

Oracle 10g Chapter-6 to 8

1: What is subquery? Write down the syntax of Subquery.

Ans: A subquery is a SELECT statement that is embedded in a clause of another SELECT statement. We can build powerful statement out of simple ones by using subquery.

The subquery (inner query) executes once before the main query(outer query). The result of the subquery is used by the main query. Subquery can be placed in (1) Where clause (2) Having clause (3) From clause.

Syntax: SELECT select_list from table where expr operator (select select_list from table);

2:Types of subqueries?

There are two types of subquery (1) Single_row subquery: Queries that return only one row from the inner SELECT statement.

(2) Multiple_row subquery: Queries that return more than one row from the inner SELECT statement.

Note: *There are also multiple-column subqueries, which are queries that return more than one column from the inner SELECT statement.*

3. Define each type and give an example for each.

(1) **Single_row subquery:** Single-row subquery is one Query that return only one row from the inner SELECT statement.

Example: select last_name, job_id, salary from employees where job_id = (select job_id from employees where employee_id = 141) and salary > (select salary from employees where employee_id = 143);

(2) **Multiple_row subquery:** Queries that return more than one row from the inner SELECT statement.

Example: select last_name, job_id, salary from employees where salary <any (select salary from employees where job_id = 'It_prog') and job_id <> 'It_prog';

4: What are the guidelines of subqueries?

1. Enclose subqueries in parentheses.
2. place subqueries on the right side of the comparison condition.
3. The ORDER BY clause in the subquery is not needed unless performing Top_N analysis.
4. Use single-row operators with single-row subqueries and multiple-row subqueries.

N.B. *One common error with subqueries occurs when more than one row is returned for a single-row subquery. A common problem with subqueries occurs when no rows are returned by the inner query.*

Describe Having Clause.

The **Oracle HAVING clause** is used in combination with the **GROUP BY clause** to restrict the groups of returned rows to only those the condition is TRUE. The Oracle server returns results into the Having Clause of the main query.

Describe ANY Query.

The ANY operator (and its synonym, the SOME operator) compares a value to *each* value returned by a subquery. <ANY means less than the maximum. >ANY means more than the minimum. =ANY is equivalent to IN.

Example: `SELECT employee_id, last_name, job_id, salary FROM employees WHERE salary < ANY (SELECT salary FROM employees WHERE job_id = 'IT_PROG') AND job_id <> 'IT_PROG';`

Describe ALL Operators.

The ALL operator compares a value to *every* value returned by a subquery. >ALL means more than the maximum, and <ALL means less than the minimum. The NOT operator can be used with IN, ANY, and ALL operators.

Example: `SELECT employee_id, last_name, job_id, salary FROM employees WHERE salary < ALL (SELECT salary FROM employees WHERE job_id = 'IT_PROG') AND job_id <> 'IT_PROG';`

Write down the characteristics of Subquery.

Subqueries have the following characteristics:

- Can pass one row of data to a main statement that contains a single-row operator, such as =, <>, >, >=, <, or <=
- Can pass multiple rows of data to a main statement that contains a multiple-row operator, such as IN
- Are processed first by the Oracle server, after which the WHERE or HAVING clause uses the results
- Can contain group functions

8: What is set operators? Describe different types of set operators with graph?

Operators combine the results of two or more component queries into one result is called set operator. Queries containing set operators are called compound queries.

Examples of Set Operators:

Union Operator: The UNION operator returns all rows that are selected by either query. Use the UNION operator to return all rows from multiple tables and eliminate any duplicate rows.

`SELECT employee_id, job_id FROM employees UNION SELECT employee_id, job_id FROM job_history;`

Guidelines

- The number of columns and the data types of the columns being selected must be identical in all the SELECT statements used in the query. The names of the columns need not be identical.
- UNION operates over all of the columns being selected.
- NULL values are not ignored during duplicate checking.

- The `IN` operator has a higher precedence than the `UNION` operator.
- By default, the output is sorted in ascending order of the first column of the `SELECT` clause.

