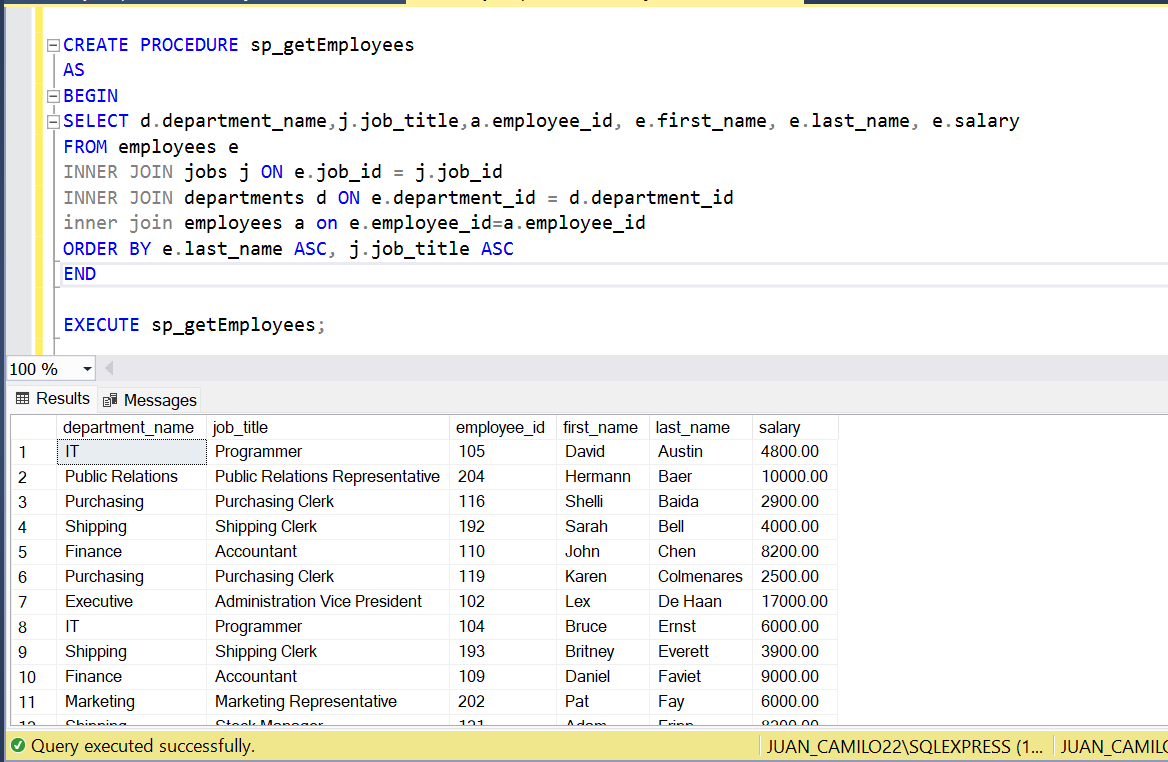
**Actividad**

Cree e importe la base de datos HR en SQL SERVER.

Relacione la sentencia SQL empleada:

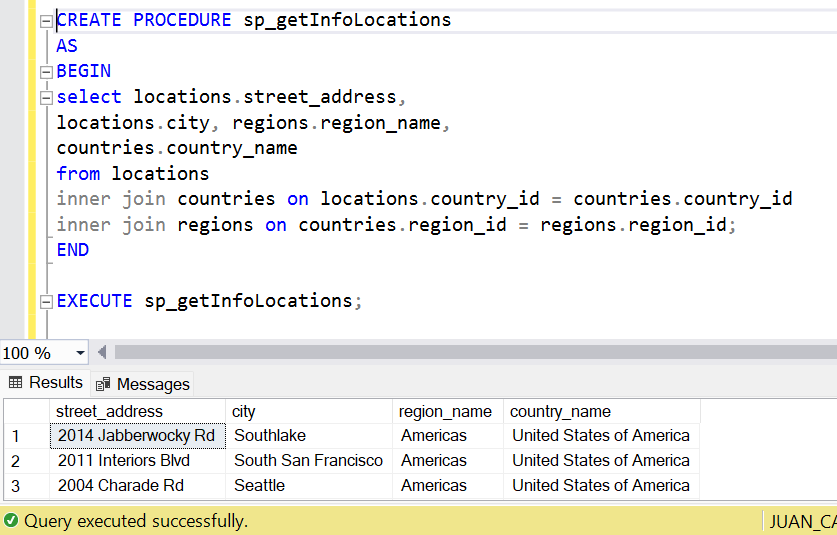
1. Cree el procedimiento almacenado sp\_getEmployees en la base de datos HR, que recupere nombre del departamento donde trabaja, cargo, id, Nombres, Apellidos y salario de cada uno de los empleados.

