

**Question # 1**

Replay

Ch

We are required to run a set of commands to process a set of records when using explicit cursors. List the correct sequence of commands to process a set of records when using explicit cursors as described above.



Revisit

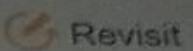
Is when using explicit  
records when using



Choose the best option

- INITIALIZE, GET, CLOSE
- CURSOR, GET, FETCH, CLOSE
- OPEN, FETCH, CLOSE
- CURSOR, FETCH, CLOSE

## Question # 2



Revisit

Choose the best option

- Left
- Right
- Center
- Both sides

If a "right outer join" in PL SQL does not find matching rows, it shows NULL results of the table on which side in this scenario?

May22-Wings1-DCA-PLSQL 1

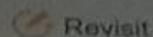
Section 1 of 1 Section #1

Total 100 Points

3

Question # 3

Which among the following scenarios is true about implicit cursors?

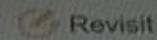


Revisit

Choose the best option

- Implicit cursors are used for SQL statements that are not named.
- Developers should use implicit cursors with great care.
- Implicit cursors are used in cursor for loops to handle data processing.
- Implicit cursors are no longer a feature in Oracle.

**Question #4**



### Revisit

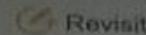
We are working with TRUNCATE statements and would like to understand which of the following are correct with regards to TRUNCATESs in SQL when compared to the DELETE statement. Which among the below options correctly highlights the comparison?

### **Choose the best option**

- It is usually slower than DELETE command
  - It is usually faster than DELETE command
  - There is no comparison between DELETE & TRUNCATE
  - Truncate command can be rolled back

Section 1 of 1 Section #1

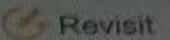
1 2 3 4 5 6 7 8 9 10 5 of 40

**Question # 5**

Revisit

**Choose the best option**

- CREATE PROCEDURE myproc (x NUMBER, y NUMBER) IS
- CREATE PROCEDURE (x NUMBER, y NUMBER) myproc IS
- CREATE PROCEDURE myproc IS (x NUMBER, y NUMBER)
- CREATE PROCEDURE IS myproc (x NUMBER, y NUMBER)

**Question # 6****Choose the best option**

We are working with TRUNCATE, DELETE and DROP statements and would like to understand which of the following statement(s) is/are true about TRUNCATE, DELETE and DROP in PL SQL?

1. *DELETE operation can be rolled back but TRUNCATE and DROP operations cannot be rolled back.*
2. *DELETE operation cannot be rolled back but TRUNCATE and DROP operations can be rolled back.*
3. *DELETE is an example of DML (Data Manipulation Language) but remaining are the examples of DDL (Data Definition Language).*
4. *All are an example of DDL*

 1 and 3 2 and 3 1 and 4 2 and 4

**Question # 7**

Revisit

**Choose the best option**

- Assume that you are required to calculate the average of the first ten natural numbers. Before finding the average, you need to calculate the sum of all the ten numbers.
- Which PL/SQL block will you select from the listed choices to print the sum of the first 10 numbers?
- DECLARE  
num NUMBER := 1, sum NUMBER = 0;  
BEGIN  
WHILE num <= 10  
LOOP sum:= sum + 1;  
DBMS\_OUTPUT.PUT\_LINE(sum);  
num := num+1;   
END LOOP;  
END;
  - DECLARE  
num NUMBER := 1; sum NUMBER := 0;  
BEGIN  
WHILE num <= 10  
LOOP sum:= sum + num;  
DBMS\_OUTPUT.PUT\_LINE(num);  
num := num+1;  
END LOOP;  
END;

average of the first ten natural numbers. Before  
the sum of all the ten numbers.

stated choices to print the sum of the first 10

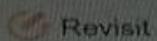
END LOOP;  
END;

DECLARE

```
num NUMBER := 1; sum NUMBER := 0;
BEGIN
  WHILE num <= 10
    LOOP sum := sum + num;
    DBMS_OUTPUT.PUT_LINE(sum);
    num := num+1; 
  END LOOP;
END;
```

DECLARE

```
num NUMBER := 1;
BEGIN
  WHILE num <= 10
    LOOP
    DBMS_OUTPUT.PUT_LINE(num);
    num := num+1;
  END LOOP;
END;
```

**Question #8**

Revisit

**Choose the best option**

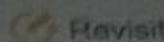
- replace function Product(x in number, y in number)  
return number  
is  
mult number(8);  
begin  
mult :=x\*y;  
return mult;  
end;
- create function Product(x in number, y in number)  
return number  
is  
mult number(8);  
begin  
mult :=x\*y;  
end;
- create function Product(x in number, y in number)  
is



sales of a product he made on a  
the product sold and the price per unit  
ion to calculate the total sales by  
m as the parameters to the function?

