1. Give an example of hierarchical ordering. (https://docs.oracle.com/cd/B19306\_01/server.102/b14200/queries003.htm)

SELECT last\_name, employee\_id, manager\_id, LEVEL

FROM employees

START WITH employee\_id = 100

CONNECT BY PRIOR employee\_id = manager\_id

ORDER SIBLINGS BY last\_name;

1. Write a query to print the last name of each employee in department 110, each manager above that employee in the hierarchy, the number of levels between manager and employee, and the path between the two.

SELECT last\_name "Employee", CONNECT\_BY\_ROOT last\_name "Manager",

LEVEL-1 "Pathlen", SYS\_CONNECT\_BY\_PATH(last\_name, '/') "Path"

FROM employees

WHERE LEVEL > 1 and department\_id = 110

CONNECT BY PRIOR employee\_id = manager\_id;

1. Write a procedure to append all the deleted records to a new table.

CREATE TABLE dept\_temp AS SELECT \* FROM departments;

DECLARE

dept\_no NUMBER(4) := 270;

BEGIN

DELETE FROM dept\_temp WHERE department\_id = dept\_no;

IF SQL%FOUND THEN -- delete succeeded

INSERT INTO dept\_temp VALUES (270, 'Personnel', 200, 1700);

END IF;

END;

/

1. Triggers fire first or constriants?

It can be specified when to fire the trigger.

1. Diff between view and materialized views

Basic difference is that view is not stored physically unlike MV.

1. Importance of materialized views

MVs are like precomputed tables that are stored in the database, these are used so that original data remains intact in tables and the computed values in views and can be obtained at reduced cost as you do not need to rerun the query every time on the table.

These MVs can further be incrementally refreshed to get the fresh data from the table.

1. Procedure vs function

|  |  |
| --- | --- |
| Function | Procedure |
| Can be called through sql queries | Cannot |
| Can be called by a procedure | Cannot be called by a function |
| DMLs are not allowed | Allowed |
| Compiled each time when called | Compiled once called multiple times |
| Returns control and value | Cannot return the result value, need to use out parameter for that |
| Doesn’t support try catch | Does |
| Can be used in select stmt | cannot |

1. Triggers vs Procedures

|  |  |
| --- | --- |
| Triggers | Procedures |
| Implicitly invoked | Explicitly called |
| Can only be nested in a table, not inside another trigger | Procedural nesting is allowed |
| Transaction stmts like commit, rollback , savepoint not allowed | Allowed |
| Used to maintain referential integrity of a record by book-keeping of these activities | Used to perform tasks defined by the user |
| We cannot pass values as parameters to triggers, and cannot return values from triggers | We can return values as out params and pass values as well with in params. |

1. Triggers vs contraints

|  |  |
| --- | --- |
| Triggers | Constraints |
| Applies only to new data | Depends on the state –  [ [ [ **NOT** ] **DEFERRABLE** ]  [ **INITIALLY** { **IMMEDIATE** | **DEFERRED** } ]  | [ **INITIALLY** { **IMMEDIATE** | **DEFERRED** } ]  [ [ **NOT** ] **DEFERRABLE** ]  ]  [ **RELY** | **NORELY** ]  [ using\_index\_clause ]  [ **ENABLE** | **DISABLE** ]  [ **VALIDATE** | **NOVALIDATE** ]  [ exceptions\_clause ] |
| Complex statements | Easier to write and less error prone |
| Can be used for complex business logic | Used for simpler usecases |

1. How can you backup deleted rows in another table without using triggers?

You can create a procedure like below:

Declare

Type array is table of table1%rowtype;

rows array;

Begin

Delete from table1 where rownum < 50

Returning <all columns separated by comma>

Bulk collect into rows

Forall row in 1 .. rows.count

Insert into table2 values rows(row);

End

1. What are different types of collections in oracle and why are these used?

Collections make it easy for us to access the elements within them by either index or key.

