1Z0-144 Q&As

Oracle Database 11g: Program with PL/SQL

Pass Oracle 1Z0-144 Exam with 100% Guarantee

Free Download Real Questions & Answers **PDF** and **VCE** file from:

https://www.certbus.com/1z0-144.html

100% Passing Guarantee
100% Money Back Assurance

Following Questions and Answers are all new published by Oracle Official Exam Center

QUESTION 1

View the Exhibit and examine the structure of the EMPLOYEES table.

Name	Null?	Type
EMPLOYEE_ID	NOT NULL	NUMBER(6)
FIRST_NAME		VARCHAR2(20)
LAST_NAME	NOT NULL	VARCHAR2(25)
HIRE_DATE	NOT NULL	DATE
JOB_ID	NOT NULL	VARCHAR2(10)
SALARY		NUMBER(8,2)
COMISSION_PCT		NUMBER(2,2)
MANAGER_ID		NUMBER(6)
DEPARTMENT_ID		NUMBER(4)

Execute the following block of code:

SQL > SET SERVEROUTPUT ON

```
SQL>DECLARE
  2 v_sum_sal NUMBER;
  3 department_id employees.department_id%TYPE := 60;
  4 BEGIN
  5       SELECT SUM(salary)
  6            INTO v_sum_sal FROM employees
  7            WHERE department_id = department_id;
  8 DBMS_OUTPUT.PUT_LINE ('The sum of salary is' || v_sum_sal);
  9* END;
    /
```

What is the outcome?

A. It gives an error because the variable name and column name are the same in the WHERE clause of the SELECT statement.

B. It executes successfully and displays the sum of salaries in department 60.

C. It gives an error because group functions cannot be used in anonymous blocks.

D. It executes successfully and displays the incorrect sum of salaries in department 60.

Correct Answer: D

QUESTION 2

Which two statements are true about the %ROWTYPE attribute? (Choose two.)

A. It is used to declare a record that can hold multiple rows of a table.

B. The attributes of fields in the record with the %ROWTYPE attribute can be modified manually.

C. The attributes of fields in the record take their names and data types from the columns of the table, view, cursor, or cursor variable.

D. It ensures that the data types of the variables that are declared with the %ROWTYPE attribute change dynamically when the underlying table is altered.

Correct Answer: CD

QUESTION 3

View the Exhibit and examine the structure of the customer table.

Name	Null?	Type
CUST_ID	NOT NULL	NUMBER
CUST_LAST_NAME	NOT NULL	VARCHAR2(40)
CUST_CITY	NOT NULL	VARCHAR2(30)
CUST_CREDIT_LIMIT		NUMBER
CUST_CATEGORY		VARCHAR2(20)

You create the following trigger to ensure that customers belonging to category "A" or "B" in the CUSTOMER table can have a credit limit of more than 8000.

```
SQL>CREATE OR REPLACE TRIGGER restrict_credit_limit
         BEFORE INSERT OR UPDATE ON customer
         FOR EACH ROW
        BEGIN
          IF (:NEW.cust_category NOT IN ('A', 'B'))
              AND :NEW.cust_credit_limit > 8000 THEN
                DBMS_OUTPUT.PUT_LINE ('Credit Limit cannot be greater
                                       than 8000 for this category');
          END IF;
        END;
/
```

You execute the following UPDATE command for CUST_ID 101 existing in the CUSTOMER table.

```
SQL> UPDATE customer SET cust_category = 'C', cust_credit_limit = 9000
         WHERE cust_id = 101;
```

What is the outcome?

A. The trigger is fired, a message is displayed, and the update is successful.

B. The trigger is fired and a message is displayed, but the update is rolled back.

C. The trigger is not fired because the WHEN clause should be used to specify the condition; however, the update is successful.

D. The trigger is not fired because column names must be specified with the UPDATE event to identify which columns must be changed to cause the trigger to fire; however, the update is successful.

