



**Universidad Tecnológica de Panamá**  
**Facultad de Ingeniería de Sistemas Computacionales**  
**BASE DE DATOS I**  
**Laboratorio N°5**



**Facilitador:** Víctor A. Fuentes T.  
**Cédula:** 8-914-587

**Estudiante:** Johel Heraclio Batista Cárdenas  
**Grupo:** 1IF-131

**A. TÍTULO DE LA EXPERIENCIA:**

Laboratorio No.5. Filas de salida condicionada con top y el uso de operadores In, Between y Like.

**B. TEMAS:**

- I. Filas de salida limitadas con top
- II. Operador In (list)
- III. Operador Between...And
- IV. Operador Like

**C. OBJETIVO(S):**

- Ampliar el uso del SELECT, utilizando en las cláusulas TOP, IN, Between y Like

**D. METODOLOGÍA:**

Para presentar el informe de los resultados obtenidos, coloque el código usado en forma de texto de forma directa y no como parte de la captura de pantalla. Use captura de pantalla desde el SQL Server para mostrar el resultado de la consulta.

Copie estas capturas de pantalla en la sección G (RESULTADOS) de esta guía, en según el número mostrado en la sección E (PROCEDIMIENTO). Corte y sólo presente el área de trabajo donde aparece el resultado obtenido, no incluya el

explorador de objetos, el menú de opciones ni la barra de herramientas estándar en su respuesta final.

**E. PROCEDIMIENTO O ENUNCIADO DE LA EXPERIENCIA:** (todo lo indicado en color verde corresponden a acciones que usted deberá ejecutar)

**EI. Inicie sesión en SQL SERVER, active la base de datos PUBS. Realice cada una de los ejemplos mostrados, a fin de que pueda asimilar el concepto. Estos ejemplos NO debe colocarlos en su informe.**

Además de los operadores de comparación (<, >, >=, <=) vistos en el laboratorio pasado, la sentencia WHERE puede usarse para condiciones especiales definidas por el sistema de gestión, en este caso, MS SQL Server. He aquí las más comunes:

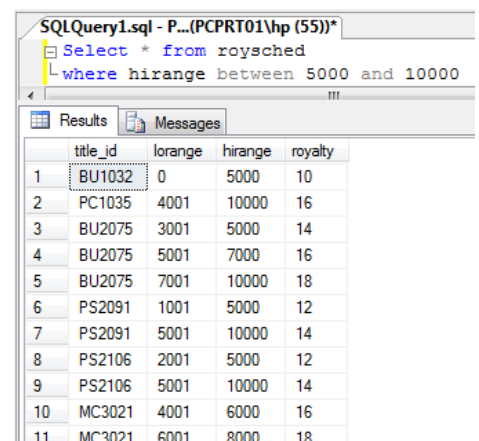
- **Between... and**

Devuelve todos los valores comprendidos entre los rangos especificados (incluye los rangos menor y mayor).

Debe especificarse primero el rango menor. En el caso de que el criterio de búsqueda sea alfanumérico o tipo fecha, debe colocarse entre comilla simple.

**Ejemplo 1.** Obtenga de la tabla roysched, todos los valores cuyo hirange están comprendidos entre 5000 y 10000.

Note que se tiene entre los datos un 6000, un 7000 y 8000 además de los extremos listados (5000 y 10000). Es decir, el Between devuelve todo lo que se encuentra entre los rangos dados. Visto de otra forma, equivale a listar los **hirange >=5000 y hirange <=10000**



The screenshot shows a SQL query window with the following text:

```
SQLQuery1.sql - P...(PCPRT01\hp (55))*  
Select * from roysched  
Where hirange between 5000 and 10000
```

Below the query window, the 'Results' tab is active, displaying a table with 11 rows and 4 columns: title\_id, lorange, hirange, and royalty.

	title_id	lorange	hirange	royalty
1	BU1032	0	5000	10
2	PC1035	4001	10000	16
3	BU2075	3001	5000	14
4	BU2075	5001	7000	16
5	BU2075	7001	10000	18
6	PS2091	1001	5000	12
7	PS2091	5001	10000	14
8	PS2106	2001	5000	12
9	PS2106	5001	10000	14
10	MC3021	4001	6000	16
11	MC3021	6001	8000	18

**Ejemplo 2:** Obtener de la tabla authors todos los valores comprendidos entre el nombre Bennet y Dull.

Para poder distinguir claramente quiénes son, inicialmente puede hacer un select de todos los campos, pero ordenado por lname.

