Online Test
Instructor and Examiner:
MD. ABDUL BARI

Test Results

surname	name	user	points
Khalid	M. M. Khalid Mamun	1147017	20.267 (51%)

test: R-18 Oracle mock test 1

start time:	2013-05-27 03:33:52
end time:	2013-05-27 04:13:28
time:	00:39:36
test time [min]:	40
basic points:	1.000
points for wrong answer:	0.000
points for no answer:	0.000
max score:	40.000
correct:	21 (53%)

R-18 Oracle mock test 1

#	points	IP	start [hh:mm:ss]	end [hh:mm:ss]	time [mm:ss]	reaction [sec]
1 S	1.000	281473913978898	03:41:46	03:41:55	00:09	9.271

RDBMS stands for…

+	1	Relational Database Management System
	2	Real Database Management System
	3	Read Database Master System
	4	Realtime Database Management System

2 S	1.000	281473913978898	03:47:52	03:49:09	01:17	75.717

Which SELECT statement should you use if you want to display unique combination of the POSITION and MANAGER values from the EMPLOYEE table?

explanation
To display a unique values in the result you can use the DISTINCT key word this will eliminate the duplicate values from the result of the query.

+	1	SELECT DISTINCT position, manager FROM employee;
	2	SELECT position, DISTINCT manager FROM employee;
	3	SELECT position, manager FROM employee;
	4	SELECT position, manager DISTINCT FROM employee;

3 M	0.600	281473913978898	03:52:17	03:53:19	01:02	62.013

Which two are attributes of /SQL*Plus? (Choose two)

explanation
SQL*Plus commands can be used to manipulate data in tables and SQL*Plus commands manipulate table definitions in the database.

Incorrect Answers
SQL*Plus commands can be abbreviated. Like command DESCRIBE can be abbreviated as DESC, or SELECT as SELE.
SQL*Plus commands are not accesses from a browser.
SQL*Plus is not only the Oracle proprietary interface for executing SQL statements.

-	1	/SQL*Plus commands are accesses from a browser.
-	2	/SQL*Plus commands manipulate table definitions in the database.
+	3	/SQL*Plus is the Oracle proprietary interface for executing SQL statements.
+	4	/SQL*Plus commands cannot be abbreviated.
+	5	/SQL*Plus commands are used to manipulate data in tables.

4 S	1.000	281473913978898	04:10:13	04:10:26	00:13	13.486

Which is an /SQL*Plus command?

explanation
There is only one SQL*Plus command in this list: DESCRIBE. It cannot be used as SQL command. This command returns a description of tablename, including all columns in that table, the datatype for each column, and an indication of whether the column permits storage of NULL values.
Incorrect Answers
INSERT is not a SQL*Plus command. It's data-manipulation language (DML) command.
UPDATE is not a SQL*Plus command. It's data-manipulation language (DML) command.
SELECT is not a SQL*Plus command.
DELETE is not a SQL*Plus command. It's data-manipulation language (DML) command.
RENAME is not a SQL*Plus command.

	1	INSERT
+	2	DESCRIBE
	3	DELETE
	4	SELECT
	5	RENAME
	6	UPDATE

5 S	0.000	0	04:12:46	--:--:--	--:--	0

You need to produce a report for mailing labels for all customers. The mailing label must have only the customer name and address. The CUSTOMERS table has these columns:
CUST_ID NUMBER(4) NOT NULL
CUST_NAME VARCHAR2(100)
CUST_ADDRESS VARCHAR2(150)
CUST_PHONE VARCHAR2(20)
Which SELECT statement accomplishes this task?

	1	SELECT* FROM customers;
	2	SELECT cust_name, cust_address FROM customers;
	3	SELECT name, address FROM customers;
	4	SELECT id, name, address, phone FROM customers;
	5	SELECT cust_id, cust_name, cust_address, cust_phone FROM customers;

6 S	1.000	281473913978898	03:26:11	04:00:08	33:57	44.819

Evaluate this SQL statement:

```
SELECT e.EMPLOYEE_ID,e.LAST_NAME,e.DEPARTMENT_ID, d.DEPARTMENT_NAME.
FROM EMP e, DEPARTMENT d
WHERE e.DEPARTMENT_ID = d.DEPARTMENT_ID;
```

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

+	1	Selection, projection, join
	2	Difference, projection, join
	3	Selection, intersection, join
	4	Intersection, projection, join
	5	Difference, projection, product

7 M	0.667	281473913978898	04:09:08	04:09:43	00:35	35.137

Which two statements are true regarding the ORDER BY clause? (Choose two)

explanation
The ORDER BY clause does sort data in ascending order by default. And the ORDER BY clause comes last in the SELECT statement: after FROM or WHERE or GROUP BY clause.

