1. DB2
2. 基本概念：
3. DB2支持以下两种类型的表空间：
4. 系统管理存储器表空间（SMS-SYSTEM   MANAGED   STORAGE）
5. 数据库管理存储器表空间（DMS-DATABASE   MANAGED   STORAGE）
6. SMS、DMS用户表空间的特性对照

特性                                                                       SMS                       DMS     
  能够在表空间中动态的增加容器数量                         n                     y               
  能够把索引数据存放到不同的表空间                         n                     y     
  能够把长型(LOB)数据存放到单独的表空间               n                     y           
  表可以分散存放到多个表空间                                     n                     y     
  只在需要的时候才分配空间                                         y                     n           
  表空间可以被定向到不同类型的磁盘空间                 y                     n       
  创建之后，区段大小（extent   size）能够改变       n                   n

1. 默认表空间: 当创建数据库的时候，DB2将按照默认方式创建三个表空间：这些表空间缺省是SMS模式：
2. SYSCATSPACE:   包含系统编目
3. TEMPSPACE1:保存临时表
4. USERSPACE1:包含用户数据
5. 实例(Instance)和数据库(Database)：在一个操作系统中，DB2数据服务可以同时运行多个实例(有别于Oracle在一个系统内只能起一个实例). 数据库定义在实例中，一个实例可以包含多个数据库。在同一个实例中的不同数据库是完全独立的，分别拥有自己独立的系统编目表。
6. 表空间(TableSpace)：表空间有两种管理方式：DMS(Database management Space)方式 ；SMS(System manegement Space)方式；DMS与SMS方式在表空间建立时指定，建好后不能转换。对于DMS方式，一个表空间对应了一个或多个容器(Container)，容器指定了数据的物理存储位置。对于SMS方式，只能够指定一个目录，不能够增加。
7. 表空间类型：
8. 系统编目表空间(SysCatSpace)
9. 系统临时表空间(SysTempSpace)
10. 用户表空间(UserSpace)
11. 用户临时表空间(UserTempSpace)
12. 一个数据库中必须存在两个系统基本的表空间，分别是系统编目表空间与系统临时表空间。在数据库中创建的任何对象都以在系统编目表空间中增加记录的方式体现，对于临时表空间，其占用磁盘大小是根据使用情况动态伸缩的，即仅在需要时才分配磁盘空间，并在使用后进行回收。此外，若用户需要创建表，则需要创建用户表空间，若需要使用临时表，则需要创建用户临时表空间。
13. 容器类型：
14. Files 文件 : 用于DMS方式的表空间
15. Devices 设备 : 用于DMS方式的表空间
16. Directory 目录 : 用于SMS方式的表空间，此种方式不需要人工管理数据存储文件，DB2可根据情况在目录中自动增加存储文件，只要磁盘空间允许。
17. 影响一个数据库的性能主要有以下因素:
18. 磁盘(Disk): 以磁盘最为显著，90%的性能瓶颈可能来自于磁盘的IO竞争;
19. 内存(Memory) : 一方面是指物理内存的总量要满足需求，另一方面是指与内存相关的配置参数应正确配置
20. 处理器(CPU) : 多路CPU会对哪些依赖计算能力的复杂SQL查询起到显著的效果
21. 网络(Network)
22. DB2的参数配置分为两个级别，一个是实例级别，另一个是数据库级别
23. 对数据服务性能影响较大的参数主要在数据库级别配置。以下是三个比较重要的内存配置参数:
24. bufferpagelocklistsortheap
25. bufferpage: 由同一个数据库中的所有对象共享。
26. sortheap: 用于排序的内存交换区，非共享，不宜设置太大，否则，很容易引起内存耗尽，因为每一个事务都会申请独立的内存用于排序。
27. locklist: 共享内存，用于记录数据服务运行中建立的锁。DB2默认使用行级锁，如果设置太小，当锁的记录太多时，则会导致内存不足，此时DB2会把多个行锁升级为一个表锁，这样就会大大降低应用程序的并发性能。如果设置太大，则多分配的内存很少会被用到，导致浪费。
28. numdb: 同时可以启动的实例数目
29. DB2的常用命令
30. 打开命令行窗口: db2cmd
31. 打开控制中心: db2cmd db2cc
32. 打开命令编辑器 db2cmd db2ce
33. 启动数据库实例 db2start
34. 停止数据库实例 db2stop
35. 如果你不能停止数据库由于激活的连接，在运行db2stop前执行db2 force application all就可以了 /db2stop force
36. 创建数据库 db2 create db [dbname]
37. 连接到数据库 db2 connect to [dbname] user[username] using [password]
38. 断开数据库连接 db2 connect reset
39. 列出所有数据库 db2 list db directory
40. 列出所有激活的数据库 db2 list active databases
41. 列出所有数据库配置 db2 get db cfg
42. 删除数据库 db2 drop database [dbname]
43. 列出所有用户表 db2 list tables
44. 列出所有系统表 db2 list tables for system
45. 列出所有表 db2 list tables for all
46. 列出系统表 db2 list tables for system
47. 列出用户表 db2 list tables for user
48. 列出特定用户表 db2 list tables for schema[user]
49. 创建一个与数据库中某个表(t2)结构相同的新表(t1) db2 create table t1 like t2
50. 将一个表t1的数据导入到另一个表t2 db2 "insert into t1 select \* from t2"
51. 查询表 db2 "select \* from tablename where ..."
52. 显示表结构 db2 describe table tablename
53. 修改列 db2 alter table [tablename]alter column [columname] set data type varchar(24)
54. 执行脚本文件 db2 -tvf scripts.sq
55. 查看命令帮助 db2 ? db2start
56. 查看错误码信息 db2 ? 22001
57. 详细命令请使用"db2 ? <command>"进行查看。
58. 备份数据库 db2 backup db <db name>
59. 在线备份数据库 db2 -v "BACKUP DATABASE <database name> ONLINE TO <path> WITH2 BUFFERS BUFFER 1024 INCLUDE LOGS WITHOUT PROMPTING"
60. 恢复数据库 db2 restore db <source db name>
61. 在线恢复数据库 db2 "RESTORE DB <database name> TO <db path> LOGTARGET<logpath> WITHOUT PROMPTING" 和db2 "ROLLFORWARD DB <database name> TO END OF LOGS AND STOP"...
62. 导出数据文件 db2move <db name> export