- begin  
mult :=x\*y;  
end;
- create function Product(x in number, y in number)  
is  
mult number(8);  
begin  
mult :=x\*y;  
return mult;  
end;
- create function Product(x in number, y in number)  
return numbe  
is  
mult number(8);  
begin  
mult :=x\*y;  
return mult;  
end;



**Question #9****Choose the best option**

What is the value of customer\_id within the nested block in the example below?

- 678
- 10000
- AP56
- 467.87

```
/* Start main block */
DECLARE
customer_id NUMBER(9) := 678;
credit_limit NUMBER(10,2) := 10000;
BEGIN
/* Start nested block */
DECLARE
customer_id VARCHAR2(9) := 'AP56';
current_balance NUMBER(10,2) := 467.87;
BEGIN
```

— what is the value of customer\_id at this point?

NULL;

END;

END;

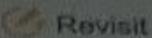
Please select the best answer.



Section 1 of 1 Section #1

1 2 3 4 5 6 7 8 9 10 10 of 10

## Question # 10



Revisit

Choose the best option

- 50000
- 0
- 55000
- 45000

What is the new salary of Manisha printed on the output screen? (Note that salary is represented using the sal attribute)

```
CREATE TABLE EMPLOYEE (
    empld INTEGER PRIMARY KEY,
    name TEXT NOT NULL,
    sal INTEGER,
    comm INTEGER
);
```

```
INSERT INTO EMPLOYEE VALUES (1, 'Ravi', 30000,250);
INSERT INTO EMPLOYEE VALUES (2, 'Manisha',50000,0);
INSERT INTO EMPLOYEE VALUES (3, 'Alekhya',60000,NULL);
select * from EMPLOYEE;
```

delimiter \$\$

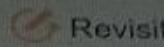
```
CREATE PROCEDURE raise()
```

BEGIN

UPDATE EMPLOYEE

set sal = sal+(10/100)\*sal

HERE comm IS NULL OR comm = 0;

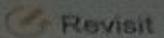
**Question # 11**

Revisit

**Choose the best option**

We are required to add  $y$  months to "x" date while writing a query in PL SQL. Which among the below functions can be used to meet this requirement?

- LAST\_DAY(x);
- ADD\_MONTHS(x, y);
- MONTHS\_BETWEEN(x, y);
- NEXT\_DAY(x, day);

**Question # 12**

Revisit

Choose the best option.

While working with **NULL** values in PL SQL, we come across a few scenarios during computation. What among the below is true for a Null value scenario in SQL?

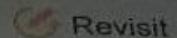
**Null + 1 = Null**

**Null + 1 = 1**

**Null \* 2 = Null**

**Null \* 2 = 0**

 1 and 3 2 and 4 1 and 4 2 and 3

**Question # 13**

Revisit

**Choose the best option**

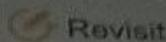
We are working with "HAVING" and "WHERE" clause statements and would like to understand which of the following statement(s) is/are true about "HAVING" and "WHERE" clause in PL SQL?

1. "WHERE" is always used before "GROUP BY" and HAVING after "GROUP BY"
2. "WHERE" is always used after "GROUP BY" and "HAVING" before "GROUP BY"
3. "WHERE" is used to filter rows but "HAVING" is used to filter groups
4. "WHERE" is used to filter groups but "HAVING" is used to filter rows

 1 and 3 1 and 4 2 and 3 2 and 4

Question # 14

We are required to support fetching of rows from start to end from a result set in PL SQL and are not required to go to the previous row in the result set. Which among the following PL SQL features can we use for the same?