Associative array (Index by table)

Nested Tables and

Variable sized array(Varray)

Index by table is a set of Key-Value pairs.

Syntax:

TYPE type\_name IS TABLE OF element\_type [NOT NULL] INDEX BY subscript\_type;

Example:

TYPE salary IS TABLE OF NUMBER INDEX BY VARCHAR2(20)

Complete example with usage:

DECLARE

TYPE salary IS TABLE OF NUMBER INDEX BY VARCHAR2(20);

salary\_list salary;

name VARCHAR2(20);

BEGIN

-- adding elements to the table

salary\_list('Rajnish') := 62000;

salary\_list('Minakshi') := 75000;

salary\_list('Martin') := 100000;

salary\_list('James') := 78000;

-- printing the table

name := salary\_list.FIRST;

WHILE name IS NOT null LOOP

dbms\_output.put\_line

('Salary of ' || name || ' is ' || TO\_CHAR(salary\_list(name)));

name := salary\_list.NEXT(name);

END LOOP;

END;

/

Nested table: A nested table is like a one-dimensional array with an arbitrary number of elements.

Syntax: TYPE type\_name IS TABLE OF element\_type [NOT NULL];

TYPE names\_table IS TABLE OF VARCHAR2(10);

Varray: Stands for variable sized array. It is a collection of variable number of similar types of elements fixed by the maximum number specified at the time of creation of the array variable.

Syntax:

CREATE OR REPLACE TYPE varray\_type\_name IS VARRAY(n) of <element\_type>

Example:

CREATE Or REPLACE TYPE namearray AS VARRAY(3) OF VARCHAR2(10);

1. What is the difference between associative arrays and simple arrays?

Simple arrays are indexed by numbers whereas associative arrays are key-value pairs and elements can be accessed by keys, which could be strings.

1. What is the role of AUTHID parameter in procedure?

It is used to identify what role/permissions should be assumed while executing the procedure, it could either be invoker or definer.

1. Scd types

Type1: the new data overwrites the existing data.

Type2: the new data creates a new row, each row has creation and expiration time.

Type3: the new data is stored in new\_value column and old\_value column is updated with older data, only 2 columns are maintained here.

If full history of data is needed, it is better to go with type2.

1. How do you calculate average of salary of each employee in each department when some values are not present for salary

Select avg(salary) over (partition by dept\_id order by hire\_date rows between 1 preceding and 1 following) from employees

1. Difference between first and first\_value analytic functions.

Say you wanted the average sal for each departement, but only for the earliest year (taken from the hiredate column) in department (that is, the column called f in the query below).

WITH got\_hireyear AS

(

SELECT deptno, ename, sal, hiredate

, EXTRACT (YEAR FROM hiredate) AS hireyear

FROM scott.emp

)

SELECT deptno, hireyear, hiredate, ename, sal

, AVG (sal) KEEP (DENSE\_RANK FIRST ORDER BY hireyear)

OVER ( PARTITION BY deptno

) AS f

, FIRST\_VALUE (sal) OVER ( PARTITION BY deptno

ORDER BY hireyear

) AS fv

, AVG (sal) OVER ( PARTITION BY deptno

, hireyear

) AS a

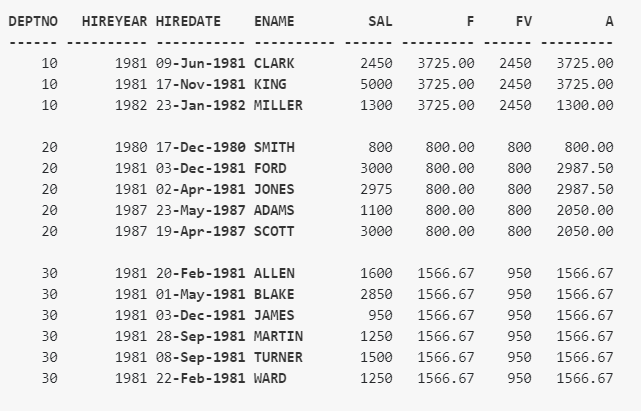
FROM got\_hireyear

ORDER BY deptno

, hireyear

, ename

;



1. Difference between rowid and rownum

Rowid remains unique, it is a unique identifier for a row.

