**Stored Procedures**

In Snowflake, **Stored Procedures** allow us to write procedural logic in JavaScript and run multiple SQL statements together. it allows us to write procedural code to execute the sql in database. we can called as Independent Statements. returning a value is optional un Stored procedures. I’ve used them to automate data loading from S3, implement data cleansing logic, and handle incremental deletes. They are especially useful for ETL workflows and when combined with tasks for scheduling.

create or replace table client\_stg

( id NUMBER(38,0), first\_name VARCHAR(16), last\_name VARCHAR(50), sex VARCHAR(1), ethinicity VARCHAR(30), ssn VARCHAR(15), street\_address VARCHAR(90),status VARCHAR(10)

);

-- delete from client\_stg ;

insert into client\_stg values (111111, 'James', 'Schwartz', 'M', 'American','342-76-9087','5676 Washington Street','ACTIVE') ;

insert into client\_stg values (222222, 'Jessica', 'Escobar', 'F', 'Hispanic','456-93-5629','3234 WateringCan Drive','INACTIVE') ;

insert into client\_stg values (333333, 'Ben', 'Hardy', 'M', 'American','876-98-3245','6578 Historic Circle','INACTIVE') ;

insert into client\_stg values (444444, 'Anjali', 'Singh', 'F', 'Indian American','435-87-6532','8978 Autumn Day Drive','ACTIVE') ;

insert into client\_stg values (555555, 'Dean', 'Tracy', 'M', 'African','767-34-7656','2343 India

Street','ACTIVE') ;

select \* from client\_stg ; -- create a target table

create or replace table client\_tgt

( id NUMBER(38,0), first\_name VARCHAR(16), last\_name VARCHAR(50), sex VARCHAR(1), ethinicity VARCHAR(30), ssn VARCHAR(15), street\_address VARCHAR(90),status VARCHAR(10)

);

-- create a procedure

create or replace procedure prc\_load\_client\_tgt(tgt\_table\_name varchar)

returns varchar

language sql

as

declare

stg\_table text default 'practice\_db.snowflake.client\_stg';

begin

insert into identifier(:tgt\_table\_name)

select \* from identifier(:stg\_table);

-- return 'Record Inserted' ;

end;

;

-- test the procedure

select \* from client\_stg ;

select \* from client\_tgt ; -- empty

call prc\_load\_client\_tgt('client\_tgt') ;

select \* from client\_tgt ; -- procedure works