+	1	The ORDER BY clause is executed first in the query execution.
+	2	The ORDER BY clause must precede the WHERE clause.
-	3	The ORDER BY clause comes last in the SELECT statement.
+	4	The sort is in descending order by default.
+	5	The sort is in ascending by order by default.
-	6	The ORDER BY clause is executed on the client side.

8 M	0.400	281473913978898	03:50:30	03:52:17	01:47	107.21

From SQL*Plus, you issue this SELECT statement:
SELECT*
From orders;
You use this statement to retrieve data from a data table for _____. (Choose all that apply)

explanation
You can use SELECT statement to display and to insert data into different table.

+	1	Viewing
+	2	Updating
-	3	Truncating
-	4	Inserting
-	5	Deleting

9 S	1.000	281473913978898	03:40:42	03:41:46	01:04	61.091

Which SQL SELECT statement performs a projection, a selection, and join when executed?

explanation
PROJECTION will select the whole column values of the table while SELECTION will gives you rows of the table and JOIN is joining the two tables on a same column. To get all these task done in one statement you can use this command
SELECT p.id_number, m.manufacturer_id, m.city
FROM product p, manufacturer m

	1	SELECT id_number, manufacturer_id

		FROM product ORDER BY manufacturer_id, id_number;
+	2	SELECT p.id_number, m.manufacturer_id, m.city FROM product p, manufacturer m WHERE p.manufacturer_id = m.manufacturer_id AND m.manufacturer_id = 'NF10032';
	3	SELECT manufacturer_id, city FROM manufacturer AND manufacturer_id = 'NF10032' ORDER BY city;
	4	SELECT id_number, manufacturer_id FROM product WHERE manufacturer_id = 'NF10032';

10 S	0.000	281473913978898	03:33:04	04:12:46	39:42	30.943

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)

Which statement finds the rows in the CUSTOMERS table that do not have a postal code?

explanation

This statement returns the rows in the CUSTOMERS table that do not have a postal code. The correct syntax to check NULL values is usage of “IS NULL” clause.

	1	SELECT customer_id, customer_name FROM customers WHERE postal_code = '_____';
-	2	SELECT customer_id, customer_name FROM customers WHERE postal code IS NVL;
	3	SELECT customer_id, customer_name FROM customers WHERE postal_code IS NULL;
	4	SELECT customer_id, customer_name FROM customers WHERE postal_code CONTAINS NULL;
	5	SELECT customer_id, customer_name FROM customers WHERE postal_code = NULL;

11 S	0.000	281473913978898	03:25:57	03:42:38	16:41	42.445

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?

explanation

These two statements produce identical results, because it is possible even to use numbers to indicate the column position where Oracle should order the output from a statement.

	1	The two statements produce identical results.
	2	The second statement returns a syntax error.
	3	There is no need to specify DESC because the results are sorted in descending order by default.
-	4	The two statements can be made to produce identical results by adding a column alias for the salary column in the second SQL statement.

12 S	0.000	0	03:26:48	--:--:--	--:--	0

Evaluate the set of SQL statements:

```
CREATE TABLE dept
(deptno NUMBER(2),
dname VARCNAR2(14),
loc VARCNAR2(13));
```

ROLLBACK;
DESCRIBE DEPT

What is true about the set?

explanation
The structure of the DEPT table will be displayed because the CREATE TABLE statement is DDL operation and it cannot be rolled back because implicit commit occurs on the database when a user exits SQL*Plus or issues a data-definition language (DDL) command such as a create table statement, user to create a database object, or an alter table statement, used to alter a database object.

	1	The DESCRIBE DEPT statement displays the structure of the DEPT table only if the us a COMMIT statement introduced before the ROLLBACK statement..
	2	The ROLLBACK statement frees the storage space occupies by the DEPT table.
	3	The DESCRIBE DEPT statement returns an error ORA-04043: object DEPT does not exist.
	4	The DESCRIBE DEPT statement displays the structure of the DEPT table.