SQLQuery1.sql - P...(PCPRT01\hp (55))\*

```
Select * from authors
where au_lname between 'Bennet' and 'Dull'
```

Results Messages

	au_id	au_lname	au_fname	phone	address
1	238-95-7766	Carson	Cheryl	415 548-7723	589 D
2	409-56-7008	Bennet	Abraham	415 658-9932	6223
3	427-17-2319	Dull	Ann	415 836-7128	3410
4	648-92-1872	Blotchet-Halls	Reginald	503 745-6402	55 Hill
5	712-45-1867	del Castillo	Innes	615 996-8275	2286
6	722-51-5454	DeFrance	Michel	219 547-9982	3 Bald

**Ejemplo 3:** Es importante señalar que la búsqueda, no hace distinción entre mayúsculas y minúsculas, aunque si distingue por las letras que siguen. Note que, al consultar los campos comprendidos entre b y d, sólo se listan tres tuplas en tanto que si la consulta se realiza por “Dull”, se listarán todos los campos tal cual el ejemplo 2. Esto se debe a que al momento que SQL ordena los datos, la ‘d’ como tal, está antes que la ‘Dull’.

SQLQuery1.sql - P...(PCPRT01\hp (55))\*

```
where au_lname between 'b' and 'd'
```

Results Messages

	au_id	au_lname	au_fname	phone
1	238-95-7766	Carson	Cheryl	415 548-77
2	409-56-7008	Bennet	Abraham	415 658-99
3	648-92-1872	Blotchet-Halls	Reginald	503 745-64

SQLQuery1.sql - P...(PCPRT01\hp (55))\*

```
where au_lname between 'b' and 'dull'
```

Results Messages

	au_id	au_lname	au_fname	phone	address
1	238-95-7766	Carson	Cheryl	415 548-7723	589
2	409-56-7008	Bennet	Abraham	415 658-9932	622
3	427-17-2319	Dull	Ann	415 836-7128	341
4	648-92-1872	Blotchet-Halls	Reginald	503 745-6402	55
5	712-45-1867	del Castillo	Innes	615 996-8275	228
6	722-51-5454	DeFrance	Michel	219 547-9982	3 B

**2. IN (list):** Encuentra y lista SÓLO los valores especificados en la lista colocada en el paréntesis. Es similar al igual, pero permite listar por varios criterios igual a la vez.

La selección funciona como un **OR**; es decir

- Si sólo hay una condición que se cumple, lista todos los campos que cumplen con esta condición
- Si se cumplen las dos condiciones, lista las tuplas que cumplen con las dos condiciones, pero
- Si no se cumple ninguna, la consulta es nula

**Ejemplo 1:** Muestre todos los campos de la tabla authors cuya city es igual a Oakland o a San Jose.

SQLQuery1.sql - P...(PCPRT01\hp (55))\*

```
Select * from authors
where city in('oakland', 'san jose')
```

	au_id	au_lname	au_fname	phone	address	city
1	213-46-8915	Green	Marjorie	415 986-7020	309 63rd St. #411	Oakland
2	267-41-2394	O'Leary	Michael	408 286-2428	22 Cleveland Av. #14	San Jose
3	274-80-9391	Straight	Dean	415 834-2919	5420 College Av.	Oakland
4	724-08-9931	Stringer	Dirk	415 843-2991	5420 Telegraph Av.	Oakland
5	724-80-9391	MacFeather	Stearns	415 354-7128	44 Upland Hts.	Oakland
6	756-30-7391	Karsen	Livia	415 534-9219	5720 McAuley St.	Oakland

**Ejemplo 2:** Muestre todos los campos de la tabla authors cuya city es igual a Oakland o a Panamá.

SQLQuery1.sql - P...(PCPRT01\hp (55))\*

```
Select * from authors
where city in('oakland', 'panama')
```

	au_id	au_lname	au_fname	phone	address	city
1	213-46-8915	Green	Marjorie	415 986-7020	309 63rd St. #411	Oakland
2	274-80-9391	Straight	Dean	415 834-2919	5420 College Av.	Oakland
3	724-08-9931	Stringer	Dirk	415 843-2991	5420 Telegraph Av.	Oakland
4	724-80-9391	MacFeather	Stearns	415 354-7128	44 Upland Hts.	Oakland
5	756-30-7391	Karsen	Livia	415 534-9219	5720 McAuley St.	Oakland

Como Panamá no aparece en la tabla authors, sólo trae Oakland.

