

DUMPSBOSS.

Oracle Database 12c: Advanced PL/SQL

Oracle 1z0-148

Version Demo

Total Demo Questions: 10

Total Premium Questions: 106

Buy Premium PDF

<https://dumpsboss.co>

support@dumpsboss.co

support@dumpsboss.co
dumpsboss.co

QUESTION NO: 1

The user MY_USER has been assigned the roles CONNECT, RESOURCE, and DBA.

Which statement generates the DDL for these privileges?

- A. SELECT DBMS_METADATA.GET_GRANTED_DDL ('SYSTEM_GRANT', 'my_user') FROM dual;
- B. SELECT DBMS_METADATA.GET_GRANTED_DDL ('ROLE_GRANT', 'my_user') FROM dual;
- C. SELECT DBMS_METADATA.GET_GRANTED_DDL ('OBJECT_GRANT', 'my_user') FROM dual;
- D. SELECT DBMS_METADATA.GET_DDL ('USER', 'my_user') FROM dual;

ANSWER: D

Explanation:

Reference: <https://www.databasedevelop.com/article/10596168/CONNECT+ROLE+and+CREATE+SESSION>

QUESTION NO: 2

Identify the two correct scenarios where a function can be optimized using the function result cache feature. (Choose two.)

- A. A function which inserts multiple records into a DEPARTMENTS table as part of one-time data setup for an HR application.
- B. A function which accesses multiple tables and calculates the commission to be given to a sales representative based on the number of products sold by that representative.
- C. A function which deletes all the records from an EMPLOYEES_AUDIT table based on their LOG_DATE.
- D. A function which updates the SALARY of all the employees in an EMPLOYEES table by a fixed percentage based on their DESIGNATION.
- E. A function which calculates the factorial of a given number without accessing any table.

ANSWER: D E

QUESTION NO: 3

Examine this function:

```
CREATE FUNCTION remap_schema RETURN CLOB IS
  h NUMBER;
  th NUMBER;
  doc CLOB;
BEGIN
  h := DBMS_METADATA.OPEN ('TABLE');
  DBMS_METADATA.SET_FILTER (h, 'SCHEMA', 'SCOTT');
  DBMS_METADATA.SET_FILTER (h, 'NAME', 'EMP');
  th := DBMS_METADATA.ADD_TRANSFORM (h, 'MODIFY');
  DBMS_METADATA.SET_REMAP_PARAM (th, 'REMAP_SCHEMA', 'SCOTT', NULL);
  DBMS_METADATA.SET_REMAP_PARAM (th, 'REMAP_TABLESPACE', 'USERS', 'SYSAUX');
  th := DBMS_METADATA.ADD_TRANSFORM (h, 'DDL');
  DBMS_METADATA.SET_TRANSFORM_PARAM (th, 'SEGMENT_ATTRIBUTES', FALSE);
  doc := DBMS_METADATA.FETCH_CLOB (h);
  DBMS_METADATA.CLOSE (h);
  RETURN doc;
END remap_schema;
```

Execute the query:

```
SELECT remap_schema FROM dual;
```

Which is the correct output from the query?