[-sn <模式名称，一般为db2admin>]

[-tn <表名，多个之间用逗号分隔>]

1. 导入数据文件 db2move <db name> import
2. 获取db2数据库管理配置环境信息 db2 get dbm cfg
3. .获取db2某个数据库数据库管理配置环境信息 db2 get db cfg for <db name>或者：连接至某个数据库以后执行db2 get db cfg
4. 更改db2日志空间的大小

备注：以下命令为了防止db2数据库过份使用硬盘空间而设，仅用于开发者自己机器上的db2，如果是服务器，则参数需要修改。 db2 UPDATE DB CFG FOR <db name> USING logretain OFF logprimary 3logsecond 2 logfilsiz 25600;如果页大小是4KB，则以上命令创建3个100M的日志文件，占用300MB硬盘空间。25600\*4KB=102400KB。

1. 创建临时表空间 DB2 CREATE USER TEMPORARY TABLESPACE STMASPACE PAGESIZE 32 K MANAGED BYDATABASE USING (FILE 'D:\DB2\_TAB\STMASPACE.F1' 10000)

EXTENTSIZE 256

1. 获取数据库管理器的快照数据 db2 –v get snapshot for dbm
2. 显示进行程号 db2 list applications show detail

以默认分隔符加载,默认为“,”号db2 "import from btpoper.txt of del insert into btpoper"； 以指定分隔符“|”加载db2 "import from btpoper.txt of del modified by coldel| insert intobtpoper"

1. 卸载一个表中全部数据

db2 "export to btpoper.txt of del select \* from btpoper"

db2 "export to btpoper.txt of del modified by coldel| select \* frombtpoper"

1. 带条件卸载一个表中数据

db2 "export to btpoper.txt of del select \* from btpoper wherebrhid='907020000'"

db2 "export to cmmcode.txt of del select \* from cmmcode wherecodtp='01'"

db2 "export to cmmcode.txt of del modified by coldel| select \* fromcmmcode where codtp='01'"

1. 查询数据结构及数据：

db2 "select \* from btpoper"

db2 "select \* from btpoper where brhid='907020000' and oprid='0001'"

db2 "select oprid,oprnm,brhid,passwd from btpoper"

1. 删除表中数据：

db2 "delete from btpoper"

db2 "delete from btpoper where brhid='907020000' orbrhid='907010000'"

1. 修改表中数据：

db2 "update svmmst set prtlines=0 where brhid='907010000' and jobtp='02'"

db2 "update svmmst set prtlines=0 where jobtp='02' or jobtp='03'"

