Koło Naukowe TechOr
testy kwalifikacyjne
Examine the structure of the EMPLOYEES table:

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>EMPLOYEE_ID</td>
<td>NUMBER</td>
<td>Primary Key</td>
</tr>
<tr>
<td>FIRST_NAME</td>
<td>VARCHAR2(25)</td>
<td></td>
</tr>
<tr>
<td>LAST_NAME</td>
<td>VARCHAR2(25)</td>
<td></td>
</tr>
</tbody>
</table>

Which three statements inserts a row into the table?

A. INSERT INTO employees
VALUES ( NULL, ‘John’,‘Smith’);

B. INSERT INTO employees( first_name, last_name)
VALUES(‘John’,‘Smith’);

C. INSERT INTO employees
VALUES (‘1000’,John’,NULL);

D. INSERT INTO employees(first_name,last_name, employee_id)
VALUES (1000, ‘John’,‘Smith’);

E. INSERT INTO employees (employee_id)
VALUES (1000);

F. INSERT INTO employees (employee_id, first_name, last_name)
VALUES (1000, ‘John’,’’);
Examine the description of the EMPLOYEES table:

- EMP_ID NUMBER(4) NOT NULL
- LAST_NAME VARCHAR2(30) NOT NULL
- FIRST_NAME VARCHAR2(30)
- DEPT_ID NUMBER(2)
- JOB_CAT VARCHAR2(30)
- SALARY NUMBER(8,2)

Which statement shows the maximum salary paid in each job category of each department?

A. SELECT dept_id, job_cat, MAX(salary) FROM employees WHERE salary > MAX(salary);

B. SELECT dept_id, job_cat, MAX(salary) FROM employees GROUP BY dept_id, job_cat;

C. SELECT dept_id, job_cat, MAX(salary) FROM employees;

D. SELECT dept_id, job_cat, MAX(salary) FROM employees GROUP BY dept_id;

E. SELECT dept_id, job_cat, MAX(salary) FROM employees GROUP BY dept_id, job_cat, salary;
Management has asked you to calculate the value $12 \times \text{salary} \times \text{commission\_pct}$ for all the employees in the EMP table. The EMP table contains these columns:

- **LAST NAME**  VARCNAR2(35)  NOT NULL
- **SALARY**  NUMBER(9,2)  NOT NULL
- **COMMISION\_PCT**  NUMBER(4,2)

Which statement ensures that a value is displayed in the calculated columns for all employees?

A. `SELECT last_name, 12*salary* commission\_pct`  
FROM emp;

B. `SELECT last_name, 12*salary* (commission\_pct,0)`  
FROM emp;

C. `SELECT last_name, 12*salary*(nvl(commission\_pct,0))`  
FROM emp;

D. `SELECT last_name, 12*salary*(decode(commission\_pct,0))`  
FROM emp;
Examine the description of the STUDENTS table:

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>STD_ID</td>
<td>NUMBER(4)</td>
</tr>
<tr>
<td>COURSE_ID</td>
<td>VARCHAR2(10)</td>
</tr>
<tr>
<td>START_DATE</td>
<td>DATE</td>
</tr>
<tr>
<td>END_DATE</td>
<td>DATE</td>
</tr>
</tbody>
</table>

Which two aggregate functions are valid on the START_DATE column?

A. SUM(start_date)
B. AVG(start_date)
C. COUNT(start_date)
D. AVG(start_date, end_date)
E. MIN(start_date)
F. MAXIMUM(start_date)
The EMPLOYEE tables has these columns:
LAST_NAME VARCHAR2(35)
SALARY NUMBER(8,2)
COMMISSION_PCT NUMBER(5,2)

You want to display the name and annual salary multiplied by the commission_pct for all employees. For records that have a NULL commission_pct, a zero must be displayed against the calculated column.