- A.** CREATE TABLE "EMP" ("EMPNO" NUMBER (4,0), "ENAME" VARCHAR2 (10), "JOB" VARCHAR2 (9), "MGR" NUMBER (4,0), "HIREDATE" DATE, "SAL" NUMBER (7,2), "COMM" NUMBER (7,2), "DEPTNO" NUMBER (2,0), CONSTRAINT "PK_EMP" PRIMARY KEY ("EMPNO") USING INDEX PCTFREE 10 INITRANS 2 MAXTRANS 255 STORAGE (INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645 PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT FLASH_CACHE DEFAULT CELL_FLASH_CACHE DEFAULT) TABLESPACE "USERS" ENABLE, CONSTRAINT "FK_DEPTNO" FOREIGN KEY ("DEPTNO") REFERENCES "DEPT" ("DEPTNO") ENABLE) SEGMENT CREATION IMMEDIATE PCTFREE 10 PCTUSED 40 INITRANS 1 MAXTRANS 255 NOCOMPRESS LOGGING STORAGE (INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645 PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT FLASH_CACHE DEFAULT CELL_FLASH_CACHE DEFAULT) TABLESPACE "USERS"
- B.** CREATE TABLE "EMP" ("EMPNO" NUMBER (4, 0), "ENAME" VARCHAR2 (10), "JOB" VARCHAR2 (9), "MGR" NUMBER (4, 0), "HIREDATE" DATE, "SAL" NUMBER (7, 2), "COMM" NUMBER (7, 2), "DEPTNO" NUMBER (2, 0), CONSTRAINT "PK_EMP" PRIMARY KEY ("EMPNO") USING INDEX ENABLE, CONSTRAINT "FK_DEPTNO" FOREIGN KEY ("DEPTNO") REFERENCES "DEPT" ("DEPTNO") ENABLE)
- C.** CREATE TABLE "SCOTT"."EMP" ("EMPNO" NUMBER (4, 0), "ENAME" VARCHAR2 (10), "JOB" VARCHAR2 (9), "MGR" NUMBER (4, 0), "HIREDATE" DATE, "SAL" NUMBER (7, 2), "COMM" NUMBER (7, 2), "DEPTNO" NUMBER (2, 0), CONSTRAINT "PK_EMP" PRIMARY KEY ("EMPNO") USING INDEX ENABLE, CONSTRAINT "FK_DEPTNO" FOREIGN KEY ("DEPTNO") REFERENCES "DEPT" ("DEPTNO") ENABLE)

```
D. CREATE TABLE "EMP" ("EMPNO" NUMBER (4,0), "ENAME" VARCHAR2 (10), "JOB" VARCHAR2 (9), "MGR" NUMBER (4,0), "HIREDATE" DATE, "SAL" NUMBER (7, 2), "COMM" NUMBER (7, 2), "DEPTNO" NUMBER (2,0), CONSTRAINT "PK_EMP" PRIMARY KEY ("EMPNO") USING INDEX PCTFREE 10 INITRANS 2 MAXTRANS 255 STORAGE (INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645 PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT FLASH_CACHE DEFAULT CELL_FLASH_CACHE DEFAULT) TABLESPACE "SYSAUX" ENABLE, CONSTRAINT "FK_DEPTNO" FOREIGN KEY ("DEPTNO") REFERENCES "DEPT" ("DEPTNO") ENABLE ) SEGMENT CREATION IMMEDIATE PCTFREE 10 PCTUSED 40 INITRANS 1 MAXTRANS 255 NOCOMPRESS LOGGING STORAGE (INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645 PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT FLASH_CACHE DEFAULT CELL_FLASH_CACHE DEFAULT) TABLESPACE "SYSAUX"
```

ANSWER: B

QUESTION NO: 4

Your company's executive team voted to give a one-time special bonus, the amount of which is to be based on number of years of service. The bonus table will be updated using a procedure called `update_bonus()`.

Examine this PL/SQL block which shows the logic to calculate the bonus:

```
DECLARE
    emp_rec employees%ROWTYPE;
    bonus1 INTEGER;
    bonus2 INTEGER;
    service_days INTEGER;
    CURSOR c1 IS SELECT * FROM employees;
BEGIN
    OPEN c1;
    LOOP
        FETCH c1 INTO emp_rec;
        EXIT WHEN c1%NOTFOUND;
        service_days := sysdate - emp_rec.hire_date;
        bonus1 := 10000;
        bonus2 := 5000;
        IF (sysdate - emp_rec.hire_date) > 10*365 THEN
            update_bonus (emp_rec.employee_id, bonus1);
        ELSIF (sysdate - emp_rec.hire_date) > 5*365 THEN
            update_bonus (emp_rec.employee_id, bonus2);
        END IF;
    END LOOP;
    CLOSE c1;
END;
```

Consider these possible optimization actions:

1. Remove the service_days calculation statement.
2. Move bonus1 and bonus2 initializations out of the loop.
3. Inline the update_bonus() subprogram.
4. Call update_bonus(), passing the literal bonus amount instead of the bonus1 or bonus2 variables.

Which set of optimizations will be performed if this block is compiled with PLSQL_OPTIMIZE_LEVEL = 2?