1. 联接数据库

db2 connect to btpdbs

1. 清除数据库联接

db2 connect reset 断开数据库连接

db2 terminate 断开数据库连接

db2 force applications all 断开所有数据库连接

1. 备份数据库

1、 db2 backup db btpdbs

2、 db2move btpdbs export

db2look -d btpdbs -e -x [-a] -o crttbl.sql

1. 恢复数据库

1、 db2 restore db btpdbs withoutrolling forward

2、 db2 -tvf crtdb.sql

crtdb.sql文件内容：create db btpdbs on /db2catalog

db2 -stvf crttbl.sql

db2move btpdbs import

1. DB2帮助命令：

db2 ?

db2 ? restroe

db2 ? sqlcode (例：db2 ? sql0803) 注：code必须为4位数，不够4位，前面补0

1. bind命令：将应用程序与数据库作一捆绑,每次恢复数据库后，建议都要做一次bind

(1) db2 bind br8200.bnd

(2) /btp/bin/bndall /btp/bnd

/btp/bin/bndall /btp/tran/bnd

1. 查看数据库参数：

db2 get dbm cfg

db2 get db cfg for btpdbs

1. 修改数据库参数：

db2 update db cfg for btpdbs using LOGBUFSZ 20

db2 update db cfg for btpdbs using LOGFILSIZ 5120

改完后，应执行以下命令使其生效：

db2 stop

db2 start

1. db2 set schema btp 修改当前模式为"btp"
2. db2 list tablespaces show detail 查看当前数据库表空间分配状况
3. db2 list tablespace containers for 2 show detail 查看tablespace id=2使用容器所在目录
4. db2 list application
5. db2 list db directory 列出所有数据库
6. db2 list active databases 列出所有活动的数据库
7. db2 list tables for all 列出当前数据库下所有的表
8. db2 list tables for schema btp 列出当前数据库中schema为btp的表
9. db2 list tablespaces show detail 显示数据库空间使用情况
10. db2 list packages for all
11. db2 "import from tab76.ixf of ixf commitcount 5000 insert intoachact"
12. db2 "create table achact\_t like achact"
13. db2 "rename table achact\_t to achact"
14. db2 "insert into achact\_t select \* from achact where txndt>=(selectlstpgdt from
15. acmact where actno=achact.actno)"
16. db2 get snapshot for dynaimic sql on jining
17. 删除一个实例

cd /usr/lpp/db2\_07\_01/instance

./db2idrop InstName

1. 列出所有DB2实例：

cd /usr/lpp/db2\_07\_01/bin

./db2ilist

1. 为数据库建立编目

db2 catalog db btpdbs on /db2catalog

1. 取消已编目的数据库btpdbs

db2 uncatalog db btpdbs

1. 查看版本

db2level

1. 显示当前数据库管理实例

db2 get instance

1. 设置实例系统启动时是否自动启动。

db2iauto -on 自动启动

db2iauto -off 不自动启动

1. 数据库优化命令：

reorg、runstats

当数据库经过一段时间使用，数据空间会变得越来越庞大。一些delete掉

的数据仍存放在数据库中，占用数据空间，影响系统性能。因此需要定期

运行reorg、runstats命令，清除已delete的数据，优化数据结构。

db2 reorg table 表名

db2 runstats on table 表名 with distribution and indexes all

因为要优化的表比较多，所以在/btp/bin目录下提供了一个sh程序runsall，

可在当天业务结束后，运行runsall，对数据库进行优化

1. 更改db2日志空间的大小

备注：以下命令为了防止db2数据库过份使用硬盘空间而设，仅用于开发者自己机器上的db2，如果是服务器，则参数需要修改。

db2 UPDATE DB CFG FOR <db name> USING logretain OFF logprimary 3logsecond 2 logfilsiz 25600;

如果页大小是4KB，则以上命令创建3个100M的日志文件，占用300MB硬盘空间。25600\*4KB=102400KB。

1. 创建临时表空间

#DB2 CREATE USER TEMPORARY TABLESPACE STMASPACE PAGESIZE 32 K MANAGED BYDATABASE USING (FILE 'D:\DB2\_TAB\STMASPACE.F1' 10000) EXTENTSIZE 256

