

UTL_ENCODE

The `UTL_ENCODE` package provides functions that encode `RAW` data into a standard encoded format so that the data can be transported between hosts.

You can use `UTL_ENCODE` functions to encode the body of email text. The package also contains the decode counterpart functions of the encode functions. The functions follow published standards for encoding to accommodate non-Oracle utilities on the sending or receiving ends.

This chapter contains the following topic:

- [Summary of UTL_ENCODE Subprograms](#)

Summary of UTL_ENCODE Subprograms

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

Table 288-1 *UTL_ENCODE Package Subprograms*

| Subprogram | Description |
|--|---|
| BASE64_DECODE Function | Reads the base 64-encoded <code>RAW</code> input string and decodes it to its original <code>RAW</code> value |
| BASE64_ENCODE Function | Encodes the binary representation of the <code>RAW</code> value into base 64 elements and returns it in the form of a <code>RAW</code> string |
| MIMEHEADER_DECODE Function | Decodes a string from mime header format |
| MIMEHEADER_ENCODE Function | Encodes a string into mime header format |
| QUOTED_PRINTABLE_DECODE Function | Reads the <code>varchar2</code> quoted printable format input string and decodes it to the corresponding <code>RAW</code> string |
| QUOTED_PRINTABLE_ENCODE Function | Reads the <code>RAW</code> input string and encodes it to the corresponding quoted printable format string |
| TEXT_DECODE Function | Decodes a character set sensitive text string |
| TEXT_ENCODE Function | Encodes a character set sensitive text string |
| UUDECODE Function | Reads the <code>RAW</code> uuencode format input string and decodes it to the corresponding <code>RAW</code> string |
| UUENCODE Function | Reads the <code>RAW</code> input string and encodes it to the corresponding uuencode format string |

BASE64_DECODE Function

This function reads the base 64-encoded RAW input string and decodes it to its original RAW value.

Syntax

```
UTL_ENCODE.BASE64_DECODE (  
    r IN RAW)  
RETURN RAW;
```

Pragmas

```
pragma RESTRICT_REFERENCES (base64_decode, WNDS, RNDS, WNPS, RNPS);
```

Parameters

Table 288-2 BASE64_DECODE Function Parameters

| Parameter | Description |
|-----------|---|
| r | The RAW string containing base 64-encoded data. There are no defaults or optional parameters. |

Return Values

Table 288-3 BASE64_DECODE Function Return Values

| Return | Description |
|--------|-----------------------------|
| RAW | Contains the decoded string |

BASE64_ENCODE Function

This function encodes the binary representation of the RAW value into base 64 elements and returns it in the form of a RAW string.

Syntax

```
UTL_ENCODE.BASE64_ENCODE (  
    r IN RAW)  
RETURN RAW;
```

Pragmas

```
pragma RESTRICT_REFERENCES (base64_encode, WNDS, RNDS, WNPS, RNPS);
```

Parameters

Table 288-4 BASE64_ENCODE Function Parameters

| Parameter | Description |
|-----------|--|
| r | The RAW value to be encoded. There are no defaults or optional parameters. |

Return Values

Table 288-5 BASE64_ENCODE Function Return Values

| Return | Description |
|--------|---------------------------------------|
| RAW | Contains the encoded base 64 elements |

MIMEHEADER_DECODE Function

This function accepts as input an "encoded word."

It uses the form:

```
=?<charset>?<encoding>?<encoded text>?=
=?ISO-8859-1?Q?Here is some encoded text?=
```

The <encoded text> is encapsulated in mime header tags which give the `MIMEHEADER_DECODE` function information about how to decode the string. The mime header metadata tags are stripped from the input string and the <encoded text> is converted to the base database character set as follows:

- If this is a UTF16 platform, convert the encoded text from UTF16 to ASCII
- If this is an EBCDIC platform, convert the encoded text from EBCDIC to ASCII
- If this is an ASCII or UTF8 platform, no conversion needed

The string is decoded using either quoted-printable or base64 decoding, as specified by the <encoding> metadata tag in the encoded word. The resulting converted and decoded text is returned to the caller as a `VARCHAR2` string.

Syntax

```
UTL_ENCODE.MIMEHEADER_DECODE (
    buf    IN  VARCHAR2 CHARACTER SET ANY_CS)
RETURN data VARCHAR2 CHARACTER SET buf%CHARSET;
```

Parameters

Table 288-6 MIMEHEADER_DECODE Function Parameters

| Parameter | Description |
|-----------|---|
| buf | The encoded text data with mime header format tags. |

Return Values

Table 288-7 MIMEHEADER_DECODE Function Return Values

| Return | Description |
|--------|--|
| data | The encoded text data with mime header format tags |

Examples

```
v2:=utl_encode.mimeheader_decode('=?ISO-8859-1?Q?Here is some encoded text?=');
```

MIMEHEADER_ENCODE Function

This function returns as an output an "encoded word".