13 S	1.000	281473913978898	03:40:07	03:40:42	00:35	35.176

Which /SQL*Plus feature can be used to replace values in the WHERE clause?

explanation
Lexical substitution variables can be used to replace values in the WHERE clause.

	1	Instead-of variables
+	2	Substitution variables
	3	Replacement variables
	4	Prompt variables
	5	This feature cannot be implemented through /SQL*Plus.

14 S	0.000	281473913978898	03:23:52	03:24:22	00:30	30.172

You are formulating queries in a SQL*Plus. Which of the following statement correctly describes how to specify a column alias?

explanation
Aliases do not describe the tables they describe columns so the alias should be place at the end of each column and separated by a space to describe the column.

	1	Place the alias at the beginning of the statement to describe the table.
	2	Place the alias after each column separated by a space to describe the column.
	3	Place the alias at the end of the statement to describe the table.
-	4	Place the alias after each column separated by a comma to describe the column.

15 S	0.000	281473913978898	04:11:13	04:11:41	00:28	27.463

You want to use a function in you column clause of a SQL statement. The NVL function accomplishes which of the following tasks?

explanation
NVL function is simple if_then operation that test column values out to see whether it is NULL and if it find it is null then NVL substitutes the specified default value for the NULL value.

	1	Enables you to specify alternated out for NULL column values.
	2	Nullifies the value of the column out put.
-	3	Enables you to specify alternate output for non-NULL column values.
	4	Assists in the distribution of output across multiple columns.

16 S	0.000	0	04:13:00	--:--:--	--:--	0

You want to use SQL*Plus to connect to the oracle database. Which of the following choices does not indicate a component you must specify when logging into the oracle?

explanation
When connecting to the database you don't need to specify the name of the database and when you are not specifying the name of the database then you will be connected to the local database.

	1	The password.
	2	The database name.
	3	The username
	4	The SQL*Plus Keyword.

17 S	1.000	281473913978898	03:45:24	03:47:52	02:28	148.01

The EMPLOYEE_HISTORY table contains these columns:

EMPLOYEE_ID NUMBER
LAST_NAME VARCHAR2(25)
FIRST_NAME VARCHAR2(25)
DEPARTMENT_ID NUMBER
POSITION VARCHAR2(30)
SALARY NUMBER(6,2)
HIRE_DATE DATE
DEPART_DATE DATE

The EMPLOYEE_HISTORY table contains only former employees.
You need to create a report to display all former employees that were hired on or

after January 1, 1996. The data should display in this format:
Former Employee Term of Employment

14837 - SMITH 10-MAY-92 / 01-JUN-01

Which SELECT statement could you use?

+	1	SELECT employee_id||' - '||last_name "Former Employee", hire_date||' / '||depart_date "Term of Employment" FROM employee_history WHERE hire_date > '31-DEC-95' AND depart_date IS NOT NULL;
	2	SELECT employee_id||' - '||last_name "AS Former Employee", hire_date||' / '||depart_date "AS Term of Employment" FROM employee_history WHERE hire_date > '31-DEC-95';
	3	SELECT employee_id||' - '||last_name 'Former Employee', hire_date||' / '||depart_date 'Term of Employment' FROM employee_history WHERE hire_date > '31-DEC-95' AND depart_date > NULL;
	4	SELECT employee_id||' - '||last_name AS Former Employee, hire_date||' / '||depart_date AS Term of Employment FROM employee_history WHERE hire_date > '31-DEC-95';
	5	SELECT employee_id||' - '||last_name "Former Employee", hire_date||' / '||depart_date "Term of Employment" FROM employee_history WHERE hire_date > '31-DEC-95' AND depart_date <> NULL;

18 S	0.000	281473913978898	04:11:41	04:11:56	00:15	15.315

Examine the structure of the EMP_DEPT_VU view:
Column Name Type Remarks
EMPLOYEE_ID NUMBER From the EMPLOYEES table
EMP_NAME VARCHAR2(30) From the EMPLOYEES table
JOB_ID VARCHAR2(20) From the EMPLOYEES table
SALARY NUMBER From the EMPLOYEES table
DEPARTMENT_ID NUMBER From the DEPARTMENTS table
DEPT_NAME VARCHAR2(30) From the DEPARTMENTS table

Which SQL statement produces an error?