### 3.Like:

Compara un campo con un carácter o cadena alfanumérica y muestra todos los datos que contienen dicha cadena o carácter. LIKE admite el uso de caracteres comodines.

Like 'm'	Trae todo lo que sea exactamente igual a m. No tiene ningún otro caracter
Like 'm%'	Trae todo lo que empieza con m, sin importar los caracteres que le siguen
Like '%m'	Trae todo lo que termina como m, sin importar los caracteres que le anteceden
Like '_m%'	El carácter de Subrayado permite omitir un carácter. En este caso trae lo que tenga como segunda letra una m, sin importar que sigue después de la m
Puede combinar los diferentes comodines	

**Ejemplo 1.** Traer de la bd authors todo el au\_fname que empiece con M

SQLQuery1.sql - P...(PCPRT01\hp (55))\*

```

Select * from authors
where au_fname like 'M%'

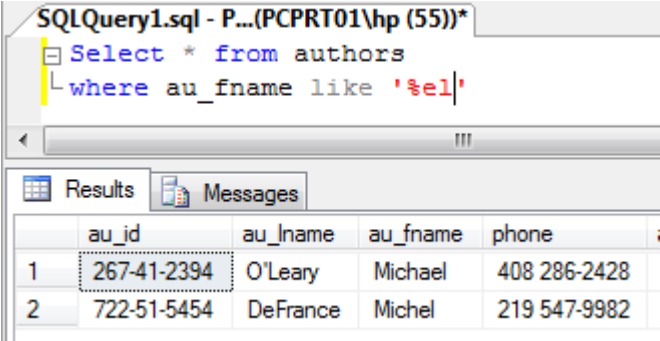
```

Results Messages

	au_id	au_lname	au_fname	phone	address	city
1	213-46-8915	Green	Marjorie	415 986-7020	309 63rd St. #411	Oakland
2	267-41-2394	O'Leary	Michael	408 286-2428	22 Cleveland Av. #14	San Jose
3	341-22-1782	Smith	Meander	913 843-0462	10 Mississippi Dr.	Lawrence
4	527-72-3246	Greene	Momingstar	615 297-2723	22 Graybar House Rd.	Nashville
5	722-51-5454	DeFrance	Michel	219 547-9982	3 Balding Pl.	Gary

## Ejemplo 2.

Traer de la tabla authors todo el au\_fname  
que termine con el



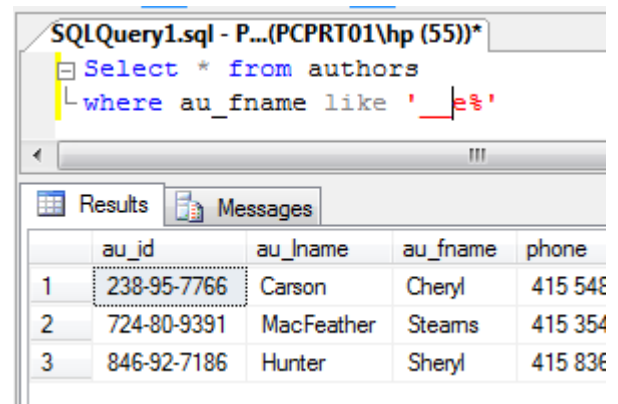
The screenshot shows a SQL query window titled "SQLQuery1.sql - P...(PCPRT01\hp (55))\*". The query is: `Select * from authors where au_fname like '%el'`. Below the query, the "Results" tab is active, displaying a table with 5 columns: au\_id, au\_lname, au\_fname, and phone. The table contains two rows of data. The first row has au\_id 1, au\_lname O'Leary, au\_fname Michael, and phone 408 286-2428. The second row has au\_id 2, au\_lname DeFrance, au\_fname Michel, and phone 219 547-9982. The au\_fname column is highlighted in the first row.

	au_id	au_lname	au_fname	phone
1	267-41-2394	O'Leary	Michael	408 286-2428
2	722-51-5454	DeFrance	Michel	219 547-9982

## Ejemplo 2

Traer de la tabla authors todo el au\_fname cuya **tercera letra sea e.**

(aunque no se ve claro en la consulta, hay dos subrayados seguidos, por lo que omite los dos primeros caracteres e inicia la comparación a partir del tercer carácter.