The output is in the following form:

```
=?<charset>?<encoding>?<encoded text>?=
=?ISO-8859-1?Q?Here is some text?=
```

The `buf` input parameter is the text to be encoded and becomes the `<encoded text>`.

The `<encoding>` value is either "Q" or "B" for quoted-printable encode or base64 encoding respectively. The `ENCODING` input parameter accepts as valid values

`UTL_ENCODE.QUOTED_PRINTABLE` or `UTL_ENCODE.BASE64` or `NULL`. If `NULL`, quoted-printable encoding is selected as a default value.

The `<charset>` value is specified as the input parameter `encode_charset`. If `NULL`, the database character set is selected as a default value.

The mimeheader encoding process includes conversion of the `buf` input string to the character set specified by the `encode_charset` parameter. The converted string is encoded to either quoted-printable or base64 encoded format. The mime header tags are appended and prepended.

Finally, the string is converted to the base character set of the database:

- If this is a UTF16 platform, convert the encoded text to UTF16
- If this is an EBCDIC platform, convert the encoded text to EBCDIC
- If this is an ASCII or UTF8 platform, no conversion needed.

Syntax

```
UTL_ENCODE.MIMEHEADER_ENCODE (  
    buf                IN  VARCHAR2 CHARACTER SET ANY_CS,  
    encode_charset IN  VARCHAR2 DEFAULT NULL,  
    encoding           IN  PLS_INTEGER DEFAULT NULL)  
RETURN string VARCHAR2 CHARACTER SET buf%CHARSET;
```

Parameters

Table 288-8 MIMEHEADER_ENCODE Function Parameters

| Parameter | Description |
|-----------------------------|---|
| <code>buf</code> | The text data. |
| <code>encode_charset</code> | The target character set. |
| <code>encoding</code> | The encoding format. Valid values are <code>UTL_ENCODE.BASE64</code> , <code>UTL_ENCODE.QUOTED_PRINTABLE</code> and <code>NULL</code> |

Return Values

Table 288-9 MIMEHEADER_ENCODE Function Return Values

| Return | Description |
|--------|---|
| string | A VARCHAR2 encoded string with mime header format tags. |

QUOTED_PRINTABLE_DECODE Function

This function reads the `varchar2` quoted printable format input string and decodes it to the corresponding `RAW` string.

Syntax

```
UTL_ENCODE.QUOTED_PRINTABLE_DECODE (  
    r IN RAW)  
RETURN RAW;
```

Pragmas

```
pragma RESTRICT_REFERENCES(quoted_printable_decode, WNDS, RNDS, WNPS, RNPS);
```

Parameters

Table 288-10 QUOTED_PRINTABLE_DECODE Function Parameters

| Parameters | Description |
|------------|---|
| r | The RAW string containing a quoted printable data string. There are no defaults or optional parameters. |

Return Values

Table 288-11 QUOTED_PRINTABLE_DECODE Function Return Values

| Return | Description |
|--------|--------------------|
| RAW | The decoded string |

QUOTED_PRINTABLE_ENCODE Function

This function reads the `RAW` input string and encodes it to the corresponding quoted printable format string.

Syntax

```
UTL_ENCODE.QUOTED_PRINTABLE_ENCODE (  
    r IN RAW)  
RETURN RAW;
```

Pragmas

```
pragma RESTRICT_REFERENCES(quoted_printable_encode, WNDS, RNDS, WNPS, RNPS);
```

Parameters

Table 288-12 QUOTED_PRINTABLE_ENCODE Function Parameters

| Parameter | Description |
|----------------|---|
| <code>r</code> | The RAW string. There are no defaults or optional parameters. |

Return Values

Table 288-13 QUOTED_PRINTABLE_ENCODE Function Return Values

| Return | Description |
|--------|--------------------------------------|
| RAW | Contains the quoted printable string |

TEXT_DECODE Function

This function converts the input text to the target character set as specified by the `encode_charset` parameter, if not NULL.

The encoded text is converted to the base character set of database, as follows:

- If this is a UTF16 platform, convert the encoded text from UTF16 to ASCII
- If this is an EBCDIC platform, convert the encoded text from EBCDIC to ASCII
- If this is an ASCII or UTF8 platform, no conversion needed

You can decode from either quoted-printable or base64 format, with regard to each `encoding` parameter. If NULL, quoted-printable is selected as a default decoding format. If `encode_charset` is not NULL, you convert the string from the specified character set to the database character set. The resulting decoded and converted text string is returned to the caller.