	1	SELECT department_id, SUM(salary) FROM emp_dept_vu GROUP BY department_id;
	2	None of the statements produce an error; all are valid.
-	3	SELECT job_id, SUM(salary) FROM emp_dept_vu WHERE department_id IN (10,20) GROUP BY job_id HAVING SUM(salary) > 20000;
	4	SELECT * FROM emp_dept_vu;
	5	SELECT department_id, job_id, AVG(salary) FROM emp_dept_vu GROUP BY department_id, job_id;

19 S	0.000	0	--:--:--	--:--:--	--:--	0

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"?

explanation
To achieve the result you must add 100 to sal before multiply with 12.
Select ename, sal, 12*(sal+100) from EMP;

	1	SELECT ename, sal+100,*12 FROM emp;
	2	No change is required to achieve the desired results.
	3	SELECT ename, sal, 12*(sal+100) FROM emp;
	4	SELECT ename, sal, (12*sal)+100 FROM emp;

20 S	1.000	281473913978898	03:53:19	03:55:59	02:40	159.634

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?

explanation		
Concatenation operator to create a resultant column that is a character expression.		
	1	SELECT 'Dear Customer ' || customer_name || ',' || FROM customers;
	2	SELECT "Dear Customer", customer_name || ',' FROM customers;
+	3	SELECT 'Dear Customer ' || customer_name || ',' FROM customers;
	4	SELECT dear customer, customer_name, FROM customers;
	5	SELECT "Dear Customer " || customer_name || "," FROM customers;
	6	SELECT 'Dear Customer ' || customer_name ',' FROM customers;

21 S	1.000	281473913978898	03:59:15	03:59:23	00:08	8.021

A SELECT statement can be used to perform these three functions:
- Choose rows from a table.
- Choose columns from a table.
- Bring together data that is stored in different tables by creating a link between them.

Which set of keywords describes these capabilities?

explanation

choose rows from a table is SELECTION,
Choose column from a table is PROJECTION Bring together data in different table by creating a link between them is JOIN.

+	1	selection, projection, join
	2	difference, projection, product
	3	intersection, projection, join
	4	difference, projection, join
	5	selection, intersection, join

22 S	1.000	281473913978898	03:32:43	03:33:04	00:21	20.597

Which statement correctly describes SQL and /SQL*Plus?

	1	/SQL*Plus recognizes SQL statements and sends them to the server; SQL is the Oracle proprietary interface for executing SQL statements.
+	2	Both SQL and /SQL*plus allow manipulation of values in the database.
	3	/SQL*Plus is a language for communicating with the Oracle server to access data; SQL recognizes SQL statements and sends them to the server.
	4	SQL manipulates data and table definitions in the database; /SQL*Plus does not allow manipulation of values in the database.

23 S	1.000	281473913978898	03:30:14	03:32:43	02:29	130.488

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?

explanation

Statement in this answer will show correct results because usage of operator LIKE and format mask ‘_A%’ extract the last names of those employees who have the letter “A” as the second character in their names. Symbol ‘_’ in format mask substitute exactly one symbol and cannot be NULL.

	1	SELECT last_name FROM EMP WHERE last name =’*A%’
	2	SELECT last_name FROM EMP WHERE last name LIKE ‘*A%’
+	3	SELECT last_name FROM EMP WHERE last_name LIKE ‘_A%’;
	4	SELECT last_name FROM EMP WHERE last name =’_A%’;

24 S	1.000	281473913978898	04:00:08	04:02:11	02:03	122.607

In which scenario would TOP N analysis be the best solution?

explanation

If you want to rank the top three sales representatives who have sold the maximum number of products TOP-N query will be the best solution. TON-N queries use inline views and are handy for displaying a short list of table data, based on “greatest” or “least” criteria.

+	1	You want to rank the top three sales representatives who have sold the maximum number of products.
	2	You want to find the manager supervising the largest number of employees.
	3	You want to identify the most senior employee in the company.
	4	You want to identify the person who makes the highest salary for all employees.

25 S	1.000	281473913978898	03:24:33	03:45:24	20:51	118.778

Evaluate this SQL statement:

```
SELECT c.customer_id, o.order_id, o.order_date, p.product_name
FROM customer c, curr_order o, product p
WHERE customer.customer_id = curr_order.customer_id
```

AND o.product_id = p.product_id
ORDER BY o.order_amount;

This statement fails when executed. Which change will correct the problem?

explanation

When an alias is define for a table name in join then you cannot use the table name instead of alias in the FROM clause while using alias in the SELECT list. An alias should be used in the WHERE clause also.