Revisit

Choose the best option

- External cursors
- Implicit Cursors
- Forward cursors
- None of the above

Section 1 of 1 Section #1

8 7 6 5 10 11 12 13 14 15 ← 15 of 45 →

What is the correct output of the given data retrieval code? Select the correct answer from the given choices.

```
CREATE TABLE EMPLOYEE (
    empId INTEGER PRIMARY KEY,
    name TEXT NOT NULL,
    dept TEXT NOT NULL
);
INSERT INTO EMPLOYEE VALUES (1, 'Don', 'Sales');
INSERT INTO EMPLOYEE VALUES (2, 'Ravi', 'Accounts');
```

Choose the best option

- 0
- 1
- 2
- Exception

```
DELIMITER $$  
CREATE PROCEDURE myProc(in_id int)  
READS SQL DATA  
BEGIN  
SELECT name,dept  
FROM EMPLOYEE  
WHERE empId=in_id;  
END$$  
DELIMITER ;
```



Muralidhar | Support +1 8002656038



+91 80 4719 0902

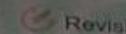


Type here to search

Section 1 of 1 Section #1

May22-Wings1-DCA-PLSQL 3

16 17 18 19 20 21 22 23 24 25 10 of 40

**Question # 16**

Tables A, B have three columns (namely: 'id', 'age', 'name') each. These tables have no 'null' values and there are 100 records in each of the table.

Below are two queries based on these two tables 'A' and 'B'.

Query1. `SELECT A.id FROM A WHERE A.age > ALL (SELECT B.age FROM B WHERE`

`B.name = 'Ankit')`

Query2. `SELECT A.id FROM A WHERE A.age > ANY (SELECT B.age FROM B WHERE`

`B.name = 'Ankit')`

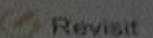
Now, which of the following statement is correct for the output of each query?

**Choose the best option**

- The number of tuples in the output of Query 1 will be more than or equal to the output of Query 2
- The number of tuples in the output of Query 1 will be equal to the output of Query 2
- The number of tuples in the output of Query 1 will be less than or equal to the output of Query 2
- None of the above



## Question # 17



Choose the best option

What is the correct output of the given code? Select the right answer from the given choices.

```
DECLARE
    loopcounter NUMBER := 1;
    loopresult NUMBER;
BEGIN
    WHILE loopcounter <= 10
LOOP
    loopresult := loopcounter + 100;
    dbms_output.put_line(loopresult);
    loopcounter := loopcounter +2;
END LOOP;
END
```

 100

102

104

106

108

110

 100

102

104

106

108

 101

103

105

107

109

 101

100

**Question # 17**

Revisit

105

106

110

 100

102

104

106

108

 101

103

105

107

109

 101

103

105

107

109

111

What is the correct output of the given code? Select the right answer from the given choices.

DECLARE

loopcounter NUMBER := 1;

loopresult NUMBER;

BEGIN

WHILE loopcounter <= 10

LOOP

loopresult := loopcounter + 100;

dbms\_output.put\_line(loopresult);

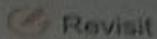
loopcounter := loopcounter +2;

END LOOP;

END

Section 1 of 1 Section #1

16 17 18 19 20 21 22 23 24 25 &lt; 18 of 45 &gt; 60

**Question # 18**

Revisit

*There is a requirement to execute a set of statements everytime we have a situation of SERVERERROR during database operations. Which of the following PL/SQL sub-program methods can be used to run these set of statements on SERVERERROR?*

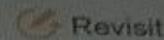
Choose the best option

- Recursive functions
- Parameter-based stored procedures
- Triggers
- Implicit cursors

Section 1 of 1 Section #1

16 17 18 19 20 21 22 23 24 25 < 19 of 40 >

Question # 19



Choose the best option

- 2147483648
- Error : Numeric Overflow
- 21474836471
- 2147483646

Assume that a software programmer has written the given code to find the sum. What will be the correct output of the given code snippet?

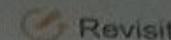
```
DECLARE
    num1 PLS_INTEGER := 2147483647;
    num2 PLS_INTEGER := 1;
    sum PLS_NUMBER;
BEGIN
    sum := num1 + num2;
END;
/
```

Select the correct answer from the given choices.



