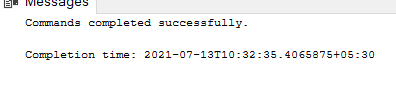
**LAB – 2: TO LEARN STORED PROCEDURES IN SQL**

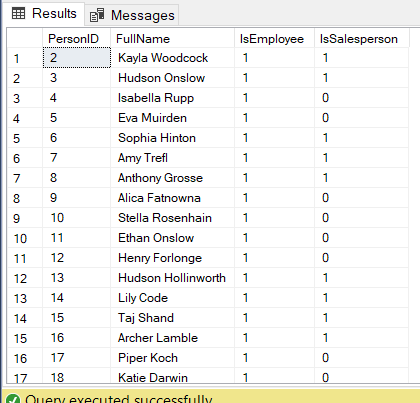
**This Document contains Screenshots of the output by executing queries & Explanation(if required). Queries are attached in another file.**

**1. Stored Procedures**

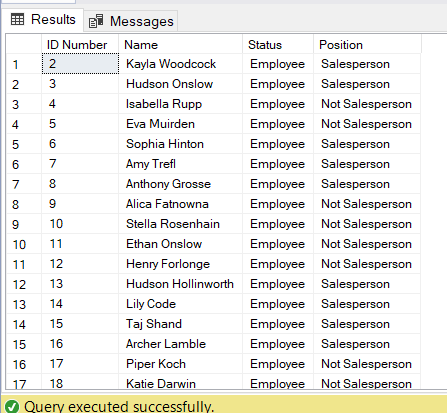
**i) Creating or Altering Stored Procedure ‘**Application.uspViewEmployees**’**



**ii) Executing the above stored procedure**



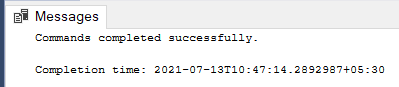
**iii) Altering Stored Procedure to mask column names**



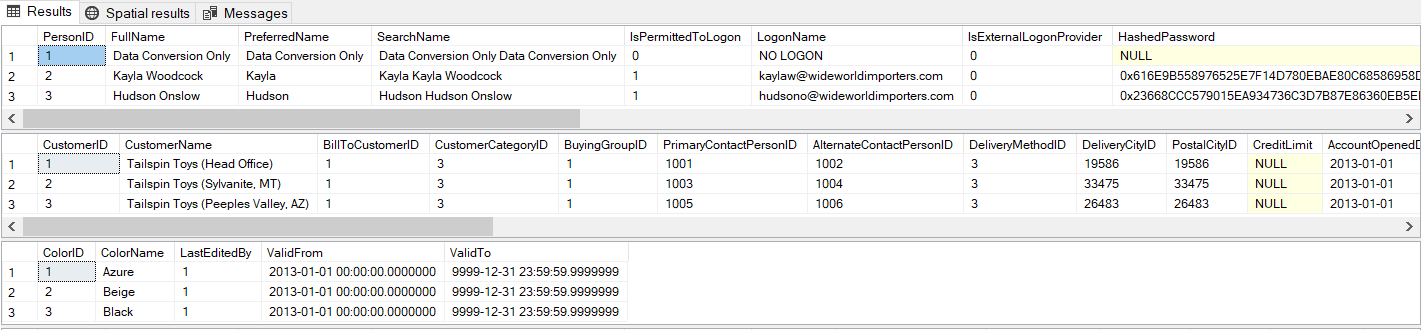
**Explanation:**

This Stored Procedure is helpful as it masks name of all columns to different name in output which is more comprehensible & user-friendly as compared to original column names. It also changes values 1 and 0 from column ‘isSalesperson’ to 2 category - 'Salesperson' & 'Not Salesperson'

