一

1独立性

2结构化

3共享

4减少冗余度

数据库是有数据表组成的

数据表的每一行称为一条数据记录，每一列为一字段，数据表是彼此相关数据的集合

数据库是用来存放数据的

数据库管理系统是用来管理和控制数据库的

数据库服务器是运行数据库管理系统对数据库实施管理的服务器

在关系数据库中数据被放在一张二维表中，一个关系数据库包含多张数据表，每一张数据表又包含行和列，数据表之间相互关联，这种关联是通关主键和外键的参照关系实现的。

数据库不仅包含数据表而且还包含了其他的数据对象，例如视图、存储过程、函数等

关键字（key）是关系模型中的一个重要概念，它是一个逻辑结构，不是数据库的物理部分，它分为主关键字和外部关键字两种，主关键字（primary key）是数据库中的某一列，它的值唯一标识一行，主关键字的值必须唯一，不允许有null出现，或重复值出现。外部关键字又称外键（foreign key）它是指数据表中含有与另外一个数据表主关键字相对应的列

某一列在A表中做主键在B表中作为普通列，那么该列称为B表的外键，我们称A为父表，称B表为子表。子表中的外键值的取值只要在父表的主键值的范围内即可。

二

Oracle默认占用端口号1521

原则上一台计算机可以安装多个oracle实例，但实际操作中只允许安装一个实例

三

1.数据库规模无限制

2.支持windows Linux Unix

3.支持64位

4.具备故障保护配置和验证windows集群功能，并通过与微软集群服务器集成的高可用性软件快速准确地自动恢复具备回闪查询功能无需耗时复杂的操作即可恢复更早版本的数据

5.具备回闪数据表、数据库和事物查询功能，诊断和撤销错误操作以缩短恢复时间

6.具备数据卫士创建，维护并监控一个或多个备用数据库功能，以保护企业数据受故障错误和灾难的影响

7.具备oracle数据库安全备份的功能，针对oracle数据库和异构文件系统的安全高性能的磁带备份管理可降低网络数据保护的成本和复杂性，

8.具备服务器管理的备份和回复功能，借助oracle回复管理器，简化自动化并提高恢复性能。

9.具备真正集群的功能，跨多个相互连接或集群的服务器运行任意未作运行的打包或定制得管理软件

10.具备集成的集群件，利用一组通用内置的集群服务创建和运行数据库集群。

11.具备自动工作负载管理，将服务连接请求发送给拥有最低负载的服务器，一旦发生故障，自动将幸存的服务器重新分配以用于服务。

12.具备java和pl/sql的本地编译功能，用java和pl/sql语言编写部署在数据库中的程序。

13.oracle data base vault限制超级用户和特权用户对管理软件和数据的访问 oracle audit vault可自动收集来自多个服务器的审计数据，是审计数据转换化为一种安全的。

14.具备oracle的高级安全性，符合法律隐私的安全性，具备oracle的标签安全性，提供线程的安全性，用户安全检查与附加在数据行上的数据。

15.具备安全应用决策，仅在用户通过了大量安全检查后才激活决策。

16.具备虚拟专用数据库，通过得很入行级的基于政策的访问控制增强安全性和隐私性

17.具备细粒度审计功能，实现非常有针对性的审计而不受背景噪声的影响

18.

19.具备数据加密工具包，通过为存储在介质上的数据提供一个额外的保护层来保护重要的数据

Dbwr将数据高速缓存区中数据写入到数据文件中，日志写进程（lgwr）将日志缓存区中数据写入到数据文件中。

※Oracle总是先写日志再做修改※

检查点进程（ckpt）

归档进程（arch）只有在归档模式下才有用

监控进程（smon）回滚未完成事物

进程监控进程（pmon）

Oracle 11g数据库存储结构是由表空间构成的，物理上是由数据和表空间构成的，段在逻辑上是有区组成，区在逻辑上是有块组成，表空间的大小就是构成它的数据文件大小之和。