The screenshot shows a SQL query window titled 'SQLQuery1.sql - P...(PCPRT01\hp (55))'. The query is: `Select * from authors where au_fname like ' _e%'`. Below the query, the 'Results' tab is active, displaying a table with 5 columns: au\_id, au\_lname, au\_fname, and phone. The results show three rows of data.

	au_id	au_lname	au_fname	phone
1	238-95-7766	Carson	Cheryl	415 548
2	724-80-9391	MacFeather	Steams	415 354
3	846-92-7186	Hunter	Sheryl	415 836

## 4. Uso de TOP

### Formato

```
Select Top NoRegistrosDeseados campo1, campo2, ...campon
From nombre tabla
Where .....
Order by ...
```

En ocasiones sólo se desea obtener un número limitado de filas o registros. El uso de la sentencia TOP NoRegistros permite controlar el número de filas que obtendrán como resultado de la instrucción.

Para comprender mejor el uso de este, observe qué ocurre cuando lo incluye en una consulta.



```
select au_lname, state from authors
order by au_lname
```

	au_lname	state
1	Bennet	CA
2	Blotch-Halls	OR
3	Carson	CA
4	DeFrance	IN
5	del Castillo	MI
6	Dull	CA
7	Green	CA
8	Greene	TN
9	Gringlesby	CA
10	Hunter	CA
11	Karsen	CA
12	Locksley	CA
13	MacFeather	CA
14	McBadden	CA
15	O'Leary	CA
16	Panteley	MD
17	Ringer	UT
18	Ringer	UT
19	Smith	KS
20	Straight	CA
21	Stringer	CA
22	White	CA
23	Yokomoto	CA

```
select top 10 au_lname, state from authors
order by au_lname
```

	au_lname	state
1	Bennet	CA
2	Blotch-Halls	OR
3	Carson	CA
4	DeFrance	IN
5	del Castillo	MI
6	Dull	CA
7	Green	CA
8	Greene	TN
9	Gringlesby	CA
10	Hunter	CA

Esta consulta limita los resultados a las 10 primeras filas, listando solo los primeros 10 apellidos de autores que encuentra al ordenar los datos.

## F. RECURSOS:

Computador con acceso a internet, Software SQL Server, acceso a plataforma [ecampus.utp.ac.pa/moodle](http://ecampus.utp.ac.pa/moodle), curso de Base de Datos 1.

## G. RESULTADOS:

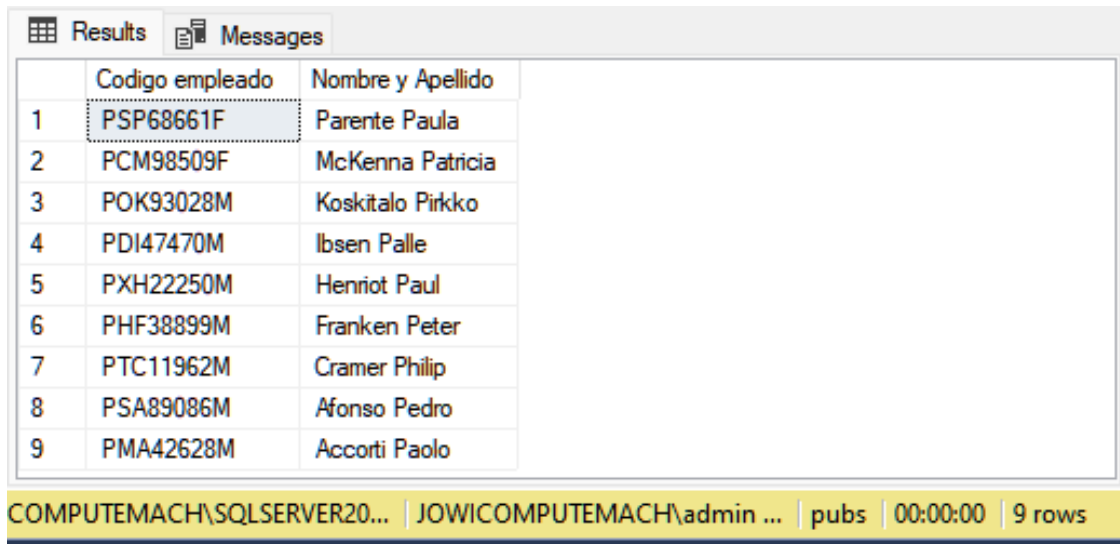
*En esta sección Usted colocará las capturas de pantalla que muestran los resultados de los procesos realizados en el punto anterior.*