1. 创建表空间

rem 创建缓冲池空间 8K

db2 connect to gather

db2 CREATE BUFFERPOOL STMABMP IMMEDIATE SIZE 25000 PAGESIZE 8K

rem 创建表空间：STMA

rem 必须确认路径正确

rem D:\DB2Container\Stma

db2 drop tablespace stma

db2 CREATE REGULAR TABLESPACE STMA PAGESIZE 8 K MANAGED BY SYSTEM USING('D:\DB2Container\Stma' ) EXTENTSIZE 8 OVERHEAD 10.5 PREFETCHSIZE 8TRANSFERRATE 0.14 BUFFERPOOL STMABMP DROPPED TABLE RECOVERY OFF

#db2 connect reset

1. 将暂挂的数据恢复到前滚状态

#db2 ROLLFORWARD DATABASE TESTDB TO END OF LOGS AND COMPLETE NORETRIEVE

1. 备份表空间

#BACKUP DATABASE YNDC TABLESPACE ( USERSPACE1 ) TO "D:\temp" WITH 2BUFFERS BUFFER 1024 PARALLELISM 1 WITHOUT PROMPTING

1. 创建db2工具数据库

#db2 create tools catalog systools create new database toolsdb

1. 如何进行增量/差量备份

增量：上一次完整备份至本次备份之间增加的数据部分；

差量(delta)：上次备份以来（可能是完整备份、增量备份或者差量备份）至本次备份之间增加的数据部分；

1. 更新所有表的统计信息

db2 -v connect to DB\_NAME

db2 -v "select tbname, nleaf, nlevels, stats\_timefromsysibm.sysindexes"

db2 -v reorgchkupdate statistics on table all

db2 -v "select tbname, nleaf, nlevels, stats\_timefromsysibm.sysindexes"

db2 -v terminate

1. 对一张表运行统计信息

#db2 -v runstatson table TAB\_NAMEand indexes all

1. 查看是否对数据库执行了RUNSTATS

db2 -v "select tbname, nleaf, nlevels,stats\_timefromsysibm.sysindexes"

1. 更改缓冲池的大小

缓冲池中，当syscat.bufferpools的npages是-1时，由数据库的配置参数bufferpage控制缓冲池的大小。

将npages的值更改为-1的命令：

#db2 -v connect to DB\_NAME

#db2 -v select \* from syscat.bufferpools

#db2 -v alter bufferpoolIBMDEFAULTBP size -1

#db2 -v connect reset

#db2 -v terminate

更改数据库配置参数BufferPages的命令如下：

#db2 -v update db cfgfor dbnameusing BUFFPAGE bigger\_value

#db2 -v terminate

1. 看数据库监视内容列表

#db2 -v get monitor switches

1. 打开某个数据库监视内容

#db2 -v update monitor switches using bufferpoolon

1. 获取数据库快照

#db2 -v get snapshot for all databases > snap.out

#db2 -v get snapshot for dbm>> snap.out

#db2 -v get snapshot for all bufferpools>> snap.out

#db2 -v terminate

1. 重置数据库快照

#db2 -v reset monitor all

1. 计算缓冲池命中率

理想情况下缓冲池命中率在95%以上，计算公式如下：

(1 -((buffer pool data physical reads + buffer pool index physical reads)

/(buffer pool data logical reads + pool index logical reads))) \*100%

1. 创建db2实例

#db2icrt <实例名称>

1. 删除db2实例

#db2idrop <实例名称>

1. 设置当前db2实例

#set db2intance=db2

1. 显示db2拥有的实例

#db2ilist

1. 恢复离线增量备份数据库的命令

#DB2 RESTORE DATABASE YNDC INCREMENTAL AUTOMATIC FROM D:\backup\autobak\db2TAKEN AT 20060314232015

1. 创建样本数据库

在unix平台，使用：

#sqllib/bin/db2sampl <path>

在windows,os/2平台，使用：db2sampl e,e是可选参数，指定将创建数据库的驱动器

1. 设置联合数据库为可用（默认联合数据库不可用）

#db2 update dbm cfg using federated yes

1. 列出数据库中所有的表

#db2 list tables

1. 数据迁移方法1

export脚本示例

#db2 connect to testdb user test password test

#db2 "export to aa1.ixf of ixf select \* from table1"

#db2 "export to aa2.ixf of ixf select \* from table2"

#db2 connect reset

import脚本示例

#db2 connect to testdb user test password test

#db2 "load from aa1.ixf of ixf replace into table1 COPY NO withoutprompting "

#db2 "load from aa2.ixf of ixf replace into table2 COPY NO withoutprompting "

