

《数据库基础》（精简版）

《数据库基础》（精简版）

一、数据库基础

1.1 基本概念

1.2 系统特征

1.3 数据模型

1.4 数据表结构

二、SQL语言

2.1 特点与分类

2.2 创建（CREATE）

三、基本查询与操作

3.1 查询（SELECT）

3.2 条件（WHERE）

3.3 排序（ORDER BY）

3.4 聚集函数

3.5 数据操作

出处：23数媒2班 陆云清

一、数据库基础

1.1 基本概念

1. **数据库 (DB)**: 有组织、可共享的数据集合。特点：**冗余低、独立性高、易扩充。**

2. **数据库管理系统 (DBMS)**: 位于用户与OS之间的数据管理软件。

◦ 功能：定义、操纵、控制、维护。

◦ 常见：`Access`, `SQLite`, `MySQL`, `SQL Server`。

3. **数据库系统 (DBS)**: 带有数据库的计算机系统。

1.2 系统特征

结构化、共享性高、独立性高、DBMS统一管理。

1.3 数据模型

1. **关系型数据库**: 基于二维表。

数据库	特点
<code>SQLite</code>	轻量嵌入式、零配置、跨平台
<code>MySQL</code>	开源小型、速度快、成本低
<code>SQL Server</code>	微软推出、易用、可伸缩
<code>Oracle</code>	甲骨文推出、大型专业

2. **非关系型 (NoSQL)**: 追求速度/扩展性。

◦ 键值（Redis）: 快查。

- **列存** (`Hbase`): 扩展强。
- **文档** (`MongoDB`): 结构松散。
- **图形** (`InfoGrid`): 图计算。

1.4 数据表结构

1. **组成**: 字段 (列/属性)、记录 (行/数据)。

2. **设计原则**: 无重复数据、单一主题。

3. **常用类型**:

- **字符型**: 文本 (姓名)。
- **数值型**: 运算 (工资)。
- **日期型**: 时间 (生日)。
- **逻辑型**: 真假 (`True` / `False`)。

二、SQL语言

结构化查询语言，用于增删改查。

2.1 特点与分类

1. **优势**: 非过程化、简洁易学、可移植。

2. **分类**:

- **DQL (查询)**: `SELECT`
- **DML (操纵)**: `INSERT`, `UPDATE`, `DELETE`
- **DDL (定义)**: `CREATE`, `DROP`, `ALTER`
- **DCL (控制)**: `COMMIT`, `ROLLBACK`

2.2 创建 (CREATE)

1. **数据库**: `CREATE DATABASE db_name`

2. **数据表**:

```
CREATE TABLE table_name (
    col1 TYPE [NOT NULL] [PRIMARY KEY],
    ...
);
```

- **类型**: `INTEGER` (整), `REAL` (浮), `TEXT` (文), `BLOB` (二进制)。
- **约束**: `PRIMARY KEY` (主键唯一), `NOT NULL` (非空)。

三、基本查询与操作

3.1 查询 (SELECT)

```
SELECT 字段 | *
FROM 表名
[WHERE 条件]
[ORDER BY 字段 [ASC|DESC]];
```

3.2 条件 (WHERE)

1. 比较: >, <, =, >=, <=, !=。
2. 逻辑: AND (与), OR (或), NOT (非)。
3. 模糊 (LIKE):
 - %: 任意多个字符。
 - _: 单个字符。

3.3 排序 (ORDER BY)

- ASC: 升序 (默认)。
- DESC: 降序。

3.4 聚集函数

函数	作用
COUNT()	计数
SUM()	求和
AVG()	平均
MAX() / MIN()	最大/最小

3.5 数据操作

1. 插入 (INSERT):

```
INSERT INTO 表 (列1...) VALUES (值1...);
```

2. 更新 (UPDATE):

```
UPDATE 表 SET 列=新值 WHERE 条件;
```

3. 删除 (DELETE):

```
DELETE FROM 表 WHERE 条件;
```

出处：23数媒2班 陆云清