Section 1 of 1 Section #1

16 17 18 19 20 21 22 23 24 25 26 &lt; 25.04 AG &gt;



Revisit

## Question # 20

For the view- Create view instructor\_info as

SELECT ID, name, building

FROM instructor, department

WHERE instructor.dept name= department.dept name;

If we insert tuple into the view as insert into instructor info values ('69987', 'White', 'Taylor');

What will be the values of the other attributes in instructor and department relations?

Choose the best option

 Default value NULL ERROR 0

May22-Wings1-DCA-PLSQL-1

Section 1 of 1 Section #1 ▾

21

22

23

24

25

26

27

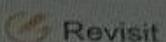
28

29

30

21 of 40 >

Question # 21



We are required to execute a set of PL SQL code lines by itself until it reaches some boundary condition so the programmers can use the same set of code any number of times. Which PL SQL utility from among the below can help achieve this?

Choose the best option

- Functions
- Stored procedure
- Recursive stored procedure
- None of the above

**Question # 22**



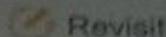
## Revisit

*Consider a scenario where an index scan is replaced by sequential scan in SQL, then what will happen or what would be the possible outcomes from the below options?*

*Note: Number of observations is equal to 1 million.*

**Choose the best option.**

- Execution will be faster
  - Execution will be slower
  - Execution will not be affected
  - None of the above

**Question # 23**

Revisit

**Choose the best option**

- Usually, there are functions that may accidentally include infinite loops instead of finite loops;
1. Which element in pl/sql is used for infinite loops in a function?
  2. How do you use the above element in pl/sql to define it?

Analyze the given choices and select the correct set of answers.

- 1. Using package called sf\_loop  
2.  
BEGIN  
loop\_killer.kill\_after(100)  
LOOP  
DBMS\_OUTPUT.put\_line(loop\_killer,  
END LOOP;  
END;
- 1. Using package called end\_loop  
2.  
BEGIN  
loop\_killer.kill\_after(100)  
LOOP  
DBMS\_OUTPUT.put\_line( curr  
END LOOP;  
END;
- 1. Using package called stop  
2.

dentally include infinite loops instead of finite loops;  
e loops in a function?

/sql to define it?

rect set of answers.

```
LOOP  
DBMS_OUTPUT.put_line( current_count);  
END LOOP;  
END;  
 1. Using package called stop_loop  
2.  
BEGIN  
stop_loop.kill_after(100)  
LOOP  
DBMS_OUTPUT.put_line(loop_killer, current_count);  
END LOOP;  
END;  
 1. Using package called sf_loop_killer  
2.  
BEGIN  
loop_killer.kill_after(100)  
LOOP  
DBMS_OUTPUT.put_line(loop_killer, current_count);  
END LOOP;  
END;
```

Section 1 of 1 Section #1

21 22 23 24 25 26 27 28 29 30 < 24 of 40 >

All

### Question # 24

Revisit

Consider the relation T1 (A, B) in which (A, B) is the primary key and the relation T2 (A, C) where A is the primary key. Assume there are no null values and no foreign keys or integrity constraints.

Now, which of the following option is correct related to following queries?

Query 1: select A from T1 where A in (select A from T2)

Query 2: select A from T2 where A in (select A from T1)

Choose the best option

- Both queries will definitely give the same result
- Both queries may give the same result
- Both queries will definitely give a different result
- None of the above



21 22 23 24 25 26 27 28 29 30 &lt; 25 of 40 &gt;

All

**Question # 25**

Revisit

We are required to convert the TIMESTAMP WITH TIMEZONE x to a TIMESTAMP containing the date and time in UTC. Which among the below methods can we use to meet this requirement?

**Choose the best option**

- LOCALTIMESTAMP();
- CURRENT\_TIMESTAMP();
- SYS\_EXTRACT\_UTC(x);
- FROM\_TZ(x, time\_zone);



26 27 28 29 30 31 32 33 34 35 36 37 38 39 40

Question # 26

In PL SQL, we are required to find all the unique students who have taken more than one course. Which of the following queries can be used in this scenario?

