Delete all contents in a schema in Oracle

Ask Question

20

9

Is possible to delete all contents in scheme in Oracle? I found this script:

```
Begin
for c in (select table_name from user_tables) loop
execute immediate ('drop table '||c.table_name||' cascade constraints);
end loop;
End;
```

But I would like to know if are there anything to drop everything in the schema, indexes,tables,contraints... but not the schema (drop user ...).

Thanks.

sql oracle plsql

edited Dec 14 '16 at 14:04

nop77svk

3,243 2 13 29

asked Apr 28 '15 at 17:38

zerosssa

148 1 1 8

If you mean don't delete the Oracle user ids, I think that script would do it. But create a test database, and try it. – CargoMeister Apr 28 '15 at 17:41

What is your goal? To clear content of all tables? Then you should not drop them. If you want just to get an empty schema, then I suppose it's easier to recreate it. You'll get a new schema from scratch with empty statistics, without any objects (tables, views, synonyms, materialized views, etc) – denied Apr 28 '15 at 18:21

2 I dont have permissions to create users so I cant drop the user and recreate it (cant use DROP USER as is commented in the question) – zerosssa Apr 28 '15 at 21:01

If you really need to drop *every* object type, you'll have to dig through the SQL Language Reference and handle most of the `DROP ...` statements. – Jon Heller Apr 30 '15 at 5:32

1 It's a shame Oracle provide a `create schema` command but not a `drop schema`. I imagine this is a pretty common requirement for automated build testing, where typically you can't create users from scratch. – William Robertson Oct 10 '17 at 17:31

5 Answers

34

Normally, it is simplest to drop and add the user. This is the preferred method if you have system or sysdba access to the database.

If you don't have system level access, and want to scrub your schema, the following sql will produce a series of drop statments, which can then be executed.

```
select 'drop '||object_type||' '|| object_name|| DECODE(OBJECT_TYPE,
CONSTRAINTS','') || ';'  from user_objects
```

Then, I normally purge the recycle bin to really clean things up. To be honest, I don't see a lot of use for oracle's recycle bin, and wish i could disable it, but anyway:

```
purge recyclebin;
```

This will produce a list of drop statements. Not all of them will execute - if you drop with cascade, dropping the PK_* indices will fail. But in the end, you will have a pretty clean schema. Confirm with:

```
select * from user_objects
```

Also, just to add, the Pl/sql block in your question will delete only tables, it doesn't delete all other objects.

ps: Copied from some website, was useful to me. Tested and working like a charm.

edited Oct 10 '17 at 13:12
Betlista
7,013 7 50 82

answered Apr 30 '15 at 3:24
venki
831 1 6 17

Edit to put semicolon on end of each drop: select 'drop '||object_type||' '|| object_name|| DECODE(OBJECT_TYPE,'TABLE',' CASCADE CONSTRAINTS','') || ';' from user_objects – rr789 Aug 28 '16 at 1:28

this answer is incomplete for more sophisticated schemas – Pancho Oct 16 '16 at 20:28

2

Yes you can. You can drop the user and thus drop the schema objects. The DROP USER statement is used to remove a user from the Oracle database and remove all objects owned by that user.

Home

PUBLIC

Stack Overflow

Tags

Users

Jobs

```
DROP USER TestDB;
```

This statement will only run properly and drop the user called TestDB only if TestDB does not own any objects in its schema. Object in the sense tables and views etc. If it contains any objects then after executing the DROP USER statement you will get the below error message

```
Error starting at line : 1 in command -
DROP USER TestDB
Error report -
SQL Error: ORA-01922: CASCADE must be specified to drop 'TESTDB'
01922. 00000 -  "CASCADE must be specified to drop '%s'"
*Cause:    Cascade is required to remove this user from the system.
           user own's object which will need to be dropped.
*Action:   Specify cascade.
```

If TestDB did own objects in its schema, you would need to run the following DROP USER statement instead:

```
DROP USER TestDB CASCADE;
```

This statement will drop all objects owned by TestDB, and all referential integrity constraints on TestDB objects would also be dropped.