Rownum is a temporary number generated on the go.

1. How do you replace a NAN value?

Use NANVL(<Binary\_float/Binary\_Double>,<NULL/0>)

1. How to get the previous salary of an employee by department without doing self join?

Use lag function with offset=1 and use ignore nulls if NULL salary is not acceptable like below.

SELECT hire\_date, name, salary, dept\_id,

LAG(salary,1,1) IGNORE NULLS OVER(partition by dept\_id order by hire\_date) prev\_sal

From my\_Table

Order by hire\_Date;

1. How do you get the list of all employees under a single manager in a single row?

Use LISTAGG

Select manager\_id, LISTAGG(name,’,’) WITHIN GROUP (ORDER BY hire\_date) as loe

From employees

Group by manager\_id

Order by manager\_id;

You can also use it as analytic function by using PARTITION BY clause with function instead of grouping at select.

Select manager\_id, LISTAGG(name,’,’) WITHIN GROUP (ORDER BY hire\_date) OVER (PARTITION by manager\_id) as loe

From employees

Order by manager\_id;

1. Differences between analyze table and gather table stats.

ANALYZE calculates global statistics for partitioned tables and indexes instead

of gathering them directly. This can lead to inaccuracies for some statistics, such as the number of distinct values. DBMS\_Stats won't do that.

Usage of histograms with method\_opt parameter will increase the speed of gathering stats of the table.

1. Difference between compute and estimate table stats

FTS happens if using compute. Only 10% of the table gets read if using the estimate.

1. What are chain rows?

Chain rows are the rows that span across multiple blocks

1. What are the 2 types of hits of oracle performance tuning

Library and buffer hit

1. Which is the quickest query method for a table?

The quickest query method for a table is to fetch by rowid

1. How to fix contention in the library cache?

Increase the shared pool size.

1. A query was performing okay for a few days, suddenly it is behaving poorly, what to do?

Gather table stats for the tables used in the query, if the table’s schema or data is modified a lot, the stats become inaccurate and the table should be analyzed and stats regenerated to make query faster.

1. How do you gather table stats of partition efficiently?

By using parameters like Incremental and Granularity

Use:

DBMS\_STATS.SET\_TABLE\_PREFS('COS\_DATA', 'LAR\_ALLOCATION\_PER\_PART', 'incremental', 'true');

But you need to gather this at default estimate percent.

You can use the Granularity option and set it to APPROX\_GLOBAL AND PARTITION which will gather the stats accurately for partitions when the table changes are less than 10%. You will have to gather table stats for entire table when the changes are higher than 10% at that time incremental stats collection would be easier and faster.

1. Oracle types of tuning

IO, DB, CPU, Application and Memory

IO Tuning: Database files must be sized correctly and located on the right place to afford supreme disk subsystem quantity. Also, look out for missing indexes, regular disk sorts, row chaining, data fragmentation, complete table scans, and so on.

Memory Tuning: Properly sizing the database buffers such as buffer cache, log buffer, shared pool, buffer cache by analyzing the ratios of buffer hit. Large objects are pinned into memory to avoid recurrent refills.

Application Tuning: About 80 percent of Oracle system performance issues are fixed using optimal SQL code. Batch tasks must be properly scheduled on time.

1. Types of optimizers

Cost based and role based

When a server does not have internal statistics with respect to the objects influenced by the statement, here RBO plays its role.

If internal statistics are available, CBO method is employed over there. CBO performs various checks on all the possible execution plans and picks one that has the bottommost charge depending on the system resources.

