

Oracle基本数据类型存储格式浅析日期类型(1) PDF转换可能丢失图片或格式，建议阅读原文

https://www.100test.com/kao_ti2020/292/2021_2022_Oracle_E5_9F_BA_E6_c67_292231.htm Oracle基本数据类型存储格式浅析 (三) 日期类型 下面通过一个例子进行说明。 SQL> create table test_date (date_col date). 表已创建。 SQL> insert into test_date values (to_date(' ' 2000-1-1 0:0:0 ' ' , ' ' yyyy-mm-dd hh24:mi:ss ' ')). 已创建 1 行。 SQL> insert into test_date values (to_date(' ' 1-1-1 0:0:0 ' ' , ' ' yyyy-mm-dd hh24:mi:ss ' ')). 已创建 1 行。 SQL> insert into test_date values (to_date(' ' -1-1-1 0:0:0 ' ' , ' ' syyyy-mm-dd hh24:mi:ss ' ')). 已创建 1 行。 SQL> insert into test_date values (to_date(' ' -101-1-1 0:0:0 ' ' , ' ' syyyy-mm-dd hh24:mi:ss ' ')). 已创建 1 行。 SQL> insert into test_date values (to_date(' ' -4712-1-1 0:0:0 ' ' , ' ' syyyy-mm-dd hh24:mi:ss ' ')). 已创建 1 行。 SQL> insert into test_date values (to_date(' ' 9999-12-31 23:59:59 ' ' , ' ' syyyy-mm-dd hh24:mi:ss ' ')). 已创建 1 行。 SQL> insert into test_date values (sysdate). 已创建 1 行。 SQL> insert into test_date values (to_date(' ' -4713-1-1 0:0:0 ' ' , ' ' syyyy-mm-dd hh24:mi:ss ' ')). insert into test_date values (to_date(' ' -4713-1-1 0:0:0 ' ' , ' ' syyyy-mm-dd hh24:mi:ss ' ')) * ERROR 位于第 1 行: ORA-01841: (全)年度值必须介于 -4713 和 9999 之间，且不为 0 SQL> insert into test_date values (to_date(' ' 0000-1-1 0:0:0 ' ' , ' ' yyyy-mm-dd hh24:mi:ss ' ')). insert into test_date values (to_date(' ' 0000-1-1 0:0:0 ' ' , ' ' yyyy-mm-dd hh24:mi:ss ' ')) * ERROR 位于第 1 行:

ORA-01841: (全)年度值必须介于 -4713 和 9999 之间, 且不为 0
SQL> col dump_date format a80 SQL> 0select
to_char(date_col, ' ' syyyy-mm-dd hh24:mi:ss ' '),
dump(date_col) dump_date from test_date.

TO_CHAR(DATE_COL, ' ' SY DUMP_DATE

2000-01-01 00:00:00 Typ=12 Len=7: 120,100,1,1,1,1,1 0001-01-01
00:00:00 Typ=12 Len=7: 100,101,1,1,1,1,1 -0001-01-01 00:00:00
Typ=12 Len=7: 100,99,1,1,1,1,1 -0101-01-01 00:00:00 Typ=12
Len=7: 99,99,1,1,1,1,1 -4712-01-01 00:00:00 Typ=12 Len=7:
53,88,1,1,1,1,1 9999-12-31 23:59:59 Typ=12 Len=7:
199,199,12,31,24,60,60 2004-12-15 13:56:19 Typ=12 Len=7:

120,104,12,15,14,57,20 已选择7行。通过最后两条语句已经可以看出Oracle的DATE类型的取值范围是公元前4712年1月1日至公元9999年12月31日。而且根据日期的特定, 要不然是在公元1年, 要不然是在公元前1年, 不会出现0年的情况。日期类型长度是7, 7个字节分别表示世纪、年、月、日、时、分和秒。由于不会出现0的情况, 月和日都是按照原值存储的, 月的范围是1~12, 日的范围是1~31。由于时、分、秒都会出现0的情况, 因此存储时采用原值加1的方式。0时保存为1, 13时保存为14, 23时保存为24。分和秒的情况与小时类似。小时的范围是0~23, 在数据库中以1~24保存。分和秒的范围都是0~59, 在数据库中以1~60保存。年和世纪的情况相对比较复杂, 可分为公元前和公元后两种情况。由于最小的世纪的值是-47(公元前4712年), 最大值是99(公元9999年)。为了避免负数的产生, oracle把世纪加100保存在数据

库中。公元2000年，世纪保存为120，公元9999年，世纪保存为199，公元前101年，世纪保存为99（100（-1）），公元前4712年，世纪保存为53（100（-47））。注意，对于公元前1年，虽然已经是公元前了，但是表示世纪的前两位的值仍然是0，因此，这时的保存的世纪的值仍然是100。世纪的范围是-47 ~ 99，保存的值是53 ~ 199。年的保存与世纪的保存方式类似，也把年的值加上100进行保存。对于公元2000年，年保持为100，公元1年保存为101，公元2004年保存为104，公元9999年保存为199，公元前1年，保存为99（100（-1）），公元前101年，保存为99（100（-1）），公元前4712年保存为88（100（-12））。对于公元前的年，保存的值总是小于等于100，对于公元后的年，保存的值总是大于等于100。年的范围是0 ~ 99，保存的值是1 ~ 199。注意：一般的世纪，都包含了100年，而对于0世纪，由于包含公元前和公元后两部分且不包含0年，因此包含了198年。100Test 下载频道开通，各类考试题目直接下载。详细请访问 www.100test.com