| PL/SQL | CREATE PROCEDURE sp\_getEmployees  AS  BEGIN  SELECT d.department\_name,j.job\_title,a.employee\_id, e.first\_name, e.last\_name, e.salary  FROM employees e  INNER JOIN jobs j ON e.job\_id = j.job\_id  INNER JOIN departments d ON e.department\_id = d.department\_id  inner join employees a on e.employee\_id=a.employee\_id  ORDER BY e.last\_name ASC, j.job\_title ASC  END  EXECUTE sp\_getEmployees; |
| --- | --- |



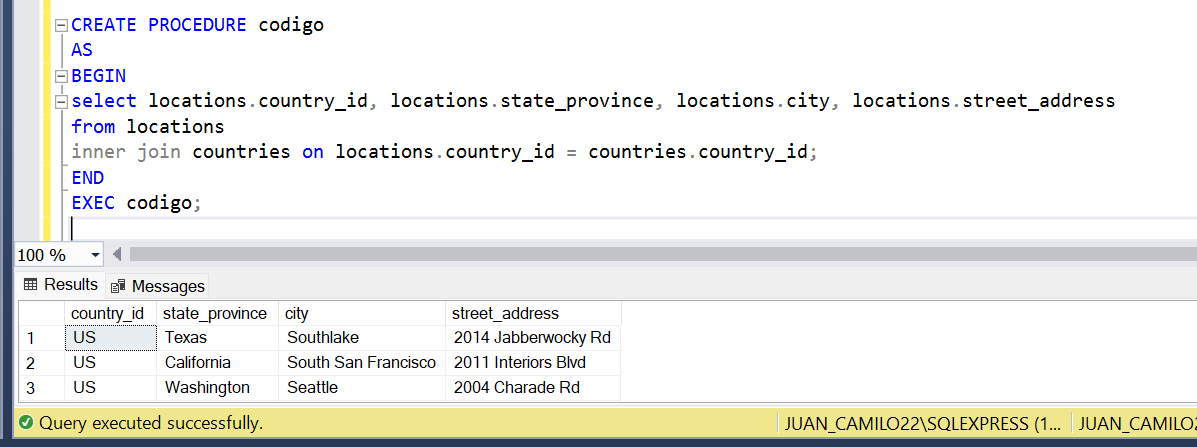
2. Cree el procedimiento almacenado sp\_getInfoLocations en la base de datos HR, que recupere el nombre del país, nombre de la ciudad, nombre de la región y dirección de cada ubicación (sucursal u oficina).

| PL/SQL | CREATE PROCEDURE sp\_getInfoLocations  AS  BEGIN  select locations.street\_address,  locations.city, regions.region\_name,  countries.country\_name  from locations  inner join countries on locations.country\_id = countries.country\_id  inner join regions on countries.region\_id = regions.region\_id;  END  EXECUTE sp\_getInfoLocations; |
| --- | --- |



