

# T estpassportQ&A



H i g h e r   Q u a l i t y

B e t t e r   S e r v i c e !

We offer free update service for one year  
[Http://www.testpassport.com](http://www.testpassport.com)

**Exam : 1Z0-007**

**Title : Introduction to Oracle9i:  
SQL**

**Version : DEMO**

1. What does the FORCE option for creating a view do?
  - A.create a view with constraints
  - B.create a view even if the underlying parent table has constraints
  - C.create a view in another schema even if you don't have privileges
  - D.create a view regardless of whether or not the base tables exist

**Answer:** D

2. What are two reasons to create synonyms? (Choose two.)

- A.You have too many tables.
- B.Your tables are too long.
- C.Your tables have difficult names.
- D.You want to work on your own tables.
- E.You want to use another schema's tables.
- F.You have too many columns in your tables.

**Answer:** CE

3. The STUDENT\_GRADES table has these columns:

STUDENT\_ID NUMBER(12)

SEMESTER\_END DATE

GPA NUMBER(4,3)

The registrar requested a report listing the students' grade point averages (GPA) sorted from highest grade point average to lowest.

Which statement produces a report that displays the student ID and GPA in the sorted order requested by the registrar?

A.SELECT student\_id, gpa

FROM student\_grades

ORDER BY gpa ASC;

B.SELECT student\_id, gpa

FROM student\_grades

SORT ORDER BY gpa ASC;

C.SELECT student\_id, gpa

FROM student\_grades

SORT ORDER BY gpa;

D.SELECT student\_id, gpa

FROM student\_grades

ORDER BY gpa;

```
E.SELECT student_id, gpa  
FROM student_grades  
SORT ORDER BY gpa DESC;  
  
F.SELECT student_id, gpa  
FROM student_grades  
ORDER BY gpa DESC;
```

**Answer:** F

4. In which three cases would you use the USING clause? (Choose three.)
- A.You want to create a nonequijoin.
  - B.The tables to be joined have multiple NULL columns.
  - C.The tables to be joined have columns of the same name and different data types.
  - D.The tables to be joined have columns with the same name and compatible data types.
  - E.You want to use a NATURAL join, but you want to restrict the number of columns in the join condition.

**Answer:** CDE

5. The CUSTOMERS table has these columns:

```
CUSTOMER_ID NUMBER(4) NOT NULL  
CUSTOMER_NAME VARCHAR2(100) NOT NULL  
STREET_ADDRESS VARCHAR2(150)  
CITY_ADDRESS VARCHAR2(50)  
STATE_ADDRESS VARCHAR2(50)  
PROVINCE_ADDRESS VARCHAR2(50)  
COUNTRY_ADDRESS VARCHAR2(50)  
POSTAL_CODE VARCHAR2(12)  
CUSTOMER_PHONE VARCHAR2(20)
```

The CUSTOMER\_ID column is the primary key for the table.

You need to determine how dispersed your customer base is. Which expression finds the number of different countries represented in the CUSTOMERS table?

- A.COUNT(UPPER(country\_address))
- B.COUNT(DIFF(UPPER(country\_address)))
- C.COUNT(UNIQUE(UPPER(country\_address)))
- D.COUNT DISTINCT UPPER(country\_address)
- E.COUNT(DISTINCT (UPPER(country\_address)))

**Answer:** E

6. Click the Exhibit button and examine the data in the EMPLOYEES table.

LAST_NAME	DEPARTMENT_ID	SALARY
Getz	10	3000
Davis	20	1500
King	20	2200
Davis	30	5000
...		

Which three subqueries work? (Choose three.)

A.SELECT \*

FROM employees

where salary > (SELECT MIN(salary))

FROM employees

GROUP BY department\_id);

B.SELECT \*

FROM employees

WHERE salary = (SELECT AVG(salary))

FROM employees

GROUP BY department\_id);

C.SELECT distinct department\_id

FROM employees

WHERE salary > ANY (SELECT AVG(salary))

FROM employees

GROUP BY department\_id);

D.SELECT department\_id

FROM employees

WHERE salary > ALL (SELECT AVG(salary))

FROM employees

GROUP BY department\_id);

E.SELECT last\_name

FROM employees

WHERE salary > ANY (SELECT MAX(salary))

FROM employees

GROUP BY department\_id);

F.SELECT department\_id

FROM employees

```
WHERE salary > ALL (SELECT AVG(salary)
FROM employees
GROUP BY AVG(SALARY));
```

**Answer:** CDE

7. A SELECT statement can be used to perform these three functions:

1. Choose rows from a table.
2. Choose columns from a table.
3. Bring together data that is stored in different tables by creating a link between them.

