296

UTL_MATCH

The `UTL_MATCH` package facilitates matching two records. This is typically used to match names, such as two First Names or two Last Names.

This chapter contains the following topics:

- Overview
- Security Model
- Summary of UTL_MATCH Subprograms

UTL_MATCH Overview

`UTL_MATCH` can use either the Edit Distance algorithm or Jaro-Winkler algorithm when determining matches.

Edit Distance, also known as Levenshtein Distance (named after the Russian scientist Vladimir Levenshtein, who devised the algorithm in 1965), is a measure of similarity between two strings, `s1` and `s2`. The distance is the number of insertions, deletions or substitutions required to transform s1 to s2.

The Edit Distance between strings `shack*le*ford` and `shack*el*ford` = 2.

The "Jaro-Winkler algorithm" is another way of calculating Edit distance between two strings. This method, developed at the U.S. Census, is a String Comparator measure that gives values of partial agreement between two strings. The string comparator accounts for length of strings and partially accounts for typical human errors made in alphanumeric strings.

The following table shows similarity values returned by Jaro-Winkler and Edit Distance

Table 296-1 Comparison between normalized values returned by Jaro-Winkler and Edit Distance algorithms

String 1	String 2	Jaro Winkler	Edit Distance
Dunningham	Cunnigham	89	80
Abroms	Abrams	92	83
Lampley	Campley	90	86
Marhta	Martha	96	67
Jonathon	Jonathan	95	88
Jeraldine	Geraldine	92	89

UTL_MATCH Security Model

The `UTL_MATCH` package runs with definer's rights. `UTL_MATCH` must be created under `SYS`. Operations provided by this package are performed with `SYS` privileges.

Summary of UTL_MATCH Subprograms

This table lists the `UTL_MATCH` subprograms and briefly describes them.

Table 296-2 UTL_MATCH Package Subprograms

Subprogram	Description
EDIT_DISTANCE Function	Calculates the number of changes required to transform string-1 into string-2
EDIT_DISTANCE_SIMILARITY Function	Calculates the number of changes required to transform string-1 into string-2, returning a value between 0 (no match) and 100 (perfect match)
JARO_WINKLER Function	Calculates the measure of agreement between string-1 and string-2
JARO_WINKLER_SIMILARITY Function	Calculates the measure of agreement between string-1 and string-2, returning a value between 0 (no match) and 100 (perfect match)

EDIT_DISTANCE Function

This function calculates the number of insertions, deletions or substitutions required to transform string-1 into string-2.

Syntax

```
UTL_MATCH.EDIT_DISTANCE (
   s1  IN  VARCHAR2,
   s2  IN  VARCHAR2)
 RETURN PLS_INTEGER;
```

Parameters

Table 296-3 EDIT_DISTANCE Function Parameters

Parameter	Description
`s1`	The string to be transformed
`s2`	The string into which `s1` is to be transformed

Examples

```
SELECT UTL_MATCH.EDIT_DISTANCE('shackleford', 'shackelford') FROM DUAL;
-------------
returns 2
```

EDIT_DISTANCE_SIMILARITY Function

This function calculates the number of insertions, deletions or substations required to transform string-1 into string-2, and returns the Normalized value of the Edit Distance between two strings.

The value is typically between 0 (no match) and 100 (perfect match).

Syntax

```
UTL_MATCH.EDIT_DISTANCE_SIMILARITY (
   s1  IN  VARCHAR2,
   s2  IN  VARCHAR2)
 RETURN PLS_INTEGER;
```

Parameters

Table 296-4 EDIT_DISTANCE_SIMILARITY Function Parameters

Parameter	Description
s1	The string to be transformed
s2	The string into which s1 is to be transformed

Examples

```
SELECT UTL_MATCH.EDIT_DISTANCE_SIMILARITY('shackleford', 'shackelford') FROM DUAL;
--------------
returns 82
```

JARO_WINKLER Function

This function calculates the measure of agreement between two strings.

Syntax

```
UTL_MATCH.JARO_WINKLER (
   s1  IN  VARCHAR2,
   s2  IN  VARCHAR2)
 RETURN BINARY_DOUBLE;
```

Parameters

Table 296-5 JARO_WINKLER Function Parameters

Parameter	Description
s1	Input
s2	input

Examples

```
SELECT UTL_MATCH.JARO_WINKLER('shackleford', 'shackelford') FROM DUAL;
--------------
returns 9.818E-001
```

JARO_WINKLER_SIMILARITY Function

This function calculates the measure of agreement between two strings, and returns a score between 0 (no match) and 100 (perfect match).

Syntax

```
UTL_MATCH.JARO_WINKLER_SIMILARITY (
   s1  IN  VARCHAR2,
   s2  IN  VARCHAR2)
 RETURN PLS_INTEGER;
```

Parameters

Table 296-6 JARO_WINKLER_SIMILARITY Function Parameters

Parameter	Description
s1	Input
s2	input

Examples

```
SELECT UTL_MATCH.JARO_WINKLER_SIMILARITY('shackleford', 'shackelford') FROM DUAL;
--------------
returns 98
```