

Created by: Ahsan Arif

INSERT Statement

The INSERT statement allows you to insert a single record or multiple records into a table.

The syntax for the INSERT statement is:

```
INSERT INTO table  
(column-1, column-2, ... column-n)  
VALUES  
(value-1, value-2, ... value-n);
```

Example #1 - Simple example

Let's take a look at a very simple example.

```
INSERT INTO suppliers  
(supplier_id, supplier_name)  
VALUES  
(24553, 'IBM');
```

This would result in one record being inserted into the suppliers table. This new record would have a `supplier_id` of 24553 and a `supplier_name` of IBM.

Example #2 - More complex example

You can also perform more complicated inserts using sub-selects.

For example:

```
INSERT INTO suppliers  
(supplier_id, supplier_name)  
SELECT account_no, name  
FROM customers  
WHERE city = 'Newark';
```

By placing a "select" in the insert statement, you can perform multiples inserts quickly.

With this type of insert, you may wish to check for the number of rows being inserted. You can determine the number of rows that will be inserted by running the following SQL statement **before** performing the insert.

```
SELECT count(*)
FROM customers
WHERE city = 'Newark';
```

Frequently Asked Questions

Question: I am setting up a database with clients. I know that you use the "insert" statement to insert information in the database, but how do I make sure that I do not enter the same client information again?

Answer: You can make sure that you do not insert duplicate information by using the EXISTS condition.

For example, if you had a table named *clients* with a primary key of *client_id*, you could use the following statement:

```
INSERT INTO clients
(client_id, client_name, client_type)
SELECT supplier_id, supplier_name, 'advertising'
FROM suppliers
WHERE not exists (select * from clients
where clients.client_id = suppliers.supplier_id);
```

This statement inserts multiple records with a subselect.

If you wanted to insert a single record, you could use the following statement:

```
INSERT INTO clients
(client_id, client_name, client_type)
SELECT 10345, 'IBM', 'advertising'
FROM dual
WHERE not exists (select * from clients
where clients.client_id = 10345);
```

The use of the **dual** table allows you to enter your values in a select statement, even though the values are not currently stored in a table.

Question: How can I insert multiple rows of explicit data in one SQL command in Oracle?

Answer: The following is an example of how you might insert 3 rows into the *suppliers* table in Oracle.

```
INSERT ALL
  INTO suppliers (supplier_id, supplier_name) VALUES (1000, 'IBM')
  INTO suppliers (supplier_id, supplier_name) VALUES (2000, 'Microsoft')
  INTO suppliers (supplier_id, supplier_name) VALUES (3000, 'Google')
SELECT * FROM dual;
```