Syntax

```
UTL_ENCODE.TEXT_DECODE(  
    buf          IN VARCHAR2 CHARACTER SET ANY_CS,  
    encode_charset IN VARCHAR2 DEFAULT NULL,  
    encoding      IN PLS_INTEGER DEFAULT NULL)  
RETURN string VARCHAR2 CHARACTER SET buf%CHARSET;
```

Parameters

Table 288-14 TEXT_DECODE Function Parameters

| Parameter | Description |
|-----------------------------|---|
| <code>buf</code> | The encoded text data. |
| <code>encode_charset</code> | The source character set. |
| <code>encoding</code> | The encoding format. Valid values are <code>UTL_ENCODE.BASE64</code> , <code>UTL_ENCODE.QUOTED_PRINTABLE</code> and NULL. |

Return Values

Table 288-15 TEXT_DECODE Function Return Values

| Return | Description |
|--------|---------------------------------|
| string | A VARCHAR2 decoded text string. |

Examples

```
v2:=UTL_ENCODE.TEXT_DECODE(  
    'Here is some text',  
    WE8ISO8859P1,  
    UTL_ENCODE.BASE64);
```

TEXT_ENCODE Function

This function converts the input text to the target character set as specified by the `encode_charset` parameter, if not `NULL`.

The text is encoded to either base64 or quoted-printable format, as specified by the `encoding` parameter. Quoted-printable is selected as a default if `ENCODING` is `NULL`.

The encoded text is converted to the base character set of the database:

- If this is a UTF16 platform, convert the encoded text to UTF16
- If this is an EBCDIC platform, convert the encoded text to EBCDIC
- If this is an ASCII or UTF8 platform, no conversion needed

The resulting encoded and converted text string is returned to the caller.

Syntax

```
UTL_ENCODE.TEXT_ENCODE (  
    buf          IN  VARCHAR2 CHARACTER SET ANY_CS,  
    encode_charset IN  VARCHAR2 DEFAULT NULL,  
    encoding      IN  PLS_INTEGER DEFAULT NULL)  
RETURN string VARCHAR2 CHARACTER SET buf%CHARSET;
```

Parameters

Table 288-16 TEXT_ENCODE Function Parameters

| Parameter | Description |
|----------------|---|
| buf | The text data. |
| encode_charset | The target character set. |
| encoding | The encoding format. Valid values are <code>UTL_ENCODE.BASE64</code> , <code>UTL_ENCODE.QUOTED_PRINTABLE</code> and <code>NULL</code> |

Return Values

Table 288-17 TEXT_ENCODE Function Return Values

| Return | Description |
|--------|---|
| string | A VARCHAR2 encoded string with mime header format tags. |

Examples

```
v2:=utl_encode.text_encode(  
    'Here is some text',  
    'WE8ISO8859P1',  
    UTL_ENCODE.BASE64);
```

UUDECODE Function

This function reads the RAW uuencode format input string and decodes it to the corresponding RAW string.

See "[UUENCODE Function](#)" for discussion of the cumulative nature of UUENCODE and UUDECODE for data streams.

Syntax

```
UTL_ENCODE.UUDECODE (  
    r IN RAW)  
RETURN RAW;
```

Pragmas

```
pragma RESTRICT_REFERENCES(uudecode, WNDS, RNDS, WNPS, RNPS);
```

Parameters

Table 288-18 UUDECODE Function Parameters

| Parameter | Description |
|-----------|--|
| r | The RAW string containing the uuencoded data string. There are no defaults or optional parameters. |

Return Values

Table 288-19 UUDECODE Function Return Values

| Return | Description |
|--------|------------------------|
| RAW | The decoded RAW string |

UUENCODE Function

This function reads the `RAW` input string and encodes it to the corresponding uuencode format string.

The output of this function is cumulative, in that it can be used to encode large data streams, by splitting the data stream into acceptably sized `RAW` values, encoded, and concatenated into a single encoded string.

Syntax

```
UTL_ENCODE.UUENCODE (  
    r            IN RAW,  
    type         IN PLS_INTEGER DEFAULT 1,  
    filename     IN VARCHAR2 DEFAULT NULL,  
    permission   IN VARCHAR2 DEFAULT NULL) RETURN RAW;
```

Pragmas

```
pragma RESTRICT_REFERENCES (uuencode, WNDS, RNDS, WNPS, RNPS);
```

Parameters

Table 288-20 UUENCODE Function Parameters

| Parameter | Description |
|-------------------------|--|
| <code>r</code> | RAW string |
| <code>type</code> | Optional number parameter containing the type of uuencoded output. Options: complete—a defined PL/SQL constant with a value of 1. (default) header_piece ...middle_piece ...end_piece |
| <code>filename</code> | Optional <code>varchar2</code> parameter containing the uuencode filename; the default is <code>uuencode.txt</code> |
| <code>permission</code> | Optional <code>varchar2</code> parameter containing the permission mode; the default is 0 (a text string zero). |

Return Values

Table 288-21 UUENCODE Function Return Values

| Return | Description |
|--------|-------------------------------------|
| RAW | Contains the uuencode format string |