CREATE TABLE employee (

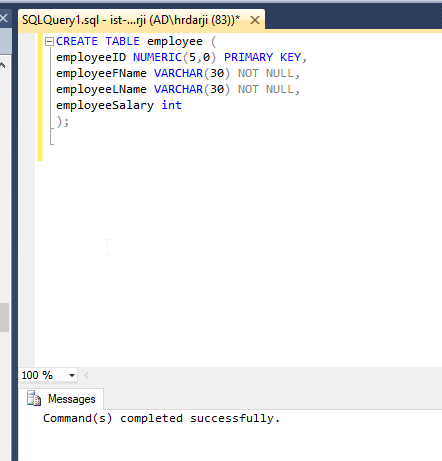
employeeID NUMERIC(5,0) PRIMARY KEY,

employeeFName VARCHAR(30) NOT NULL,

employeeLName VARCHAR(30) NOT NULL,

employeeSalary int

);



CREATE TABLE project (

projectID NUMERIC(4,0) PRIMARY KEY,

projectDesc VARCHAR(200) NOT NULL,

projectStartDate DATETIME DEFAULT getdate(),

projectDuration VARCHAR(30)

);

CREATE TABLE projectAssignment(

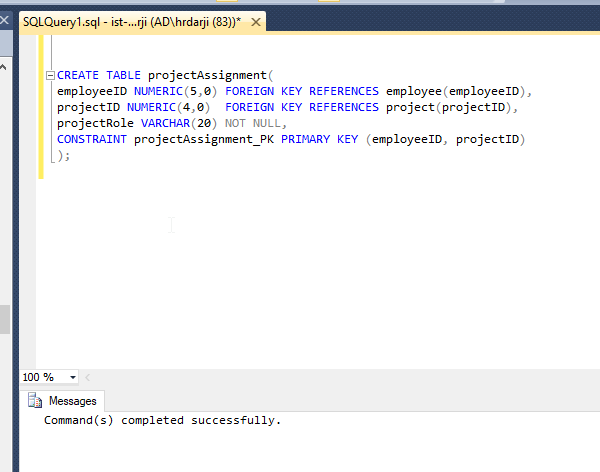
employeeID NUMERIC(5,0) FOREIGN KEY REFERENCES employee(employeeID),

projectID NUMERIC(4,0)  FOREIGN KEY REFERENCES project(projectID),

projectRole VARCHAR(20) NOT NULL,

CONSTRAINT projectAssignment\_PK PRIMARY KEY (employeeID, projectID)

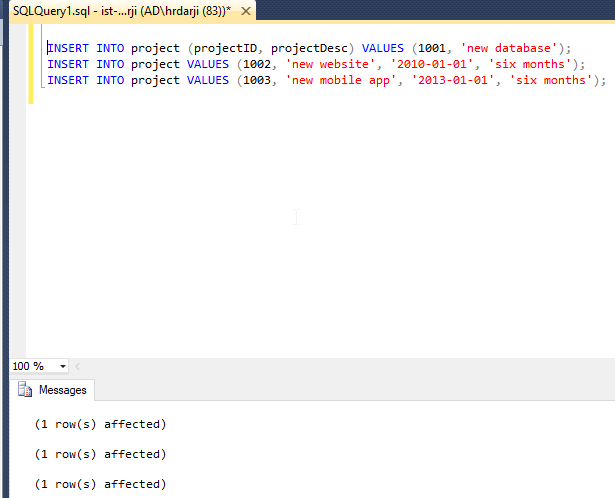
);



INSERT INTO project (projectID, projectDesc) VALUES (1001, 'new database');

INSERT INTO project VALUES (1002, 'new website', '2010-01-01', 'six months');

INSERT INTO project VALUES (1003, 'new mobile app', '2013-01-01', 'six months');



INSERT INTO employee VALUES (11111, 'James', 'Smith', 30000);

INSERT INTO employee VALUES (11112, 'Ada', 'Zack', 40000);