## ACTIVE LA BD PUBS.

1. Conociendo que el código del empleado (emp\_id) de la tabla employee tiene como primera letra la referencia del departamento en el que trabaja el empleado, liste el código del empleado, el nombre y apellido en orden

**descendente, de todos los empleados que laboran en el departamento de Producción (P).**

```
select [emp_id] 'Codigo empleado',  
       [lname]+' '+'[fname] 'Nombre y Apellido'  
from [dbo].[employee]  
where [emp_id] like 'P%'  
order by 2 desc
```



	Codigo empleado	Nombre y Apellido
1	PSP68661F	Parente Paula
2	PCM98509F	McKenna Patricia
3	POK93028M	Koskitalo Pirkko
4	PDI47470M	Ibsen Palle
5	PXH22250M	Henriot Paul
6	PHF38899M	Franken Peter
7	PTC11962M	Cramer Philip
8	PSA89086M	Afonso Pedro
9	PMA42628M	Accorti Paolo

COMPUTEMACH\SQLSERVER20... | JOWICOMPUTEMACH\admin ... | pubs | 00:00:00 | 9 rows

**2. Liste descendente el nombre completo del empleado y el job\_id de todos los empleados de la tabla employee, con job\_id igual a 5, 6 y 7. Etiqueta la columna del nombre completo con el título EMPLEADO y la columna job\_id como CODIGO\_DEPTO**

```
select [fname] + ' '+'[lname] 'Empleado',  
       [job_id] 'Codigo depto'  
from [dbo].[employee]  
where [job_id] in (5,6,7)  
order by 1 desc
```

Results		Messages
	Empleado	Codigo depto
1	Victoria Ashworth	6
2	Sven Ottlieb	5
3	Rita Muller	5
4	Paul Henriot	5
5	Palle Ibsen	7
6	Matti Karttunen	6
7	Maria Pontes	5
8	Maria Larsson	7
9	Lesley Brown	7
10	Laurence Leblan	5
11	Janine Labrune	5
12	Helvetius Nagy	7
13	Diego Roel	6
14	Carlos Hemadez	5
15	Annette Roulet	6

✓ JOWICOMPUTEMACH\SQLSERVER20... JOWICOMPUTEMACH\admin ... pubs 00:00:00 15 rows

3. Liste ascendentemente el nombre y apellido de todos los empleados de la tabla employee, cuyo nombre (fname) empiece con m y termina con a.

```
select [fname] + ' ' + [lname] 'Empleado'
from [dbo].[employee]
where [fname] LIKE 'M%A'
```

Results		Messages
	Empleado	
1	Maria Larsson	
2	Maria Pontes	

SQLSERVER20... JOWICOMPUTEMACH\admin ... pubs 00:00:00 2 rows

4. Liste de la tabla titleauthor el title\_id y el royaltyper de los royaltyper mayores de 70 ordenados por title\_id.

```
select [title_id],
       [royaltyper]
from [dbo].[titleauthor]
```

```

where [royaltyper] between 70 and 100
order by 1

```

	title_id	royaltyper
1	BU2075	100
2	BU7832	100
3	MC2222	100
4	MC3021	75
5	PC1035	100
6	PC9999	100
7	PS1372	75
8	PS2106	100
9	PS3333	100
10	PS7777	100
11	TC3218	100
12	TC4203	100

JOWICOMPUTEMACH\SQLSERVER20... | JOWICOMPUTEMACH\admin ... | pubs | 00:00:00 | 12 rows