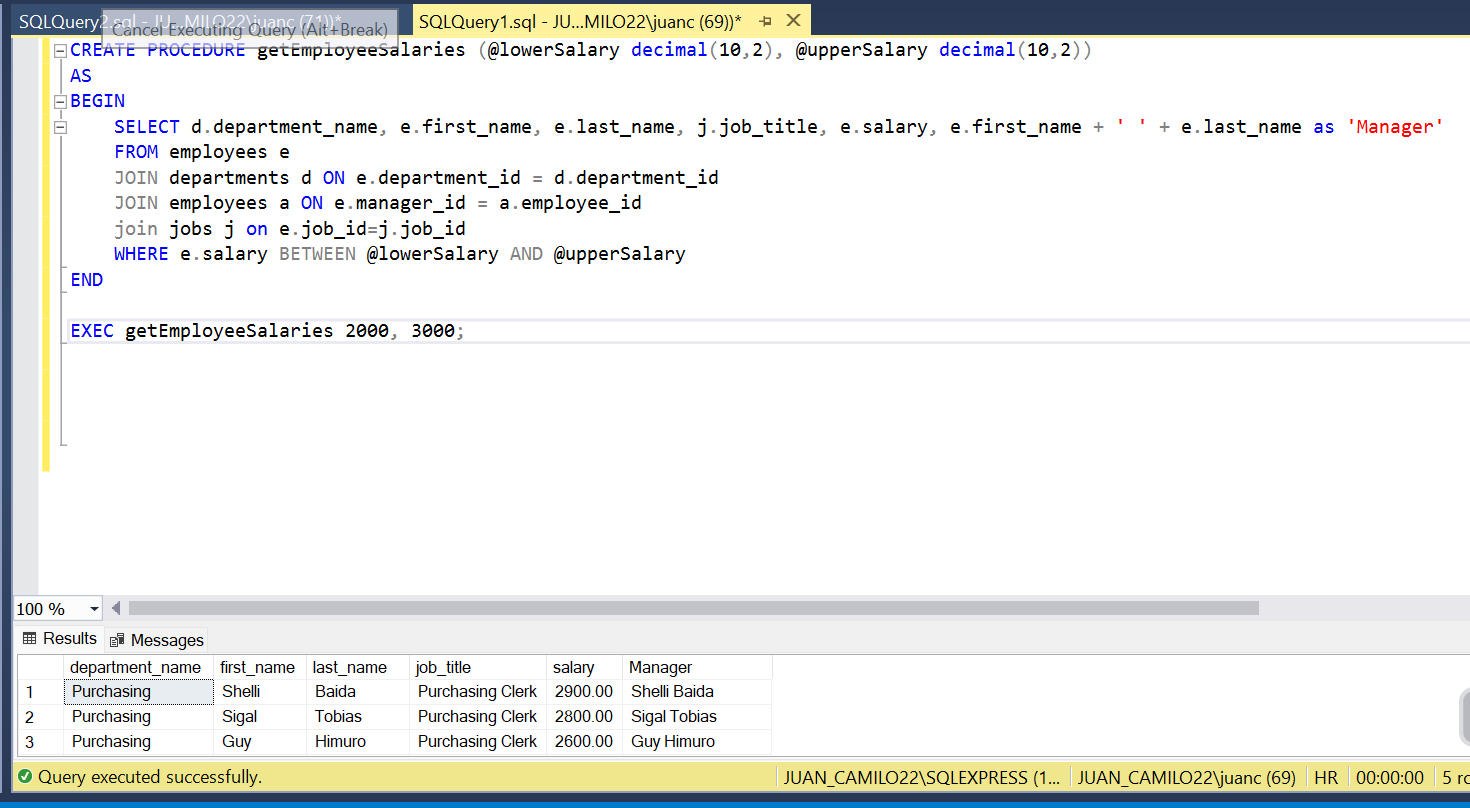
3. Cree un procedimiento almacenado que recupere el estado, la ciudad y la dirección de las oficinas recibiendo por parámetro in el código del país.

| PL/SQL | CREATE PROCEDURE codigo  AS  BEGIN  select locations.country\_id, locations.state\_province, locations.city, locations.street\_address  from locations  inner join countries on locations.country\_id = countries.country\_id;  END  EXEC codigo; |
| --- | --- |



4. Desarrollar un procedimiento almacenado que recupere departamento, nombres, apellidos, cargo, salario y nombre del jefe de los empleados que tengan salario dentro de un rango determinado. El procedimiento debe recibir como parámetros el límite inferior y superior del rango.

