工程系

(a) 学生表 S

视图的查询

■ 【例6-24】 设已创建一个视图CSS，它给出计算机科学与工程系学生的基本情况。对视图CSS查询计算机科学与工程系年龄大于22岁学生的学号、姓名和年龄。

■ `SELECT 学号, 姓名, 年龄 FROM CSS WHERE 年龄 > 22`

■ 本例转换后的查询为:

■ `SELECT 学号, 姓名, 年龄 FROM S`

■ `WHERE 系别 = '计算机科学与工程系' AND 年龄 > 22`

学号	姓 名	年龄	性别	系 别
20022501	张卫国	21	男	计算机科学与工程系
20022502	李建峰	23	男	计算机科学与工程系
20022601	赵丽	21	女	信息与控制工程系
20022701	钱华	20	男	电气工程系
20022702	王小平	22	男	电气工程系

(a) 学生表 S

视图的更新操作

❖ 视图的更新操作:

➤ *INSERT*

➤ *DELETE*

➤ *UPDATE*

■ 【例6-25】 设已创建一个视图CSS，它给出计算机科学与工程系学生的基本情况。将视图CSS中学号为20022602的学生的姓名修改为赵立

■ UPDATE CSS SET 姓名 = ‘赵立’

■ WHERE 学号 = ‘20022602’

■ 该更新语句将被转换为对基本表S的更新:

■ UPDATE S SET 姓名 = ‘赵立’

■ WHERE 系别 = ‘计算机科学与工程系’ AND 学号 = ‘20022602’

视图的撤销

- ❖ 视图撤销语句:

- DROP VIEW <视图名>

- ❖ 注意:

- DROP VIEW语句的执行将把指定视图的定义从数据字典中删除，由该视图导出的视图其他视图也将自动被删除。
 -
- 如果导出某一视图的基本表被删除，则该视图也将自动被删除。
- 【例6-26】撤销已创建的视图CSS:
 - DROP VIEW CSS

分布式数据库

- ❖ 分布式数据库 (Distributed Database, DDB)：是物理上分散在计算机网络各结点上，但在逻辑上属于同一系统的数据集合。
- ❖ 分布式数据库的特点：
 - 数据分布性
 - 逻辑相关性
 - 局部自治与全局共享性
 - 数据的冗余性
 - 数据的独立性
 - 系统的透明性
- ❖ 分布式数据库管理系统 (Distributed Database Management System, DDBMS)：是一个支持分布式数据库的建立、操纵与维护的软件系统，负责实现局部数据管理、数据通信、分布数据管理以及数据字典管理等功能。

多媒体数据库

- ❖ 多媒体数据库 (Multimedia Database, MDB)：是以数据库的方式合理地存储在计算机中的多媒体信息（包括文字、图形、图像、音频和视频等）的集合。
- ❖ 多媒体数据库的特点：
 - 媒体多样性
 - 信息量大
 - 管理复杂
- ❖ 多媒体数据库管理系统 (Multimedia Database Management System, MDBMS)：是一个支持多媒体数据库的建立、操纵与维护的软件系统，负责实现对多媒体对象的存储、处理、检索和输出等功能。
- ❖ 主要研究内容：多媒体的数据模型、MDBMS的体系结构、多媒体数据的存取与组织技术、多媒体查询语音、MDB的同步控制以及多媒体数据压缩技术等。

并行数据库

- ❖ 并行数据库 (Parallel Database, PDB)：是数据库技术与并行技术相结合的产物，它在并行体系结构的支持下，实现数据库操作处理的并行化，以提高数据库的效率。
- ❖ 并行数据库技术的主要研究内容：
 - 并行数据库体系结构
 - 并行数据库机
 - 并行操作算法
 - 并行查询优化
 - 并行数据库的物理设计
 - 并行数据库的数据加载和再组织技术

演绎数据库

- ❖ 演绎数据库（Deductive Database, DeDB）：是数据库技术与逻辑理论相结合的产物，它是一种支持演绎推理功能的数据库。
- ❖ 演绎数据库的组成：由用关系组成的外延数据库EDB和由规则组成的内涵数据库IDB两部分组成，并具有一个演绎推理机构，从而实现数据库的推理演绎功能。
- ❖ 演绎数据库技术主要研究内容：
 - 逻辑理论与逻辑语言
 - 递归查询处理与优化算法
 - 演绎数据库体系结构

主动数据库

- ❖ **主动数据库**（Active Database, Active DB）：它除了具有传统数据库的被动服务功能之外，还提供主动进行服务的功能，即数据库系统在某种情况下能够根据当前状态主动地作出反应，执行某些操作，向用户提供所需的信息。
- ❖ **常用的方法**：在传统的数据库系统中嵌入“事件—条件—动作”规则。当某一事件发生后引发数据库系统去检测数据库当前状态是否满足所设定的条件，若条件满足则触发规定动作的执行。
- ❖ **需完善的技术问题**：主动数据库中的知识模型、执行模型、事件监测和条件检测方法、事务调度、安全性和可靠性、体系结构和系统效率等。

数据仓库

- ❖ **数据仓库**：是支持管理决策的面向主题的、集成的、稳定的、定期更新的数据集合。
- ❖ **数据仓库系统的功能**：对异构数据源中数据的提取、过滤、加工和存储以及响应用户的查询和决策分析请求。
- ❖ **数据仓库管理系统的组成**：由监控器、转换器、集成器、元数据管理等部件组成，分别完成对源数据库数据更新的监控、异构数据的转换、数据加载以及对元数据进行管理等功能。

作业:

- 1. 数据库系统由哪几部分组成?
- 2. 试阐述数据库、数据库管理系统以及数据库系统的定义。
- 3. 数据管理技术经历了哪几个阶段? 各个阶段的主要特点是什么?
- 4. 什么是分布式数据库? 它有哪些主要的特点?
- 5. 目前数据管理的主要方法是 () ?
- A. 文件系统和数据库系统 B. 文件系统和操作系统 C. 文件系统和批处理系统 D. 数据库系统和批处理系统
- 6. 数据库系统的核心是 () ?
- A. 数据模型 B. 数据库 C. 数据库开发工具 D. 数据库管理系统
- 7. SQL是一种 () 的语言
- A. 格式化 B. 过程化 C. 非过程化 D. 导航式
- 8. SQL有两种使用方式, 分别称为自含式和 ()
- A. 外含式 B. 交互式 C. 嵌入式 D. 解释式
- 9. 设有一关系R(NO,NAME,SEX,AGE,CLASS), 其中NO为学号, NAME为姓名, SEX为性别, AGE为年龄, CLASS为班级, 试写出实现下列功能的SQL语句。
 - (1) 插入一个记录 (31, '张云', '女', 20, '200232')。
 - (2) 将学号为20的学生姓名改为 '孙大光' ;
 - (3) 将所有 '2002032' 班级号改为 '2002082' ;
 - (4) 删除学号为30的记录;
 - (5) 删除班级号为 '1998032' 的所有记录。