

GOTO (不建議使用)

GOTO語句提供無條件跳轉到在同一個子程序的GOTO標籤的語句

不建議使用原因：

程序會變得很亂、難以理解

GOTO使用限制：

- goto語句不能跳轉到IF語句，CASE語句，循環語句或子塊。
- goto語句不能從一個分支IF語句子句到另一個或從一個CASE語句WHEN子句到另一個。
- goto語句不能從外部塊分支成子塊(即一個內BEGIN-END塊)。
- goto語句不能分支出來一個子程序。提前結束子程序，要麼使用RETURN語句或子程序結束前在正確的GOTO分支的地方。
- goto語句不能從一個異常處理分支回到當前BEGIN-END塊。但是，GOTO語句可以從一個異常處理程序跳轉到一個封閉塊。

=====

範例：

當a = 15時，a = a+3，並且跳到”loopstart”這個LOOP中

```
DECLARE
  a number(2) := 10;
BEGIN

  --宣告一個LOOP 階層名稱
  <<loopstart>>
  -- while loop execution

  WHILE a < 20 LOOP
    dbms_output.put_line ('value of a: ' || a);
```

```

a := a + 1;

IF a = 15 THEN
    a := a + 4;

    GOTO loopstart;

END IF;
END LOOP;
END;
/

```

db<>fiddle

By using db<>fiddle, you agree to license everything you submit by [Creative Commons CC0](https://creativecommons.org/licenses/by/4.0/).

```

1 DECLARE
2   a number(2) := 10;
3 BEGIN
4
5   --宣告一個LOOP 階層名稱
6   <<loopstart>>
7   -- while loop execution
8
9   WHILE a < 20 LOOP
10    dbms_output.put_line ('value of a: ' || a);
11    a := a + 1;
12
13    IF a = 15 THEN
14      a := a + 4;
15
16      GOTO loopstart;
17
18    END IF;
19  END LOOP;
20 END;
21 /

```

1 rows affected

dbms_output:
value of a: 10
value of a: 11
value of a: 12
value of a: 13
value of a: 14
value of a: 19