Union ALL Operator:

Use the `UNION ALL` operator to return all rows from multiple queries.

Guidelines

The guidelines for `UNION` and `UNION ALL` are the same, with the following two exceptions that pertain to `UNION ALL`:

- Unlike `UNION`, duplicate rows are not eliminated and the output is not sorted by default.
- The `DISTINCT` keyword cannot be used.

Example: `SELECT employee_id, job_id, department_id FROM employees UNION ALL SELECT employee_id, job_id, department_id FROM job_history ORDER BY employee_id;`

N.B. The `UNION ALL` operator does not eliminate duplicate rows. `UNION` returns all distinct rows selected by either query. `UNION ALL` returns all rows selected by either query, including all duplicates.

Example: `SELECT employee_id, job_id, department_id FROM employees UNION SELECT employee_id, job_id, department_id FROM job_history ORDER BY employee_id;`

Intersect Operator: The `INTERSECT` operator returns rows that are common to both queries. [*the `INTERSECT` operator to return all rows that are common to multiple queries.*]

Example: `SELECT employee_id, job_id FROM employees INTERSECT SELECT employee_id, job_id FROM job_history;`

{*Display the employee IDs and job IDs of those employees who currently have a job title that is the same as a previous job title.*}

Guidelines

- The number of columns and the data types of the columns being selected by the `SELECT` statements in the queries must be identical in all the `SELECT` statements used in the query. The names of the columns need not be identical.
- Reversing the order of the intersected tables does not alter the result.
- `INTERSECT` does not ignore `NULL` values.

Minus Operator: The `MINUS` operator returns rows in the first query that are not present in the second query.

For Example: `SELECT employee_id FROM employees MINUS SELECT employee_id FROM job_history;`

Set Operator Guidelines

- The expressions in the SELECT lists must match in number and data type.
- Parentheses can be used to alter the sequence of execution.
- The ORDER BY clause:
 - Can appear only at the very end of the statement
 - Will accept the column name, aliases from the first SELECT statement, or the positional notation.

Oracle Server and Set Operators

- Duplicate rows are automatically eliminated except in UNION ALL.
- Column names from the first query appear in the result.
- The output is sorted in ascending order by default except in UNION ALL.

Chapter-8

What is DML statement? When does it execute?

Data manipulation language (DML) is a core part of SQL. When we want to add, update, or delete data in the database, we execute a DML statement. A collection of DML statements that form a logical unit of work is called a *transaction*.

How to add a new row in a table?

We can add new rows to a table by using the INSERT statement. `INSERT INTO table [(column [, column...])] VALUES (value [, value...]);`

With this syntax, only one row is inserted at a time.

Or

- Insert a new row containing values for each column.
- List values in the default order of the columns in the table.
- Optionally, list the columns in the INSERT clause.
- Enclose character and date values in single quotation marks.

Example: `INSERT INTO departments(department_id, department_name, manager_id, location_id) VALUES (70, 'Public Relations', 100, 1700);`

Method for inserting Null values:

Implicit Omit the column from the column list.

Explicit Specify the NULL keyword in the VALUES list; specify the empty string (' ') in the VALUES list for character strings and dates.

What are the common errors that occur during user input?

Common errors that can occur during user input:

- Mandatory value missing for a NOT NULL column
- Duplicate value violates uniqueness constraint
- Foreign key constraint violated

- CHECK constraint violated
- Data type mismatch
- Value too wide to fit in column

How to update a table?

- Modify existing rows with the UPDATE statement.
- Update more than one row at a time (if required).
- Specific row or rows are modified if you specify the WHERE clause.
- All rows in the table are modified if you omit the WHERE clause.

Syntax: `UPDATE table SET column = value [, column = value, ...] [WHERE condition];`

Example: `UPDATE employees SET department_id = 70 WHERE employee_id = 113;`

How to delete rows from a table?

We can remove existing rows from a table by using the `DELETE` statement. Specific rows are deleted if we specify the `WHERE` clause. All rows in the table are deleted if you omit the `WHERE` clause.

For Example: `DELETE FROM departments WHERE department_name = 'Finance';`

What is Truncate Statement?

