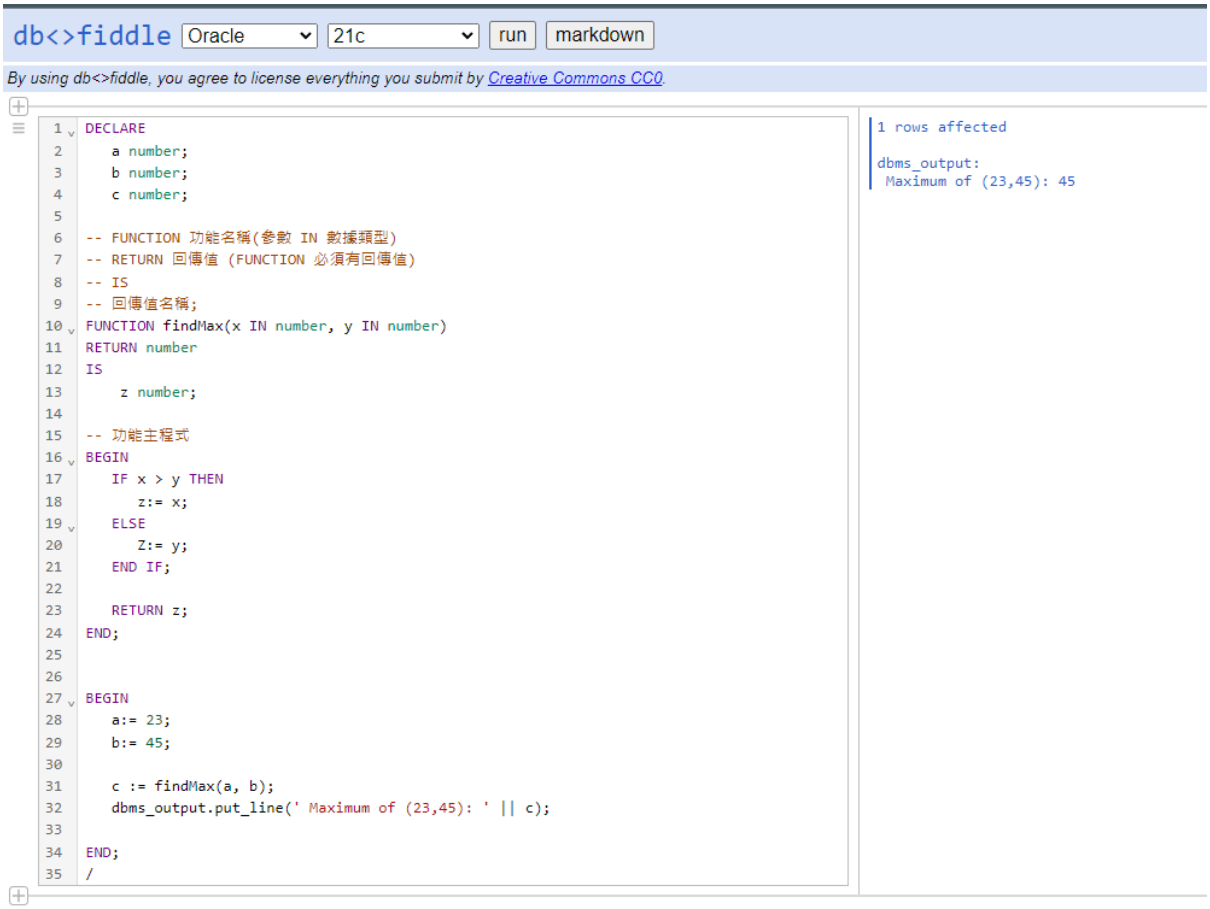
定義/呼叫函數

範例1.：

```
DECLARE
   a number;
   b number;
   c number;

-- FUNCTION 功能名稱(參數 IN 數據類型)
-- RETURN 回傳值 (FUNCTION 必須有回傳值)
-- IS
-- 回傳值名稱;
FUNCTION findMax(x IN number, y IN number)
RETURN number
IS
    z number;

-- 功能主程式
BEGIN
   IF x > y THEN
      z:= x;
   ELSE
      Z:= y;
   END IF;

   RETURN z;
END;


BEGIN
   a:= 23;
   b:= 45;

   c := findMax(a, b);
   dbms_output.put_line(' Maximum of (23,45): ' || c);

END;
/
```



==========

範例2.

邏輯概念：

一開始輸入num := 6，呼叫功能"fact(6)"，進入ELSE得到6 * fact(5)，此時執行fact(5)

得到5 * fact(4)，此時執行fact(4)，以此類推，執行到fact(1)，執行得到1 * fact(0)

fact(0) = 1，整趟執行結束

最後變成6 * 5 * 4 * 3 * 2 * 1 * 1(fact(0)) = 720

```
DECLARE
   num number;
   factorial number;
```

```
FUNCTION fact(x number)
RETURN number
IS
   f number;
BEGIN
   IF x=0 THEN
      f := 1;
   ELSE
      f := x * fact(x-1);
   END IF;
RETURN f;
END;


BEGIN
   num:= 6;
   factorial := fact(num);
   dbms_output.put_line(' Factorial '|| num || ' is ' || factorial);
END;
/
```

db<>fiddle Oracle 21c run markdown

By using db<>fiddle, you agree to license everything you submit by Creative Commons CC0.

```
DECLARE
   num number;
   factorial number;

FUNCTION fact(x number)
RETURN number
IS
   f number;
BEGIN
   IF x=0 THEN
      f := 1;
   ELSE
      f := x * fact(x-1);
   END IF;
RETURN f;
END;


BEGIN
   num:= 6;
   factorial := fact(num);
   dbms_output.put_line(' Factorial '|| num || ' is ' || factorial);
END;
/
```

```
1 rows affected

dbms_output:
 Factorial 6 is 720
```