

Data Dictionary Catalogue Tables

Querying to learn about your indexes, procedures, packages, etc.

This catalogue describes how to use SQL data dictionary tables in Oracle SQLPLUS like the `user_views`, `user_indexes`, `user_constraints`, `user_procedures` and `user_Objects` (to find the functions or any other DB objects).

1. User_Views

Column name	Uniqueness	Data type
VIEW_NAME	NOT NULL	VARCHAR2(30)
TEXT_LENGTH		NUMBER
TEXT		LONG
TYPE_TEXT_LENGTH		NUMBER
TYPE_TEXT		VARCHAR2(4000)

Example:

The following example explains how to retrieve the all views that are owned by the current user who is logged on to SQLPLUS.

```
Select View_Name
From User_Views;
```

2. User_Indexes

Column name	Uniqueness	Data type
INDEX_NAME	NOT NULL	VARCHAR2(30)
INDEX_TYPE		VARCHAR2(27)
TABLE_OWNER	NOT NULL	VARCHAR2(30)
TABLE_NAME	NOT NULL	VARCHAR2(30)
TABLE_TYPE		VARCHAR2(11)
UNIQUENESS		VARCHAR2(9)

Example:

The following example explains how to retrieve all the indexes that are owned by the current user who is logged on to SQLPLUS, the table name and table owner of the table that the index is built on;

```
Select Index_Name, Table_owner, Table_Name
From User_Indexes;
```

3. User_Constraints

Column name	Uniqueness	Data type	De
OWNER	NOT NULL	VARCHAR2(30)	Owner of the constraint definition
CONSTRAINT_NAME	NOT NULL	VARCHAR2(30)	Name of the constraint definition
CONSTRAINT_TYPE		VARCHAR2(1)	Type of constraint definition:

			C (check constraint on a table) P (primary key) U (unique key) R (referential integrity) V (with check option, on a view) O (with read only, on a view)
TABLE_NAME	NOT NULL	VARCHAR2(30)	Name associated with the table (or view) with constraint definition
STATUS		VARCHAR2(8)	Enforcement status of constraint (ENABLED or DISABLED)

Example:

The following example explains how to retrieve all the Constraints that are owned by the current user who is logged on to SQLPLUS, Constraint type, table name that the constraint is built on and the status of each constraint.

```
Select Constraint_Name, Constraint_Type, Table_Name ,Status
From User_Constraints;
```

4. User_Procedures

Column name	Uniqueness	Data type	Description
OWNER	NOT NULL	VARCHAR2(30)	Owner of the procedure
PROCEDURE_NAME	NOT NULL	VARCHAR2(30)	Name of the procedure

Example:

The following example explains how to retrieve all the Procedure names that are owned by the current user who is logged on to SQLPLUS

```
Select Procedure_Name
From User_Procedures;
```

5. User_Objects

Column name	Uniqueness	Data type	Description
OWNER	NOT NULL	VARCHAR2(30)	Owner of the object
OBJECT_NAME	NOT NULL	VARCHAR2(128)	Name of the object
OBJECT_TYPE		VARCHAR2(18)	Type of the object (such as TABLE, INDEX, FUNCTION, PROCEDURE)

We can use the User_Objects table to retrieve some information about any kind of Database objects like table, indexes, functions, packages ..etc

Example:

The following example explains how to retrieve all the Function names that are owned by the current user who is logged on to SQLPLUS

```
Select OBJECT_NAME
From User_Objects
Where Object_Type = 'FUNCTION ';
```

6. User_source	
Name	Type
-----	-----
NAME	VARCHAR2(30)
TYPE	VARCHAR2(12)
LINE	NUMBER
TEXT	VARCHAR2(4000)

We can use the User_source table to retrieve some information on the source code text of Database objects like procedures, functions, packages ..etc

Eg., to get the source code of procedure DISCOUNT, use:

```
SELECT TEXT FROM USER_SOURCE WHERE NAME LIKE '%DISCOUNT%';
```

7. Information About Errors

1. DESC USER_ERRORS
[Used to determine details of a compilation error]
2. SHO ERR
[displays the line number the error occurred in USER_SOURCE view]
3. DESC <packagename>
[for describing details of a package].
4. From tables User_objects, User_procedures, etc., you can check the status of a procedure, function or package.

If the status of any of these objects is valid, it means that it has been compiled without error and is ready to be called by any block. However, if the status is invalid, it needs to be re-compiled.

5.To re-compile a procedure, function or package, use:

```
Alter Procedure Procedure_name Compile;
```

Or

```
Alter Function Function_name Compile;
```

Or

```
Alter Package Package_name Compile;
```

6. To drop any of these objects in SQLPLUS, use statement like:

```
DROP PROCEDURE Procedure_name;
```

7. You can also execute a compiled procedure or package in SQLPLUS with the instruction:

```
EXECUTE Procedure_name;
```

Or

```
EXECUTE Package_name;
```

References:

Oracle Data Dictionary - <https://netfiles.uiuc.edu/jstrode/www/oradd/index.html>.