最新Oracle 和 mysql 的对比参照

|  |  |  |
| --- | --- | --- |
|  | **Oracle** | **mysql** |
| 对比版本 | Release 10.2.0.1.0 XE windowsXP | 5.0.45-community-nt-log MySQL Community Edition (GPL) |
| 当作计算器 | SQL> select 1+1 from dual; | mysql> select 1+1; |
| mysql> select 1+1 from dual; |
| 显示表结构 | SQL> desc 表名 | mysql> desc 表名; |
| SQL> describe 表名 | mysql> describe 表名; |
|  | mysql> explain 表名;  注意：explain 也可以用来获取执行计划。 |
| 需要访问：  User\_tables、user\_tab\_cols、USER\_COL\_COMMENTS、USER\_INDEXES等等字典表。 | mysql> show columns from 表名; |
|  | bin>mysqlshow 库名表名 |
| 日期的概念 | 一个完全不同于字符串的类型。日期类型没有所谓的格式，日期就是日期，只有在显示的时候可以指定转换成的字符串的格式。绝对不会允许2008-00-00这种情况出现。对于日期转换为另一个日期相对比较简单，比如select trunc(sysdate) fromdual; | 可以方便的和字符串进行比较、转换。允许0000-00-00这种情况出现作为null的替代品方便处理（个人不认为这样比较方便，反而给我造成困惑）。对于字符串转换为日期比较简单，比如CREATETABLE`datetime` (   `dt` **datetime**NOTNULLdefault'0000-00-00 00:00:00',   PRIMARYKEY (`dt`)  );  INSERTINTO `datetime` VALUES('2008-01-01 00:00:00');  select\*from `datetime`; |
| 日期类型 | date | datetime |
| 无 | Date  日期 |
| 无 | Time  时间 |
| 无 | Timestamp  如果表中其他字段变化，此类型的字段自动更新为当前系统时间。 |
| Timestamp  保存了毫秒级别的时间 | 不清楚 |
| 日期函数 | SQL> select trunc(sysdate)from dual; | mysql> select curdate();  mysql> select current\_date; |
| SQL> select sysdate from dual; | mysql> select sysdate(); |
| mysql> select now(); |
| SQL> select to\_char(sysdate,'HH24:Mi:SS')from dual;  注意此处返回结果为字符串，而不是日期类型。Oracle不提供和mysql中time类型相应的类型。 | mysql> select current\_time;  mysql> select curtime();  注意：此处返回的是time类型。就是时间17:22:14这种时间。Oracle中没有此类型。 |
| 毫秒级别：  SQL> selectcurrent\_timestampfrom dual; | 需要使用函数MICROSECOND。目前还不会。 |
| 日期格式化 | SQL> select to\_char(sysdate,'yyyy-mm-dd') from dual; | mysql> select date\_format(now(),'%Y-%m-%d'); |
| SQL> select to\_char(sysdate,'hh24-mi-ss') from dual; | mysql> select time\_format(now(),'%H-%i-%S'); |
| 日期函数  （增加一天） | SQL> select sysdate+1from dual;  结果：2008-2-20 19:34:27 | mysql> select date\_add(now(),interval 1 day); |
| mysql>select now()+interval 1 day; |
| 日期函数  （增加一个月） | SQL> select add\_months(sysdate,1)from dual;  结果：2008-3-19 19:34:27 | mysql> select date\_add(now(),interval 1 month); |
| mysql>select now()+interval 1 month; |
| 别名 | SQL> select1as a from dual; | mysql> select**1**as a; |
| SQL> select1 a from dual; | mysql> select**1** a; |
| 字符串截取函数 | SQL> select substr('abcdefg',1,5)from dual;  结果：abcde | mysql> selectsubstr('abcdefg',1,5);  结果：abcde |
| 去除空格：  select trim('abc   ')from dual; | 去除空格：  select trim('abc   ')fromdual; |
| 字符串拼接：  SELECT CONCAT('a',' test')from dual;  结果：a test | SELECT CONCAT('a',' test');  结果：a test |
| 注意oracle中||为字符串拼接  select'a'||' test'from dual;  结果：a test | 注意mysql中||为或操作符。  select 0||1;  结果1；  Select 0||0;  结果0。 |
| 判断是否包含：  select instr('abcd','bcd')from dual;  结果：2 | mysql> SELECT INSTR('foobarbar', 'bar');  结果：4 |
| 另有正则表达式。 | 另有SUBSTRING\_INDEX(str,delim,count)函数 |
| 执行sql脚本 | SQL >@a.sql | 1：mysql> source a.sql |
| 执行外部shell脚本 | SQL>host test.sh | 目前还不会 |
| SQL>! cd .. |
| 改表名 | SQL> altertable T renameto T1; | mysql> alter table t rename t1; |
| mysql> alter table T1 rename to T; |
| 执行命令 | ;<回车> | ;<回车> |
| / | go |
| r | ego |
| run |  |
| distinct用法 | SQL> select distinct 列1 from 表1; | mysql> select distinct 列1 from 表1; |