Correct Answer: A

QUESTION 4

Examine the following PL/SQL code: The serveroutput is on for the session. Which statement is true about the execution of the code?

```
DECLARE
   emp_rec employees%ROWTYPE;
BEGIN
   SELECT * INTO emp_rec FROM employees WHERE employee_id=123;
   IF SQL%NOTFOUND THEN
     DBMS_OUTPUT.PUT_LINE('Record Not found');
   ELSE
     DBMS_OUTPUT.PUT_LINE('Employee ' ||emp_rec.first_name||''||
                  emp_rec.last_name||' Salary is '||emp_rec.salary);
   END IF;
END;
/
```

A. It displays NULL if no employee with employee_id 123 exists.

B. It produces the ORA-01403: no data found error if no employee with employee_id 123 exists.

C. It displays an error because the SELECT * INTO clause cannot be used to populate the PL/SQL record type.

D. The code executes successfully even if no employee with employee_id 123 exists and displays Record Not Found.

Correct Answer: B

QUESTION 5

Which two guidelines are recommended by Oracle to reduce invalidation of dependent objects? (Choose two.)

A. Reference tables indirectly by using views.

B. Reference tables directly and avoid using views.

C. When adding new items to a package, add them to the end of the package.

D. When adding new items to a package, add them to the beginning of the package.

Correct Answer: AC

QUESTION 6

Examine the following code: What is the outcome?

```
SQL> SET SERVEROUTPUT ON
SQL > VARIABLE n1 NUMBER
SQL > VARIABLE n2 NUMBER
SQL>CREATE OR REPLACE PROCEDURE proc1
        (:n1 IN OUT NUMBER, :n2 IN OUT NUMBER) IS
    BEGIN
      :n1 := 20;
      DBMS_OUTPUT.put_line(:n1);
      :n2 := 30;
      DBMS_OUTPUT.put_line(:n2);

    END;
   /
```

A. The procedure is created successfully and displays the values 20 and 30 when it is called.

B. The procedure gives errors because the parameters should be in OUT mode.

C. The procedure gives errors because the host variables cannot be referenced anywhere in the definition of a PL/SQL stored procedure.

D. The procedure is created successfully, but does not display any values when it is called because the host variables cannot be displayed inside the procedure.

Correct Answer: C

QUESTION 7

View Exhibit 1 and examine the structure of the EMPLOYEES table.

Name	Null?	Type
EMPLOYEE_ID	NOT NULL	NUMBER(6)
FIRST_NAME		VARCHAR2(20)
LAST_NAME	NOT NULL	VARCHAR2(25)
HIRE_DATE	NOT NULL	DATE
JOB_ID	NOT NULL	VARCHAR2(10)
SALARY		NUMBER(8,2)
COMISSION_PCT		NUMBER(2,2)
MANAGER_ID		NUMBER(6)
DEPARTMENT_ID		NUMBER(4)

View Exhibit 2 and examine the code.

```
DECLARE
   emp_num NUMBER(6) := 120;
   sal NUMBER;
   FUNCTION increase (emp_num NUMBER)
     RETURN number IS
     inc_amt NUMBER;
     BEGIN
      SELECT salary INTO sal FROM employees WHERE employee_id = emp_num;
      inc_amt := sal * .10;
      RETURN inc_amt;
     END;
   PROCEDURE raise_salary (emp_id NUMBER) IS
    amt NUMBER;
    BEGIN
        amt := increase (emp_num);
        UPDATE employees SET salary = salary + amt
                        WHERE employee_id = emp_id;
    END raise_salary;
 BEGIN
   raise_salary(emp_num);
   COMMIT;
 END;
/
```

What would be the outcome when the code is executed?

A. It executes successfully.

B. It gives an error because the SAL variable is not visible in the INCREASE function.

C. It gives an error because the INCREASE function cannot be called from the RAISE_SALARY procedure.

D. It gives an error because the INCREASE function and the RAISE_SALARY procedure should be declared at the beginning of the DECLARE section before all the other declarations.

Correct Answer: A

QUESTION 8

You want to create a trigger that fires whenever rows are deleted from the customer table and that displays the number of rows remaining in the table.

Which two statements are correct about the trigger to be created for the above requirement? (Choose two.)

A. It should be an after trigger.

B. It should be a before trigger.

C. It should be a row-level trigger.

D. It should be a statement-level trigger.

E. It can be a before or an after trigger.

Correct Answer: AD

QUESTION 9

Which two statements are true about the usage of the cursor for loops? (Choose two.)

