Explicit:

DECLARE

CURSOR prod\_cursor is select pid, pname, price from product;

v\_id product.pid%TYPE;

v\_name product.pname%TYPE;

v\_price product.price%TYPE;

BEGIN

OPEN prod\_cursor;

LOOP

FETCH prod\_cursor into v\_id, v\_name, v\_price;

EXIT when prod\_cursor%NOTFOUND;

DBMS\_OUTPUT.PUT\_LINE('ID: ' || v\_id || ', Name: ' || v\_name || ', Price: ' || v\_price);

END LOOP;

CLOSE prod\_cursor;

END;

/

Implicit:

declare

p\_price product.price%type;

begin

select price into p\_price from product where pid=101;

if sql%found then

dbms\_output.put\_line('Price fetched:'||p\_price);

else

dbms\_output.put\_line('Product not found.');

end if;

end;

/

Parameterized.sql

declare

cursor price\_cursor(p\_min\_price NUMBER) is select pid, pname, price from product where price > p\_min\_price;

v\_id product.pid%TYPE;

v\_name product.pname%TYPE;

v\_price product.price%TYPE;

v\_min\_price NUMBER;

begin

v\_min\_price := &min\_price;

open price\_cursor(v\_min\_price);

loop

fetch price\_cursor into v\_id, v\_name, v\_price;

exit when price\_cursor%NOTFOUND;

DBMS\_OUTPUT.PUT\_LINE('ID: ' || v\_id || ', Name: ' || v\_name || ', Price: ' || v\_price);

end loop;

close price\_cursor;

end;

/

Update.sql

DECLARE

v\_increase NUMBER := 2.5;

BEGIN

FOR rec IN (SELECT pid, price FROM product WHERE price < 20000) LOOP

UPDATE product

SET price = rec.price \* v\_increase

WHERE pid = rec.pid;

END LOOP;

COMMIT;

DBMS\_OUTPUT.PUT\_LINE('Prices updated.');

END;

/

Output:

SQL\*Plus: Release 21.0.0.0.0 - Production on Sun Sep 7 15:24:15 2025

Version 21.3.0.0.0

Copyright (c) 1982, 2021, Oracle. All rights reserved.

Enter user-name: system

Enter password:

Last Successful login time: Sun Sep 07 2025 14:54:23 +05:30

Connected to:

Oracle Database 21c Express Edition Release 21.0.0.0.0 - Production

Version 21.3.0.0.0

SQL> insert into product values(104,'mobile',30000);

1 row created.

SQL> insert into product values(108,' laptop ',70000);

1 row created.

SQL> commit;

Commit complete.

SQL> select \* from product

2 ;

PID PNAME PRICE

---------- -------------------------------------------------- ----------

104 mobile 30000

108 laptop 70000

SQL> set serveroutput on

SQL> @ " C:\Users\Raj\OneDrive\Desktop\cursor\output.docx".sql

Price fetched:70000

PL/SQL procedure successfully completed.

SQL> @"C:\Users\Raj\OneDrive\Desktop\cursor\output.docx".sql

ID: 104, Name: mobile, Price: 30000

PL/SQL procedure successfully completed.

SQL> insert into product values(109,'headphones',5000);

SQL> select \* from product;

PID PNAME PRICE

---------- -------------------------------------------------- ----------

108 laptop 70000

104 mobile 30000

109 headphones 5000

SQL> set serveroutput on

SQL> @ "C:\Users\Raj\OneDrive\Desktop\cursor\output.docx".sql

Enter value for enter\_price: 5000

old 14: v\_input := &enter\_price;

new 14: v\_input := 5000;

Starting to fetch products above price 5000...

pid: 108, name: laptop, price: 70000

pid: 104, name: mobile, price: 30000

Done fetching products.

PL/SQL procedure successfully completed.

SQL> set serveroutput on

SQL> @”C:\Users\Raj\OneDrive\Desktop\cursor\output.docx".sql

Prices updated.

PL/SQL procedure successfully completed.

SQL> select \* from product;

PID PNAME PRICE

---------- -------------------------------------------------- ----------

108 laptop 70000

104 mobile 30000

109 headphone 7500