The **TRUNCATE** TABLE statement is used to remove all records from a table in **Oracle**. It performs the same function as a `DELETE` statement without a `WHERE` clause.

N.B. The **TRUNCATE** TABLE statement cannot be rolled back.

What is the characteristic of Truncate Statement?

- The **TRUNCATE** statement is a data definition language (DDL) statement and generates no rollback information.
- Truncating a table does not fire the delete triggers of the table.
- If the table is the parent of a referential integrity constraint, you cannot truncate the table. You need to disable the constraint before issuing the **TRUNCATE** statement. Disabling constraints is covered in a subsequent lesson.

What is Database Transactions?

A **transaction** is a logical unit of work that contains one or more SQL statements. A **transaction** is an atomic unit. The effects of all the SQL statements in a **transaction** can be either all committed (applied to the **database**) or all rolled back (undone from the **database**).

A database transaction consists of one of the following:

- DML statements that constitute one consistent change to the data
- One DDL statement
- One data control language (DCL) statement

When Does a Transaction Start and End?

A transaction begins when the first DML statement is encountered and ends when one of the following occurs:

- A `COMMIT` or `ROLLBACK` statement is issued.
- A DDL statement, such as `CREATE`, is issued.
- A DCL statement is issued.
- The user exits SQL Developer or SQL*Plus.
- A machine fails or the system crashes.

What is Rollback?

In **Oracle**, the **ROLLBACK** statement is used to undo the work performed by the current transaction or a transaction that is in doubt.

What are the advantages of COMMIT and ROLLBACK Statements?**Advantages of COMMIT and ROLLBACK Statements**

With `COMMIT` and `ROLLBACK` statements, you can:

- Ensure data consistency
- Preview data changes before making changes permanent
- Group logically related operations

How to control Transaction?

We can control the logic of transactions by using the `COMMIT`, `SAVEPOINT`, and `ROLLBACK` Statements.

Note: In SQL*Plus, the `AUTOCOMMIT` command can be toggled ON or OFF. If set to ON, each individual DML statement is committed as soon as it is executed. You cannot roll back the changes. If set to OFF, the `COMMIT` statement can still be issued explicitly.

How many ways a user can access the database?

Database users access the database in two ways:

- Read operations (`SELECT` statement)
- Write operations (`INSERT`, `UPDATE`, `DELETE` statements)

Chapter-9

What are the Objects of Database?

Database Objects are:

- **Table:** Stores data
- **View:** Subset of data from one or more tables
- **Sequence:** Generates numeric values
- **Index:** Improves the performance of some queries
- **Synonym:** Gives alternative names to objects

What are the rules of naming Database Table?

Table names and column names:

- Must begin with a letter
- Must be 1–30 characters long
- Must contain only A–Z, a–z, 0–9, _, \$, and #
- Must not duplicate the name of another object owned by the same user
- Must not be an Oracle server–reserved word

What is Schema?

A schema is a collection of objects such as tables, views, and sequences. The schema is owned by a database user and has the same name as that user. Schema objects include tables, views, synonyms, sequences, stored procedures, indexes, clusters, and database links.

What are the guidelines for creating a table?

Guidelines

- A `LONG` column is not copied when a table is created using a subquery.
- A `LONG` column cannot be included in a `GROUP BY` or an `ORDER BY` clause.
- Only one `LONG` column can be used per table.
- No constraints can be defined on a `LONG` column.
- You might want to use a `CLOB` column rather than a `LONG` column.

What is Timestamp Data Type?

The `TIMESTAMP` data type is an extension of the `DATE` data type. It stores the year, month, and day of the `DATE` data type plus hour, minute, and second values. This data type is used for storing precise time values.

What is Constraint?

A `CONSTRAINT` clause is an optional part of a `CREATE TABLE` statement or `ALTER TABLE` statement. A constraint is a rule to which data must conform. Constraint names are optional.

What are the Constraints Types?

