

What is Stored Procedures in SQL ?

Stored Procedures are created to perform one or more DML operations on Database. It is nothing but the group of SQL statements that accepts some input in the form of parameters and performs some task and may or may not returns a value.

Syntax : Creating a Procedure

```
CREATE or REPLACE PROCEDURE name(parameters)
IS
variables;
BEGIN
//statements;
END;
```

The most important part is parameters. Parameters are used to pass values to the Procedure. There are 3 different types of parameters, they are as follows:

1. **IN:**
This is the Default Parameter for the procedure. It always receives the values from calling program.
2. **OUT:**
This parameter always sends the values to the calling program.
3. **IN OUT:**
This parameter performs both the operations. It receives value from as well as sends the values to the calling program.

Example:

Imagine a table named with emp_table stored in Database. We are writing a Procedure to update a Salary of Employee with 1000.

```
CREATE or REPLACE PROCEDURE INC_SAL(eno IN NUMBER, up_sal OUT NUMBER)
```

```
IS
BEGIN
UPDATE emp_table SET salary = salary+1000 WHERE emp_no = eno;
COMMIT;
SELECT sal INTO up_sal FROM emp_table WHERE emp_no = eno;
END;
```

Steps to execute the procedure:

- Declare a Variable to Store the value coming out from Procedure

```
VARIABLE v NUMBER;
```

- Execution of the Procedure:

```
EXECUTE INC_SAL(1002, :v);
```

- To check the updated salary use SELECT statement:

```
SELECT * FROM emp_table WHERE emp_no = 1002;
```

- or Use print statement :

```
PRINT :v
```

Stored Procedure With One Parameter

The following SQL statement creates a stored procedure that selects Customers from a particular City from the "Customers" table:

Example

```
CREATE PROCEDURE SelectAllCustomers @City nvarchar(30)
AS
SELECT * FROM Customers WHERE City = @City
GO;
```

Execute the stored procedure above as follows:

Example

```
EXEC SelectAllCustomers @City = 'London';
```

Stored Procedure With Multiple Parameters

Setting up multiple parameters is very easy. Just list each parameter and the data type separated by a comma as shown below.

The following SQL statement creates a stored procedure that selects Customers from a particular City with a particular PostalCode from the "Customers" table:

Example

```
CREATE PROCEDURE SelectAllCustomers @City nvarchar(30), @PostalCode
nvarchar(10)
AS
SELECT * FROM Customers WHERE City = @City AND PostalCode = @PostalCode
GO;
```

Execute the stored procedure above as follows:

Example

```
EXEC SelectAllCustomers @City = 'London', @PostalCode = 'WA1 1DP';
```