DZ_DICT

- Release:
- Commit Date: Mon Oct 10 16:40:21 2016 -0400

Utilities for the manipulation of the Oracle data dictionary.

Summary

DZ_DICT

Functions

dz_dict_main.drop_type_quietly	Utility function which drops a type without returning an error if the type does not exist.
dz_dict_main.drop_table_quietly	Utility function which drops a table without returning an error if the table not exist.
dz_dict_main.table_exists	Utility function to determine if a table (or view) exists and is selectable.
dz_dict_main.tables_exists	Utility function to determine if one or more tables exist in a given schema.
dz_dict_main.mview_exists	Utility function to determine if a materialized view exists in a given schema.
dz_dict_main.tablespace_exists	Utility function to determine if a tablespace exists.
dz_dict_main.table_privileges	Utility function to dump the privledges you currently have on the provided table.
dz_dict_main.table_privileges_dml	Utility function to determine if user has full SELECT, UPDATE, INSERT and DELETE rights on a given table.
dz_dict_main.sequence_exists	Utility function to determine if a sequence exists in a given schema.
dz_dict_main.drop_sequence	Utility to drop one or more sequences matching a name value either exactly or via wildcard LIKE selection.
dz_dict_main. sequence_from_max_column	Utility to generate a sequence with a “start with” value equal to the next integer value in an existing table and column.
dz_dict_main.quick_sequence	Utility function to quickly generate a simple sequence.
dz_dict_main.object_exists	Utility function to interrogate all_objects for existence a given object.
dz_dict_main.object_is_valid	Utility function to interrogate all_objects for existence and validity of a given object.
dz_dict_main.get_column_number	Utility function to interrogate all_objects for existence and validity of a given object.
dz_dict_main.rename_to_x	Utility function to rename a given table with X suffix, usually used in preparation for rebuilding or redefining the table into a new table with the old name.
dz_dict_main.fast_not_null	Utility to quickly add a not null constraint to a column of a given table.
dz_dict_main.drop_indexes	Utility to drop all indexes from a given table.
dz_dict_main.drop_index	Utility to drop all indexes matching a given name or LIKE wildcard.
dz_dict_main.index_exists	Utility function to verify existence a given index.
dz_dict_main.drop_constraints	Utility to drop all constraints from a given table.
dz_dict_main.drop_constraint	Utility to drop all constraints matching a given name or LIKE wildcard.
dz_dict_main.drop_ref_constraints	Utility to drop all referential constraints from a given table.
dz_dict_main.get_index_name	Procedure to extract the index name from a given table and column.
dz_dict_main.new_index_name	Utility function to generate a standardized and unique name for a single-column index on a given table.
dz_dict_main.fast_index	Procedure to quickly and with no fuss build a simple index with a standardized name on a given table and column.
dz_dict_main.new_index_name	Utility function to dump all index DBMS_METADATA.GET_DDL information on a given table.

Functions

dz_dict_main.drop_type_quietly

Utility function which drops a type without returning an error if the type does not exist.

Parameters

p_owner	optional owner of the type, default is current user.
p_type_name	name of type to drop.

Returns

Nothing

dz_dict_main.drop_table_quietly

Utility function which drops a table without returning an error if the table not exist.

Parameters

p_owner	optional owner of the table, default is current user.
p_table_name	name of table to drop.

Returns

Nothing

dz_dict_main.table_exists

Utility function to determine if a table (or view) exists and is selectable. Test may also include one or more column names as part of the test.

Parameters

p_owner	optional owner of the table or view, default is current user.
p_table_name	name of table or view to verify
p_column_name	optional column name or names to verify is part of table in question

Returns

TRUE or FALSE as VARCHAR2 text string

dz_dict_main.tables_exists

Utility function to determine if one or more tables exist in a given schema.

Parameters

p_owner	optional owner of the tables or views, default is current user.
p_table_names	array of table names or views to verify

Returns

TRUE or FALSE as VARCHAR2 text string

dz_dict_main.mview_exists

Utility function to determine if a materialized view exists in a given schema.

Parameters

p_owner	optional owner of the materialized view, default is current user.
p_mview_name	materialized view name to verify

Returns

TRUE or FALSE as VARCHAR2 text string

dz_dict_main.tablespace_exists

Utility function to determine if a tablespace exists.

Parameters

p_tablespace_name	tablespace name to verify

Returns

TRUE or FALSE as VARCHAR2 text string

dz_dict_main.table_privileges

Utility function to dump the privledges you currently have on the provided table.

Parameters

p_owner	optional owner of the table or view, default is current user.
p_table_name	table or view name to verify

Returns

MDSYS.SDO_STRING2_ARRAY of privledges as text strings.

dz_dict_main.table_privileges_dml

Utility function to determine if user has full SELECT, UPDATE, INSERT and DELETE rights on a given table.

Parameters

p_owner	optional owner of the table or view, default is current user.
p_table_name	table or view name to verify

Returns

TRUE or FALSE as VARCHAR2 text string

dz_dict_main.sequence_exists

Utility function to determine if a sequence exists in a given schema.

Parameters

p_owner	optional owner of the sequence, default is current user.
p_sequence_name	sequence name to verify

Returns

TRUE or FALSE as VARCHAR2 text string

dz_dict_main.drop_sequence

Utility to drop one or more sequences matching a name value either exactly or via wildcard LIKE selection.

Parameters

p_owner	optional owner of the sequences, default is current user.
p_sequence_name	sequence name or wilcard value to verify
p_like_flag	optional flag to trigger LIKE verse equals sequence name matching.

Returns

Nothing

dz_dict_main.sequence_from_max_column

Utility to generate a sequence with a “start with” value equal to the next integer value in an existing table and column.