- A. 3 and 4 only
- B. 1, 2, 3, and 4
- C. 1, 2, and 4 only
- D. 1 and 2 only
- E. 1 and 3 only

ANSWER: D

QUESTION NO: 5

Examine this code (all compiled into schema HR):

```
CREATE PROCEDURE format_call_stack_12c IS
BEGIN
  DBMS_OUTPUT.PUT_LINE ('LexDepth   Dynamic Depth   Name');
  DBMS_OUTPUT.PUT_LINE ('-----   -----   -----');
  FOR the_depth IN REVERSE 1 .. UTL_CALL_STACK.DYNAMIC_DEPTH LOOP
    DBMS_OUTPUT.PUT_LINE (RPAD (UTL_CALL_STACK.LEXICAL_DEPTH
(the_depth), 10)||
  RPAD (the_depth, 15)||
  UTL_CALL_STACK.CONCATENATE_SUBPROGRAM
(UTL_CALL_STACK.SUBPROGRAM (the_depth) ));
  END LOOP;
END;
/
CREATE PACKAGE test_pkg IS
  PROCEDURE do_stuff;
END;
/
CREATE PACKAGE BODY test_pkg IS
  PROCEDURE do_stuff IS
    PROCEDURE np1 IS
      PROCEDURE NP2 IS
        PROCEDURE np3 IS
          BEGIN
            format-call_stack_12c;
          END;
        BEGIN;
        np3;
      END;
    BEGIN;
    np2;
  END;
  BEGIN
    np1;
  END;
END;
/
```

Execute this code:

```
SET SERVEROUTPUT ON;
```

```
EXEC test_pkg.do_stuff
```

What is the correct output?

A. LexDepth Dynamic Depth Name

```
-----
0 6 __anonymous_block
1 5 package body HR.TEST_PKG
2 4 package body HR.TEST_PKG
```

```
3 3 package body HR.TEST_PKG
4 2 package body HR.TEST_PKG
0 1 procedure HR. FORMAT_CALL_STACK_12C
```

B. LexDepth Dynamic Depth Name

```
-----
0 6 __anonymous_block
1 5 TEST_PKG.DO_STUFF
2 4 TEST_PKG.DO_STUFF.NP1
2 3 TEST_PKG.DO_STUFF.NP1.NP2
2 2 TEST_PKG.DO_STUFF.NP1.NP2.NP3
0 1 FORMAT_CALL_STACK_12C
```

C. LexDepth Dynamic Depth Name

```
-----
0 6 __anonymous_block
1 5 TEST_PKG.DO_STUFF
2 4 TEST_PKG.DO_STUFF.NP1
3 3 TEST_PKG.DO_STUFF.NP1.NP2
4 2 TEST_PKG.DO_STUFF.NP1.NP2.NP3
0 1 FORMAT_CALL_STACK_12C
```

D. LexDepth Dynamic Depth Name

```
----- 0 6 __anonymous_block
1 5 DO_STUFF
2 4 NP1
3 3 NP2
4 2 NP3
0 1 FORMAT_CALL_STACK_12C
```

E. LexDepth Dynamic Depth Name

```
-----
0 6 __anonymous_block
1 5 package body HR.TEST_PKG
2 4 package body HR.TEST_PKG
2 3 package body HR.TEST_PKG
2 2 package body HR.TEST_PKG
0 1 procedure HR. FORMAT_CALL_STACK_12C
```

ANSWER: E

QUESTION NO: 6

This result cache is enabled for the database instance.

Examine this code for a PL/SQL function:

```
CREATE FUNCTION get_hire_date (emp_id NUMBER) RETURN VARCHAR2
  RESULT_CACHE
IS
  date_hired DATE;
BEGIN
  SELECT hire_date INTO date_hired
  FROM HR.EMPLOYEES
  WHERE EMPLOYEE_ID = emp_id;
  RETURN TO_CHAR (date_hired);
END;
```

Which two actions would ensure that the same result will be consistently returned for any session when the same input value is passed to the function? (Choose two.)