#db2 connect reset

1. MYSQL

**1、连接Mysql**

格式： mysql -h主机地址 -u用户名 －p用户密码

1、连接到本机上的MYSQL。

首先打开DOS窗口，然后进入目录mysql\bin，再键入命令mysql -u root -p，回车后提示你输密码.注意用户名前可以有空格也可以没有空格，但是密码前必须没有空格，否则让你重新输入密码。

如果刚安装好MYSQL，超级用户root是没有密码的，故直接回车即可进入到MYSQL中了，MYSQL的提示符是： mysql>

2、连接到远程主机上的MYSQL。假设远程主机的IP为：110.110.110.110，用户名为root,密码为abcd123。则键入以下命令：

mysql -h110.110.110.110 -u root -p 123;（注:u与root之间可以不用加空格，其它也一样）

3、退出MYSQL命令： exit （回车）

**2、修改密码**

格式：mysqladmin -u用户名 -p旧密码 password 新密码

1、给root加个密码ab12。

首先在DOS下进入目录mysql\bin，然后键入以下命令

mysqladmin -u root -password ab12

注：因为开始时root没有密码，所以-p旧密码一项就可以省略了。

2、再将root的密码改为djg345。

mysqladmin -u root -p ab12 password djg345

**3、增加新用户**

注意：和上面不同，下面的因为是MYSQL环境中的命令，所以后面都带一个分号作为命令结束符

格式：grant select on 数据库.\* to 用户名@登录主机 identified by “密码”

1、增加一个用户test1密码为abc，让他可以在任何主机上登录，并对所有数据库有查询、插入、修改、删除的权限。首先用root用户连入MYSQL，然后键入以下命令：

grant select,insert,update,delete on \*.\* to [email=test1@”%]test1@”%[/email]” Identified by “abc”;

但增加的用户是十分危险的，你想如某个人知道test1的密码，那么他就可以在internet上的任何一台电脑上登录你的mysql数据库并对你的数据可以为所欲为了，解决办法见2。

2、增加一个用户test2密码为abc,让他只可以在localhost上登录，并可以对数据库mydb进行查询、插入、修改、删除的操作（localhost指本地主机，即MYSQL数据库所在的那台主机），这样用户即使用知道test2的密码，他也无法从internet上直接访问数据库，只能通过MYSQL主机上的web页来访问了。

grant select,insert,update,delete on mydb.\* to [email=test2@localhost]test2@localhost[/email] identified by “abc”;

如果你不想test2有密码，可以再打一个命令将密码消掉。

grant select,insert,update,delete on mydb.\* to [email=test2@localhost]test2@localhost[/email] identified by “”;

**4、数据库**

4.1 创建数据库

注意：创建数据库之前要先连接Mysql服务器

命令：create database <数据库名>

例1：建立一个名为xhkdb的数据库

mysql> create database xhkdb;

例2：创建数据库并分配用户

①CREATE DATABASE 数据库名;

②GRANT SELECT,INSERT,UPDATE,DELETE,CREATE,DROP,ALTER ON 数据库名.\* TO 数据库名@localhost IDENTIFIED BY '密码';

③SET PASSWORD FOR '数据库名'@'localhost' = OLD\_PASSWORD('密码');

依次执行3个命令完成数据库创建。注意：中文 “密码”和“数据库”是户自己需要设置的。

4.2 显示数据库

命令：show databases （注意：最后有个s）

mysql> show databases;

注意：为了不再显示的时候乱码，要修改数据库默认编码。以下以GBK编码页面为例进行说明：

1、修改MYSQL的配置文件：my.ini里面修改default-character-set=gbk

2、代码运行时修改：

①Java代码：jdbc:mysql://localhost:3306/test?useUnicode=true&characterEncoding=gbk

②PHP代码：header("Content-Type:text/html;charset=gb2312");

③C语言代码：int mysql\_set\_character\_set( MYSQL \* mysql, char \* csname)；

该函数用于为当前连接设置默认的字符集。字符串csname指定了1个有效的字符集名称。连接校对成为字符集的默认校对。该函数的工作方式与SET NAMES语句类似，但它还能设置mysql- > charset的值，从而影响了由mysql\_real\_escape\_string() 设置的字符集。

4.3 删除数据库

命令：drop database <数据库名>

例如：删除名为 xhkdb的数据库

mysql> drop database xhkdb;

例子1：删除一个已经确定存在的数据库

mysql> drop database drop\_database;

Query OK, 0 rows affected (0.00 sec)

例子2：删除一个不确定存在的数据库

mysql> drop database drop\_database;