Which SQL statement displays the desired results?
A. SELECT last_name, (salary * 12) * commission_pct
FROM EMPLOYEES;
B. SELECT last_name, (salary * 12) * IFNULL(commission_pct, 0)
FROM EMPLOYEES;
C. SELECT last_name, (salary * 12) * NVL2(commission_pct, 0)
FROM EMPLOYEES;
D. SELECT last_name, (salary * 12) * NVL(commission_pct, 0)
FROM EMPLOYEES;
Evaluate the SQL statement:

1 SELECT a.emp_name, a.sal, a.dept_id, b.maxsal
2 FROM employees a,
3 (SELECT dept_id, MAX(sal) maxsal
4 FROM employees
5 GROUP BY dept_id) b
6 WHERE a.dept_id = b.dept_id
7 AND a.sal < b.maxsal;

What is the result of the statement?
A. The statement produces an error at line 1.
B. The statement produces an error at line 3.
C. The statement produces an error at line 6.
D. The statement returns the employee name, salary, department ID, and maximum salary earned in the department of the employee for all departments that pay less salary then the maximum salary paid in the company.
E. The statement returns the employee name, salary, department ID, and maximum salary earned in the department of the employee for all employees who earn less than the maximum salary in their department.
Evaluate the set of SQL statements:
CREATE TABLE dept
(deptno NUMBER(2),
dname VARCHAR2(14),
loc VARCHAR2(13));
ROLLBACK;
DESCRIBE DEPT

What is true about the set?
A. The DESCRIBE DEPT statement displays the structure of the DEPT table.
B. The ROLLBACK statement frees the storage space occupies by the DEPT table.
C. The DESCRIBE DEPT statement returns an error ORA-04043: object DEPT does not exist.
D. The DESCRIBE DEPT statement displays the structure of the DEPT table only if there is a COMMIT statement introduced before the ROLLBACK statement.
The EMP table contains these columns:
LAST NAME VARCHAR2(25)
SALARY NUMBER(6,2)
DEPARTMENT_ID NUMBER(6)

You need to display the employees who have not been assigned to any department. You write the SELECT statement:
SELECT LAST_NAME, SALARY, DEPARTMENT_ID
FROM EMP
WHERE DEPARTMENT_ID = NULL;

What is true about this SQL statement?
A. The SQL statement displays the desired results.
B. The column in the WHERE clause should be changed to display the desired results.
C. The operator in the WHERE clause should be changed to display the desired results.
D. The WHERE clause should be changed to use an outer join to display the desired results.
Examine the description of the MARKS table:
STD_ID NUMBER(4)
STUDENT_NAME VARCHAR2(30)
SUBJ1 NUMBER(3)
SUBJ2 NUMBER(3)

SUBJ1 and SUBJ2 indicate the marks obtained by a student in two subjects.

Examine this SELECT statement based on the MARKS table:
SELECT subj1+subj2 total_marks, std_id
FROM marks
WHERE subj1 > AVG(subj1) AND subj2 > AVG(subj2)
ORDER BY total_marks;

What is the result of the SELECT statement?
A. The statement executes successfully and returns the student ID and sum of all marks for each student who obtained more than the average mark in each subject.
B. The statement returns an error at the SELECT clause.
C. The statement returns an error at the WHERE clause.
D. The statement returns an error at the ORDER BY clause.
You want to display the titles of books that meet these criteria:
1. Purchased before January 21, 2001
2. Price is less then $500 or greater than $900

You want to sort the results by their data of purchase, starting with the most recently bought book.

Which statement should you use?