- A. Add a parameter, fmt, and change the RETURN statement to: RETURN TO_CHAR (date_hired, fmt);
- B. Set the RESULT_CACHE_MODE parameter to FORCE.
- C. Increase the value for the RESULT_CACHE_MAX_SIZE parameter.
- D. Change the return type of GET_HIRE_DATE to DATE and have each session invoke the TO_CHAR function.
- E. Set the RESULT_CACHE_MAX_RESULT parameter to 0.

ANSWER: B D

QUESTION NO: 7

You execute this PL/SQL block:

```
SQL> DECLARE
2     TYPE varchar_type1 IS VARRAY(3) OF VARCHAR2(15);
3     TYPE varchar_type2 IS VARRAY(3) OF VARCHAR2(15);
4     TYPE varchar_type3 IS VARRAY(3) OF VARCHAR2(15);
5     TYPE nested_typ IS TABLE OF varchar_type3;
6     n_table1 nested_typ := varchar_type3('AB1', 'AB2', 'AB3');
7     list_A varchar_type1 := varchar_type1('Seattle', 'Tokyo', 'Paris');
8     list_B varchar_type1;
9     list_C varchar_type2;
10 BEGIN
11   list_B := list_A;
12   list_C := list_A;
13 END;
14 /
```

What will be the outcome?

- A. It will fail to compile because of errors at lines 11 and 12.

- B. It will fail to compile because of errors at lines 6 and 12.
- C. It will fail to compile because of an error at line 7.
- D. It will compile and execute successfully.
- E. It will fail to compile because of errors at lines 5 and 6.

ANSWER: B

QUESTION NO: 8

Which two statements are true regarding edition-based redefinition (EBR)? (Choose two.)

- A. There is no default edition defined in the database.
- B. EBR does not let you upgrade the database components of an application while in use.
- C. You never use EBR to copy the database objects and redefine the copied objects in isolation.
- D. Editions are non-schema objects.
- E. When you change an editioned object, all of its dependents remain valid.
- F. Tables are not editionable objects.

ANSWER: E F

Explanation:

Reference: https://docs.oracle.com/cd/E11882_01/appdev.112/e41502/adfns_editions.htm#BABEHGAF

QUESTION NO: 9

Which two statements are true about the DBMS_LOB package? (Choose two.)

- A. DBMS_LOB.COMPARE can compare parts of two LOBs.
- B. DBMS_LOB.COMPARE returns the size difference of the compared LOBs.
- C. DBMS_LOB.COMPARE is overloaded and can compare CLOBs with BLOBs.
- D. If the destination LOB is a temporary LOB, the row must be locked before calling DBMS_LOB.CONVERTTOBLOB.
- E. Before calling DBMS_LOB.CONVERTTOBLOB, both the source and destination LOB instances must exist.

ANSWER: A E

Explanation:

Reference: https://docs.oracle.com/cd/B19306_01/appdev.102/b14258/d_lob.htm#BABDDFDH

QUESTION NO: 10

The STUDENTS table with column LAST_NAME of data type VARCHAR2 exists in your database schema. Examine this PL/SQL block:

```
DECLARE
    CURSOR name_cur IS
        SELECT last_name FROM students WHERE last_name LIKE 'A%';
    TYPE l_name_type IS VARRAY(25) OF students.last_name%TYPE;
    names_varray l_name_type;
    v_index INTEGER := 0;
BEGIN
    FOR name_rec IN name_cur LOOP
        v_index := v_index + 1;
        names_varray(v_index) := name_rec.last_name;
        DBMS_OUTPUT.PUT_LINE (names_varray(v_index));
    END LOOP;
END;
```

Which two actions must you perform for this PL/SQL block to execute successfully? (Choose two.)

- A. Replace the FOR loop with FOR name_rec IN names_varray.FIRST .. names_varray.LAST LOOP.
- B. Replace the L_NAME_TYPE declaration with TYPE l_name_type IS VARRAY (25) OF SYS_REFCURSOR;
- C. Add name_rec name_cur%ROWTYPE; at the end of the DECLARE section.
- D. Replace the NAMES_VARRAY declaration with names_varray l_name_type := l_name_type ();
- E. Replace the NAMES_VARRAY declaration with names_varray l_name_type := null;
- F. Add names_varray.EXTEND after the FOR ...LOOP statement.

ANSWER: E F