ERROR 1008 (HY000): Can't drop database 'drop\_database'; database doesn't exist

//发生错误，不能删除'drop\_database'数据库，该数据库不存在。

mysql> drop database if exists drop\_database;

Query OK, 0 rows affected, 1 warning (0.00 sec)//产生一个警告说明此数据库不存在

mysql> create database drop\_database;

Query OK, 1 row affected (0.00 sec)

mysql> drop database if exists drop\_database;//if exists 判断数据库是否存在，不存在也不产生错误

Query OK, 0 rows affected (0.00 sec)

4.4 连接数据库

命令： use <数据库名>

例如：如果xhkdb数据库存在，尝试存取它：

mysql> use xhkdb;

屏幕提示：Database changed

use 语句可以通告MySQL把db\_name数据库作为默认（当前）数据库使用，用于后续语句。该数据库保持为默认数据库，直到语段的结尾，或者直到发布一个不同的USE语句：

mysql> USE db1;

mysql> SELECT COUNT(\*) FROM mytable; # selects from db1.mytable

mysql> USE db2;

mysql> SELECT COUNT(\*) FROM mytable; # selects from db2.mytable

使用USE语句为一个特定的当前的数据库做标记，不会阻碍您访问其它数据库中的表。下面的例子可以从db1数据库访问作者表，并从db2数据库访问编辑表：

mysql> USE db1;

mysql> SELECT author\_name,editor\_name FROM author,db2.editor

-> WHERE author.editor\_id = db2.editor.editor\_id;

USE语句被设立出来，用于与Sybase相兼容。

有些网友问到，连接以后怎么退出。其实，不用退出来，use 数据库后，使用show databases就能查询所有数据库，如果想跳到其他数据库，用 use 其他数据库名字就可以了。

4.5 当前选择的数据库

命令：mysql> select database();

MySQL中SELECT命令类似于其他编程语言里的print或者write，你可以用它来显示一个字符串、数字、数学表达式的结果等等。如何使用MySQL中SELECT命令的特殊功能？

1.显示MYSQL的版本

mysql> select version();+-----------------------+

| version() | +-----------------------+

| 6.0.4-alpha-community | +-----------------------+ 1 row in set (0.02 sec)

2. 显示当前时间

mysql> select now();+---------------------+

| now() | +---------------------+

| 2009-09-15 22:35:32 | +---------------------+ 1 row in set (0.04 sec)

3. 显示年月日

SELECT DAYOFMONTH(CURRENT\_DATE); +--------------------------+

| DAYOFMONTH(CURRENT\_DATE) | +--------------------------+

| 15 | +--------------------------+ 1 row in set (0.01 sec)

SELECT MONTH(CURRENT\_DATE); +---------------------+

| MONTH(CURRENT\_DATE) | +---------------------+ | 9 |

+---------------------+ 1 row in set (0.00 sec)

SELECT YEAR(CURRENT\_DATE); +--------------------+

| YEAR(CURRENT\_DATE) | +--------------------+ | 2009 |

+--------------------+ 1 row in set (0.00 sec)

4. 显示字符串

mysql> SELECT "welecome to my blog!"; +----------------------+

| welecome to my blog! | +----------------------+

| welecome to my blog! | +----------------------+ 1 row in set (0.00 sec)

5. 当计算器用

select ((4 \* 4) / 10 ) + 25; +----------------------+ | ((4 \* 4) / 10 ) + 25 | +----------------------+ | 26.60 |

+----------------------+ 1 row in set (0.00 sec)

6. 串接字符串

select CONCAT(f\_name, " ", l\_name) AS Name

from employee\_data where title = 'Marketing Executive'; +---------------+

| Name | +---------------+

| Monica Sehgal | | Hal Simlai | | Joseph Irvine | +---------------+ 3 rows in set (0.00 sec)

注意：这里用到CONCAT()函数，用来把字符串串接起来。另外，我们还用到以前学到的AS给结果列'CONCAT(f\_name, " ", l\_name)'起了个假名。

**5.1 创建数据表**

命令：create table <表名> ( <字段名1> <类型1> [,..<字段名n> <类型n>]);  
例如，建立一个名为MyClass的表，

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 字段名 | 数字类型 | 数据宽度 | 是否为空 | 是否主键 | 自动增加 | 默认值 |
| id | int | 4 | 否 | primary key | auto\_increment |  |
| name | char | 20 | 否 |  |  |  |
| sex | int | 4 | 否 |  |  | 0 |
| degree | double | 16 | 是 |  |  |  |

mysql> create table MyClass(  
> id int(4) not null primary key auto\_increment,  
> name char(20) not null,  
> sex int(4) not null default '0',  
> degree double(16,2));