A. SELECT book_title
FROM books
WHERE price between 500 and 900
AND purchase_date < '21-JAN-2001'
ORDER BY purchase_date;

B. SELECT book_title
FROM books
WHERE price IN (500,900)
AND purchase_date < '21-JAN-2001'
ORDER BY purchase_date ASC;

C. SELECT book_title
FROM books
WHERE price < 500 or > 900
AND purchase_date < '21-JAN-2001'
ORDER BY purchase date DESC;

D. SELECT book_title
FROM books
WHERE (price < 500 OR price > 900)
AND purchase_date < '21-JAN-2001'
ORDER BY purchase date DESC;
Examine the description of the EMPLOYEES table:

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Constraints</th>
</tr>
</thead>
<tbody>
<tr>
<td>EMP_ID</td>
<td>NUMBER(4)</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>LAST_NAME</td>
<td>VARCHAR2(30)</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>FIRST_NAME</td>
<td>VARCHAR2(30)</td>
<td></td>
</tr>
<tr>
<td>DEPT_ID</td>
<td>NUMBER(2)</td>
<td></td>
</tr>
<tr>
<td>JOB_CAT</td>
<td>VARCHAR2(30)</td>
<td></td>
</tr>
<tr>
<td>SALARY</td>
<td>NUMBER(8,2)</td>
<td></td>
</tr>
</tbody>
</table>

Which statement shows the department ID, minimum salary, and maximum salary paid in that department, only if the minimum salary is less than 5000 and the maximum salary is more than 15000?

A. SELECT dept_id, MIN(salary), MAX(salary) FROM employees
   WHERE MIN(salary) < 5000 AND MAX(salary) > 15000;

B. SELECT dept_id, MIN(salary), MAX(salary) FROM employees
   WHERE MIN(salary) < 5000 AND MAX(salary) > 15000 GROUP BY dept_id;

C. SELECT dept_id, MIN(salary), MAX(salary) FROM employees
   HAVING MIN(salary) < 5000 AND MAX(salary) > 15000;

D. SELECT dept_id, MIN(salary), MAX(salary) FROM employees
   GROUP BY dept_id HAVING MIN(salary) < 5000 AND MAX(salary) < 15000;
In a SELECT statement that includes a WHERE clause, where is the GROUP BY clause placed in the SELECT statement?

A. Immediately after the SELECT clause

B. Before the WHERE clause

C. Before the FROM clause

D. After the ORDER BY clause

E. After the WHERE clause
What is necessary for your query on an existing view to execute successfully?

A. The underlying tables must have data.
B. You need SELECT privileges on the view.
C. The underlying tables must be in the same schema.
D. You need SELECT privileges only on the underlying tables.
The EMP table has these columns:

- ENAME VARCHAR2(35)
- SALARY NUMBER(8,2)
- HIRE_DATE DATE

Management wants a list of names of employees who have been with the company for more than five years. Which SQL statement displays the required results?

A. SELECT ENAME
   FROM EMP
   WHERE SYSDATE-HIRE_DATE > 5;

B. SELECT ENAME
   FROM EMP
   WHERE HIRE_DATE-SYSDATE > 5;

C. SELECT ENAME
   FROM EMP
   WHERE (SYSDATE-HIRE_DATE)/365 > 5;

D. SELECT ENAME
   FROM EMP
   WHERE (SYSDATE-HIRE_DATE)* 365 > 5;
Examine the data in the EMPLOYEES table.

<table>
<thead>
<tr>
<th>EMP_ID</th>
<th>EMP_NAME</th>
<th>DEPT_ID</th>
<th>MGR_ID</th>
<th>JOB_ID</th>
<th>SALARY</th>
</tr>
</thead>
<tbody>
<tr>
<td>101</td>
<td>Smith</td>
<td>20</td>
<td>120</td>
<td>SA_REP</td>
<td>4000</td>
</tr>
<tr>
<td>102</td>
<td>Martin</td>
<td>10</td>
<td>105</td>
<td>CLERK</td>
<td>2500</td>
</tr>
<tr>
<td>105</td>
<td>Diana</td>
<td>30</td>
<td>108</td>
<td>IT_ADMIN</td>
<td>5000</td>
</tr>
<tr>
<td>106</td>
<td>Smith</td>
<td>40</td>
<td>110</td>
<td>AD.ASST</td>
<td>3000</td>
</tr>
<tr>
<td>108</td>
<td>Jennifer</td>
<td>30</td>
<td>110</td>
<td>HR(DIR</td>
<td>6500</td>
</tr>
<tr>
<td>110</td>
<td>Bob</td>
<td>40</td>
<td></td>
<td>EK_DIR</td>
<td>8000</td>
</tr>
<tr>
<td>120</td>
<td>Revi</td>
<td>20</td>
<td>110</td>
<td>SA_DIR</td>
<td>6500</td>
</tr>
</tbody>
</table>

On the EMPLOYEES table, EMPLOYEE_ID is the primary key. MGR_ID is the ID of managers and refers to the EMPLOYEE_ID. The JOB_ID column is a NOT NULL column.