1. Exceptions in PLSQL

TimesTen or Predefined errors/exceptions:

NO\_DATA\_FOUND

VALUE\_ERROR

ZERO\_DIVIDE

TOO\_MANY\_ROWS

SYS\_INVALID\_ROWID

SUBSCRIPT\_OUTSIDE\_LIMIT

SUBSCRIPT\_BEYOND\_COUNT

STORAGE\_ERROR

PROGRAM\_ERROR  
ROWTYPE\_MISMATCH

INVALID\_CURSOR

INVALID\_NUMBER

DUP\_VAL\_ON\_INDEX

CURSOR\_ALREADY\_OPENED

COLLECTION\_IS\_NULL

ACCESS\_INTO\_NULL

1. How to raise user defined exceptions
2. By declaring EXCEPTION variable in DECLARE section and RAISE exception in BEGIN part and catch it with EXCEPTION
3. By using RAISE\_APPLICATION\_ERROR(Error\_code, Error\_message). Use an error number between -20,000 and -20,999.
4. Improve SQL Query Performance

* Avoid Multiple Joins in a Single Query
* Eliminate Cursors from the Query
* Avoid Use of non correlated Scalar Sub Query
* Avoid Multi statement Table Valued Functions (TVFs)
* Creation and Use of Indexes
* Create a Highly Selective Index
* Position a Column in an Index
* Drop Unused Indexes
* Statistic Creation and Updates
* Revisit Your Schema Definitions

1. Why not to use nchar and nvarchar?

Both the data types take just double memory as char and varchar.

1. Why does padding not work on varchar2 bind variables?
2. Implications of using trim on char columns in database

It makes it impossible to use existing indexes on that column.

1. Issues with CHAR columns in database.

While you can compare literals with TRIM or RPAD on the literals. But TRIM makes it impossible to use existing indexes on that column and If the size of the field changes, then the application is impacted while using RPAD, as it must change its field width.

1. Why are bind variables treated differently than literals in sql compare statement?

Because in ANSI, “character string literals will be promoted to the type they are being compared” and

"respect what the programmer binds - if they bind a varchar2(10) - that is what they meant to do It would be wrong if the varchar2 containing '1'

matched the char containing '1 ‘.

1. Count number of letters in a string

SELECT REGEXP\_COUNT ('TechOnTheNet is a great resource', 't')

FROM dual;

Result: 2

By default, search is case sensitive.

1. Count average, min or max of salaries with null values for each employee in each department.

Select dept\_id, emp\_id, avg/min/max(nvl(sal,0)) over (partition by dept\_id);

1. Gather stats on partitioned tables

You can gather the global statistics on a partitioned table, or you can gather global and partition-level statistics. It has two options. They are: AUTO and GLOBAL AND PARTITION.

When the AUTO option is specified, the procedure determines the granularity based on the partitioning type. Oracle collects global, partition-level, and sub-partition level statistics if sub-partition method is LIST. For other partitioned tables, only the global and partition level statistics are generated.

When the GLOBAL AND PARTITION option is specified, Oracle gathers the global and partition level statistics. No sub-partition level statistics are gathered even it is composite partitioned object.

1. Types of cursors?

Implicit and explicit

1. Implicit Cursor attributes

FOUND  
NOTFOUND  
ROWCOUNT

ISOPEN

BULK\_ROWCOUNT

BULK\_EXCEPTIONS

1. Steps for explicit cursor

Declaring the cursor with IS Select

Opening the cursor

Loop over and Fetching cursor into variables

Closing cursor

Example:

DECLARE

v\_jobid employees.job\_id%TYPE; -- variable for job\_id

v\_lastname employees.last\_name%TYPE; -- variable for last\_name

CURSOR c1 IS SELECT last\_name, job\_id FROM employees

WHERE REGEXP\_LIKE (job\_id, 'S[HT]\_CLERK');

v\_employees employees%ROWTYPE; -- record variable for row

CURSOR c2 is SELECT \* FROM employees

WHERE REGEXP\_LIKE (job\_id, '[ACADFIMKSA]\_M[ANGR]');