5. Liste el title\_id, el código de la tienda (stor\_id) y las cantidades (qty) de las 5 mayores ventas (qty mayores). Utilice para tal fin la tabla Sales

```

select      top 5 [title_id],
            [stor_id],
            [qty]
            from [dbo].[sales]
order by 3 desc

```

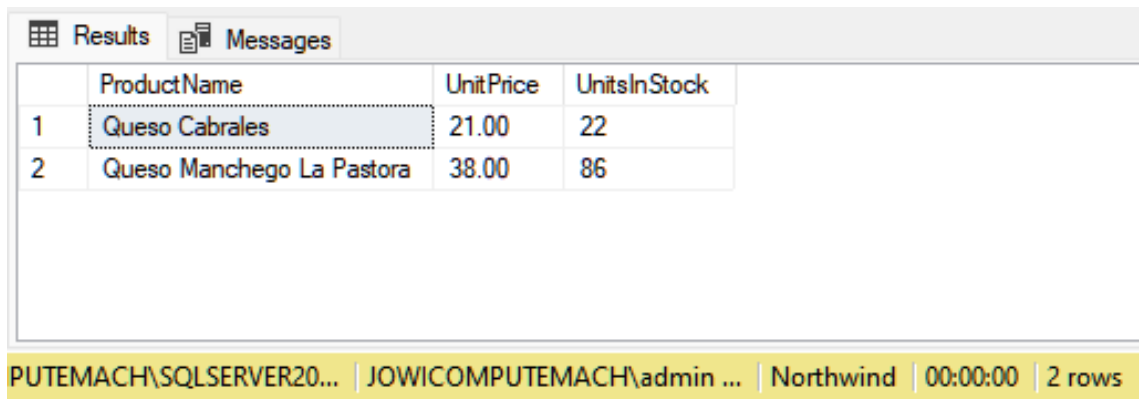
	title_id	stor_id	qty
1	PS2091	7066	75
2	PC8888	7066	50
3	TC3218	7067	40
4	BU2075	7896	35
5	PC1035	8042	30

ICOMPUTEMACH\SQLSERVER20... | JOWICOMPUTEMACH\admin ... | pubs | 00:00:00 | 5 rows

6. Active la Base de Datos NORTWIND. Liste el Nombre del Producto, el precio y las unidades en Stock de todos los Quesos o Chess que se venden. Utilice

**para tal fin la tabla Products.** (Dependiendo de si los datos están en inglés o español, realizará la búsqueda por Chess o por Queso)

```
select [ProductName],[UnitPrice],[UnitsInStock]
      from [dbo].[Products]
     where [ProductName] like 'q%'
```



The screenshot shows the SQL Server Enterprise Manager interface. At the top, there are tabs for 'Results' and 'Messages'. Below the tabs is a table with the following data:

	ProductName	UnitPrice	UnitsInStock
1	Queso Cabrales	21.00	22
2	Queso Manchego La Pastora	38.00	86

At the bottom of the window, a status bar displays the following information: PUTEMACH\SQLSERVER20... | JOWICOMPUTEMACH\admin ... | Northwind | 00:00:00 | 2 rows

**7. Se le ha solicitado mostrar el ID del Producto, el nombre del producto y Cantidades por unidad (QuantityPerUnit) de los productos que vienen en botellas (bottle o bottles). Muestre los datos ordenados ascendentemente por Nombre del producto.**

```
select [ProductID],
       [ProductName],
       [QuantityPerUnit]
     from [dbo].[Products]
    where [QuantityPerUnit] like '%bottles'
 order by 2
```

Results		Messages	
	ProductID	ProductName	QuantityPerUnit
1	3	Aniseed Syrup	12 - 550 ml bottles
2	2	Chang	24 - 12 oz bottles
3	38	Côte de Blaye	12 - 75 cl bottles
4	15	Genen Shouyu	24 - 250 ml bottles
5	67	Laughing Lumberjack Lager	24 - 12 oz bottles
6	65	Louisiana Fiery Hot Pepper Sauce	32 - 8 oz bottles
7	70	Outback Lager	24 - 355 ml bottles
8	75	Rhönbräu Klosterbier	24 - 0.5 l bottles
9	34	Sasquatch Ale	24 - 12 oz bottles
10	61	Sirup d'érable	24 - 500 ml bottles
11	35	Steeleye Stout	24 - 12 oz bottles