A. The cursor needs to be closed after the iteration is complete.

B. The implicit open, fetch, exit, and close of the cursor happen.

C. The record type must be explicitly declared to control the loop.

D. The PL/SQL creates a record variable with the fields corresponding to the columns of the cursor result set.

Correct Answer: BD

QUESTION 10

Identify the scenario in which you would use the CURRENT OF clause for an UPDATE or DELETE statement to rows fetched from a cursor.

A. when you want to lock the rows fetched by the cursor

B. when you want to update or delete the result set without affecting the rows in the table

C. when you want the database not to wait if the requested rows are locked by another user

D. when you want to ensure that the current rows fetched by the cursor are updated or deleted

Correct Answer: D

References: https://www.techonthenet.com/oracle/cursors/current_of.php

QUESTION 11

View the Exhibit and examine the structure of the SALGRADE table.

```
SQL> desc salgrade
Name          Null?      Type
------------ --------- -------
GRADE         NOT NULL   NUMBER
LOSAL                    NUMBER
HISAL                    NUMBER
```

Examine the following code:

```
SQL>VARIABLE min_sal NUMBER
SQL>VARIABLE max_sal NUMBER

SQL>CREATE OR REPLACE FUNCTION sal ok(salary NUMBER, jobgrade NUMBER)
     RETURN BOOLEAN AS
    BEGIN
     SELECT losal, hisal INTO :min_sal, :max_sal FROM salgrade
     WHERE grade = jobgrade;
     RETURN (salary >= min_sal) AND (salary <= max_sal);
    END sal_ok;
/
```

What is the outcome?

A. It is created successfully.

B. It gives an error because the return clause condition is invalid.

C. It gives an error because the usage of the host variables is invalid.

D. It gives an error because the data type of the return clause is invalid.

Correct Answer: C

QUESTION 12

View the Exhibit and examine the structure of the employees table.

```
Name                 Null?     Type
-------------------- --------- --------
EMPLOYEE_ID          NOT NULL  NUMBER(6)
FIRST_NAME                     VARCHAR2(20)
LAST_NAME            NOT NULL  VARCHAR2(25)
HIRE_DATE            NOT NULL  DATE
JOB_ID               NOT NULL  VARCHAR2(10)
SALARY                         NUMBER(8,2)
COMMISSION_PCT                 NUMBER(2,2)
MANAGER_ID                     NUMBER(6)
DEPARTMENT_ID                  NUMBER(4)
```

Examine the following block of code:

```
SQL>DECLARE
2       v_sal NUMBER;
3       v_name VARCHAR2(30);
4       v_tenure NUMBER;
5       v_hire_date DATE;
6     BEGIN
7       SELECT AVG(salary)INTO v_sal FROM employees;
8       SELECT hire_date,DECODE(salary, v_sal, last_name,'NA')
9                       INTO v_hire_date,v_name
10                      FROM employees
11                      WHERE employee_id = 195;
12      v_tenure := MONTHS_BETWEEN (CURRENT_DATE, v_hire_date);
13    END;
/
```

What is the outcome when the above code is executed?

A. It executes successfully.

B. It gives an error because decode cannot be used in a PL/SQL block.

C. It gives an error because the AVG function cannot be used in a PL/SQL block

D. It gives an error because the MONTHS_BETWEEN function cannot be used in a PL/SQL block.

E. It gives an error because both the AVG and decode functions cannot be used in a PL/SQL block.

Correct Answer: A

QUESTION 13

View the Exhibits and examine the structure of the EMPLOYEES, DEPARTMENTS AND EMP_BY_DEPT tables.

EMPLOYEES

Name	Null?	Type
EMPLOYEE_ID	NOT NULL	NUMBER(6)
FIRST_NAME		VARCHAR2(20)
LAST_NAME	NOT NULL	VARCHAR2(25)
HIRE_DATE	NOT NULL	DATE
JOB_ID	NOT NULL	VARCHAR2(10)
SALARY		NUMBER(8,2)
COMISSION_PCT		NUMBER(2,2)
MANAGER_ID		NUMBER(6)
DEPARTMENT_ID		NUMBER(4)

DEPARTMENTS

Name	Null?	Type
DEPARTMENT_ID	NOT NULL	NUMBER(4)
DEPARTMENT_NAME	NOT NULL	VARCHAR2(30)
MANAGER_ID		NUMBER(6)
LOCATION_ID		NUMBER(4)