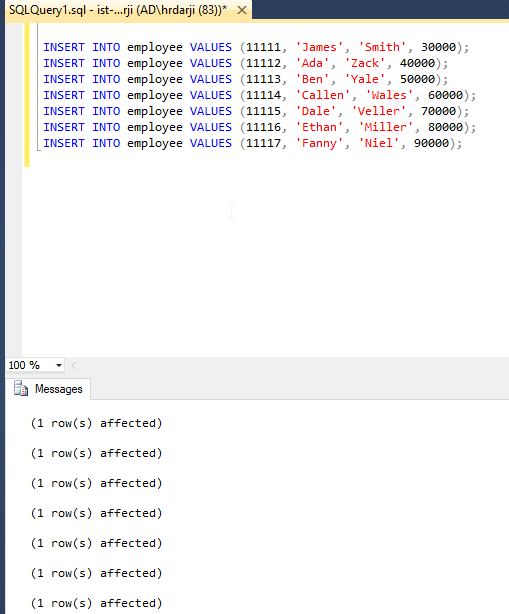
INSERT INTO employee VALUES (11113, 'Ben', 'Yale', 50000);

INSERT INTO employee VALUES (11114, 'Callen', 'Wales', 60000);

INSERT INTO employee VALUES (11115, 'Dale', 'Veller', 70000);

INSERT INTO employee VALUES (11116, 'Ethan', 'Miller', 80000);

INSERT INTO employee VALUES (11117, 'Fanny', 'Niel', 90000);



INSERT INTO projectAssignment VALUES (11111, 1001, 'manager');

INSERT INTO projectAssignment VALUES (11112, 1001, 'PHP programmer');

INSERT INTO projectAssignment VALUES (11113, 1001, 'Oracle DBA');

INSERT INTO projectAssignment VALUES (11114, 1001, 'quality assurance');

INSERT INTO projectAssignment VALUES (11115, 1001, 'test engineer');

INSERT INTO projectAssignment VALUES (11117, 1001, 'Oracle DBA');

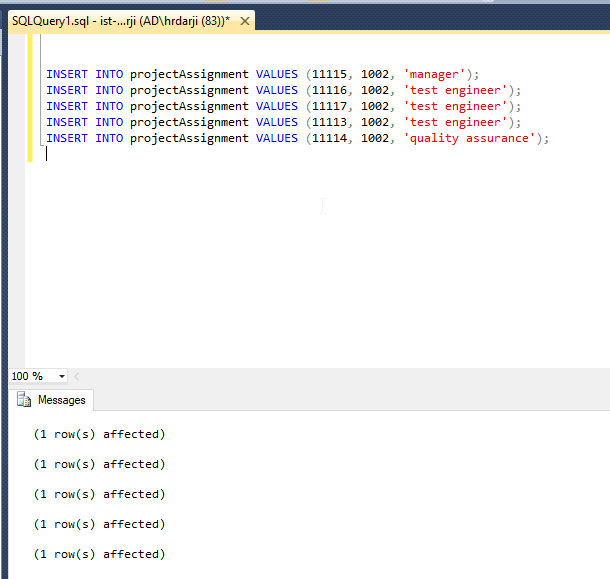
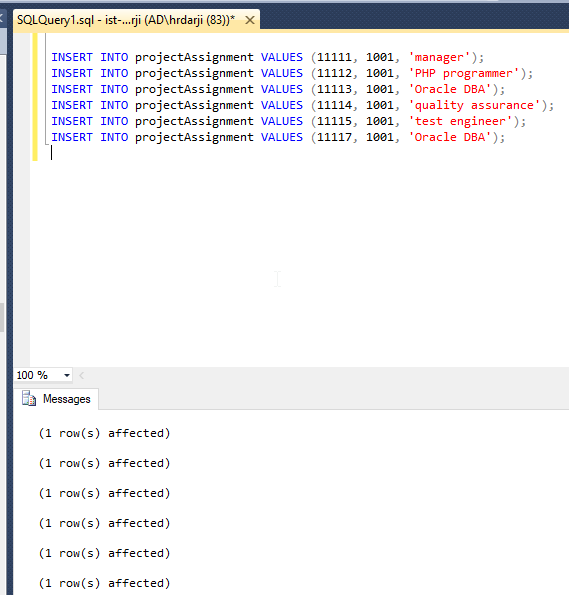
INSERT INTO projectAssignment VALUES (11115, 1002, 'manager');