BEGIN

OPEN c1; -- open the cursor before fetching

LOOP

FETCH c1 INTO v\_lastname, v\_jobid; -- fetches 2 columns into variables

EXIT WHEN c1%NOTFOUND;

DBMS\_OUTPUT.PUT\_LINE( RPAD(v\_lastname, 25, ' ') || v\_jobid );

END LOOP;

CLOSE c1;

DBMS\_OUTPUT.PUT\_LINE( '-------------------------------------' );

OPEN c2;

LOOP

FETCH c2 INTO v\_employees; -- fetches entire row into the v\_employees record

EXIT WHEN c2%NOTFOUND;

DBMS\_OUTPUT.PUT\_LINE( RPAD(v\_employees.last\_name, 25, ' ') ||

v\_employees.job\_id );

END LOOP;

CLOSE c2;

END;

/

1. What is a ref cursor?

A REF CURSOR is a pointer or a handle to a result set on the database or something that refers to a memory address on the database.

These are not updatable.

Is not backward scrollable.

It is a PLSQL datatype.

REFCURSOR implements lazy-loading, only loading the data at client side when the client opens the cursor and requests for it.

1. Difference between REF CURSOR and CURSOR

A cursor is really any SQL statement that runs DML (select, insert, update, delete) on your database. A ref cursor is a pointer to a result set. This is normally used to open a query on the database server, then leave it up to the client to fetch the result it needs.

1. Difference between cursor and view

View is a database object similar to table so it can be used with both SQL and PL/SQL. Cursor is defined and used within the block of stored procedure which means it can be only used with PL/SQL.

1. Difference between %TYPE and %ROWTYPE

% TYPE provides the data type of a variable or a database column to that variable.

% ROWTYPE provides the record type that represents an entire row of a table or view or columns selected in the cursor.

1. Difference between TYPE RECORD and %ROWTYPE

% ROWTYPE is to be used whenever query returns a entire row of a table or view.

TYPE customer IS RECORD is to be used whenever query returns columns of different table or views and variables.

DECLARE

TYPE user\_preferences\_rt IS RECORD

(

show\_full\_name BOOLEAN,

autologin BOOLEAN

);

l\_user user\_preferences\_rt;

1. What is a cursor for loop?

Cursor for loop implicitly declares %ROWTYPE as loop index,opens a cursor, fetches rows of values from active set into fields in the record and closes when all the records have been processed.

1. Explain the usage of WHERE CURRENT OF clause in cursors ?

WHERE CURRENT OF clause in an UPDATE,DELETE statement refers to the latest row fetched from a cursor.

1. What is an Exception? What are types of Exception?

Exception is the error handling part of PL/SQL block. The types are Predefined and user\_defined. Some of Predefined execptions are.

CURSOR\_ALREADY\_OPEN

DUP\_VAL\_ON\_INDEX

NO\_DATA\_FOUND

TOO\_MANY\_ROWS

INVALID\_CURSOR

INVALID\_NUMBER

LOGON\_DENIED

NOT\_LOGGED\_ON

PROGRAM-ERROR

STORAGE\_ERROR

TIMEOUT\_ON\_RESOURCE

VALUE\_ERROR

ZERO\_DIVIDE

OTHERS.

1. What is difference between a Cursor declared in a procedure and Cursor declared in a package specification?

Cursor in package – globally accessible

Cursor in procedure – locally accessible

1. What are some use cases of triggers?

Logging events

Preventing DMLs after business hours

Implementing complex business constraints which cannot be handled by constraints

Prevent invalid transactions

Publish/Push information about db events to subscribers.

1. Is it possible to use Transaction control Statements such a ROLLBACK or COMMIT in Database Trigger? Why?

It is not possible. As triggers are defined for each table, if you use COMMIT of ROLLBACK in a trigger, it affects logical transaction processing.

