

プロシージャやファンクションなどをすべてコンパイルする  
 オラクルのバージョンを表示する  
 初期化パラメータファイルの互換性値 (compatibility) を表示する  
 DDL文 (Create、Alter) を実行する  
 オブジェクト間の依存関係を出力する  
 指定した初期化パラメータ値を取得する

DBMS\_UTILITY パッケージは、その他のパッケージです。

多種の機能を提供しています。

それぞれのプロシージャには、相互の関係性はありません。

パッケージのプロシージャは、以下の通りです

パッケージ名 : DBMS\_UTILITY.

プロシージャ名	サービス提供の内容説明
COMPILE_SCHEMA	指定したスキーマが所有するすべてのプロシージャやファンクションなどをすべてコンパイルする
DB_VERSION	オラクルのバージョンを表示する 初期化パラメータファイルの互換性値 (compatibility) を表示する
EXEC_DDL_STATEMENT	DDL文 (Create、Alter) を実行する
GET_DEPENDENCY	指定されたオブジェクト間の依存関係を出力する
GET_PARAMETER_VALUE	指定した初期化パラメータ値を取得する

構成プロシージャについて

プロシージャ名 : COMPILE\_SCHEMA

機能説明 : 指定したスキーマが所有するすべてのプロシージャやファンクションなどをすべてコンパイルする

受渡し引数

引数名	I/O	データ型	説明
schema	IN	VARCHAR2	コンパイル対象のスキーマ名を指定
compile_all	IN	BOOLEAN	常に省略して使用 DEFAULT TRUE

プロシージャの呼出し方

/\* パッケージの起動 \*/

SQL> EXECUTE DBMS\_UTILITY.COMPILE\_SCHEMA ('user1') ;

## 構成プロシージャについて

プロシージャ名：DB\_VERSION

機能説明：オラクルのバージョンを表示する

初期化パラメータファイルの互換性値（compatibility）を表示する

受渡し引数

引数名	I/O	データ型	説 明
version	OUT	VARCHAR2	バージョンを表す文字列
compatibility	OUT	VARCHAR2	初期化パラメータ compatibility の値

## プロシージャの呼出し方

```
/* OUT 用引数の受け取り変数の定義 */
SQL> VARIABLE ver VARCHAR2( 16 )
SQL> VARIABLE comp VARCHAR2( 16 )

/* パッケージの起動 */
/* OUT 用パラメータには、バインド変数としてコロン「:」を付ける */
SQL> EXECUTE DBMS_UTILITY.DB_VERSION ( :ver , :comp ) ;
/* オラクルのバージョンと初期化パラメータファイルの互換性値（compatibility）
   を表示するジョブ番号の出力表示 */
SQL> PRINT ver
SQL> PRINT comp
```

## 構成プロシージャについて

プロシージャ名：EXEC\_DDL\_STATEMENT

機能説明：DDL 文（Create、Alter）を実行する

受渡し引数

引数名	I/O	データ型	説 明
parse_string	IN	VARCHAR2	実行したい DDL 文の文字列

## プロシージャの呼出し方

```
/* パッケージの起動 */
SQL> DECLARE str VARCHAR2 ( 128 );
      BEGIN
        str := 'CREATE TABLE TableA ( Kye_F NUMBER, Data_F VARCHAR(16))';
        DBMS_UTILITY.EXEC_DDL_STATEMENT ( str ) ;
      END ;
SQL> /
```

## 構成プロシージャについて

プロシージャ名：GET\_DEPENDENCY

機能説明：指定されたオブジェクト間の依存関係を出力する

受渡し引数

引数名	I/O	データ型	説 明
type	IN	VARCHAR2	対象のオブジェクトの型 'TABLE'、'FUNCTION'、'INDEX'、'PACKAGE'、 'PROCEDURE'、'TRIGGER'、'VIEW'
schema	IN	VARCHAR2	対象となるオブジェクトを所有しているスキーマ
name	IN	VARCHAR2	オブジェクトの名前を指定

## プロシージャの呼出し方

```
/* パッケージの起動 */
```

```
SQL> EXECUTE DBMS_UTILITY.GET_DEPENDENCY ('タイプ', 'スキーマ', 'オブジェクト名') ;
```

### 【 注意事項 】

調査対象のオブジェクトを所有しているユーザーで SQL/Plus を接続していないと、エラーが発生する

これは、例え system ユーザーであっても一般ユーザーのオブジェクトの調査ができない

— SYSTEM ユーザーの接続で、一般ユーザーのオブジェクトを調査した時のエラー —

```
SQL> EXECUTE DBMS_UTILITY.GET_DEPENDENCY ('TABLE', 'KOZUE', 'EMP') ;
```

\*

行 1 でエラーが発生しました。:

ORA-20000: ORU-10013: TABLE KOZUE.EMP was not found.

ORA-06512: "SYS.DBMS\_UTILITY", 行 826

ORA-06512: 行 1

## 実行例

```
SQL> EXECUTE DBMS_UTILITY.GET_DEPENDENCY ('TABLE', 'KOZUE', 'EMP') ;
```

-

DEPENDENCIES ON KOZUE.EMP

-----

\*TABLE KOZUE.EMP()

PL/SQL プロシージャが正常に完了しました。

```
SQL> EXECUTE DBMS_UTILITY.GET_DEPENDENCY ('TABLE', 'SYSTEM', 'AQ$_INTERNET_AGENT_PRIVS') ;
```

-

DEPENDENCIES ON SYSTEM.AQ\$\_INTERNET\_AGENT\_PRIVS

-----

\*TABLE SYSTEM.AQ\$\_INTERNET\_AGENT\_PRIVS()

\* VIEW SYS.USER\_AQ\_AGENT\_PRIVS()

PL/SQL プロシージャが正常に完了しました。

## 構成~~プロセス~~ファンクションについて

ファンクション名：GET\_PARAMETER\_VALUE

機能説明：指定した初期化パラメータ値を取得する

受渡し引数

引数名	I/O	データ型	説 明
parnam	IN	VARCHAR2	初期化パラメータの名称
intval	I/O	BINARY_INTEGER	数値の場合の値の戻り場所
strval	I/O	VARCHAR2	文字列の場合の値の戻り場所
RETURN			戻り値が、 整数 or BOOLEAN 型の場合は、0 文字列 or ファイルパラメータの場合は、1

## プロセスの呼出し方

```
SQL> DECLARE ret NUMBER ;  
        intval NUMBER ;  
        strval VARCHAR2( 256 ) ;  
BEGIN  
    ret := DBMS_UTILITY.GET_PARAMETER_VALUE ('パラメータ名',  
        intval, strval ) ;  
    IF ret = 0 THEN  
        DBMS_OUTPUT.PUT_LINE('intval = ' || TO_CHAR(intval));  
    ELSE  
        DBMS_OUTPUT.PUT_LINE('strval = ' || strval);  
    END IF ;  
END ;
```

SQL> /

(例) パラメータ名：sort\_area\_size、 user\_dump\_dest