Which set of keywords describes these capabilities?

- A.difference, projection, join
- B.selection, projection, join
- C.selection, intersection, join
- D.intersection, projection, join
- E.difference, projection, product

**Answer:** B

8. Evaluate this SQL statement:

```
SELECT e.EMPLOYEE_ID,e.LAST_NAME,e.DEPARTMENT_ID, d.DEPARTMENT_NAME
FROM EMPLOYEES e, DEPARTMENTS d
WHERE e.DEPARTMENT_ID = d.DEPARTMENT_ID;
```

In the statement, which capabilities of a SELECT statement are performed?

- A.selection, projection, join
- B.difference, projection, join
- C.selection, intersection, join
- D.intersection, projection, join
- E.difference, projection, product

**Answer:** A

9. Evaluate this SQL statement:

```
SELECT e.employee_id, (.15* e.salary) + (.5 * e.commission_pct)
+ (s.sales_amount * (.35 * e.bonus)) AS CALC_VALUE
FROM employees e, sales s
WHERE e.employee_id = s.emp_id;
```

What will happen if you remove all the parentheses from the calculation?

- A.The value displayed in the CALC\_VALUE column will be lower.
- B.The value displayed in the CALC\_VALUE column will be higher.

- C.There will be no difference in the value displayed in the CALC\_VALUE column.
- D.An error will be reported.

**Answer:** C

10. Which SQL statement generates the alias Annual Salary for the calculated column SALARY\*12?

A.SELECT ename, salary\*12 'Annual Salary'  
FROM employees;  
  
B.SELECT ename, salary\*12 "Annual Salary"  
FROM employees;  
  
C.SELECT ename, salary\*12 AS Annual Salary  
FROM employees;  
  
D.SELECT ename, salary\*12 AS INITCAP("ANNUAL SALARY")  
FROM employees

**Answer:** B

11. Evaluate this SQL statement:

SELECT ename, sal, 12\*sal+100  
FROM emp;

The SAL column stores the monthly salary of the employee. Which change must be made to the above syntax to calculate the annual compensation as "monthly salary plus a monthly bonus of \$100, multiplied by 12"?

A.No change is required to achieve the desired results.  
  
B.SELECT ename, sal, 12\*(sal+100)  
FROM emp;  
  
C.SELECT ename, sal, (12\*sal)+100  
FROM emp;  
  
D.SELECT ename, sal+100,\*12  
FROM emp;

**Answer:** B

12. The CUSTOMERS table has these columns:

CUSTOMER\_ID NUMBER(4) NOT NULL  
  
CUSTOMER\_NAME VARCHAR2(100) NOT NULL  
  
CUSTOMER\_ADDRESS VARCHAR2(150)  
  
CUSTOMER\_PHONE VARCHAR2(20)

You need to produce output that states "Dear Customer customer\_name, ".

The customer\_name data values come from the CUSTOMER\_NAME column in the CUSTOMERS table.

Which statement produces this output?

- A.SELECT dear customer, customer\_name,
- B.SELECT "Dear Customer", customer\_name || ','  
FROM customers;
- C.SELECT 'Dear Customer ' || customer\_name ','  
FROM customers;
- D.SELECT 'Dear Customer ' || customer\_name || ','  
FROM customers;
- E.SELECT "Dear Customer " || customer\_name || ","  
FROM customers;
- F.SELECT 'Dear Customer ' || customer\_name || ',' ||  
FROM customers;

**Answer:** D

13. Which two are attributes of iSQL\*Plus? (Choose two.)

- A.iSQL\*Plus commands cannot be abbreviated.
- B.iSQL\*Plus commands are accessed from a browser.
- C.iSQL\*Plus commands are used to manipulate data in tables.
- D.iSQL\*Plus commands manipulate table definitions in the database.
- E.iSQL\*Plus is the Oracle proprietary interface for executing SQL statements.