| SQL> select distinct 列1,列2 from 表1; | mysql> select distinct 列1,列2 from 表1; |
| 注释 | *--* | -- |
| */\* 与\*/* | /\*与\*/ |
| *REM* | # |
| 限制返回记录条数为5条 | SQL> select \* from 表名 where rownum<=5; | mysql> select \* from 表名 limit 5; |
| 分页查询 | select\*  from (  select row\_.\*, rownum rownum\_  from (  yourSqlHere ) row\_  where rownum <= 100)  where rownum\_ > 20; | select\*from **t**limit**20**,**100**; |
| 外连接 | (+) | left join |
| LEFTOUTERJOIN | left outer join |
| RIGHTOUTERJOIN |  |
| 查询索引 | SQL> select index\_name,table\_name from user\_indexes; | mysql> show index from 表名 [FROM 库名]; |
| 通配符 | “%” | “%”和“\_” |
| 如果存在则更新，否则插入 | Merge into | Replace into  参见，mysql的replace没有oracle的merge严格:  [http://www.bitbybit.dk/carsten/blog/?p=128#comment-50048](http://www.bitbybit.dk/carsten/blog/?p=128%23comment-50048) |
| SQL语法 | SELECT selection\_list 选择哪些列 | SELECT selection\_list 选择哪些列 |
| FROM table\_list 从何处选择行 | FROM table\_list 从何处选择行 |
| WHERE primary\_constraint 行必须满足什么条件 | WHERE primary\_constraint 行必须满足什么条件 |
| GROUP BY grouping\_columns 怎样对结果分组  注意：oracle如果需要排序必须指定order by 子句。 | GROUP BY grouping\_columns 怎样对结果分组  注意：即使不存在order by子句，mysql也会按照group by 的列进行排序。甚至还能为group by 语句指定asc/desc子句。 |
| HAVING secondary\_constraint 行必须满足的第二条件 | HAVING secondary\_constraint 行必须满足的第二条件 |
| ORDER BY sorting\_columns 怎样对结果排序 | ORDER BY sorting\_columns 怎样对结果排序 |
| Oracle的结果集限定语句见：”分页查询”  注意：oracle的rownum和mysql的Limit完全不同。 | LIMIT count 结果限定 |
| 对not in null的处理，是关于三值逻辑的问题。 | CREATETABLE a  ( code INT);  INSERTINTO a       VALUES(2);  INSERTINTO a       VALUES(NULL);  SELECT1   FROM DUAL   WHERE1NOTIN(SELECT code FROM a);  结果：无返回结果。 | CREATETABLE a  ( code INT);  INSERTINTO a       VALUES(2);  INSERTINTO a       VALUES(NULL);  SELECT1   FROM DUAL   WHERE1NOTIN(SELECTcode FROM a);  结果：无返回结果 |
| 对not exists的处理 | CREATETABLE a  ( code INT);  INSERTINTO a       VALUES(2);  INSERTINTO a       VALUES(NULL);  SELECT1   FROM DUAL   WHERENOTEXISTS(SELECT1                       FROM a WHERE a.code =1);  结果：1 | CREATETABLE a  ( code INT);  INSERTINTO a       VALUES(2);  INSERTINTO a       VALUES(NULL);  SELECT1   FROM DUAL   WHERENOTEXISTS(SELECT1                       FROM aWHERE a.code =1);  结果：1 |
| 获取sql语句执行计划 | SQL>EXPLAIN PLAN FOR   select \* from t;  然后  SQL>select \* from table(DBMS\_XPLAN.DISPLAY) | Mysql> explainselect\*from t; |
| 执行计划对照 | 全表扫描: TABLE ACCESS FULL | Type=all |
| Where条件过滤:FILTER | Extra=Using where |
| 排序：SORT ORDER BY | Extra=Using filesort |
|  |  |
|  |  |
| 事务管理 | 默认不自动提交 | innoDB支持事务 |
| 默认自动提交autocommit |
| 阻塞读取方式 |  |  |
| Unique index的区别。  又是关于三值逻辑的问题。 | CREATETABLE hr.t (     IDINTNOTNULL,     DATACHAR(30)DEFAULTNULL,     UNIQUE(ID,DATA)   );    INSERTINTO hr.t       VALUES(1,NULL);    INSERTINTO hr.t       VALUES(1,NULL); | mysql> CREATE TABLE t (     ID INT NOT NULL,     DATA CHAR(30) DEFAULT NULL,     UNIQUE (ID, DATA)   );  INSERT INTO t       VALUES (1, NULL);  INSERT INTO t       VALUES (1, NULL);  select \* from t;  Query OK, 0 rows affected  Query OK, 1 row affected  Query OK, 1 row affected  +----+------+  | ID | DATA |  +----+------+  | 1 | NULL |  | 1 | NULL |  +----+------+  2 rows in set |
| ORA-00001: 违反唯一约束条件 (HR.SYS\_C003999) |