Evaluate this DELETE statement:

DELETE employee_id, salary, job_id
FROM employees
WHERE dept_id = 90;

Why does the DELETE statement fail when you execute it?
A. There is no row with dept_id 90 in the EMPLOYEES table.
B. You cannot delete the JOB_ID column because it is a NOT NULL column.
C. You cannot specify column names in the DELETE clause of the DELETE statement.
D. You cannot delete the EMPLOYEE_ID column because it is the primary key of the table.
Evaluate these two SQL statements:
SELECT last_name, salary, hire_date
FROM EMPLOYEES
ORDER BY salary DESC;
SELECT last_name, salary, hire_date
FROM EMPLOYEES
ORDER BY 2 DESC;

What is true about them?
A. The two statements produce identical results.
B. The second statement returns a syntax error.
C. There is no need to specify DESC because the results are sorted in descending order by default.
D. The two statements can be made to produce identical results by adding a column alias for the salary column in the second SQL statement.
The CUSTOMERS table has these columns:
CUSTOMER_ID NUMBER(4) NOT NULL
CUSTOMER_NAME VARCHAR2(100) NOT NULL
ADDRESS VARCHAR2(250)
POSTAL_CODE VARCHAR2(12)
CUSTOMER_PHONE VARCHAR2(20)

Which statement finds the rows in the CUSTOMERS table that do not have a postal code?
A. SELECT customer_id, customer_name FROM customers
   WHERE postal_code CONTAINS NULL;
B. SELECT customer_id, customer_name FROM customers
   WHERE postal_code = '________';
C. SELECT customer_id, customer_name FROM customers
   WHERE postal_code IS NULL;
D. SELECT customer_id, customer_name FROM customers
   WHERE postal_code IS NVL;
E. SELECT customer_id, customer_name FROM customers
   WHERE postal_code = NULL;
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.

Which two statements find the number of customers?

- A. SELECT TOTAL(*) FROM customers;
- B. SELECT COUNT(*) FROM customers;
- C. SELECT TOTAL(customer_id) FROM customers;
- D. SELECT COUNT(customer_id) FROM customers;
- E. SELECT COUNT(customers) FROM customers;
- F. SELECT TOTAL(customer_name) FROM customers;
Which two statements are true regarding the ORDER BY clause?

A. The sort is in ascending by order by default.

B. The sort is in descending order by default.

C. The ORDER BY clause must precede the WHERE clause.

D. The ORDER BY clause is executed on the client side.

E. The ORDER BY clause comes last in the SELECT statement.

F. The ORDER BY clause is executed first in the query execution
Question No: 20

Which operator can be used with a multiple-row subquery?

A. =
B. LIKE
C. BETWEEN
D. NOT IN
E. IS
F. <>
You need to display the last names of those employees who have the letter “A” as the second character in their names.

Which SQL statement displays the required results?

A. SELECT last_name
   FROM EMP
   WHERE last_name LIKE '_A%';

B. SELECT last_name
   FROM EMP
   WHERE last_name = '*A%';

C. SELECT last_name
   FROM EMP
   WHERE last_name = '_A%';

D. SELECT last_name
   FROM EMP
   WHERE last_name LIKE '*A%';
Question No: 22

Which two are character manipulation functions?

A. TRIM
B. REPLACE
C. TRUNC
D. TO_DATE
E. MOD
F. CASE
You need to calculate the total of all salaries in the accounting department. Which group function should you use?