answered Apr 28 '15 at 18:19
UUIUI
6,073 9 35 54

8 This is **explicitly** what the OP did **not** ask for. – zb226 Apr 26 '16 at 17:22

Just go with:

0

```
select 'drop '||object_type||' '|| object_name || ';' from user_obje
object_type in ('VIEW','PACKAGE','SEQUENCE', 'PROCEDURE', 'FUNCTION'
```

answered Jan 30 at 8:21

Tomok
3 2

This is what I used:

0

```
set echo off feedback off serverout on

spool drop_all_objects.sql

declare  l_object varchar2(32000);

begin

  for i in (select object_name, object_type from dba_objects where o

    if i.object_type='JOB' then

          l_object := 'begin dbms_scheduler.drop_job (job_name =>
''<owner>'||i.object_name||'''); end;';

elsif i.object_type='PROGRAM' then

      l_object := 'begin dbms_scheduler.drop_program (program_name =
''<owner>'||i.object_name||'''); end;';

elsif i.object_type='RULE' then

      l_object := 'begin dbms_rule_adm.drop_rule (rule_name =>
''<owner>'||i.object_name||''', force => TRUE); end;';

elsif i.object_type='RULE SET' then

      l_object := 'begin dbms_rule_adm.drop_rule_set (rule_set_name
''<owner>'||i.object_name||''', delete_rules => TRUE); end;';
```

```
elsif i.object_type='CHAIN' then

      l_object := 'begin dbms_scheduler.drop_chain (chain_name =>
''<owner>'||i.object_name||''', force => TRUE); end;';

elsif i.object_type='RULE' then

      l_object := 'begin dbms_rule_adm.drop_evaluation_context (eval
=> ''<owner>'||i.object_name||''', force => TRUE); end;';

else

          l_object := 'drop '||i.object_type||'<owner>'||i.object_na

end if;

dbms_output.put_line(i_object);

dbms_output.put_line('/');

end loop;

end;

/

@drop_all_objects
```

edited Nov 9 '16 at 13:52

captainsac
2,199 2 17 37

answered Nov 9 '16 at 12:48

John Taylor
1

The following SQLplus script generates the SQL statements needed to delete all schema objects from the desired user:

0

```
set heading off
set pagesize 0
set feedback off

-- wipe out all scheduler jobs
select 'exec dbms_scheduler.drop_job(job_name =>
'''||j.job_creator||'.'||j.job_name||''');'
from user_scheduler_jobs j
/

-- wipe out all XML schemas
select 'exec dbms_xmlschema.deleteSchema(schemaURL =>
'''||s.qual_schema_url||''',delete_option => dbms_xmlschema.DELETE_C
from user_xml_schemas s
/

-- wipe out all remaining objects
select 'drop '
       ||o.object_type
       ||' '||object_name
       ||case o.object_type when 'TABLE' then ' cascade constraints'
force' else '' end
       ||';'
from user_objects o
where o.object_type not in ('JOB','LOB','PACKAGE BODY','INDEX','TRIG
and not exists (select 1
                from user_objects r
                where r.object_name = o.object_name
                and   r.object_type = 'MATERIALIZED VIEW'
                and   o.object_type != 'MATERIALIZED VIEW'
               )
/

-- empty the recycle bin
select 'purge recyclebin;' from dual
/
```

The script works 100% for me as is - but if for some reason it is not complete for you then it is easily enhanced using a virtual machine (VM) as follows:

1. log on as [your schema user to empty]
2. Take a snapshot of your VM

3. run the above script to create the deletion statements
4. run the deletion statements *(you can ignore any "object does not exist" errors, as some objects will be automatically removed prior to the script removal statement. This occurs as a result of owning objects being removed)*
5. log off
6. log on as SYS and execute "drop user [your schema user to empty];" -- WITHOUT the cascade option

If step 6 fails then you need to identify the remaining objects preventing your user from being deleted and add them to the above script. Repeat until your user drops (ie. your script is comprehensive) then save your script

Roll back your VM to your snapshot and repeat steps 3 and 4 (using your updated script) - and you should now have a 100% empty schema.

edited Oct 16 '16 at 20:25

answered Oct 16 '16 at 19:18

Pancho
875 9 27