What is an Oracle Synonym?

A synonym is an object in a database that represents an alternative name for other objects, such as tables, views, sequences, and stored procedures.

Why Would you Use a Synonym?

The main reason is to **simplify the user access to an object**. You can provide access to an object to a user using a synonym, and you don't need to worry about who owns the original object.

It also simplifies access to objects, especially when multiple databases and schemas are involved.

Oracle Synonym vs View

As mentioned above, a synonym is an alias or alternative name for an object.

How is this different from a view?

A view object is an SQL query that is saved and run when other queries use that view. It works like a table.

Why would you use a synonym instead of a view?

If you just want to provide an alias for a table or another object, you can use a synonym. If you want to SELECT data from a table and use WHERE conditions or other logic, then a view is better.

Also, a view can only be created on objects with data, such as tables and other views. Synonyms can be created on tables, views, sequences, stored procedures, and more.

Difference Between Public and Private Synonym in Oracle

There are two types of synonyms that can be created on an Oracle database: public and private.

- **Public** synonym: can be **accessed by any user on the database**. The user who creates the synonym it does not own it – it's owned by the PUBLIC user group.
- **Private** synonym: can only be **accessed by the person who created the synonym**. This user is also the owner. The synonym name must be unique within the schema.
- This is how public synonyms and private synonyms compare:

Criteria	Public	Private
Who owns it?	PUBLIC user group	The user that created it
Scope of the unique name	All database objects	The schema of the user who created it
Who can use it?	All database users	Only the user who created it

Order of precedence	Third, after schema objects and private synonyms	Second, after schema objects

Let’s see some examples of creating a synonym.

Example 1 – Create private synonym

```
CREATE SYNONYM emp
FOR hr.employees;
```

This creates a new private synonym called emp, which refers to the employees object in the hr schema.

Example 2 – Create public synonym

```
CREATE PUBLIC SYNONYM cust
FOR sales.customers;
```

This will create a new public synonym for the customer object in the sales schema. All users can query the cust synonym and access the sales.customers table (if they have access to the underlying sales.customers table).

Example 3 – Same name

```
CREATE SYNONYM product
FOR sales.product;
```

This will create a private synonym. Notice that both objects have the same name. A query such as this will look for the local object first (which does not exist), and then the synonym:

```sql
SELECT * FROM product;
```

However, querying on the original object will also access that object:

```sql
SELECT * FROM sales.product;
```

If a new table was created in the user’s current schema called product, then this query will look at that table instead of the synonym:

```sql
SELECT * FROM product;
```

Example 1 – Drop a public synonym

```sql
DROP PUBLIC SYNONYM emp;
```

This will drop the public synonym emp.

Example 2 – Drop a private synonym

```sql
DROP SYNONYM customer;
```

This will drop the private synonym called customer.

Example 3 – Drop a synonym with a schema

```sql
DROP SYNONYM hr.emp;
```

This will drop the synonym emp that exists in the hr schema.