Revisit

Choose the best option

- SELECT DISTINCT e1.sid FROM enrolled AS e1, enrolled AS e2 WHERE e1.sid = e2.sid  
AND e1.cid != e2.cid
- SELECT DISTINCT e1.sid FROM enrolled AS e1, enrolled AS e2 WHERE e1.sid = e2.sid  
AND e1.cid = e2.cid
- SELECT DISTINCT e1.sid FROM enrolled AS e1, enrolled AS e2 WHERE e1.sid != e2.sid  
AND e1.cid != e2.cid
- SELECT DISTINCT e1.sid FROM enrolled AS e1, enrolled AS e2 WHERE e1.sid = e2.sid  
AND e1.cid != e2.cid

## Question # 27

We are required to write a SQL query to get the third-highest salary of an employee from the `employee_table`. Which among the below queries can we use in this case?

Revisit

## Choose the best option

- SELECT TOP 1 salary  
FROM(  
SELECT TOP 3 salary  
FROM employee\_table  
ORDER BY salary DESC) AS emp  
ORDER BY salary ASC;
- SELECT TOP 3 salary  
FROM  
FROM employee\_table  
ORDER BY salary DESC
- SELECT TOP 3 salary  
FROM employee\_table  
ORDER BY salary ASC;
- None of the above

Section 1 of 1 Section Art -

26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80 81 82 83 84 85 86 87 88 89 90 91 92 93 94 95 96 97 98 99 100

**Question # 28**

Revisit

Choose the best option:

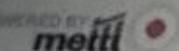
- 12
- 16
- 0
- 24

What is the correct output of the given dynamic SQL statement?

```
CREATE PROCEDURE MathOper (
    x NUMBER,
    y NUMBER,
    z NUMBER )
IS
BEGIN
    DBMS_OUTPUT.PUT_LINE((x + y) * z);
END;
/
```

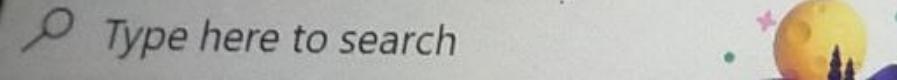
```
DECLARE
    a NUMBER := 2;
    b NUMBER := 4;
    plsql_block VARCHAR2(100);
BEGIN
    plsql_block := 'BEGIN MathOper(:x, :x, :y); END;';
    EXECUTE IMMEDIATE plsql_block USING a, b;
END;
```

Analyze the listed options and select the correct answer.

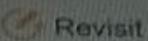


Muralidhar | Support +1 8002656038

+91 80 4719 0902



## Question # 29



Revisit

Choose the best option

Consider there is a Student table as shown in the figure. Assume that you have to insert a component such that the ages are all automatically updated with a default value of 19 irrespective of the insert value. Which option from the given choices will perform this task appropriately?

STID	SNAME	ADDRESS	AGE	GENDER
D		address4	20	male
A		address1	20	Female
B		address2	20	male
C		address3	20	female

- CREATE TRIGGER st\_age\_trig  
FOR INSERT  
ON student  
COMPOUND TRIGGER  
BEFORE EACH ROW IS  
BEGIN  
:old\_age=19;  
END BEFORE EACH ROW;  
END st\_age\_trig;  
/
- CREATE TRIGGER st\_age  
ON student  
COMPOUND TRIGGER  
BEGIN  
:new.age:=19;  
END BEFORE EACH ROW;  
END st\_age\_trig;  
/

wn in the figure. Assume that you have to insert a  
automatically updated with a default value of 19  
ion from the given choices will perform this task

ADDRESS	AGE	GENDER
20		male
20		Female
20		male
20		female

END st\_age\_trig;

- CREATE TRIGGER st\_age\_trig FOR INSERT  
ON student

BEGIN

:new.age:=19;

END BEFORE EACH ROW;

END st\_age\_trig;

/

- CREATE TRIGGER st\_age\_trig  
FOR INSERT

ON student

BEFORE EACH ROW IS

BEGIN

:new.age:=19;

END BEFORE EACH ROW;

END

/



May22-Wings1-DCA-PLSQL

Section 1 of 1 Section #1 >

**Question # 30**

What is the proper output of the given pl/sql code? Select the correct answer from the given choices.

