

# Oracle DROP TABLESPACE Statement

**Summary:** In this lab, you will learn how to remove a tablespace from the database by using the Oracle `DROP TABLESPACE` statement.

## Oracle DROP TABLESPACE statement

The `DROP TABLESPACE` allows you to remove a `tablespace` from the database. Here is the basic syntax of the `DROP TABLESPACE` statement:

```
DROP TABLESPACE tablespace_name
    [INCLUDING CONTENTS [AND | KEEP] DATAFILES]
    [CASCADE CONSTRAINTS];
```

In this syntax:

- First, specify the name of the tablespace that you want to drop after the `DROP TABLESPACE` keywords.
- Second, use the `INCLUDING CONTENTS` to delete all contents of the tablespace. If the tablespace has any objects, you must use this option to remove the tablespace. Any attempt to remove a tablespace that has objects without specifying the `INCLUDING CONTENTS` option will result in an error.
- Third, use `AND DATAFILES` option to instruct Oracle to delete the datafiles of the tablespace and `KEEP DATAFILES` option to leave the datafiles untouched.
- Fourth, if the tablespace has objects such as tables whose primary keys are referenced by [referential integrity] constraints from tables outside the tablespace, you must use the `CASCADE CONSTRAINTS` option to drop these constraints. If you omit the `CASCADE CONSTRAINTS` clause in such situations, Oracle returns an error and does not remove the tablespace.

You can use the `DROP TABLESPACE` to remove a tablespace regardless of whether it is online or offline. However, it's good practice to take the tablespace offline before removing it to ensure that no sessions are currently accessing any objects in the tablespace.

Note that you cannot drop the `SYSTEM` tablespace and only can drop the `SYSAUX` tablespace when you start the database in the `MIGRATE` mode.

You need to have the `DROP TABLESPACE` system privilege to execute the `DROP TABLESPACE` statement. To drop the `SYSAUX` tablespace, you need to have the `SYSDBA` system privilege.

## Oracle DROP TABLESPACE statement examples

Let's take some examples of using the `DROP TABLESPACE` statement.

### 1) Using Oracle DROP TABLESPACE to remove an empty tablespace example

First, create a new tablespace named `tbs1` if not exists already:

```
CREATE TABLESPACE tbs1dev
    DATAFILE 'tbs1dev_data.dbf'
    SIZE 10m;
```

Second, use the `DROP TABLESPACE` to remove the `tbs1` tablespace:

```
DROP TABLESPACE tbs1dev;
```

## 2) Using Oracle DROP TABLESPACE to remove a non-empty tablespace example

First, create a new tablespace named `tbs2` :

```
CREATE TABLESPACE tbs2
  DATAFILE 'tbs2_data.dbf'
  SIZE 5m;
```

Second, create a new table `t2` in the tablespace `tbs2` :

```
CREATE TABLE t2 (
  c1 INT
) TABLESPACE tbs2;
```

Third, use the `DROP TABLESPACE` statement to drop the `tbs2` tablespace:

```
DROP TABLESPACE tbs2;
```

Oracle issued the following error:

```
ORA-01549: tablespace not empty, use `INCLUDING CONTENTS` option
```

To drop the `tbs2` tablespace, we need to use the `INCLUDING CONTENTS` option:

```
DROP TABLESPACE tbs2
  INCLUDING CONTENTS;
```

Oracle issued the following message indicating that the tablespace has been dropped:

```
Tablespace dropped.
```

## 3) Using Oracle DROP TABLESPACE to remove a tablespace whose tables are referenced by referential constraints

First, create two tablespaces named `tbs3` and `tbs4` :

```
CREATE TABLESPACE tbs3
  DATAFILE 'tbs3_data.dbf'
  SIZE 5m;

CREATE TABLESPACE tbs4
  DATAFILE 'tbs4_data.dbf'
  SIZE 5m;
```

Next, create a new table in the `tbs3` tablespace:

```
CREATE TABLE t3 (
  c1 INT PRIMARY KEY
) TABLESPACE tbs3;
```

Then, create a new table in the `tbs4` tablespace:

```
CREATE TABLE t4 (  
    c1 INT PRIMARY KEY,  
    c2 INT NOT NULL,  
    FOREIGN KEY (c2) REFERENCES t3 (c1)  
) TABLESPACE tbs4;
```

After that, drop the tablespace `tbs3` :

```
DROP TABLESPACE tbs3  
INCLUDING CONTENTS;
```

Oracle issued the following error:

```
ORA-02449: unique/primary keys in table referenced by foreign keys
```

Finally, use the `DROP TABLESPACE` that includes the `CASCADE CONSTRAINTS` option to drop the tablespace:

```
DROP TABLESPACE tbs3  
INCLUDING CONTENTS AND DATAFILES  
CASCADE CONSTRAINTS;
```

It worked as expected.

**Important:** Run following command to drop tablespace `tbs1` :

```
DROP TABLESPACE tbs1  
INCLUDING CONTENTS AND DATAFILES  
CASCADE CONSTRAINTS;
```

In this lab, you have learned how to use the Oracle `DROP TABLESPACE` statement to remove a tablespace from the database.