**5.3 删除数据表**

命令：drop table <表名>  
例如：删除表名为 MyClass 的表  
   mysql> drop table MyClass;  
DROP TABLE用于取消一个或多个表。您必须有每个表的DROP权限。所有的表数据和表定义会被取消，所以使用本语句要小心！  
注意：对于一个带分区的表，DROP TABLE会永久性地取消表定义，取消各分区，并取消储存在这些分区中的所有数据。DROP TABLE还会取消与被取消的表有关联的分区定义（.par）文件。  
对与不存在的表，使用IF EXISTS用于防止错误发生。当使用IF EXISTS时，对于每个不存在的表，会生成一个NOTE。  
RESTRICT和CASCADE可以使分区更容易。目前，RESTRICT和CASCADE不起作用。

**5.4 表插入数据**

命令：insert into <表名> [( <字段名1>[,..<字段名n > ])] values ( 值1 )[, ( 值n )]  
例如：往表 MyClass中插入二条记录, 这二条记录表示：编号为1的名为Tom的成绩为96.45, 编号为2 的名为Joan 的成绩为82.99， 编号为3 的名为Wang 的成绩为96.5。  
mysql> insert into MyClass values(1,'Tom',96.45),(2,'Joan',82.99), (2,'Wang', 96.59);  
注意：insert into每次只能向表中插入一条记录。

**5.5 查询表中的数据**

1)、查询所有行  
命令： select <字段1，字段2，...> from < 表名 > where < 表达式 >  
例如：查看表 MyClass 中所有数据  
mysql> select \* from MyClass;  
2）、查询前几行数据  
例如：查看表 MyClass 中前2行数据  
mysql> select \* from MyClass order by id limit 0,2;  
select一般配合where使用，以查询更精确更复杂的数据。

**5.6 删除表中数据**

命令：delete from 表名 where 表达式  
例如：删除表 MyClass中编号为1 的记录  
mysql> delete from MyClass where id=1;  
下面是一个删除数据前后表的对比。

|  |  |  |
| --- | --- | --- |
| **FirstName** | **LastName** | **Age** |
| Peter | Griffin | 35 |
| Glenn | Quagmire | 33 |

下面以PHP代码为例删除 "Persons" 表中所有 LastName='Griffin' 的记录：

<?php

$con = mysql\_connect("localhost","peter","abc123");

if (!$con)

{

die('Could not connect: ' . mysql\_error());

}

mysql\_select\_db("my\_db", $con);

mysql\_query("DELETE FROM Persons WHERE LastName='Griffin'"); mysql\_close($con);

?>

在这次删除之后，表是这样的：

|  |  |  |
| --- | --- | --- |
| **FirstName** | **LastName** | **Age** |
| Glenn | Quagmire | 33 |

**5.7 修改表中数据**

语法：update 表名 set 字段=新值,… where 条件  
mysql> update MyClass set name='Mary' where id=1;  
例子1：单表的MySQL UPDATE语句：  
UPDATE [LOW\_PRIORITY] [IGNORE] tbl\_name SET col\_name1=expr1 [, col\_name2=expr2 ...] [WHERE where\_definition] [ORDER BY ...] [LIMIT row\_count]  
例子2：多表的UPDATE语句：  
UPDATE [LOW\_PRIORITY] [IGNORE] table\_references SET col\_name1=expr1 [, col\_name2=expr2 ...] [WHERE where\_definition]  
UPDATE语法可以用新值更新原有表行中的各列。SET子句指示要修改哪些列和要给予哪些值。WHERE子句指定应更新哪些行。如果没有WHERE子句，则更新所有的行。如果指定了ORDER BY子句，则按照被指定的顺序对行进行更新。LIMIT子句用于给定一个限值，限制可以被更新的行的数目。

