

PL/SQL QUESTIONS

Write a PL/SQL block that takes an integer N as input and calculates the sum of the first N natural numbers. Use a FOR loop.

<pre>1 DECLARE 2 n1 NUMBER := 10; 3 sum NUMBER := 0; 4 5 BEGIN 6 FOR n1 IN 1..10 LOOP 7 sum := sum + n1; 8 END LOOP; 9 DBMS_OUTPUT.PUT_LINE(sum); 10 END;</pre>	Output: 55
---	---------------

Write a PL/SQL block to print all even numbers from 1 to 20 using a FOR loop.

<pre>1 DECLARE 2 n1 NUMBER := 10; 3 4 BEGIN 5 FOR n1 IN 1..20 LOOP 6 IF n1 MOD 2 = 0 THEN 7 DBMS_OUTPUT.PUT_LINE(n1); 8 END IF; 9 END LOOP; 10 END;</pre>	Output: 2 4 6 8 10 12 14 16 18 20
---	---

Write a PL/SQL block to calculate the factorial of a given number N using a WHILE loop.

<pre>1 DECLARE 2 n1 NUMBER := 5; 3 fact NUMBER := 1; 4 5 BEGIN 6 WHILE n1 >= 1 LOOP 7 fact := fact * n1; 8 n1 := n1 - 1; 9 END LOOP; 10 DBMS_OUTPUT.PUT_LINE(fact); 11 END;</pre>	Output: 120
---	----------------

Write a PL/SQL block that calculates the sum of digits of a given number N using a WHILE loop.

<pre>1 DECLARE 2 n1 NUMBER := 123; 3 sum NUMBER := 0; 4 rem NUMBER := 0; 5 BEGIN 6 WHILE n1 > 0 LOOP 7 rem := MOD(n1,10); 8 sum := sum + rem; 9 n1 := FLOOR(n1 / 10); 10 END LOOP; 11 DBMS_OUTPUT.PUT_LINE(sum); 12 END;</pre>	Output: 6
---	--------------

Write a PL/SQL block to reverse given number N using WHILE loop.

<pre>1 DECLARE 2 n1 NUMBER := 123; 3 rev NUMBER := 0; 4 rem NUMBER := 0; 5 BEGIN 6 WHILE n1 > 0 LOOP 7 rem := MOD(n1,10); 8 rev := rev * 10 + rem; 9 n1 := FLOOR(n1 / 10); 10 END LOOP; 11 DBMS_OUTPUT.PUT_LINE(rev); 12 END;</pre>	Output: 321
--	----------------

Write a PL/SQL block to print the multiplication table for a given number N up to 10.

<pre> 1 DECLARE 2 n1 NUMBER := 2; 3 i NUMBER := 1; 4 mult NUMBER; 5 BEGIN 6 WHILE i <= 10 LOOP 7 mult := n1 * i; 8 DBMS_OUTPUT.PUT_LINE(n1 ' x ' i ' = ' mult); 9 i := i + 1; 10 END LOOP; 11 END; 12 </pre>	<p>Output:</p> <pre> 2 x 1 = 2 2 x 2 = 4 2 x 3 = 6 2 x 4 = 8 2 x 5 = 10 2 x 6 = 12 2 x 7 = 14 2 x 8 = 16 2 x 9 = 18 2 x 10 = 20 </pre>
--	--

Write a PL/SQL block to check if a given number N is prime. Use a FOR loop.

<pre> 1 DECLARE 2 n1 NUMBER := 7; 3 i NUMBER := 1; 4 flag NUMBER := 0; 5 BEGIN 6 for i in 2..(n1/2) LOOP 7 if n1 mod i = 0 then 8 flag := 1; 9 EXIT; 10 end if; 11 END LOOP; 12 if flag = 0 then 13 DBMS_OUTPUT.PUT_LINE('Prime'); 14 else 15 DBMS_OUTPUT.PUT_LINE('Not prime'); 16 end if; 17 END; 18 </pre>	<p>Output:</p> <p>Prime</p>
--	-----------------------------

<pre> 1 DECLARE 2 n1 NUMBER := 27; 3 i NUMBER := 1; 4 flag NUMBER := 0; 5 BEGIN 6 for i in 2..(n1/2) LOOP 7 if n1 mod i = 0 then 8 flag := 1; 9 EXIT; 10 end if; 11 END LOOP; 12 if flag = 0 then 13 DBMS_OUTPUT.PUT_LINE('Prime'); 14 else 15 DBMS_OUTPUT.PUT_LINE('Not prime'); 16 end if; 17 END; 18 </pre>	<p>Output:</p> <p>Not prime</p>
---	---------------------------------

Write a PL/SQL block to print the Fibonacci series up to N terms. Use a WHILE loop.

<pre> 1 DECLARE 2 n1 NUMBER := 10; 3 t1 NUMBER := 1; 4 t2 NUMBER := 0; 5 t3 NUMBER := 0; 6 BEGIN 7 WHILE n1 > 0 LOOP 8 t3 := t1 + t2; 9 DBMS_OUTPUT.PUT_LINE(t3); 10 t2 := t1; 11 t1 := t3; 12 n1 := n1 - 1; 13 END LOOP; 14 END; 15 </pre>	<p>Output:</p> <pre> 1 2 3 5 8 13 21 34 55 89 </pre>
--	--

Write a PL/SQL block to calculate base^exponent using a FOR loop.

<pre> 1 DECLARE 2 base NUMBER := 2; 3 exp NUMBER := 10; 4 result NUMBER := 1; 5 BEGIN 6 FOR i IN 1..exp LOOP 7 result := result * base; 8 END LOOP; 9 DBMS_OUTPUT.PUT_LINE(result); 10 END; 11 </pre>	<p>Output:</p> <p>1024</p>
---	----------------------------