The following constraint types are valid:

- NOT NULL (*Specifies that the column cannot contain a null value*)
- UNIQUE (*Specifies a column or combination of columns whose values must be unique for all rows in the table*)
- PRIMARY KEY (*Uniquely identifies each row of the table*)
- FOREIGN KEY (*Establishes and enforces a foreign key relationship between the column and a column of the referenced table*)
- CHECK (*Specifies a condition that must be true*)

What are the guidelines of Constraint?

- You can name a constraint, or the Oracle server generates a name with the SYS_Cn format.
- Create a constraint at either of the following times:
 - At the same time as the table is created
 - After the table has been created
- Define a constraint at the column or table level.
- View a constraint in the data dictionary.

What is not null constraint?

The NOT NULL constraint ensures that the column contains no null values. Columns without the NOT NULL constraint can contain null values by default. NOT NULL constraints must be defined at the column level.

What is Unique Constraint?

A UNIQUE key integrity constraint requires that every value in a column or set of columns (key) be unique—that is, no two rows of a table can have duplicate values in a specified column or set of columns. The column (or set of columns) included in the definition of the UNIQUE key Constraint is called the *unique key*.

What is PRIMARY KEY Constraint?

A PRIMARY KEY constraint creates a primary key for the table. Only one primary key can be created for each table. The PRIMARY KEY constraint is a column or set of columns that uniquely identifies each row in a table. This constraint enforces uniqueness of the column or column combination and ensures that no column that is part of the primary key can contain a null value.

What is FOREIGN KEY Constraint?

The FOREIGN KEY (or referential integrity) constraint designates a column or combination of columns as a foreign key and establishes a relationship between a primary key or a unique key in the same table or a different table.

Guidelines

- A foreign key value must match an existing value in the parent table or be NULL.
- Foreign keys are based on data values and are purely logical, rather than physical, pointers.

What is the foreign key Constraint?

FOREIGN KEY Constraint:

Keywords

- **FOREIGN KEY:** Defines the column in the child table at the table-constraint level
- **REFERENCES:** Identifies the table and column in the parent table
- **ON DELETE CASCADE:** Deletes the dependent rows in the child table when a row in the parent table is deleted
- **ON DELETE SET NULL:** Converts dependent foreign key values to null

What is Check Constraint?

CHECK Constraint

The CHECK constraint defines a condition that each row must satisfy. The condition can use the same constructs as query conditions, with the following exceptions:

- References to the CURRVAL, NEXTVAL, LEVEL, and ROWNUM pseudo columns
- Calls to SYSDATE, UID, USER, and USERENV functions
- Queries that refer to other values in other rows

N.B. *[You cannot delete a row that contains a primary key that is used as a foreign key in another table.]*

What are the uses of ALTER TABLE Statement?

Use the ALTER TABLE statement to:

- Add a new column
- Modify an existing column
- Define a default value for the new column
- Drop a column

What are the Guidelines of Dropping Table?

Guidelines

- All data is deleted from the table.
- Any views and synonyms remain but are invalid.
- Any pending transactions are committed.
- Only the creator of the table or a user with the DROP ANY TABLE privilege can remove a table.

Chapter-10

What are the Database Objects?

The Objects of Database are views, sequences, indexes, and synonyms.

What is view?

A view is a logical table based on a table or another view. A view contains no data of its own but is like a window through which data from tables can be viewed or changed. The tables on which a view is based are called *base tables*. The view is stored as a `SELECT` statement in the data dictionary.

What are the Advantages of View?

Advantages of View:

1. We use View to restrict data access.
2. We use View to makes complex queries.
3. We use View to provide data independence.
4. We use View to present different views of the same data.

What are the differences between simple views and complex views?

Simple Views and Complex Views

There are two classifications for views: simple and complex. The basic difference is related to the DML (INSERT, UPDATE, and DELETE) operations.

• A simple view is one that:

- Derives data from only one table
- Contains no functions or groups of data
- Can perform DML operations through the view

• A complex view is one that:

- Derives data from many tables
- Contains functions or groups of data
- Does not always allow DML operations through the view.

What are the rules for Performing DML Operations on a View?