	1	Include the ORDER_AMOUNT column in the SELECT list.
	2	Remove the table aliases from the WHERE clause.
	3	Remove the table alias from the ORDER BY clause and use only the column name.
	4	Use the table name in the ORDER BY clause.
+	5	Use the table aliases instead of the table names in the WHERE clause.

26 S	0.000	281473913978898	03:49:09	03:50:29	01:20	80.838

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?

	1	SELECT book_title FROM books WHERE price between 500 and 900 AND purchase_date < '21-JAN-2001' ORDER BY purchase_date;
	2	SELECT book_title FROM books WHERE (price < 500 OR price > 900) AND purchase_date < '21-JAN-2001' ORDER BY purchase date DESC;
-	3	SELECT book_title FROM books WHERE price < 500 or > 900 AND purchase_date < '21-JAN-2001' ORDER BY purchase date DESC;
	4	SELECT book_title FROM books WHERE price IN (500,900) AND purchase_date < '21-JAN-2001' ORDER BY purchase date ASC;

27 S	0.000	281473913978898	04:10:26	04:11:13	00:47	46.767

For which task would you use the WHERE clause in a SELECT statement?

explanation

You can use the WHERE clause in the SELECT statement to implement the condition on the statement by comparing values.

	1	to designate the ORDER table location
-	2	to restrict the rows returned by a GROUP BY clause
	3	to display only unique PRODUCT_ID values
	4	to compare PRODUCT_ID values to 7382

28 S	0.000	0	03:27:08	--:--:--	--:--	0

The STUDENT_GRADES table has these columns:
STUDENT_ID NUMBER(12)
SEMESTER_END DATE
GPA NUMBER(4,3)
The registrar has requested a report listing the students' grade point averages (GPA), sorted from highest grade point average to lowest within each semester, starting from the earliest date. Which statement accomplishes this?

	1	SELECT student_id, semester_end, gpa FROM student_grades ORDER BY gpa DESC, semester_end DESC;
	2	SELECT student_id, semester_end, gpa FROM student_grades ORDER BY gpa DESC, semester_end ASC;.
	3	SELECT student_id, semester_end, gpa FROM student_grades ORDER BY semester_end ASC, gpa ASC;
	4	SELECT student_id, semester_end, gpa FROM student_grades ORDER BY semester_end, gpa DESC;

29 M	0.600	281473913978898	04:13:24	04:13:28	00:04	4.112

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 nnmher, 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? (Choose two.)

explanation

Answers C and E provide correct results to show. You can use BETWEEN or comparison operations to retrieve data.

+	1	SELECT customer_id, order_id, order_total FROM orders WHERE order_total >= 100 and <= 2000;
-	2	SELECT customer_id, order_id, order_total FROM orders WHERE order_total BETWEEN 100 and 2000;
-	3	SELECT customer_id, order_id, order_total FROM orders WHERE order_total >= 100 and order_total <= 2000;
+	4	SELECT customer_id, order_id, order_total FROM orders RANGE ON order_total (100 AND 2000) INCLUSIVE;
+	5	SELECT customer_id, order_id, order_total FROM orders HAVING order_total BETWEEN 100 and 2000;

30 S	0.000	281473913978898	04:08:12	04:09:08	00:56	55.912

Examine the structure of the PRODUCT table.
PRODUCT Table
PRODUCT _ID NUMBER NOT NULL, Primary Key
PRODUCT_NAME VARCHAR2
(25)
SUPPLIER_ID NUMBER Foreign key to SUPPLIER_ID of the
SUPPLIER table
CATERORY_ID NUMBER
QTY_PER_UNIT NUMBER
LIST_RRICE NUMBER (5,2)
COST NUMBER (5,2)
You want to display all product identification numbers of products for which there are 500 or more available for immediate sale. You want the product numbers displayed alphabetically by supplier, then by product number from lowest to highest. Which statement should you use to achieve the required results?

explanation

This statement will give the product_id from product table where qty_per_unit will be equal to and greater than 500 and it will sort it in ascending order by default.