A. MAX
B. MIN
C. SUM
D. COUNT
E. TOTAL
F. LARGEST
Which constraint can be defined only at the column level?

A. UNIQUE
B. NOT NULL
C. CHECK
D. PRIMARY KEY
E. FOREIGN KEY
The database administrator of your company created a public synonym called HR for the HUMAN_RESOURCES table of the GENERAL schema, because many users frequently use this table. As a user of the database, you created a table called HR in your schema. What happens when you execute this query?

SELECT *
FROM HR;

A. You obtain the results retrieved from the public synonym HR created by the database administrator.

B. You obtain the results retrieved from the HR table that belongs to your schema.

C. You get an error message because you cannot retrieve from a table that has the same name as a public synonym.

D. You obtain the results retrieved from both the public synonym HR and the HR table that belongs to your schema, as a Cartesian product.

E. You obtain the results retrieved from both the public synonym HR and the HR table that belongs to your schema, as a FULL JOIN.
You need to modify the STUDENTS table to add a primary key on the STUDENT_ID column. The table is currently empty.

Which statement accomplishes this task?

A. ALTER TABLE students
   ADD PRIMARY KEY student_id;

B. ALTER TABLE students
   ADD CONSTRAINT PRIMARY KEY (student_id);

C. ALTER TABLE students
   ADD CONSTRAINT stud_id_pk PRIMARY KEY student_id;

D. ALTER TABLE students
   ADD CONSTRAINT stud_id_pk PRIMARY KEY (student_id);

E. ALTER TABLE students
   MODIFY CONSTRAINT stud_id_pk PRIMARY KEY (student_id);
Evaluate the SQL statement:

```sql
SELECT ROUND(TRUNC(MOD(1600,10),-1),2)
FROM dual;
```

What will be displayed?

A. 0
B. 1
C. 0.00
D. An error statement
For which two constraints does the Oracle Server implicitly create a unique index?

A. NOT NULL
B. PRIMARY KEY
C. FOREIGN KEY
D. CHECK
E. UNIQUE
The STUDENT_GRADES table has these columns:
STUDENT_ID NUMBER(12)
SEMESTER_END DATE
GPA NUMBER(4,3)
The registrar has asked for a report on the average grade point average (GPA) for students enrolled during semesters that end in the year 2000.
Which statement accomplish this?
A. SELECT AVERAGE(gpa)
   FROM student_grades
   WHERE semester_end > '01-JAN-2000' and semester_end < '31-DEC-2000';

B. SELECT COUNT(gpa)
   FROM student_grades
   WHERE semester_end > '01-JAN-2000' and semester_end < '31-DEC-2000';

C. SELECT MIN(gpa)
   FROM student_grades
   WHERE semester_end > '01-JAN-2000' and semester_end < '31-DEC-2000';

D. SELECT AVG(gpa)
   FROM student_grades
   WHERE semester_end BETWEEN '01-JAN-2000' and '31-DEC-2000';
The ORDERS table has these columns:
ORDER_ID NUMBER(4) NOT NULL
CUSTOMER_ID NUMBER(12) NOT NULL
ORDER_TOTAL NUMBER(10,2)

The ORDERS table tracks the Order number, the order total, and the customer to whom the Order belongs. Which two statements retrieve orders with an inclusive total that ranges between 100.00 and 2000.00 dollars?

A. SELECT customer_id, order_id, order_total
   FROM orders
   RANGE ON order_total (100 AND 2000) INCLUSIVE;

B. SELECT customer_id, order_id, order_total
   FROM orders
   HAVING order_total BETWEEN 100 and 2000;

C. SELECT customer_id, order_id, order_total
   FROM orders
   WHERE order_total BETWEEN 100 and 2000;

D. SELECT customer_id, order_id, order_total
   FROM orders
   WHERE order_total >= 100 and <= 2000;

E. SELECT customer_id, order_id, order_total
   FROM orders
   WHERE order_total >= 100 and order_total <= 2000;