

列への単純データをセットしたテスト用データの作成

```
create table test_easy(col1 number, col2 number, col3 varchar2(20));

begin
  for i in 1..1000 loop
    insert into test_easy values( i, mod(i, 5), 'dummy' ) ;
  end loop ;
end ;
/
```

```
select * from test_easy order by col1 ;
```

col1	col2	col3
1	1	dummy
2	2	dummy
3	3	dummy
4	4	dummy
5	5	dummy
6	1	dummy
7	2	dummy
8	3	dummy

```
update test_easy set col3 = '関東' where col2 = 1
```

これで、全体の 20%のデータが、col3=関東となる

列へのランダム・データをセットしたテスト用データの作成

```
create table test_num(col1 number, col2 number);
```

```
begin
  dbms_random.SEED(1) ;
  for i in 1..100000 loop
    insert into test_num values(i, ceil(dbms_random.value(1, 1000000000))) ;
  end loop ;
end ;
/
```

The diagram consists of two arrows. The first arrow starts at the text '乱数の SEED を設定' (Set random SEED) and points to the 'dbms_random.SEED(1) ;' line in the code. The second arrow starts at the text '乱数の発生' (Random number generation) and points to the 'dbms_random.value(1, 1000000000))' part of the 'insert into test_num values...' line in the code.

```
select count(*) from (
  select col2, count(*) from test_num
  group by col2 having count(*) >= 2);

  count(*)
  -----
         3
```

1～10億の値の種類で、10万件のレコードを生成すると、重複しているレコードは3件

配列変数を使った INSERT 処理の高速化 (バルク Insert 処理)

あらかじめ配列変数に複数レコードの値をセットしておき、それを1回のSQL処理でINSERTするバルクINSERT処理

```
-- テーブル定義
CREATE TABLE <テーブル名>
  ( 列名 1 データ型,
    列名 2 データ型 ) ;

-- バルク INSERT 実行

DECLARE
-- 1. LARGE_TBL レコード形式の配列宣言
  TYPE <ユーザー定義型名> IS TABLE OF <テーブル名>%ROWTYPE
    INDEX BY <ユーザー定義型名> binary_integer ;
  <配列変数名> <ユーザー定義型名> ;

BEGIN
-- 2. 配列に値をセット
  FOR i IN 1 .. 1000000 LOOP
    → <配列変数名>(i).列名 1 := i ;
    → <配列変数名>(i).列名 2 := '値' ;
  END LOOP;

-- 3. バルク INSERT 実行
  FORALL i in 1 .. 1000000 INSERT INTO <テーブル名>
    VALUES <配列変数名>(i) ;
  COMMIT ;

END ;
/
```

使用 例)

-- テーブル定義

```
CREATE TABLE LARGE_TBL (ID NUMBER, VALUE VARCHAR2(50));
```

-- バルク INSERT 実行

```
DECLARE
```

-- 1. LARGE_TBL レコード形式の配列宣言

```
TYPE tbl_ins IS TABLE OF LARGE_TBL%ROWTYPE
```

```
INDEX BY BINARY_INTEGER;
```

```
w_ins tbl_ins ;
```

```
BEGIN
```

-- 2. 配列に値をセット

```
FOR i IN 1 .. 1000000 LOOP
```

```
  w_ins(i).ID := i;
```

```
  w_ins(i).VALUE := 'foobar';
```

```
END LOOP;
```

-- 3. バルク INSERT 実行

```
FORALL i in 1 .. 1000000 INSERT INTO LARGE_TBL
```

```
  VALUES w_ins(i);
```

```
COMMIT;
```

```
END;
```

```
/
```

-- 長時間 CPU を使用する SQL 文の例

```
Select A1.value
```

```
  from LARGE_TBL A1, LARGE_TBL A2, LARGE_TBL A3
```

```
Where A1.Value = A2.Value and A1.Value = A2.Value
```

```
  and A2.Value = A3.Value
```

```
  and A1.ID Between 6 and 80000
```

```
  and MOD( A2.ID, 5) = 3;
```