| PL/SQL | CREATE PROCEDURE getEmployeeSalaries (@lowerSalary decimal(10,2), @upperSalary decimal(10,2))  AS  BEGIN  SELECT d.department\_name, e.first\_name, e.last\_name, j.job\_title, e.salary, e.first\_name + ' ' + e.last\_name as 'Manager'  FROM employees e  JOIN departments d ON e.department\_id = d.department\_id  JOIN employees a ON e.manager\_id = a.employee\_id  join jobs j on e.job\_id=j.job\_id  WHERE e.salary BETWEEN @lowerSalary AND @upperSalary  END  para ejecutar:  EXEC getEmployeeSalaries 2000, 3000; (se pueden cambiar estos valores ) |
| --- | --- |



5. Se desea dar un bono a los empleados. De acuerdo a la siguiente escala salarial:

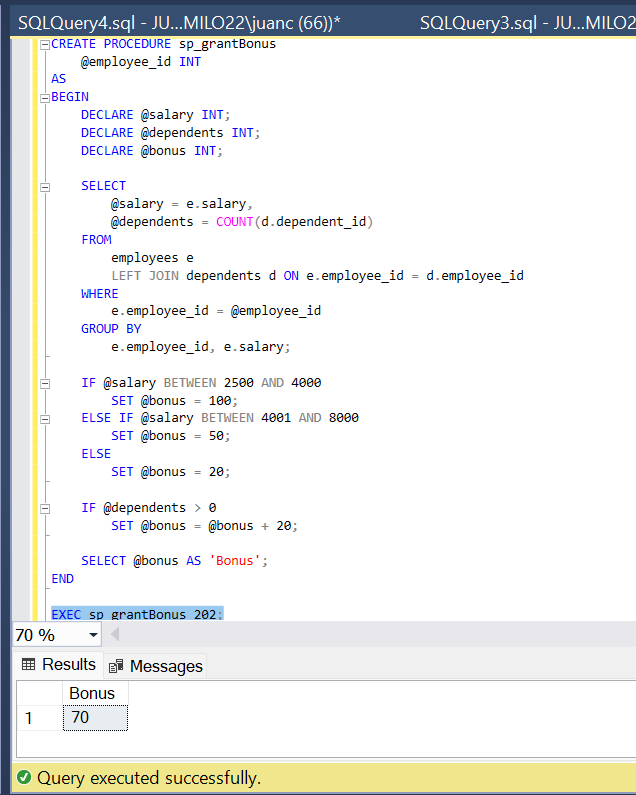
De 2500 dólares a 4000 dólares un bono de 100 dólares.

De 4001 dólares a 8000 dólares un bono de 50 dólares.

De 8001 dólares en adelante un bono de 20 dólares

Si el empleado tiene dependientes (hijos) se le darán 20 dólares adicionales. El procedimiento debe tener como parámetro de entrada el employee\_id.

| Sql | CREATE PROCEDURE sp\_grantBonus  @employee\_id INT  AS  BEGIN  DECLARE @salary INT;  DECLARE @dependents INT;  DECLARE @bonus INT;  SELECT  @salary = e.salary,  @dependents = COUNT(d.dependent\_id)  FROM  employees e  inner JOIN dependents d ON e.employee\_id = d.employee\_id  WHERE  e.employee\_id = @employee\_id  GROUP BY  e.employee\_id, e.salary;  IF @salary BETWEEN 2500 AND 4000  SET @bonus = 100;  ELSE IF @salary BETWEEN 4001 AND 8000  SET @bonus = 50;  ELSE  SET @bonus = 20;  IF @dependents = 0  SET @bonus = @bonus + 20;  (SELECT @bonus AS Bonus);  END\*/  execute sp\_grantBonus 115; |
| --- | --- |



1. Desarrollar un procedimiento almacenado que calcule la antigüedad de un empleado recibiendo como parámetro el employee\_id.

| Sql | CREATE PROCEDURE calcular\_antiguedad @employee\_id INT  AS  BEGIN  DECLARE @fecha\_contratacion DATE  DECLARE @antiguedad INT  SELECT @fecha\_contratacion = hire\_date  FROM employees  WHERE employee\_id = @employee\_id  SELECT @antiguedad = DATEDIFF(YEAR, @fecha\_contratacion, GETDATE())  SELECT @antiguedad AS antiguedad  END  EXEC calcular\_antiguedad 115; |
| --- | --- |

