

Created by Ahsan Arif

## Named System Exceptions

---

Named system exceptions are exceptions that have been given names by PL/SQL. They are named in the STANDARD package in PL/SQL and do not need to be defined by the programmer.

Oracle has a standard set of exceptions already named as follows:

| Oracle Exception Name  | Oracle Error | Explanation   |
|------------------------|--------------|---|
| DUP_VAL_ON_INDEX       | ORA-00001    | You tried to execute an INSERT or UPDATE statement that has created a duplicate value in a field restricted by a unique index.                                    |
| TIMEOUT_ON_RESOURCE    | ORA-00051    | You were waiting for a resource and you timed out.  |
| TRANSACTION_BACKED_OUT | ORA-00061    | The remote portion of a transaction has rolled back.  |
| INVALID_CURSOR         | ORA-01001    | You tried to reference a cursor that does not yet exist. This may have happened because you've executed a FETCH cursor or CLOSE cursor before OPENing the cursor. |
| NOT_LOGGED_ON          | ORA-01012    | You tried to execute a call to Oracle before logging in.  |
| LOGIN_DENIED           | ORA-01017    | You tried to log into Oracle with an invalid username/password combination.   |
| NO_DATA_FOUND          | ORA-01403    | You tried one of the  |

|                                  |           |   |
|----------------------------------|-----------|---|
|                                  |           | <p>following:</p> <ol style="list-style-type: none"> <li>1. You executed a <code>SELECT INTO</code> statement and no rows were returned.</li> <li>2. You referenced an uninitialized row in a table.</li> <li>3. You read past the end of file with the <code>UTL_FILE</code> package.</li> </ol> |
| <code>TOO_MANY_ROWS</code>       | ORA-01422 | You tried to execute a <code>SELECT INTO</code> statement and more than one row was returned.   |
| <code>ZERO_DIVIDE</code>         | ORA-01476 | You tried to divide a number by zero.   |
| <code>INVALID_NUMBER</code>      | ORA-01722 | You tried to execute an SQL statement that tried to convert a string to a number, but it was unsuccessful.  |
| <code>STORAGE_ERROR</code>       | ORA-06500 | You ran out of memory or memory was corrupted.  |
| <code>PROGRAM_ERROR</code>       | ORA-06501 | This is a generic "Contact Oracle support" message because an internal problem was encountered.   |
| <code>VALUE_ERROR</code>         | ORA-06502 | You tried to perform an operation and there was a error on a conversion, truncation, or invalid constraining of numeric or character data.  |
| <code>CURSOR_ALREADY_OPEN</code> | ORA-06511 | You tried to open a cursor that is already  |

|  |  |       |
|--|--|-------|
|  |  | open. |
|--|--|-------|

The syntax for the Named System Exception in a procedure is:

```
CREATE [OR REPLACE] PROCEDURE procedure_name
  [ (parameter [,parameter]) ]
IS
  [declaration_section]
BEGIN
  executable_section

EXCEPTION
  WHEN exception_name1 THEN
    [statements]

  WHEN exception_name2 THEN
    [statements]

  WHEN exception_name_n THEN
    [statements]

  WHEN OTHERS THEN
    [statements]

END [procedure_name];
```

The syntax for the Named System Exception in a function is:

```
CREATE [OR REPLACE] FUNCTION function_name
  [ (parameter [,parameter]) ]
  RETURN return_datatype
IS | AS
  [declaration_section]
BEGIN
  executable_section

EXCEPTION
  WHEN exception_name1 THEN
    [statements]

  WHEN exception_name2 THEN
    [statements]

  WHEN exception_name_n THEN
    [statements]
```

```
WHEN OTHERS THEN
    [statements]
```

```
END [function_name];
```

Here is an example of a procedure that uses a Named System Exception:

```
CREATE OR REPLACE PROCEDURE add_new_supplier
    (supplier_id_in IN NUMBER, supplier_name_in IN VARCHAR2)
IS
BEGIN
    INSERT INTO suppliers (supplier_id, supplier_name )
    VALUES ( supplier_id_in, supplier_name_in );

EXCEPTION
    WHEN DUP_VAL_ON_INDEX THEN
        raise_application_error (-20001,'You have tried to insert a duplicate supplier_id.');
```

```
    WHEN OTHERS THEN
        raise_application_error (-20002,'An error has occurred inserting a supplier.');
```

```
END;
```

In this example, we are trapping the Named System Exception called **DUP\_VAL\_ON\_INDEX**. We are also using the **WHEN OTHERS** clause to trap all remaining exceptions.