- You can usually perform DML operations on simple views.
- You cannot remove a row if the view contains the following:
 - Group functions
 - A GROUP BY clause
 - The DISTINCT keyword
 - The pseudocolumn ROWNUM keyword

Rules for Performing DML Operations on a View

You cannot modify data in a view if it contains:

- Group functions
- A GROUP BY clause
- The DISTINCT keyword
- The pseudocolumn ROWNUM keyword
- Columns defined by expressions

Rules for Performing DML Operations on a View

You cannot add data through a view if the view includes:

- Group functions
- A GROUP BY clause
- The DISTINCT keyword
- The pseudocolumn ROWNUM keyword
- Columns defined by expressions
- NOT NULL columns in the base tables that are not selected by the view

How to remove a view?

You can remove a view without losing data because a view is based on underlying tables in the database.

Syntax:

```
DROP VIEW view;
```

What is Sequence?

A sequence is a database object that creates integer values. You can create sequences and then use them to generate numbers.

What are the features/characteristics of Sequence?

A sequence:

- Can automatically generate unique numbers
- Is a sharable object
- Can be used to create a primary key value
- Replaces application code
- Speeds up the efficiency of accessing sequence values when cached in memory.

What are the guidelines for modifying a Sequence?

- You must be the owner or have the ALTER privilege for the sequence.
- Only future sequence numbers are affected.
- The sequence must be dropped and re-created to restart the sequence at a different number.
- Some validation is performed.
- To remove a sequence, use the DROP statement.

What is index?

An Oracle server index is a schema object that can speed up the retrieval of rows by using a pointer. Indexes can be created explicitly or automatically. An index provides direct and fast access to rows in a table. After an index is created, no direct activity is required by the user.

How many kinds of index are there? What are they?

Two types of indexes can be created.

Unique index: The Oracle server automatically creates this index when you define a column in a table to have a PRIMARY KEY or a UNIQUE key constraint. The name of the index is the name that is given to the constraint.

Nonunique index: This is an index that a user can create. For example, you can create a FOREIGN KEY column index for a join in a query to improve retrieval speed.

When to Create an Index?

We should create indexes only if:

- The column contains a wide range of values
- The column contains a large number of null values
- One or more columns are frequently used together in a WHERE clause or join condition
- The table is large and most queries are expected to retrieve less than 2% to 4% of the rows

What is Synonym?

Synonyms are database objects that enable you to call a table by another name. You can create synonyms to give an alternative name to a table.

Guidelines

- The object cannot be contained in a package.
- A private synonym name must be distinct from all other objects that are owned by the same user.

Chapter-11

Page 11-15

Data Access Using Views

When you access data using a view, the Oracle server performs the following operations:

- It retrieves the view definition from the data dictionary table USER_VIEWS.
- It checks access privileges for the view base table.
- It converts the view query into an equivalent operation on the underlying base table

What is Sequence?

A **sequence** is an object in **Oracle** that is used to generate a number **sequence**. This can be useful when you need to create a unique number to act as a primary key.

How to add comment on a Table?

Adding Comments to a Table

We can add a comment of up to 4,000 bytes about a column, table, view, or snapshot by using the comment statement. The comment is stored in the data dictionary and can be viewed in one of the following data dictionary views in the comments column.

- all_col_comments

- user_col_comments
- all_tab_comments
- user_tab_comments

Syntax

comment on table *table* | column *table.column* is 'text';

Book-3

Chapter-1

What are the DBA System Privileges?

- More than 100 privileges are available.
- The database administrator has high-level system privileges for tasks such as:
 - 1) CREATE USER.
 - 2) DROP USER.
 - 3) DROP ANY TABLE.
 - 4) BACKUP ANY TABLE.
 - 5) SELECT ANY TABLE.
 - 6) CREATE ANY TABLE.

How to Create a User?

Creating a User

The DBA creates a user by executing the CREATE USER statement. The user does not have any privileges at this point. The DBA can then grant privileges to that user. These privileges determine what the user can do at the database level.

I have left some Questions. I will add that after Eid-Vacation.