



Database Startup and Shutdown

Starting the Database

tbboot - Starts the Tibero instance.

Syntax:

```
tbboot [options]
```

Example:

```
tbboot (Starts the database with default parameters)
```

tbboot -d - Starts a specific database.

Example:

```
tbboot -d tbdb (Starts the database named 'tbdb')
```

Tips:

- Ensure the environment variables like TB_HOME and TB_SID are correctly set before starting the database.
- Check the tbboot log file for any startup errors.

Shutting Down the Database

tbdown - Shuts down the Tibero instance.

Syntax:

```
tbdown [options]
```

Example:

```
tbdown (Shuts down the database with default parameters)
```

tbdown normal - Performs a normal shutdown, waiting for all sessions to disconnect.

Example:

```
tbdown normal
```

tbdown immediate - Performs an immediate shutdown, rolling back active transactions.

Example:

```
tbdown immediate
```

tbdown abort - Performs an abort shutdown, without any rollback or cleanup (use with caution).

Example:

```
tbdown abort
```

Tips:

- Always prefer `tbdown normal` for a clean shutdown.
- Use `tbdown immediate` if a faster shutdown is needed and you can tolerate transaction rollback.
- Avoid `tbdown abort` unless absolutely necessary, as it can lead to database corruption.

Common Errors and Solutions

Error: TBM-00001: Unable to connect to the database.

Solution: Verify that the database instance is running and the TB_SID environment variable is correctly set.

Error: TBM-00002: Insufficient privileges.

Solution: Ensure you are connecting with a user that has the necessary privileges to start or stop the database.

User and Schema Management

Creating Users

CREATE USER - Creates a new database user.

Syntax:

```
CREATE USER <username> IDENTIFIED BY <password>;
```

Example:

```
CREATE USER testuser IDENTIFIED BY password123;
```

Tips:

- Always choose strong passwords.
- Consider using password profiles to enforce password policies.

Granting Privileges

GRANT - Grants privileges to users.

Syntax:

```
GRANT <privilege> TO <username>;
```

Example:

```
GRANT CONNECT, RESOURCE TO testuser; (Grants connect and resource roles to testuser)
```

GRANT SELECT ON <table_name> TO ; - Grants SELECT privilege on a specific table.

Example:

```
GRANT SELECT ON employees TO testuser;
```

Revoking Privileges

REVOKE - Revokes privileges from users.

Syntax:

```
REVOKE <privilege> FROM <username>;
```

Example:

```
REVOKE CONNECT FROM testuser;
```

Dropping Users

DROP USER - Drops a database user.

Syntax:

```
DROP USER <username>;
```

Example:

```
DROP USER testuser;
```

DROP USER CASCADE; - Drops a user and all objects owned by the user.

Example:

```
DROP USER testuser CASCADE;
```

Caution: Using CASCADE will drop all objects owned by the user. Make sure this is the desired outcome before executing the command.

Tablespace and Storage Management

Creating Tablespaces

CREATE TABLESPACE - Creates a new tablespace.

Syntax:
`CREATE TABLESPACE <tablespace_name>
DATAFILE '<file_path>' SIZE <size>
[AUTOEXTEND ON NEXT <size> MAXSIZE
<size>];`

Example:
`CREATE TABLESPACE tbs_data DATAFILE
'/tiberero/data/tbs_data01.dtf' SIZE 100M
AUTOEXTEND ON NEXT 50M MAXSIZE 500M;`

Tips:

- Plan the size of your tablespaces based on expected data growth.
- Use AUTOEXTEND to automatically increase tablespace size as needed.

Altering Tablespaces

ALTER TABLESPACE - Alters an existing tablespace.

Syntax:
`ALTER TABLESPACE <tablespace_name> ADD
DATAFILE '<file_path>' SIZE <size>
AUTOEXTEND ON NEXT <size> MAXSIZE <size>;`

Example:
`ALTER TABLESPACE tbs_data ADD DATAFILE
'/tiberero/data/tbs_data02.dtf' SIZE 100M
AUTOEXTEND ON NEXT 50M MAXSIZE 500M;`

**ALTER TABLESPACE <tablespace_name>
RESIZE DATAFILE '<file_path>' SIZE ;** - Resizes an existing datafile.

Example:
`ALTER TABLESPACE tbs_data RESIZE DATAFILE
'/tiberero/data/tbs_data01.dtf' SIZE 200M;`

Dropping Tablespaces

DROP TABLESPACE - Drops a tablespace.

Syntax:
`DROP TABLESPACE <tablespace_name>
[INCLUDING CONTENTS AND DATAFILES];`

Example:
`DROP TABLESPACE tbs_data INCLUDING
CONTENTS AND DATAFILES;`

Caution: INCLUDING CONTENTS AND DATAFILES will delete all data and datafiles in the tablespace. Use with care.

Tablespace Usage

```
SELECT tablespace_name, SUM(bytes) / 1024  
/ 1024 AS size_mb  
FROM dba_data_files  
GROUP BY tablespace_name;
```

Shows the tablespace and their sizes.

SQL and PL/SQL Execution

Executing SQL Statements

tbsql - Executes SQL statements interactively.

Syntax:
`tbsql
<username>/<password>@<database_name>`

Example:
`tbsql sys/tiberero@tbdb`

@<file_path> - Executes SQL statements from a file.

Example:
`@/home/tiberero/sql/create_table.sql`

Tips:

- Use `tbsql` for interactive querying and administration.
- Use SQL files for executing complex or repetitive SQL tasks.

Executing PL/SQL Blocks

PL/SQL Blocks - Can be executed directly within `tbsql`.

Example:

```
DECLARE  
    v_count NUMBER;  
BEGIN  
    SELECT COUNT(*) INTO v_count FROM  
employees;  
    DBMS_OUTPUT.PUT_LINE('Number of  
employees: ' || v_count);  
END;  
/
```

Common Commands in tbsql

SHOW ERRORS - Shows errors for the last compiled PL/SQL object.

Example:
`SHOW ERRORS;`

SET SERVEROUTPUT ON - Enables output from DBMS_OUTPUT.

Example:
`SET SERVEROUTPUT ON;`

DESCRIBE <table_name> - Describes the structure of a table.

Example:
`DESCRIBE employees;`