## Revisit

**Choose the best option**

```
DECLARE
  TYPE Ar_Type IS VARRAY(10) OF NUMBER;
  v_Num Ar_Type := Ar_Type();
BEGIN
  v_Num.EXTEND(4);
```

- 1 1  
○ 3 2  
○ 2 3  
○ 4 -1

```
v_Num(1) := 11;  
v_Num(2) := 21;  
v_Num(3) := 31;  
v_Num(4) := 41;
```

```
DBMS_OUTPUT.PUT_LINE(NVL(v_Num.prior (3400), -1));
DBMS_OUTPUT.PUT_LINE(NVL(v_Num.next (3400), -1));
END;
```



31 32 33 34 35 36 37

Topic 0079-01 Thread

## Question #34



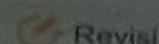
Consider the student table as shown in the figure. Assume that you need to update the age such that it triggers to call a message that displays the change in age.

STID	NAME	ADDRESS	AGE	GENENDER
student1	John Doe	123 Main St	20	male
student2	Jane Doe	123 Main St	20	Female
student3	John Doe	123 Main St	20	male
student4	Jane Doe	123 Main St	20	female

Which option from the listed choices will perform this task appropriately?

Choose the best option

- CREATE OR REPLACE TRIGGER age\_update  
BEFORE DELETE OR INSERT OR UPDATE ON student  
FOR EACH ROW  
WHEN (NEW.age > 0)  
DECLARE  
age\_diff number;  
BEGIN  
age\_diff := :NEW.age - :OLD.age;  
dbms\_output.put\_line('Old age: ' || :OLD.age);  
dbms\_output.put\_line('New age: ' || :NEW.age);  
dbms\_output.put\_line('age difference: ' || age\_diff);  
END;  
/
- CREATE OR REPLACE TRIGGER  
BEFORE DELETE OR INSERT OR UPDATE ON student  
WHEN (NEW.age > 0)  
DECLARE



ent table as shown in the figure. Assume that you need to update the age to call a message that displays the change in age.

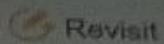
NAME	ADDRESS	AGE	GENDER
JohnDoe	123 Main St	20	male
JaneDoe	456 Elm St	20	Female
MikeSmith	789 Oak St	20	male
SarahSmith	123 Elm St	20	female

ed choices will perform this task appropriately?

- CREATE TRIGGER INSERT OR UPDATE ON student FOR EACH ROW
 

```
WHEN (NEW.age > 0)
DECLARE
age_diff number;
BEGIN
age_diff := :NEW.age - :OLD.age;
dbms_output.put_line('Old age: ' || :OLD.age);
dbms_output.put_line('New age: ' || :NEW.age);
dbms_output.put_line('age difference: ' || age_diff);
END;
/
```
- CREATE TRIGGER WHEN (NEW.age > 0)
 

```
DECLARE
age_diff number;
BEGIN
age_diff := :age - age;
dbms_output.put_line('Old age: ' || :OLD.age);
dbms_output.put_line('New age: ' || :NEW.age);
dbms_output.put_line('age difference: ' || age_diff);
```

**Question # 32**

Choose the best option

- T1>T2
- T2>T1
- T1-T2
- None of the above

Suppose we have a table 'Employee'. In Employee table, we have a column named Salary.  
Now, we apply Query1 on Employee table.

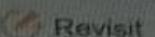
Query 1: `SELECT * FROM Employee where Salary*100 > 5000;`

After that, we create an index on Salary columns and then we re-run the Query 2 (same as  
Query 1).

Query 2: `SELECT * FROM Employee where Salary*100 > 5000;`

Here, Query 1 is taking T1 time and Query 2 is taking T2 time.

Which of the following is true for the queries time?

**Question # 33****Choose the best option:**

- No Data returned
- Returns 1 row from Employee table
- Deletes 1 row from EmpLog table
- Displays 2 rows from EmpLog table