	1	SELECT product_id FROM product WHERE qty_per_unit > 500 SORT BY supplier_id, product_id;
	2	SELECT product_id FROM product WHERE qty_per_unit >= 500 ORDER BY supplier_id, product_id;
-	3	SELECT product_id FROM product WHERE qty_per_unit >= 500 ORDER BY supplier_id, product_id DESC;
	4	SELECT product_id FROM product WHERE qty_per_unit >= 500 SORT BY supplier_id, product_id;

31 S	0.000	281473913978898	04:06:11	04:08:07	01:56	6.194

The ITEM table contains these columns:
ITEM_ID NUMBER(9)
COST NUMBER(7,2)
RETAIL NUMBER(7,2)
You need to create a report that displays the cost, the retail price, and the profit for item number 783920. To calculate the profit, subtract the cost of the item from its retail price, and then deduct an administrative fee of 25 percent of this derived value.
Which SELECT statement produces the desired results?

-	1	SELECT cost, retail, (retail - cost - retail - cost) * .25 "Profit" FROM item WHERE item_id = 783920;
	2	SELECT cost, retail, retail - cost - retail - cost * .25 "Profit" FROM item WHERE item_id = 783920;

	3	SELECT cost, retail, (retail - cost) - ((retail - cost) * .25) "Profit" FROM item WHERE item_id = 783920;
	4	SELECT cost, retail, (retail - cost) - retail - (cost * .25) "Profit" FROM item WHERE item_id = 783920;

32 S	0.000	281473913978898	04:02:11	04:05:58	03:47	227.306

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?

explanation

The operator in the WHERE clause should be changed to display the desired results. There are times when you want to substitute a value in place of NULL. Oracle provides this functionality with a special function, called NVL(). You cannot use operation equal with NULL, but you can achieve desired results using NVL() function after the WHERE clause.

	1	The WHERE clause should be changed to use an outer join to display the desired results.
	2	The column in the WHERE clause should be changed to display the desired results.
-	3	The SQL statement displays the desired results.
	4	The operator in the WHERE clause should be changed to display the desired results.

33 M	1.000	281473913978898	03:33:20	03:36:54	03:34	213.938

Which two statements are true about WHERE and HAVING clauses? (Choose two)

explanation

HAVING clause to specify which groups are to be displayed and thus further restrict the groups on the basis of aggregate information.The Oracle server performs the following steps when you use the Having clause
1. rows are grouped
2. the group function is applied to the group
3. the group that match the criteria in the Having clause are displayed.
WHERE clause cannot be use to restrict groups
HAVING clause use to restrict groups
WHERE clause cannot be use when there is group functions.

+	1	A WHERE clause CANNOT be used in a query of the query uses a HAVING clause.
+	2	A HAVING clause can be used to restrict groups only.
+	3	A HAVING clause CANNOT be used in subqueries.
+	4	A WHERE clause can be used to restrict both rows and groups.
+	5	A HAVING clause can be used to restrict both rows and groups.
+	6	A WHERE clause can be used to restrict rows only.

34 S	0.000	281473913978898	03:42:38	03:43:25	00:47	46.604

You are sorting data in a table in you SELECT statement in descending order. The column you are sorting on contains NULL records, where will the NULL record appears?

explanation

When sorting a column with null values in ascending order then the oracle places the Null values at the end of the list if the sorting is in descending order the oracle places the null values at the start of the list.

-	1	At the same location they are listed in the unordered table.
	2	In the middle of the list.
	3	At the beginning of the list.
	4	At the end of the list.

35 S	1.000	281473913978898	04:09:43	04:10:13	00:30	29.27

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?

	1	SELECT student_id, gpa FROM student_grades SORT ORDER BY gpa;
	2	SELECT student_id, gpa FROM student_grades ORDER BY gpa ASC;
+	3	SELECT student_id, gpa FROM student_grades ORDER BY gpa DESC;
	4	SELECT student_id, gpa FROM student_grades SORT ORDER BY gpa DESC;
	5	SELECT student_id, gpa FROM student_grades ORDER BY gpa;
	6	SELECT student_id, gpa FROM student_grades SORT ORDER BY gpa ASC;

36 S	1.000	281473913978898	03:37:31	03:38:43	01:12	72.276

The EMPLOYEES table contains these columns:
EMPLOYEE_ID NUMBER(4)
LAST_NAME VARCHAR2 (25)
JOB_ID VARCHAR2(10)
You want to search for strings that contain 'SA_' in the JOB_ID column. Which SQL statement do you use?