表空间分为系统表空间，用户表空间，索引表空间，临时表空间，撤销表空间。

用户表空间：数据表、视图、存储过程、索引函数

索引表空间：存放数据库中的所有索引

临时表空间：

表空间是一个数据库的逻辑区域，

方案是存放用户所拥有的数据对象的集合，方案与用户是一一对应的，即每个方案是保存在每个用户下的，oracle 11g是靠用户来管理所有数据的

只有数据库管理员才能创建其他用户

*create user 用户名 identified by "密码"*

*default tablespace 用户表空间*

*temporary tablespace 用户表空间*

*drop user 用户 cascade 删除用户*

*alter user 用户 identified by "新密码" 修改密码*

*alter user 用户 account lock/\*锁\*/ /\*unlock 解锁\*/ 冻结/解冻用户*

权限可以分为系统权限和对象权限，系统权限是oracle自带权限比如创建数据表。对象权限指某用户对某个数据对象的可操作权限，例如插入删除数据

Connect角色：作用：连接数据库

Resource：开发最基本角色

Dba：数据库管理员，只有DBA有权为普通用户分配权限与角色

只有拥有了connect和resource角色才能登陆和操作数据库只有DBA能回收

*grant 角色to 用户 分配角色*

*revoke/\*回收\*/ 角色 from 用户*

*grant 系统权限B to 用户A*

*revoke系统权限B from 用户A*

*grant 对象权限C on 数据表B to 用户A with grant option（权限传递）*

*create table emp（*

*id int，*

*no varchar2（50）*

*）*

*grant all on 数据表B to 用户A*

*revoke 对象权限C on 数据表B from 用户A*

*revoke all on 数据表B from 用户A*

*create tablespace 表空间 datafile‘路径+文件名1.dbf’ size 值 1，*

*create temporary tablespace 表空间 tempfile‘路径+文件名1.dbf’ size 值 1，*

*alter database datafile/tempfile’路径+文件名.dbf’ resize 值*

*alter tablespace 表空间 add datafile/tempfile’路径+文件名.dbf’ size 值*

*drop database 表空间（删除空表空间）*

*drop tablespace 表空间 including contents and datafiles*

ddl（database define language）数据库定义语言（包括：create table、create procedure）

dml（database management language）数据库管理语言（包括：增加，删除，修改）

dcl（database control language）数据库控制语言（包括 grant、revoke）

dql（database query language）数据库查询语言

*set autocommit on/off 开启/关闭commit命令*

*savepoint 名称 设置存储点*

*rollback to 保存点 回滚到保存点*

*create table 表名（*

*字段名1 数据类型【约束】*

*字段名2数据类型【约束】*

*）*

*[tablespace 表空间]*

*Primary key：主键约束，用来标明记录在数据表中的唯一性，即被主键修饰的字段值不能为空且在表中不能重复*

*Unique：唯一约束，用来标明记录在数据表中的唯一性，字段值可以为空*

*not null：非空约束*

*check：检查约束，字段值必须满足该约束的条件限制*

*default：默认约束，如果不给该字段赋值那么该字段的值将为default约束限定值*

*foreign key：外键约束，用来建立数据表之间的关系，被修饰字段在一张数据表中做主键（主表或父表），在另一张数据表中做外键（从表或子表）*

*alter table 数据表 add 字段 类型【约束】*

*alter table 数据表 modify 字段 类型* *修改某一项的数据类型*

*alter table 数据表 drop column 字段 删除某一字段*

如果两表建立了主外键关系，必须先删除子表再删除父表，否则会报错

*alter table 数据表 add constraint 主键约束名（默认pk\_+字段名）primary key(字段)--加主键*

*alter table 数据表 add constraint 唯一约束名（默认pk\_+字段名）unique(字段)--加唯一约束*

*alter table 数据表 add constraint 检查约束名（默认pk\_+字段名）check(表达式)--加检查约束*

*alter table 数据表1 add constraint 外键约束名 foreign key（字段）references 数据表2（字段）*

*alter table 数据表 drop constraint 约束名*

*alter table 数据表 disable constraint 约束名*

*alter table 数据表 enable constraint 约束名*

*create table 数据表 as select 语句 （数据表数据迁移）*

*delete from 数据表名【where条件表达式】 （删除数据）*

*update 数据表名 set 字段1=值1，字段2=值2……【where条件表达式】 （修改字段值）*

*select 字段1，字段2…… from 数据表1，数据表2……【结果集】【where条件表达式】【group by 字段 having 条件表达式】【order by 排序条件表达式】*