**Answer:** BE

14. Which is an iSQL\*Plus command?

- A.INSERT
- B.UPDATE
- C.SELECT
- D.DESCRIBE
- E.DELETE
- F.RENAME

**Answer:** D

15. Which are iSQL\*Plus commands? (Choose all that apply.)

- A.INSERT
- B.UPDATE
- C.SELECT
- D.DESCRIBE
- E.DELETE

## F.RENAME

### **Answer:** D

16. Which two statements are true about constraints? (Choose two.)
- A.The UNIQUE constraint does not permit a null value for the column.
  - B.A UNIQUE index gets created for columns with PRIMARY KEY and UNIQUE constraints.
  - C.The PRIMARY KEY and FOREIGN KEY constraints create a UNIQUE index.
  - D.The NOT NULL constraint ensures that null values are not permitted for the column.

### **Answer:** BD

17. Which three statements correctly describe the functions and use of constraints? (Choose three.)
- A.Constraints provide data independence.
  - B.Constraints make complex queries easy.
  - C.Constraints enforce rules at the view level.
  - D.Constraints enforce rules at the table level.
  - E.Constraints prevent the deletion of a table if there are dependencies.
  - F.Constraints prevent the deletion of an index if there are dependencies.

### **Answer:** CDE

18. Which SQL statement defines a FOREIGN KEY constraint on the DEPTNO column of the EMP table?

A.CREATE TABLE EMP  
(empno NUMBER(4),  
ename VARCHAR2(35),  
deptno NUMBER(7,2) NOT NULL,  
CONSTRAINT emp\_deptno\_fk FOREIGN KEY deptno  
REFERENCES dept deptno);

B.CREATE TABLE EMP  
(empno NUMBER(4),  
ename VARCHAR2(35),  
deptno NUMBER(7,2)  
CONSTRAINT emp\_deptno\_fk REFERENCES dept (deptno));

C.CREATE TABLE EMP  
(empno NUMBER(4),  
ename VARCHAR2(35),  
deptno NUMBER(7,2) NOT NULL,  
CONSTRAINT emp\_deptno\_fk REFERENCES dept (deptno)  
FOREIGN KEY (deptno));

```
D.CREATE TABLE EMP  
(empno NUMBER(4),  
ename VARCHAR2(35),  
deptno NUMBER(7,2) FOREIGN KEY  
CONSTRAINT emp_deptno_fk REFERENCES dept (deptno));
```

**Answer:** B

19. Which view should a user query to display the columns associated with the constraints on a table owned by the user?

- A.USER\_CONSTRAINTS
- B.USER\_OBJECTS
- C.ALL\_CONSTRAINTS
- D.USER\_CONS\_COLUMNS
- E.USER\_COLUMNS

**Answer:** D

20. You need to design a student registration database that contains several tables storing academic information.

The STUDENTS table stores information about a student. The STUDENT\_GRADES table stores information about the student's grades. Both of the tables have a column named STUDENT\_ID. The STUDENT\_ID column in the STUDENTS table is a primary key.

You need to create a foreign key on the STUDENT\_ID column of the STUDENT\_GRADES table that points to the STUDENT\_ID column of the STUDENTS table. Which statement creates the foreign key?

```
A.CREATE TABLE student_grades  
(student_id NUMBER(12),  
semester_end DATE,  
gpa NUMBER(4,3),  
CONSTRAINT student_id_fk REFERENCES (student_id)  
FOREIGN KEY students(student_id));
```

```
B.CREATE TABLE student_grades  
(student_id NUMBER(12),  
semester_end DATE,  
gpa NUMBER(4,3),  
student_id_fk FOREIGN KEY (student_id)  
REFERENCES students(student_id));
```

```
C.CREATE TABLE student_grades
```

```
(student_id NUMBER(12),
semester_end DATE,
gpa NUMBER(4,3),
CONSTRAINT FOREIGN KEY (student_id)
REFERENCES students(student_id));

D.CREATE TABLE student_grades
(student_id NUMBER(12),
semester_end DATE,
gpa NUMBER(4,3),
CONSTRAINT student_id_fk FOREIGN KEY (student_id)
REFERENCES students(student_id));
```

**Answer:** D