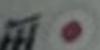
In the given SQL statement, what is the output of the final select statement (SELECT \* FROM EmpLog?)

```
CREATE TABLE Employee  
(Empid NUMBER(10) NOT NULL,  
EmpName VARCHAR2(50) NOT NULL,  
Salary NUMBER(10) NOT NULL,  
Dept NUMBER(3) NOT NULL  
);
```

```
INSERT INTO Employee VALUES(1,'Meenu',30000, 10);  
INSERT INTO Employee VALUES(2,'ANIKA',45000, 10);  
INSERT INTO Employee VALUES(3,'JEFF',67500, 40);  
COMMIT;
```

```
CREATE TABLE Emplog  
(IEmpid NUMBER(10) NOT NULL,  
IEmpName VARCHAR2(50) NOT NULL,  
IDept NUMBER(3) NOT NULL
```

```
CREATE OR REPLACE TRIGGER UtrigEmpLog
```



```
INSERT INTO Employee VALUES(2,'ANIKA',45000, 10);
INSERT INTO Employee VALUES(3,'JEFF',67500, 40);
COMMIT;
```

```
CREATE TABLE Emplog
(IEmpid NUMBER(10) NOT NULL,
IEmpName VARCHAR2(50) NOT NULL,
IDept NUMBER(3) NOT NULL
);
CREATE OR REPLACE TRIGGER UtrigEmpLog
BEFORE DELETE ON Employee
FOR EACH ROW
BEGIN
INSERT INTO Emplog VALUES(:OLD.EmpId,:OLD.EmpName, :OLD.Dept);
END;
/
DELETE FROM Employee
WHERE dept = 10;
COMMIT;
SELECT * FROM EmpLog;"
```

Select the correct answer from the choices listed.

Choose the best option

- No Data returned
- Returns 1 row from Employee table
- Deletes 1 row from EmpLog table
- Displays 2 rows from EmpLog table

**Question # 34**

Revisit

**Choose the best option**

We have a table "Loan\_Records" with the following data -

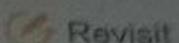
Table header - (Borrower, Bank Manager, Loan Amount)

Table rows - (Ramesh, Sundar, 10000),(Suresh, Ram,5000),(Mahesh,Sundar,7000)

What would be the output of the following SQL query-

```
SELECT Count(*) FROM ( (SELECT Borrower, Bank_Manager FROM Loan_Records) AS S NATURAL JOIN (SELECT Bank_Manager, Loan_Amount FROM Loan_Records) AS T );
```

 4 5 8 10

**Question # 35****Choose the best option**

Consider a scenario where we run the following Queries in the below order:

Create a table "Me" using the below SQL query.

Query1: Create table Me(name varchar(20), salary int);

Next, we create a view based on "Me" table by using the following query.

Query2: Create view me\_view as select name from me;

Finally, we run the following query:

Query3: DROP TABLE Me CASCADE;

Query4: select \* from me\_view;

Which of the following statements are true in this scenario?

1. Query3 will give an error
2. Query3 will run smoothly
3. Query4 will give an error

Query4 will run smoothly

 1 and 3 1 and 4 2 and 3 2 and 4

**Question # 36**

Revisit

Which option from the listed choices indicates the correct code that calls the given code and prints the output as "Hi Welcome ALL".

```
public class Welcome
{
    public static String greet()
    {
        return "Welcome ALL";
    }
}
```

**Choose the best option**

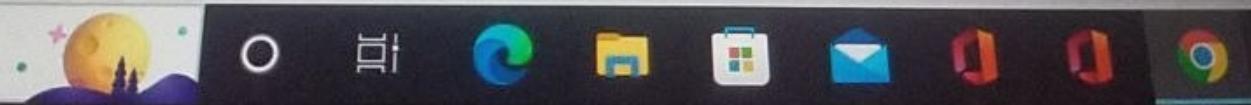
- create or replace
 

```
FUNCTION welcome RETURN VARCHAR2 AS
LANGUAGE JAVA NAME 'Welcome.greet () return java.lang.String';
```

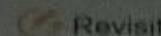
 Declare
 my\_string varchar2(200 char);
 begin
 my\_string:=welcome();
 dbms\_output.put\_line('Hi ' || my\_string);
 end;
- create or replace
 

```
FUNCTION welcome RETURN VARCHAR2 AS
LANGUAGE JAVA NAME 'Welcome.greet () return java.lang.String';
```

 Declare
 my\_string varchar2(200 char);
 begin
 dbms\_output.put\_line(my\_string);



n # 36



ion from the listed choices indicates the correct code that calls the given code and  
output as "Hi Welcome ALL".