**iv) Creating stored procedure Application.uspViewData**



**v) Executing Stored Procedure ‘Application.uspViewData ‘**

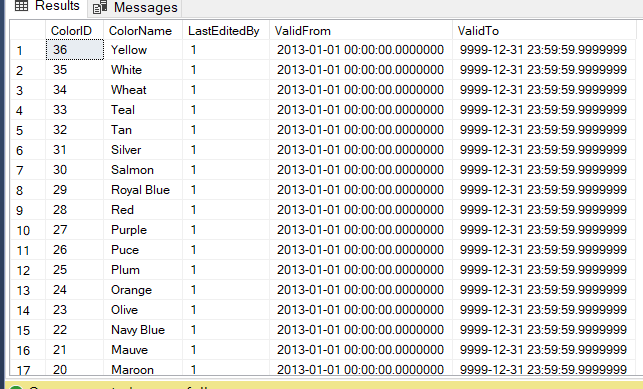


**Explanation:**

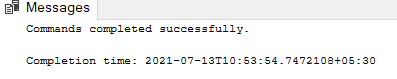
Stored Procedures helps in executing multiple queries in a single statement. This saves lot of time. Hence, we can group bunch of queries and name them with functionality they achieve and execute it directly.

**2. Stored procedure input parameters**

**i) Query to view Data from Warehouse.Colors table in desc order by colorID**



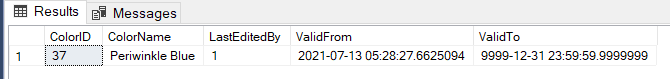
**ii) Creating Stored Procedure named ‘Warehouse.uspInsertColor’ for inserting row**



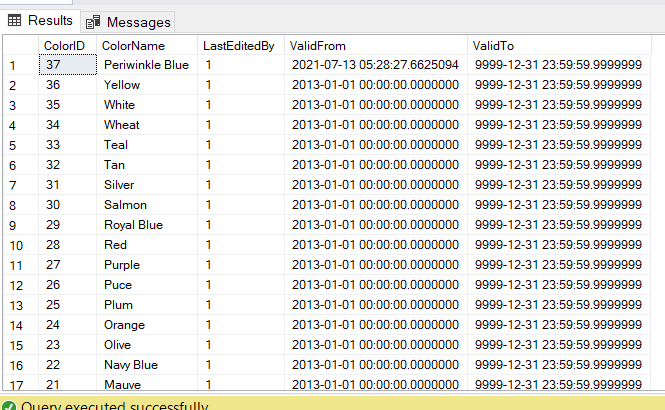
**Explanation:**

Here, ColorId for new entry is generated by taking max id from colorid column and adding 1 to it. Hence, this assumes that new and unique color is inserted. Moreover, Color is taken as input from user in stored procedure

**iii) Executing Stored Procedure for inserting new Color named ‘Periwinkle Blue’**

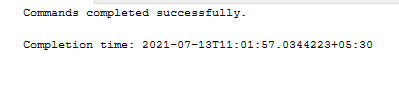


**iv) Query to see the updated Warehouse.Colors table**

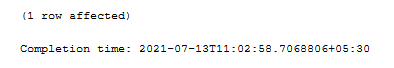


As we can see, our row has been added to the table ‘Warehouse.Colors’.

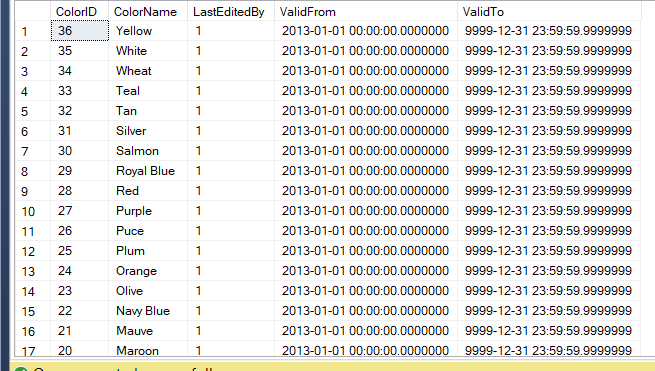
**v) Creating Stored Procedure named ‘Warehouse.uspRemoveLastColor’ for deleting last record from table**



**vi) Executing above Stored Procedure and delete our inserted row**



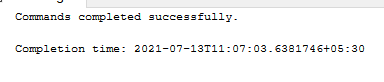
**vii) Query to see the updated Warehouse.Colors table again**



As we can see, our row has been deleted from the table ‘Warehouse.Colors’.

**3. Stored procedures output parameters**

**i) Creating Stored Procedure ‘Application.uspSimpleProcedure’ to pass message to calling application**



**i) Executing above Stored Procedure and printing the message returned by it.**