1. How many types of triggers are there?

Before row, After row, Before statement, After statement

1. Virtual tables for triggers?

Insert: New

Delete: OLD  
Update: Old and New

1. What happens if a procedure that updates a column of table X is called in a database trigger of the same table?

Mutation of table occurs due to inconsistent data.

1. Limitations of using inout parameters in functions

Functions with inout parameters get marked as Deterministic or get used as result-cached functions, preventing them to be used in SQLs.

1. Major difference between functions and procedures

Function needs to return something, return keyword is important.

Functions can be called by SQLs, procedures cannot be.

1. Order employees by their salary in their department(descending) , with continuous numbers.

SELECT department\_id, last\_name, salary,

DENSE\_RANK() OVER (PARTITION BY department\_id ORDER BY salary) DENSE\_RANK

FROM employees

ORDER BY DENSE\_RANK, last\_name;

Use rank if continuous numbers are not required.

1. How do you use pivot on below table?

age gender hours

25 Male 10

55 Female 5

45 Female 12

select age, nvl(male, 0) male, nvl(female, 0) female

from

(

select age, gender, hours

from table1

)

pivot

(

sum(hours) for gender in ('Male' as male, 'Female' as female)

);

1. How to use unpivot

SELECT \*

FROM unpivot\_test

UNPIVOT (quantity FOR product\_code IN (product\_code\_a AS 'A', product\_code\_b AS 'B', product\_code\_c AS 'C', product\_code\_d AS 'D'));

1. Rank vs rownum difference

Continuous numbers even for same values.

Salary Rank Dense Rank Row\_number

1000 1 1 1

2000 2 2 2

3000 3 3 3

3000 3 3 4

4000 5 4 5

1. Nth highest salary

select \*

from

(

select

sal

,dense\_rank() over (order by sal desc) ranking

from table

)

where ranking = n

1. How to search for multiple values: WHERE something LIKE ('bla%', '%foo%', 'batz%')

WHERE CONTAINS(t.something, 'bla OR foo OR batz', 1) > 0

1. Difference between FIRST and FIRST\_VALUE

FIRST requires min function and denser\_rank and order by clause

FIRST\_VALUE does not require it

1. LAST\_VALUE does not give correct last value

Use “rows between unbound preceding and unbound following” clause after order by .

1. How to use LISTAGG function

LISTAGG(Column, Seperator) WITHIN GROUP (ORDER BY column)

1. Difference between SYS\_REFCURSOR and REF CURSOR

SYS\_REFCURSOR is a REF CURSOR defined by oracle to limit the declaration of cursor as ref cursor.

1. What is a ref cursor?

REF CURSOR is a pointer or a handle to a result set on the database.

1. Difference between ref cursor and cursor?

* A "normal" plsql cursor is static in definition.

Ref cursors may be dynamically opened or opened based on logic.

* A ref cursor can be returned to a client.

A plsql "cursor cursor" cannot be returned to a client.

* A cursor can be global -- a ref cursor cannot (you cannot define them OUTSIDE of a procedure / function)
* A ref cursor can be passed from subroutine to subroutine -- a cursor cannot be.
* Static sql (not using a ref cursor) is much more efficient then using ref cursors and that use of ref cursors should be limited to  
  -- returning result sets to clients  
  -- when there is NO other efficient/effective means of achieving the goal
* You cannot update the database by using a REF CURSOR.

