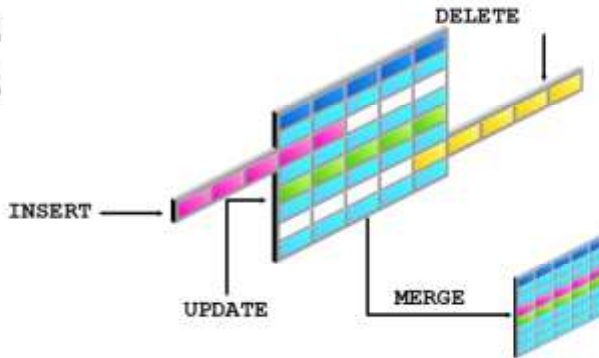


## Using PL/SQL to Manipulate Data

Make changes to database tables by using DML commands:

- INSERT
- UPDATE
- DELETE
- MERGE



ORACLE

Copyright © 2010, Oracle and/or its affiliates. All rights reserved.

### Using PL/SQL to Manipulate Data

You manipulate data in the database by using DML commands. You can issue DML commands such as INSERT, UPDATE, DELETE, and MERGE without restriction in PL/SQL. Row locks (and table locks) are released by including the COMMIT or ROLLBACK statements in the PL/SQL code.

The INSERT statement adds new rows to the table.

The UPDATE statement modifies existing rows in the table.

The DELETE statement removes rows from the table.

The MERGE statement selects rows from one table to update or insert into another table. The decision whether to update or insert into the target table is based on a condition in the ON clause.

Note: MERGE is a deterministic statement. That is, you cannot update the same row of the target table multiple times in the same MERGE statement. You must

## Updating Data: Example

Increase the salary of all employees who are stock clerks.

```
DECLARE
  quantity_increase inventories.quantity_on_hand%TYPE := 100;
BEGIN
  UPDATE      inventories
  SET         quantity_on_hand = quantity_on_hand + quantity_increase
  WHERE warehouse_id = 8;
END;
/
```

ORACLE

Copyright © 2010, Oracle and/or its affiliates. All rights reserved.

have INSERT and UPDATE object privileges on the target table and SELECT privilege on the source table.

### Inserting Data

In the example in the slide, an INSERT statement is used within a PL/SQL block to insert a record into the employees table. While using the INSERT command in a PL/SQL block, you can:

- Use SQL functions such as USER and CURRENT\_DATE

- Generate primary key values by using existing database sequences

- Derive values in the PL/SQL block

### Updating Data

There may be ambiguity in the SET clause of the UPDATE statement because, although the identifier on the left of the assignment operator is always a database column, the identifier on the right can be either a database column or a PL/SQL variable. Recall that if column names and identifier names are identical

## Deleting Data: Example

Delete rows that belong to department 10 from the `employees` table.

```
DECLARE
  ordid orders.order_id%TYPE := 2348;
BEGIN
  DELETE FROM orders
  WHERE order_id = ordid;
END ;
/
```

ORACLE

Copyright © 2010, Oracle and/or its affiliates. All rights reserved.

in the WHERE clause, the Oracle Server looks to the database first for the name.

Remember that the WHERE clause is used to determine the rows that are affected. If no rows are modified, no error occurs (unlike the SELECT statement in PL/SQL).

Note: PL/SQL variable assignments always use `:=`, and SQL column assignments always use `=`.

### Deleting Data

The DELETE statement removes unwanted rows from a table. If the WHERE clause is not used, all the rows in a table can be removed if there are no integrity constraints.