INSERT INTO projectAssignment VALUES (11116, 1002, 'test engineer');

INSERT INTO projectAssignment VALUES (11117, 1002, 'test engineer');

INSERT INTO projectAssignment VALUES (11113, 1002, 'test engineer');

INSERT INTO projectAssignment VALUES (11114, 1002, 'quality assurance');



ANSWER THE FOLLOWING QUESTIONS

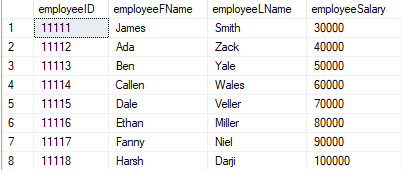
1. Add an employee using your own name and create a project assignment for yourself using existing project id.

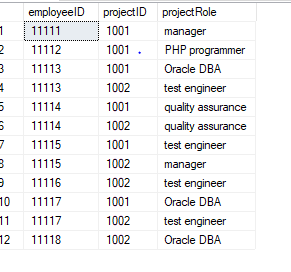
**SQL QUERY:**

INSERT INTO EMPLOYEE VALUES(11118,'Harsh','Darji',100000);

INSERT INTO projectAssignment VALUES(11118,1002,'Oracle DBA');

**SCREENSHOT:**





1. Write a scalar function that returns the average salary of the Employees

**Sql query:**

GO

CREATE FUNCTION EMPLOYEE\_AVG\_SAL ()

RETURNS INT AS

BEGIN

DECLARE @ret INT;

SELECT @ret = AVG (employeeSalary)

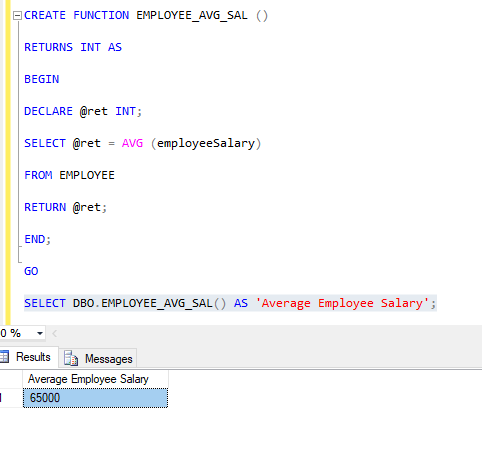
FROM EMPLOYEE

RETURN @ret;

END;

GO

SELECT DBO.EMPLOYEE\_AVG\_SAL() AS 'Average Employee Salary';

**SCREENSHOT:** 

1. Write a table-valued function that returns the Projects given an EmployeeID as a parameter and
   1. Show the function created
   2. return the results for your own project

**SQL QUERY:**

GO

CREATE FUNCTION RETURN\_PROJECTS (@EMP\_ID INT)

RETURNS TABLE

AS

RETURN

(

SELECT \*

FROM project

WHERE projectID IN

(

SELECT projectID

FROM projectAssignment

WHERE employeeID= @EMP\_ID

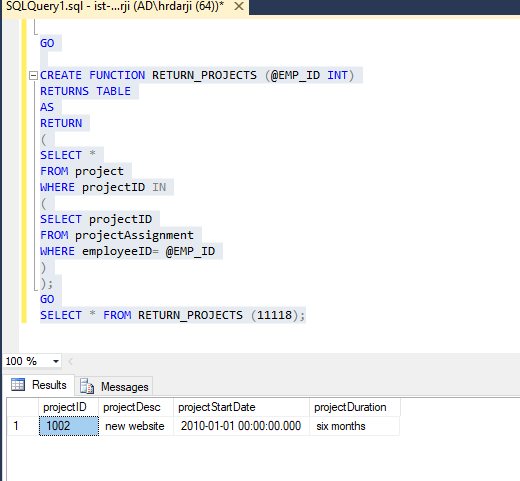
)

