EXPERIMENT 8

**Aim**: To study the different types of cursors and its implementation

**Theory**:

**Cursor** is a Temporary Memory or Temporary Work Station. It is Allocated by Database Server at the Time of Performing DML operations on Table by User. Cursors are used to store Database Tables. There are 2 types of Cursors: Implicit Cursors, and Explicit Cursors.

**Implicit Cursors:** Implicit Cursors are also known as Default Cursors of SQL SERVER. These Cursors are allocated by SQL SERVER when the user performs DML operations.

**Explicit Cursors:** Explicit Cursors are Created by Users whenever the user requires them. Explicit Cursors are used for Fetching data from Table in Row-By-Row Manner.

**Implicit Cursors Attributes**

|  |  |
| --- | --- |
| **S.No** | **Attribute & Description** |
| 1 | %FOUND  Returns TRUE if an INSERT, UPDATE, or DELETE statement affected one or more rows or a SELECT INTO statement returned one or more rows. Otherwise, it returns FALSE. |
| 2 | %NOTFOUND  The logical opposite of %FOUND. It returns TRUE if an INSERT, UPDATE, or DELETE statement affected no rows, or a SELECT INTO statement returned no rows. Otherwise, it returns FALSE. |
| 3 | %ISOPEN  Always returns FALSE for implicit cursors, because Oracle closes the SQL cursor automatically after executing its associated SQL statement. |
| 4 | %ROWCOUNT  Returns the number of rows affected by an INSERT, UPDATE, or DELETE statement, or returned by a SELECT INTO statement. |

**Creating Explicit Cursors**

1. **Creating cursor object**

Syntax: DECLARE cursor\_name CURSOR FOR SELECT \* FROM table\_name;

1. **Open Cursor Connection**

Syntax: OPEN cursor\_connection;

1. **Fetching Data from Cursors**

There are a total of 6 methods to access data from the cursor. They are as follows:

**FIRST** is used to fetch only the first row from the cursor table.

**LAST** is used to fetch only the last row from the cursor table.

**NEXT** is used to fetch data in forward direction from the cursor table.

**PRIOR** is used to fetch data in backward direction from the cursor table.

**ABSOLUTE** n is used to fetch the exact nth row from the cursor table.

**RELATIVE** n is used to fetch the data in incremental as well as decremental ways.

Syntax: FETCH NEXT/FIRST/LAST/PRIOR/ABSOLUTE n/RELATIVE n FROM cursor\_name

1. **Close Cursor Connection**

**Syntax**: CLOSE cursor\_name

1. **Deallocate cursor memory.**

**Syntax**:DEALLOCATE cursor\_name

**Code**

**DECLARE**

**total\_rows number(2);**

**BEGIN**

**UPDATE customers**

**SET salary = salary + 500;**

**IF sql%notfound THEN**

**dbms\_output.put\_line('no customers selected');**

**ELSIF sql%found THEN**

**total\_rows := sql%rowcount;**

**dbms\_output.put\_line( total\_rows || ' customers selected ');**

**END IF;**

**END;**

**/**

**OUTPUT**

**6 customers selected**

**PL/SQL procedure successfully completed.**