**5.8 增加字段**

命令：alter table 表名 add字段 类型 其他;  
例如：在表MyClass中添加了一个字段passtest，类型为int(4)，默认值为0  
mysql> alter table MyClass add passtest int(4) default '0'  
加索引  
mysql> alter table 表名 add index 索引名 (字段名1[，字段名2 …]);  
例子： mysql> alter table employee add index emp\_name (name);  
加主关键字的索引  
mysql> alter table 表名 add primary key (字段名);  
例子： mysql> alter table employee add primary key(id);  
加唯一限制条件的索引  
mysql> alter table 表名 add unique 索引名 (字段名);  
例子： mysql> alter table employee add unique emp\_name2(cardnumber);  
删除某个索引  
mysql> alter table 表名 drop index 索引名;  
例子： mysql>alter table employee drop index emp\_name;  
增加字段：  
mysql> ALTER TABLE table\_name ADD field\_name field\_type;  
修改原字段名称及类型：  
mysql> ALTER TABLE table\_name CHANGE old\_field\_name new\_field\_name field\_type;  
删除字段：  
MySQL ALTER TABLE table\_name DROP field\_name;

**5.9 修改表名**

命令：rename table 原表名 to 新表名;  
例如：在表MyClass名字更改为YouClass  
mysql> rename table MyClass to YouClass;  
当你执行 RENAME 时，你不能有任何锁定的表或活动的事务。你同样也必须有对原初表的 ALTER 和 DROP 权限，以及对新表的 CREATE 和 INSERT 权限。  
如果在多表更名中，MySQL 遭遇到任何错误，它将对所有被更名的表进行倒退更名，将每件事物退回到最初状态。  
RENAME TABLE 在 MySQL 3.23.23 中被加入。

**6、备份数据库**

命令在DOS的[url=file://\\mysql\\bin]\\mysql\\bin[/url]目录下执行  
1.导出整个数据库  
导出文件默认是存在mysql\bin目录下  
mysqldump -u 用户名 -p 数据库名 > 导出的文件名  
mysqldump -u user\_name -p123456 database\_name > outfile\_name.sql  
2.导出一个表  
mysqldump -u 用户名 -p 数据库名 表名> 导出的文件名  
mysqldump -u user\_name -p database\_name table\_name > outfile\_name.sql  
3.导出一个数据库结构  
    mysqldump -u user\_name -p -d –add-drop-table database\_name > outfile\_name.sql  
    -d 没有数据 –add-drop-table 在每个create语句之前增加一个drop table  
4.带语言参数导出  
    mysqldump -uroot -p –default-character-set=latin1 –set-charset=gbk –skip-opt database\_name > outfile\_name.sql  
例如，将aaa库备份到文件back\_aaa中：  
[root@test1 root]# cd　/home/data/mysql  
[root@test1 mysql]# mysqldump -u root -p --opt aaa > back\_aaa

**7.1 一个建库和建表的实例1**

drop database if exists school; //如果存在SCHOOL则删除  
create database school; //建立库SCHOOL  
use school; //打开库SCHOOL  
create table teacher //建立表TEACHER  
(  
    id int(3) auto\_increment not null primary key,  
    name char(10) not null,  
    address varchar(50) default ‘深圳’,  
    year date  
); //建表结束  
//以下为插入字段  
insert into teacher values(”,’allen’,'大连一中’,'1976-10-10′);  
insert into teacher values(”,’jack’,'大连二中’,'1975-12-23′);  
如果你在mysql提示符键入上面的命令也可以，但不方便调试。  
1、你可以将以上命令原样写入一个文本文件中，假设为school.sql，然后复制到c:\\下，并在DOS状态进入目录[url=file://\\mysql\\bin]\\mysql\\bin[/url]，然后键入以下命令：  
 mysql -uroot -p密码 < c:\\school.sql  
如果成功，空出一行无任何显示；如有错误，会有提示。（以上命令已经调试，你只要将//的注释去掉即可使用）。  
2、或者进入命令行后使用 mysql> source c:\\school.sql; 也可以将school.sql文件导入数据库中。

**7.2 一个建库和建表的实例2**

drop database if exists school; //如果存在SCHOOL则删除  
create database school; //建立库SCHOOL  
use school; //打开库SCHOOL  
create table teacher //建立表TEACHER  
(  
id int(3) auto\_increment not null primary key,  
name char(10) not null,  
ddress varchar(50) default ''深圳'',  
ear date  
); //建表结束  
//以下为插入字段  
insert into teacher values('''',''glchengang'',''深圳一中'',''1976-10-10'');  
insert into teacher values('''',''jack'',''深圳一中'',''1975-12-23'');  
注：在建表中  
1、将ID设为长度为3的数字字段:int(3)；并让它每个记录自动加一:auto\_increment；并不能为空:not null；而且让他成为主字段primary key。  
2、将NAME设为长度为10的字符字段  
3、将ADDRESS设为长度50的字符字段，而且缺省值为深圳。  
4、将YEAR设为日期字段。