JOWICOMPUTEMACH\SQLSERVER20... | JOWICOMPUTEMACH\admin ... | Northwind | 00:00:00 | 11 rows

## 8. Liste los productos que tienen algún tipo de salsa (Sauce).

```
select [ProductID],
       [ProductName],
       [QuantityPerUnit]
from [dbo].[Products]
where [ProductName] like '%sauce'
order by 2
```

Results		Messages	
	ProductID	ProductName	QuantityPerUnit
1	65	Louisiana Fiery Hot Pepper Sauce	32 - 8 oz bottles
2	8	Northwoods Cranberry Sauce	12 - 12 oz jars

COMPUTEMACH\SQLSERVER20... | JOWICOMPUTEMACH\admin ... | Northwind | 00:00:00 | 2 rows

## 9. Liste el código del empleado (EmployeeID) y el freight de aquellos empleados que facturaron entre 50 y 100 de flete (freight). Use para tal fin la tabla ORDERS de la BD Northwind.

```
select [EmployeeID],
       [Freight]
from [dbo].[Orders]
where [Freight] between 50 and 100
```

Results Messages		
	EmployeeID	Freight
1	4	65.83
2	4	51.30
3	3	58.17
4	4	81.91
5	4	55.09
6	2	55.28
7	8	66.29
8	6	98.03
9	3	76.07
10	8	92.69
11	3	84.81
12	4	76.56
13	1	76.83
14	8	79.70
15	1	63.79
16	1	74.16
17	7	64.50
18	1	64.86
19	4	77.92

Query e... | JOWICOMPUTEMACH\SQLSERVER20... | JOWICOMPUTEMACH\admin ... | Northwind | 00:00:00 | 173 rows

**10. Liste el nombre del producto y el precio unitario de los 10 productos más costosos.**

```
select top 10 [ProductName],
              [UnitPrice]
from [dbo].[Products]
order by 2 desc
```

Results Messages		
	ProductName	UnitPrice
1	Côte de Blaye	263.50
2	Thüringer Rostbratwurst	123.79
3	Mishi Kobe Niku	97.00
4	Sir Rodney's Marmalade	81.00
5	Camaron Tigers	62.50
6	Raclette Courdavault	55.00
7	Manjimup Dried Apples	53.00
8	Tarte au sucre	49.30
9	Ipoh Coffee	46.00
10	Rössle Sauerkraut	45.60

Query e... | JOWICOMPUTEMACH\SQLSERVER20... | JOWICOMPUTEMACH\admin ... | Northwind | 00:00:00 | 10 rows

## H. CONSIDERACIONES FINALES:

R/. Es importante recalcar, que uno de los aspectos más importantes de este laboratorio fue el manejo de cadenas de caracteres o strings al momento de realizar búsquedas dentro de una Base de Datos, utilizando el SQL Server Management Studio, pero a lo anteriormente mencionado; debo hacer algunas acotaciones:

- Es de suprema importancia observar el lenguaje en el que se encuentran almacenados los datos dentro de la Base de Datos, ya que cuando se está realizando una búsqueda por ejemplo de “Gato” y no nos aparece resultado, pero posteriormente ejecutamos el dicho query con la palabra “Cat”, es probable que nos aparezcan resultados que estamos buscando.
  - De ahí nace la importancia de los algoritmos de normalización dentro de una Base de Datos, a razón que, al momento de realizar las consultas, se conozca bien la estructura en la que están organizados y el tipado de los datos.
- Importante considerar ¿Cuáles son los datos que deseo buscar? y ¿De qué manera deseo que estos estén ordenados al momento de devolver el resultado? Ya que en muchos casos asumir que dichos datos se encontrarán, por ejemplo ordenados alfabéticamente de forma descendente, puede llevar a una incorrecta aplicación del comando Sort.

## I. BIBLIOGRAFIA:

- *A fondo SQL Server, Kalen Delaney, Serie de programación Microsoft, McGraw Hill profesional*
- [http://www.aulaclit.es/sqlserver/t\\_1\\_1.htm](http://www.aulaclit.es/sqlserver/t_1_1.htm)

## J. RÚBRICA:

- Cada uno de los problemas tienen un valor de 10 puntos, lo que hace un total de 100 puntos. Se evaluará la existencia del código y las evidencias de los resultados a través de las capturas de pantalla.