EMP_BY_DEPT

```
Name                  Null?      Type
--------------------- ---------- --------
EMPLOYEE_ID           NOT NULL   NUMBER(6)
DEPARTMENT_ID         NOT NULL   NUMBER(4)
```

Examine the following code:

```
DECLARE
  TYPE dept_tab IS TABLE OF departments.department_id%TYPE;
  deptnums dept_tab;
BEGIN
  SELECT department_id BULK COLLECT INTO deptnums FROM departments;
  FORALL i IN 1..deptnums.COUNT
     INSERT INTO emp_by_dept
        SELECT employee_id, department_id FROM employees
          WHERE department_id = deptnums(i);
     DBMS_OUTPUT.PUT_LINE(SQL%BULK_ROWCOUNT(deptnums.COUNT));
     DBMS_OUTPUT.PUT_LINE(SQL% ROWCOUNT);
END;
/
```

What is the outcome on execution of the above code?

A. It executes successfully but the output statements show different values.

B. It executes successfully and both output statements show the same values.

C. It gives an error because the SQL%ROWCOUNT attribute cannot be used with BULK COLLECT.

D. It gives an error because the INSERT SELECT construct cannot be used with the FORALL

Correct Answer: A

QUESTION 14

User SCOTT has been granted CREATE ANY TRIGGER and ALTER ANY TABLE by the DBA. HR is an

existing schema in the database.

SCOTT creates the following trigger:

CREATE OR REPLACE TRIGGER drop_trigger

BEFORE DROP ON hr.SCHEMA

BEGIN

RAISE_APPLICATION_ERROR (-20000, `Cannot drop object\\');

END;

SCOTT does not grant the EXECUTE privilege on this trigger to any other users.

For which user(s) would this trigger fire by default when they drop an object in the HR schema?

A. Only HR

B. SCOTT and HR

C. Only SCOTT

D. SCOTT, HR, and SYS

Correct Answer: A

QUESTION 15

View Exhibit 1 and examine the structure of the EMP and DEPT tables.

```
SQL> DESC emp
 Name                      Null?     Type
 ------------------------- --------- --------
 EMPNO                     NOT NULL  NUMBER(4)
 ENAME                               VARCHAR2(10)
 JOB                                 VARCHAR2(9)
 MGR                                 NUMBER(4)
 HIREDATE                            DATE
 SAL                                 NUMBER(7,2)
 COMM                                NUMBER(7,2)
 DEPTNO                              NUMBER (2)

SQL> DESC dept
 Name                      Null?     Type
 ------------------------- --------- --------
 DEPTNO                    NOT NULL  NUMBER(2)
 DNAME                               VARCHAR2(14)
 LOC                                 VARCHAR2(13)
```

View Exhibit 2 and examine the trigger code that is defined on the DEPT table to enforce the UPDATE and DELETE RESTRICT referential actions on the primary key of the DEPT table.

```
CREATE OR REPLACE TRIGGER Dept_restrict
   BEFORE DELETE OR UPDATE OF Deptno ON dept
   DECLARE
     dummy INTEGER;
     employees_present       EXCEPTION;
     employees_not_present   EXCEPTION;
     CURSOR Dummy_cursor (dn NUMBER) IS
        SELECT deptno FROM emp WHERE deptno = dn;
     BEGIN
        OPEN Dummy_cursor (:OLD.Deptno);
        FETCH Dummy_cursor INTO Dummy;
        IF Dummy_cursor%FOUND THEN
              RAISE employees_present;
        ELSE
              RAISE employees_not_present;
        END IF;
        CLOSE Dummy_cursor;
        EXCEPTION
        WHEN employees_present THEN
             CLOSE Dummy_cursor;
             RAISE_APPLICATION_ERROR(-20001, 'Employees Present in'
                                     || 'Department' || TO_CHAR(:OLD.DEPTNO));
        WHEN employees_not_present THEN
             CLOSE Dummy_cursor;
END;
/
```

What is the outcome on compilation?

A. It compiles and executes successfully.

B. It gives an error on compilation because it is not a row-level trigger.

C. It gives an error on compilation because the EXCEPTION section is used in the trigger.

D. It compiles successfully but gives an error on execution because it is not a row-level trigger.

Correct Answer: B