Parameters

p_owner	optional owner of the table, default is current user.
p_table_name	table name containing the column to interrogate
p_column_name	column name to interrogate to determine the sequence start with value.
p_sequence_owner	optional owner of the new sequence, default is current user.
p_sequence_name	name of the sequence to create. If NULL is provided then this value will contain a randomly generated name used for resulting sequence.

Returns

Nothing

Notes

- Random sequence name created using dz_dict_util.unique_name() function.
- If source table is empty, the sequence is create with a start with value of 1.

dz_dict_main.quick_sequence

Utility function to quickly generate a simple sequence.

Parameters

p_owner	optional owner of the sequence, default is current user.
p_start_with	optional start with value, default is 1.
p_prefix	optional prefix value for unique name generation.
p_suffix	optional suffix value for unique name generation.

Returns

VARCHAR2 text name of the sequence created.

dz_dict_main.object_exists

Utility function to interrogate all_objects for existence a given object.

Parameters

p_owner	optional owner of the object, default is current user.
p_object_type_name	name of the object to verify.

Returns

TRUE or FALSE as VARCHAR2 text string.

dz_dict_main.object_is_valid

Utility function to interrogate all_objects for existence and validity of a given object.

Parameters

p_owner	optional owner of the object, default is current user.
p_object_type_name	name of the object to verify.

Returns

TRUE or FALSE as VARCHAR2 text string.

dz_dict_main.get_column_number

Utility function to interrogate all_objects for existence and validity of a given object.

Parameters

p_owner	optional owner of the table, default is current user.
p_table_name	table or view name to inspect.
p_column_name	column name in table to inspect.

Returns

Number indicating position of column in table.

dz_dict_main.rename_to_x

Utility function to rename a given table with X suffix, usually used in preparation for rebuilding or redefining the table into a new table with the old name.
E.g. Renames MyTable to MyTable_X

Parameters

p_owner	optional owner of the table, default is current user.
p_table_name	table to rename.
p_flush_objects	optional TRUE or FALSE flag to drop all indexes and constraints on the table.

Returns

VARCHAR2 string of the new name of the renamed table.

Notes

- The function is smart enough to rename to X1, X2, etc if X already exists.

dz_dict_main.fast_not_null

Utility to quickly add a not null constraint to a column of a given table.

Parameters

p_owner	optional owner of the table, default is current user.
p_table_name	table name containing the column to set as not null.
p_column_name	column name to set as not null.

Returns

Nothing

dz_dict_main.drop_indexes

Utility to drop all indexes from a given table.

Parameters

p_owner	optional owner of the table, default is current user.
p_table_name	table name from which to drop all indexes.

Returns

Nothing

dz_dict_main.drop_index

Utility to drop all indexes matching a given name or LIKE wildcard.

Parameters

p_owner	optional owner of the index, default is current user.
p_index_name	index name or LIKE wildcard to match indexes in owner's schema.
p_like_flag	optional flag to trigger LIKE verse equals index name matching.

Returns

Nothing

dz_dict_main.index_exists

Utility function to verify existence a given index.

Parameters

p_owner	optional owner of the index, default is current user.
p_index_name	name of the index to verify.

Returns

TRUE or FALSE as VARCHAR2 text string.

dz_dict_main.drop_constraints

Utility to drop all constraints from a given table.

Parameters

p_owner	optional owner of the table, default is current user.
p_table_name	table name from which to drop all constraints.

Returns

Nothing

dz_dict_main.drop_constraint

Utility to drop all constraints matching a given name or LIKE wildcard.

Parameters

p_owner	optional owner of the constraint, default is current user.
p_index_name	constraint name or LIKE wildcard to match constraints in owner's schema.
p_like_flag	optional flag to trigger LIKE verse equals constraint name matching.

Returns

Nothing

dz_dict_main.drop_ref_constraints

Utility to drop all referential constraints from a given table.

Parameters

p_owner	optional owner of the table, default is current user.
p_table_name	table name from which to drop all referential constraints.

Returns

Nothing

dz_dict_main.get_index_name

Procedure to extract the index name from a given table and column. This only applies to single column indexes.

Parameters

p_owner	optional owner of the table, default is current user.
p_table_name	table name to inspect.
p_column_name	column name to inspect for single-column index.

Returns

p_index_owner	index owner
p_index_name	index name

dz_dict_main.new_index_name

Utility function to generate a standardized and unique name for a single-column index on a given table.

Parameters

p_owner	optional owner of the table, default is current user.
p_table_name	table name to inspect.
p_column_name	column name to inspect.
p_suffix_ind	optional value in suffix to indicate index type, default is "I".
p_full_suffix	optional full value for the index suffix.

Returns

VARCHAR2 string of the index name.

Notes

- This function does not actually create the index, it's just intended to provide a standardized name for whatever index you decide to create.

dz_dict_main.fast_index

Procedure to quickly and with no fuss build a simple index with a standardized name on a given table and column.

Parameters

p_owner	optional owner of the table, default is current user.
p_table_name	table name to build the index upon.
p_column_name	column name to build the index upon.
p_index_type	optional flag to build alternative index types such as BITMAP.
p_tablespace	optional value for specific index tablespace.
p_logging	optional flag to mark index as LOGGING or NOLOGGING, default is to not specify (required for situations such as temp tables whose indexes cannot be either LOGGING or NOLOGGING).

Returns

Nothing

dz_dict_main.new_index_name

Utility function to dump all index DBMS_METADATA.GET_DDL information on a given table. Useful if you need to drop all indexes for some purpose and then recreate them afterwards.

Parameters

p_owner	optional owner of the table, default is current user.
p_table_name	table name to inspect.

Returns

MDSYS.SDO_STRING2_ARRAY of string values of each DDL statement extracted from the table.