	1	SELECT employee_id, last_name, job_id FROM employees WHERE job_id LIKE '%SA_';
	2	SELECT employee_id, last_name, job_id FROM employees WHERE job_id LIKE '%SA_' ESCAPE "\";
+	3	SELECT employee_id, last_name, job_id FROM employees WHERE job_id LIKE '%SA_%' ESCAPE '\';
	4	SELECT employee_id, last_name, job_id FROM employees WHERE job_id = '%SA_';

37 S	1.000	281473913978898	03:38:43	03:40:07	01:24	83.361

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)
A promotional sale is being advertised to the customers in France. Which WHERE clause identifies customers that are located in France?

	1	WHERE lower(country_address) IS 'france'
	2	WHERE lower(country_address) = '%france%'
	3	WHERE lower(country_address) = "france"
+	4	WHERE lower(country_address) = 'france'
	5	WHERE lower(country_address) LIKE %france%

38 S	1.000	281473913978898	03:24:22	03:30:14	05:52	124.502

The PRODUCTS table has these columns:
PRODUCT_ID NUMBER(4)
PRODUCT_NAME VARCHAR2(45)
PRICE NUMBER(8,2)
Evaluate this SQL statement:
SELECT * FROM PRODUCTS ORDER BY price, product_name;
What is true about the SQL statement?

explanation
the result is sort by price which is numeric and follow by product_name which is alphabetically.

	1	The results are not sorted.
+	2	The results are sorted numerically and then alphabetically.
	3	The results are sorted numerically.
	4	The results are sorted alphabetically.

39 S	0.000	281473913978898	04:12:00	04:12:15	00:15	14.591

Evaluate these two SQL statements:

SELECT last_name, salary, hire_dateFROM EMPLOYEES ORDER BY salary DESC;
SELECT last_name, salary, hire_dateFROM EMPLOYEES ORDER BY 2 DESC;

What is true about them?

explanation
the two statement produce identical results as ORDER BY 2 will take the second column as sorting column.

	1	The two statements produce identical results.
-	2	The two statements can be made to produce identical results by adding a column alias for the salary column in the second SQL statement.
	3	There is no need to specify DESC because the results are sorted in descending order by default.
	4	The second statement returns a syntax error.

40 S	0.000	0	04:11:56	--:--:--	--:--	0

The STUDENT_GRADES table has these columns
STUDENT_ID NUMBER(12)
SEMESTER_END DATE
GPA NUMBER(4,3)
Which statement finds students who have a grade point average (GPA) greater than 3.0 for the calendar year 2001?

	1	SELECT student_id, gpa FROM student_grades WHERE semester_end BETWEEN '01-JAN-2001' AND '31-DEC-2001'

		OR gpa > 3.0;
	2	SELECT student_id, gpa FROM student_grades WHERE semester_end BETWEEN '01-JAN-2001' AND '31-DEC-2001' AND gpa > 3.0;
	3	SELECT student_id, gpa FROM student_grades WHERE semester_end > '01-JAN-2001' OR semester_end < '31-DEC-2001' AND gpa >= 3.0;
	4	SELECT student_id, gpa FROM student_grades WHERE semester_end BETWEEN '01-JAN-2001' AND '31-DEC-2001' OR gpa > 3.;
	5	SELECT student_id, gpa FROM student_grades WHERE semester_end BETWEEN '01-JAN-2001' AND '31-DEC-2001' AND gpa gt 3.0;

topics

points	correct points	module correct	topic
20.267 / 40 (51%)	21 / 40 (53%)	**Oracle9i**	
	1 / 1 (100%)	1 / 1 (100%)	Ora 4-1
	3 / 9 (33%)	3 / 9 (33%)	Writing Basic SQL Select Statements 4-1
	1 / 2 (50%)	1 / 2 (50%)	Writing Basic SQL Select Statements 5-2
	2 / 2 (100%)	2 / 2 (100%)	Writing Basic SQL Select Statements 6-1
	4 / 7 (57%)	4 / 7 (57%)	Writing Basic SQL Select Statements 5-1
	0.667 / 1 (67%)	1 / 1 (100%)	Writing Basic SQL Select Statements 6-2
	4 / 12 (33%)	4 / 12 (33%)	Restricting and Sorting Data 4-1
	2 / 3 (67%)	2 / 3 (67%)	Restricting and Sorting Data 5-1
	0.6 / 1 (60%)	1 / 1 (100%)	Restricting and Sorting Data 5-2
	1 / 1 (100%)	1 / 1 (100%)	Restricting and Sorting Data 6-2
	1 / 1 (100%)	1 / 1 (100%)	Restricting and Sorting Data 6-1