Welcome

String greet()

ome ALL";

Declare

```
my_string varchar2(200 char);
begin
dbms_output.put_line(my_string);
end;
```

create or replace

```
FUNCTION welcome RETURN VARCHAR2 AS
LANGUAGE JAVA NAME 'Welcome.greet ()';
```

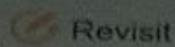
Declare

```
my_string varchar2(400 char);
begin
my_string:=welcome();
dbms_output.put_line(my_string);
end;
```

create or replace

```
FUNCTION welcome RETURN VARCHAR2 AS
LANGUAGE JAVA NAME
return java.lang.String';
```

Declare



Revisit

ices indicates the correct code that calls the given code and  
'ALL'.

create or replace

```
FUNCTION welcome RETURN VARCHAR2 AS
LANGUAGE JAVA NAME 'Welcome.greet ()';
```

Declare

```
my_string varchar2(400 char);
begin
my_string:=welcome();
dbms_output.put_line(my_string);
end;
```

create or replace

```
FUNCTION welcome RETURN VARCHAR2 AS
LANGUAGE JAVA NAME
```

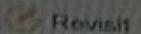


```
return java.lang.String';
```

Declare

```
my_string varchar2(400 char);
begin
my_string:=welcome();
dbms_output.put_line('Hi ' || my_string);
end;
```

## Question # 37

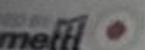


What is the correct output of the listed code? Select the appropriate answer from the given choices.

```
DECLARE
  TYPE StuRec IS RECORD (
    Stu_id  NUMBER(4) NOT NULL := 1001,
    Stu_name VARCHAR2(30) NOT NULL := 'John',
    Adm_id   NUMBER(6) := 800,
    Addr_id  NUMBER(4) := 1565
  );
  Stu_rec StuRec;
BEGIN
  DBMS_OUTPUT.PUT_LINE('Student_id: ' || Stu_rec.Stu_id);
  DBMS_OUTPUT.PUT_LINE('Student_name: ' || Stu_rec.Stu_name);
  DBMS_OUTPUT.PUT_LINE('Admission_id: ' || Stu_rec.Adm_id);
  DBMS_OUTPUT.PUT_LINE('Address_id: ' || Stu_rec.Addr_id);
END;
```

Choose the best option.

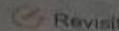
- Displays one student row.
- Displays only the student's name
- Displays two rows of students
- Displays three rows of students



31 32 33 34 35 36 37 38 39 40 &lt; 20 of 20

Total 0% Time Left

## Question # 38



We are required to identify, which of the following column "A" or "C" given in the below table is a "Primary Key" or "Foreign Key"?  
Table header - (A,C)

Table rows - (2,4),(3,4),(4,3),(5,2),(7,2),(9,5),(6,4)

Note: We have defined 'Foreign Key' and 'Primary Key' in a single table

Choose the best option

- Column 'A' is Foreign Key and Column 'C' is Primary Key
- Column 'C' is Foreign Key and Column 'A' is Primary Key
- Both can be 'Primary Key'
- Based on the above table, we cannot tell which column is 'Primary Key' and which is 'Foreign Key'

Muralidhar | Support +1 8002656038



+91 80 4719 0902



Prev Ques

Type here to search

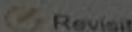


20°C



014 W 207

## Question # 39



Choose the best option

- 1 and 3
- 1 and 4
- 2 and 3
- 2 and 4

Consider a scenario where we run the following Queries in the below order:  
Create a table "Me" using the below SQL query.

Query1: Create table Me(name varchar(20), salary int);

Next, we create a view based on "Me" table by using the following query.

Query2: Create view me\_view as select name from me;

Finally, we run the following query:

Query3: Drop table Me;

Which of the following statements are true in this scenario?

1. Query3 will give an error
2. Query3 will run smoothly
3. Query2 will give an error.
4. Query2 will run smoothly



31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48

5

**Question # 40**

Suppose that you were creating a row level trigger, then which of the following is the correct possibility for old or new values in it?

Revisit

Choose the best option(s)

- 1 For Insert, New: Available  
2 Old: NULL
- 1 For Update, New: Available  
2 Old: NULL
- 1 For Delete, New: NULL  
2 Old: Available
- 1 For Delete, New: NULL  
2 Old: NULL



**Question # 1**

Replay

Ch

We are required to run a set of commands to process a set of records when using explicit cursors. List the correct sequence of commands to process a set of records when using explicit cursors as described above.