*%为通配符表示任意长度的字符串，like配合%使用表示模糊搜索*

*nvl（表达式一，表达式二）如果表达式1不为空则uvl值等于表达式一的值，如果表达式一为空，则uvl值为表达式二的值*

*rownum表示前n行记录*

*count 统计记录条数*

*max找最大的数*

*avg（）求平均值*

*sum（）求和*

*distinct 去除重复记录*

*dual 虚拟表*

*length 长度*

*initcap（）首字母大写 //select 字段 from表*

*lpad（表达式1，表达式2，表达式3）用表达式3左填充表达式1使长度为表达式2*

*rpad同上右填充*

*replace（表达式1，表达式2，表达式3）表达式2包含于表达式1，用表达式3替换表达式2，表达式3可以为空*

*instr（表达式1，表达式2）查找表达式2在表达式1中的位置找到返回位置数，未找到返回0*

*trim（表达式）去掉表达式两边的空格*

*ltrim（表达式）去掉左侧空格*

*rtrim（表达式）去掉右侧空格*

*ascii（字符）取字符的ascii码值，如括号内为字符串则取第一个字符*

*abs（表达式）取绝对值*

*mod（表达式1，表达式2）表达式1对表达式2做去模运算*

*trunc（表达式，m）保留m位小数※trunc函数不四舍五入※*

*round（表达式，m）同上※四舍五入※*

*sysdate 取系统时间*

*add\_months（表达式，m）为日期加上m个月*

*months\_between（表达式1，表达式2）返回日期型变量，求表达式2与表达式1的月份差*

*last\_day（表达式）返回表达式最后一天*

*next\_day（表达式1，表达式2）从表达式1开始到下一个表达式2是几天，返回日期型*

*to char（表达式，’格式’）格式转换，转成字符型*

*--yyyy四位数表示年份*

*--yy两位数表示年份*

*--mm表示月份*

*--dd表示天*

*--表示星期几*

*--hh12显示时间12进制*

*-- hh24显示时间24进制*

*--mi表示分钟*

*--ss表示秒*

*To number（表达式）将表达式转换为数字*

*子查询是指在一个select中使用另一个*

*不相关子查询：子查询可以不依赖主查询可以独立运行，求解方式：由内向外将子查询结果作为条件*

*相关子查询：子查询可以依赖主查询不能独立运行，将主查询作为结果*

*All是指满足子查询的所有条件*

*Create or replace synonym 同义词名 for 表名*

*匿名块：*

*Declare*

*变量声明*

*Begin*

*执行体；*

*（条件选择语句和顺序结构和循环语句的结合体）*

*异常处理；*

*End；*

*Dbms\_output.put\_line（表达式） --输出表达式*

*变量名 表.列%type； --定义列变量*

*变量名 表%rowtype； --定义行变量*

*变量名 ：=表达式； --为变量赋值*

*变量名 ：=&变量名； --交互式赋值*

*Select 字段列表 into 变量列表 from 数据表列表 where 条件表达式*

*Dbms\_output.put\_line(‘c=’||c); 输出语句*

*If 表达式 then*

*执行体;*

*End if;*

*If 表达式1 then*

*执行体;*

*Elsif 表达式n then*

*执行体;*

*End if;*

*Case 表达式1*

*When 值1 then 执行体1;*

*……*

*When 值m then 执行体 m；*

*Else*

*执行体n；*

*Endcase；*

*Loop*

*Exit when 表达式；*

*执行体；*

*End loop；*

*While 表达式*

*Loop*

*执行体；*

*End loop；*

*For 变量 in 最小值..最大值*

*Loop*

*执行体；*

*End loop；*

*操作游标：1.声明游标2.打开游标3.提取游标4.关闭游标*

*Cursor 游标名 is select 语句 声明游标并指向查询结果集中的第一条记录*

*Open 游标名 打开游标*

*Fetch 游标名 into 变量 提取游标*

*Close 游标名*

*游标名%found*

*游标名%notfound*

*游标名%rowcount 游标读取记录数*

*游标名%isopen 判断游标是否打开*

*Cursor 游标名（变量 类型（不需要指明长度）） is select 语句 where 条件 = 变量 带参数游标，更精确指向*

*Open 游标名（实参名）*

*Fetch 游标名 into 变量 fetch一次就向后一位*

*Create or replace procedure 存储过程名（参数列表）*

*as|is*

*变量声明部分*

*Begin*

*执行体*

*End；*

*无论存储过程还是函数，参数类型共三种1.in类型2.out类型3.in out类型*

*无论是调用存储过程还是调用函数，都采用匿名块的方式调用*

*Create or replace function 函数名（参数列表）return*

*as|is*

*变量声明部分*

*Begin*

*执行体；*

*Return表达式；*

*异常处理；*

*End；*

*触发器（一个特殊的存储过程，不传进任何参数，不传出任何参数，由特定事件触发，分为两类DML触发器（根据时间可分为before触发和after触发，根据级别可分为行级触发器和语句级别触发器）和代替触发器（主要针对视图））*

*Create or replace trigger 触发器名*

*Instead of |Before|after insert|delete|update on 数据表*

*【for each row】*

*Declare*

*声明变量；*

*Begin*

*执行部分*

*End；*

*对于insert触发器只有新值没有旧值，update既有新值又有旧值，delete只有旧值没有新值*

*伪列：：old代表触发之前的值，：new代表触发之后的值*