);

GO

SELECT \* FROM RETURN\_PROJECTS (11118);

**SCREENSHOT:**



1. Alter the Employee table to add a new column called ‘Num of Projects’ which can be **INTEGER** data type. Write a procedure that updates employee table with the total projects assigned to each employee (to the newly created column)

**SQL QUEY:**

ALTER table employee

add Number\_Projects varchar(23) ;

select \* from employee;

go

CREATE PROCEDURE TOTAL\_PROJ

AS

BEGIN

UPDATE employee

SET Number\_Projects = projCount.total\_count

FROM

(

SELECT employeeID,count(projectID) AS total\_count

FROM projectAssignment

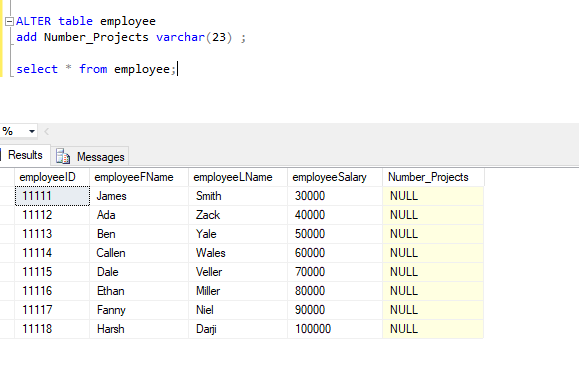
GROUP BY employeeID

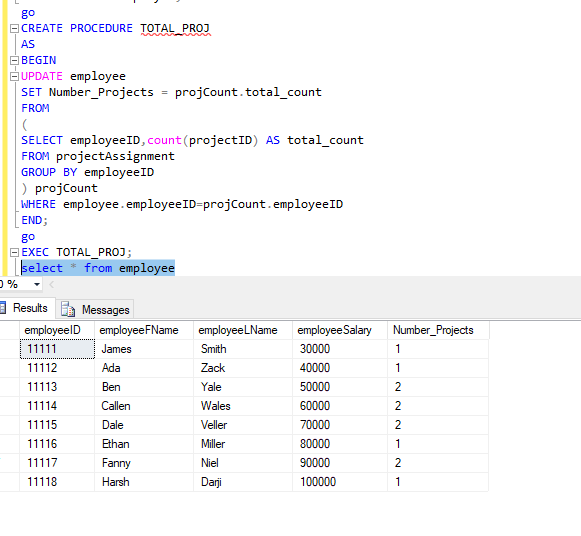
) projCount

WHERE employee.employeeID=projCount.employeeID

END;

**SCREENSHOT:**





1. Create a trigger that can update the num of projects whenever a new project is assigned to an employee.
   1. Test the trigger with the below insert

INSERT INTO projectAssignment VALUES (11114, 1003, ‘quality assurance');

INSERT INTO projectAssignment VALUES (11115, 1003,'test engineer');

**SQL QUEY:**

go

CREATE TRIGGER NumofProjectsTrigger

ON projectAssignment

FOR INSERT,UPDATE

AS

IF @@ROWCOUNT >= 1

BEGIN

UPDATE EMPLOYEE

SET Number\_Projects = projCount.total\_count

FROM

(

SELECT employeeID,count(projectID) AS total\_count

FROM projectAssignment

GROUP BY employeeID

) projCount

WHERE employee.employeeID=projCount.employeeID

END;

go

**SCREENSHOT:**

