

Oracle中生成GUID类型Oracle认证考试 PDF转换可能丢失图片或格式，建议阅读原文

[https://www.100test.com/kao\\_ti2020/644/2021\\_2022\\_Oracle\\_E4\\_B8\\_AD\\_E7\\_c102\\_644878.htm](https://www.100test.com/kao_ti2020/644/2021_2022_Oracle_E4_B8_AD_E7_c102_644878.htm) "mkhghigh"> Oracle8i引入

了SYS\_GUID这个概念，它同Oracle管理员所使用的传统的序列（sequence）相比具有诸多优势。一个序列生成器只是简单地创建从给定的起点开始的一系列整数值，而且它被用在选择陈述式的时候自动地递增该系列。序列生成器所生成的数字只能保证在单个实例里是唯一的，这就不适合将它用作并行或者远程环境里的主关键字，因为各自环境里的序列可能会生成相同的数字，从而导致冲突的发生。SYS\_GUID会保证它创建的标识符在每个数据库里都是唯一的。此外，序列必须是DML陈述式的一部分，因此它需要一个到数据库的往返过程（否则它就不能保证其值是唯一的）。SYS\_GUID源自不需要对数据库进行访问的时间戳和机器标识符，这就节省了查询的消耗。

```
create table use_seq_table(id integer). create
sequence use_seq_sequence. insert into use_seq_table values
(use_seq_sequence_value.nextval). REM - for some reason, the
documentation uses raw(32) create table use_guid_table(id
raw(16)). insert into use_guid_table(sys_guid()).
```

很多应用程序都依靠序列生成器来创建数据行的主关键字，这些数据行没有一个明显的主值，这也就是说，在这样的数据集里一条记录的创建就会让数据列发生改变。因此，管理员可能会对在表格中将SYS\_GUID用作主关键字而不使用序列数感兴趣。这在对象在不同机器的不同数据库里生成以及需要在后来合并到一起的情况下很有用。但是，SYS\_GUID所生成的值是一

个16位的原始值。序列所生成的整数不会使用16位（的值），除非它达到了10的30次方（每个字节有两位），而且数字是相当独特的：SQL> select

```
dump(123456789012345678901234567890) from dual.
```

```
DUMP(123456789012345678901234567890) Typ=2 Len=16:
```

207,13,35,57,79,91,13,35,57,79,91,13,35,57,79,91 较短的值就意味着用于表格和索引的存储空间更少，以及查询访问的时间更短。使用SYS\_GUID或者序列会在数据库使用周期里的某些地方造成性能上的消耗；问题就是在哪里。对于SYS\_GUID而言，性能上的影响在查询时间和创建时间上（在表格里要创建更多的块和索引以容纳数据）。对序列而言，性能上的影响在查询期间，在这个时候，SGA序列的缓冲区被用光。在缺省情况下，一个序列一次会缓冲20个值。如果数据库没有使用这些值就关闭了，它们就会被丢失。SYS\_GUID生成的值的另一个显著的不足之处是，管理这些值会变得困难得多。你必须（手动）输入它们或者通过脚本来填充它们，或者将它们作为Web参数来传递。出于这些原因，将SYS\_GUID作为一个主关键字不是一个很好主意，除非是在一个并行的环境里或者希望避免使用管理序列生成器的情况下。在这里，我们不讨论 guid和序列器的优势，每个都会有特定环境下的需求。在Oracle9i和Oracle 10g 里SYS\_GUID产生得到的数据是32 位的，如：234E45F0077881AAE0430AA3034681AA 我这里要做的功能是将guid分割成为 Windows系统 ObjectId 的格式：{8-4-4-4-12}，下面给出两种解决方法：方法一：利用substr进行分割，非常简单，具体代码如下：/\*\* \* 创建系统Object Id 格式的字符串，返回的结果如下

: {234E45F0-077A-81AA-E043-0AA3034681AA} \*/ function  
CreateGUID return varchar2 is guid varchar(64). begin guid :=  
SYS\_GUID(). return {||substr(guid,1,8)||-||substr(guid,9,4)||  
-||substr(guid,13,4)||-||substr(guid,17,4) ||-||substr(guid,21,12)||}.  
end CreateGUID. 方法二：利用CONCAT函数进行连接，具体  
代码如下： CREATE OR REPLACE FUNCTION GET\_GUID  
RETURN CHAR IS v\_guid CHAR (36). v\_guid\_part\_one CHAR  
(8). v\_guid\_part\_two CHAR (4). v\_guid\_part\_three CHAR (4).  
v\_guid\_part\_four CHAR (4). v\_guid\_part\_five CHAR (12). BEGIN  
SELECT SYS\_GUID () INTO v\_guid FROM DUAL.  
v\_guid\_part\_one := SUBSTR (v\_guid, 0, 8). v\_guid\_part\_two :=  
SUBSTR (v\_guid, 8, 4). v\_guid\_part\_three := SUBSTR (v\_guid, 12,  
4). v\_guid\_part\_four := SUBSTR (v\_guid, 16, 4). v\_guid\_part\_five  
:= SUBSTR (v\_guid, 20, 12). v\_guid := CONCAT (CONCAT  
(CONCAT (CONCAT (CONCAT (CONCAT  
(CONCAT (v\_guid\_part\_one,-), v\_guid\_part\_two ), - ),  
v\_guid\_part\_three ), - ), v\_guid\_part\_four ), - ), v\_guid\_part\_five ).  
RETURN (v\_guid). END GET\_GUID. 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](http://www.100test.com)