## 数据库简介

### 什么是数据库？

存放数据的仓库;

### 数据库的分类？

根据存储方式不同可划分为：

1. 关系型数据库：结构化的数据库，创建在关系模型（二维表格模型）基础上的数据库;常见的关系型数据库有Access、Sql Sever、Oracle、DB2、Mysql（免费）；
2. 非关系型数据库：常见的非关系性数据库有Redis、MongBD、Hbase;

### 关系型数据库中的字段和实体

1.字段:表中的列;

2.实体：表中的一行对于一个实体，或者叫做一条记录;

### 什么是SQL？

SQL 是一种数据库查询和程序设计语言，用于存取数据以及查询、更新和管理关系数据库系统。

### SQL语言的分类

SQL分为数据操纵语言（DML）、数据定义语言(DDL)、数据控制语言（DCL）；

DML: 用于查询修改数据记录；主要包括INSERT 、UPDATE、DELETE、SELECT；

DDL：定义数据库的结构,如创建、修改或删除数据库对象；

主要包括CREATE TABLE（创建表）、ALTER TABLE(更改表结构、添加、删除、修改列长度)、DROP TABLE(删除表)、CREATE INDEX(创建索引)、DROP INDEX(删除索引)等

DCL：用于控制数据库的访问；主要包括GRANT(授予访问权限)、REVOKE(撤销访问权限)、COMMIT(提交事务处理)、ROLLBACK(事务处理回退)、SAVEPOINT(设置保存点)、LOCK(对数据库特定部分进行锁定)等；

### SQL语言的注意事项

1. SQL语言的大小写不敏感
2. 可以写在一行或多行下

## Oracle安装和配置

## 表的创建和管理

### Oracle支持的基本数据类型

|  |  |  |
| --- | --- | --- |
| 数据类型 | 对应Java中的类型 | 说明 |
| varchar2(length) | String | 可变字符串，不会用空格填充至最大长度； |
| char(length) | String | 定长字符串，当位数不足自动用空格填充来达到其最大长度; |
| NUMBER( p,s ) | Integer/Double/Float/double/int; | p表示精度（总长度） s表示小数位数且四舍五入; |
| Date | Date |  |
| TIMESTAMP | Date |  |
| BLOB | 二进制（电子表格、字处理文档、图像文件等） |  |
| CLOB | 文本数据（4G） |  |

### Oracle支持的约束

|  |  |  |
| --- | --- | --- |
| 约束 | 语法 | 说明 |
| 主键约束 | primary key(字段) | 非空+唯一 |
| 唯一约束 | unique(字段) |  |
| 非空约束 | not null |  |
| 检查约束 | check(约束) |  |
| 外键约束 | constraint FK\_外键字段\_主键表 foreign key(外键列) references 主键表（主键） |  |

### 创建表

语法：

CREATE TABLE 表名(

字段 数据类型 [约束],

字段 数据类型 [约束]

);

实例：

CREATE TABLE t\_b\_user(

id number primary key,

userName VARCHAR2(18),

pass varchar2(18),

sex char(1)

);

## 约束的创建和管理

### 创建约束的两种方式:

1. 创建表的的时候创建约束(不推荐)；

CREATE TABLE t\_b\_user(

id number primary key,

userName VARCHAR2(18) UNIQUE NOT NULL,

pass varchar2(18) NOT NULL,

sex char(1) CHECK(SEX='0' OR SEX='1')

);

1. 创建表之后单独创建约束;(但是一般主键在创建表的时候创建)

a.添加约束语法：

ALTER TABLE 表名 ADD constraint 约束名称 约束

实例：

ALTER TABLE t\_b\_user ADD constraint uq\_username unique(userName);

## SQL基础查询

#### 简单查询

简单查询语法：

select [distinct]\*|列名称[别名],列名称[别名]... from 表名 [别名]；

执行顺序：

1. from子句,然后select子句;

说明：distinct 如果查询的是多个列,只有在多个列都完全重复的时候才可以消除重复;

distinct

常量

四则运算

限定查询语法：

select [distinct]\*|列名称[别名],列名称[别名]... from 表名 [别名] [where] 条件

执行顺序：

1.确定数据来源 from

2.确定满足数据的数据行 where

3.控制要显示的数据列 select

#### 限定查询

限定查询的符合

1.关系运算符：>,<,<=,>=,<>(!=);

2.逻辑运算符：AND,OR,NOT

3.范围运算符：BETWEEN...AND,IN,NOT IN;

4.空判断：IS NULL,IS NOT NULL

5.模糊查询：LIKE

说明：

"="可以在数字上使用,也可以在字符串中使用

NOT IN和NULL的问题

1.在使用NOT IN 进行范围判断的时候，如果范围里面包含了NULL,不会有任何的结果返回;

--6.模糊查询：LIKE

'-':匹配任意的一个字符；

'%':匹配任意的零位、一位或多位字符;

like 可以应用在各种类型上；

like 如果不设置查询关键字,那么表示查询所有;

#### 排序查询

order by 排序查询

select [distinct]\*|列名称[别名],列名称[别名]... from 表名 [别名] [where] 条件 order by 字段[ASC(默认)/DESC,字段[ASC/DESC],....]

1.确定数据来源 from

2.确定满足数据的数据行 where

3.控制要显示的数据列 select

4.针对查询结果排序 可以使用Select语句中的别名;order by

## SQL常用函数

### 字符串函数

### 数值函数

### 日期函数

### 转换函数

### 通用的函数

## SQL多表查询

### 多表查询的实现形式

【3.确定要使用的数据列】select [distnct]\*|列[别名],....|常量

【1.确定数据来源】from 表名[别名],表名[别名]

【2.对数据进行筛选】[where 条件]

【4.针对返回的结果进行排序】order by 字段[asc|desc,...]

### 笛卡尔积的处理

### 数据表的连接操作

### 数据的集合操作

## SQL分组统计查询

### 基本的统计函数

count(\*|字段|[distinct]|常量)

max(字段（日期或数字）)

min(字段)

sum(数字字段)

avg（数字字段)

### count(\*),count(字段),count(distinct 字段)的区别;

1.count(\*):明确的返回表中的数据个数;是最准确的;

2.count(字段)：不统计为null的数据个数,如果某一列的数据不可能为空，

那么结果与 count(\*)相同;

3.count(distinct 字段):统计消除重复数据后的数据个数

4.COUNT(常量) 和 COUNT(\*)表示的是直接查询符合条件的数据库表的行数，

推荐使 用count(\*);

## SQL子查询

## 数据更新和事务管理

## 数据伪列

## 序列

## 视图

## 索引

## 存储过程

## 触发器

## 游标

## 用户管理

## 数据库备份

## 数据库设计