1. Write a simple procedure in a subprogram and invoke it.

DECLARE

first\_name employees.first\_name%TYPE;

last\_name employees.last\_name%TYPE;

email employees.email%TYPE;

employer VARCHAR2(8) := 'AcmeCorp';

-- Declare and define procedure

PROCEDURE create\_email ( -- Subprogram heading begins

name1 VARCHAR2,

name2 VARCHAR2,

company VARCHAR2

) -- Subprogram heading ends

IS

-- Declarative part begins

error\_message VARCHAR2(30) := 'Email address is too long.';

BEGIN -- Executable part begins

email := name1 || '.' || name2 || '@' || company;

EXCEPTION -- Exception-handling part begins

WHEN VALUE\_ERROR THEN

DBMS\_OUTPUT.PUT\_LINE(error\_message);

END create\_email;

BEGIN

first\_name := 'John';

last\_name := 'Doe';

create\_email(first\_name, last\_name, employer); -- invocation

DBMS\_OUTPUT.PUT\_LINE ('With first name first, email is: ' || email);

create\_email(last\_name, first\_name, employer); -- invocation

DBMS\_OUTPUT.PUT\_LINE ('With last name first, email is: ' || email);

first\_name := 'Elizabeth';

last\_name := 'MacDonald';

create\_email(first\_name, last\_name, employer); -- invocation

END;

/

1. What is a RAW datatype?

It is used to store data in binary format, like images. It can store upto 32767 bytes and there can be only 1 RAW datatype in a table.

1. Why to analyze a table?

When an oracle table contents are frequently updated, the statistics of the table need to be updated so that the cost based optimizer can take the latest statistics for faster processing.

The first way the ANALYZE command is used is to analyze a table. ANALYZE TABLE causes Oracle to determine how many rows are in the table and how storage is allocated. It also calculates the number of chained rows.

1. Translate vs replace

Translate does character by character substitution and replace will find the substring and replace, else ignore if string is not found.

1. What is VArray?

VArray is an oracle data type used to have columns containing multivalued attributes and it can hold bounded array of values.

1. What are the differences between LOV and List Item?

LOV is property whereas list items are considered as single item. List of items is set to be a collection of list of items. A list item can have only one column, LOV can have one or more columns.

1. Maximum number of triggers on a table ? 12
2. Types of triggers.

A crossedition trigger is a DML trigger for use only in edition-based redefinition.

If the trigger is created on a schema or the database, then the triggering event is composed of either DDL or database operation statements, and the trigger is called a system trigger.

A conditional trigger is a DML or system trigger that has a WHEN clause that specifies a SQL condition that the database evaluates for each row that the triggering statement affects.

Instead of trigger is a statement that is fired instead of the actual triggering statement.

1. What are DML triggers?

Triggers created on table or view. Triggering event is a DML.

1. Conditional predicates in DML triggers.

When one of the DML triggering event fires the DML trigger, the trigger can determine which one by using these conditional predicates.

Example:

CREATE OR REPLACE TRIGGER t

BEFORE

INSERT OR

UPDATE OF salary, department\_id OR

DELETE

ON employees

BEGIN

CASE

WHEN INSERTING THEN

DBMS\_OUTPUT.PUT\_LINE('Inserting');

WHEN UPDATING('salary') THEN

DBMS\_OUTPUT.PUT\_LINE('Updating salary');

WHEN UPDATING('department\_id') THEN

DBMS\_OUTPUT.PUT\_LINE('Updating department ID');

WHEN DELETING THEN

DBMS\_OUTPUT.PUT\_LINE('Deleting');

END CASE;

END;

/

1. Difference between EXCEPTION\_INIT and raise\_application\_error.

Exception\_Init is used to associate an exception name with oracle error, while raise is used to raise a user defined error which is not internally handled by Oracle.

1. Statement vs row level triggers

Row level triggers executes once for each and every row in the transaction. Statement level triggers executes only once for each single transaction.

1. What is the order in which triggers are fired?

All BEFORE STATEMENT triggers

All BEFORE EACH ROW triggers

All AFTER EACH ROW triggers

All AFTER STATEMENT triggers

Refrences:

<https://www.techonthenet.com/oracle/functions/listagg.php>

<https://oracle-base.com/articles/misc/rollup-cube-grouping-functions-and-grouping-sets#concatenated_groupings>

<http://programmer2programmer.net/tips/interview_question/oracle/oracle_package_procedure_and_function_interview_questions_answers.aspx>