

UNERSIDAD PRIVADA DE TACNA



INGENIERIA DE SISTEMAS

TITULO:

**INFORME DE LABORATORIO No 01**

**CURSO:**

BASE DE DATOS II

**DOCENTE(ING):**

Patrick Cuadros Quiroga

Integrantes:

Jhony Mamani Limache	(2013046566)
Colque Ticona Carlos	(2013046500)
Nombre Estudiante No 3	(Codigo 03)
Nombre Estudiante No 4	(Codigo 04)
Nombre Estudiante No 5	(Codigo 05)

# Índice

# 1. Actividad No 01 – Revisión de Sintaxis

De los siguientes comandos ¿Cuál es el resultado? ¿En caso de ser error cual sería la sentencia correcta?

- SELECT last\_name, job\_id, salary AS Sal FROM employees;  
Es correcta

	last_name	job_id	Sal
1	King	AD_PRES	24000.00
2	Kochhar	AD_VP	17000.00
3	De Haan	AD_VP	17000.00
4	Hunold	IT_PROG	9000.00
5	Ernst	IT_PROG	6000.00
6	Austin	IT_PROG	4800.00
7	Pataballa	IT_PROG	4800.00
8	Lorentz	IT_PROG	4200.00
9	Greenberg	FI_MGR	12008.00
10	Faviet	FI_ACCOUNT	9000.00
11	Chen	FI_ACCOUNT	8200.00
12	Sciarra	FI_ACCOUNT	7700.00

- SELECT \* FROM job\_grades;  
Es incorrecta, la sentencia correcta sería:  
SELECT \* FROM jobs;

job_id	job_title	min_salary	max_salary
AC_ACCOUNT	Public Accountant	4200	9000
AC_MGR	Accounting Manager	8200	16000
AD_ASST	Administration Assistant	3000	6000
AD_PRES	President	20080	40000
AD_VP	Administration Vice President	15000	30000
FI_ACCOUNT	Accountant	4200	9000
FI_MGR	Finance Manager	8200	16000
HR_REP	Human Resources Representative	4000	9000
IT_PROG	Programmer	4000	10000
MARK_MGR	Marketing Manager	9000	15000

- SELECT employee\_id, last\_name sal x 12 ANNUAL SALARY FROM employees;  
Es incorrecta, la sentencia correcta sería:  
SELECT employee\_id, last\_name, salary \* 12 'ANNUAL SALARY' FROM employees;

employee_id	last_name	ANNUAL SALARY
100	King	288000.00
101	Kochhar	204000.00
102	De Haan	204000.00
103	Hunold	108000.00
104	Ernst	72000.00
105	Austin	57600.00
106	Pataballa	57600.00
107	Lorentz	50400.00
108	Greenberg	144096.00
109	Faviet	108000.00

## 2. Actividad No 02 – Reconociendo la estructura

1. Se requiere determinar la estructura de la tabla DEPARTMENTS y sus datos.

SP\_HELP 'DEPARTMENTS'

Name	Owner	Type	Created_datetime
departments	dbo	user table	2018-04-21 14:47:27.640

  

Column_name	Type	Computed	Length	Prec	Scale	Nullable	Trim	TrailingBlanks	FixedLen	NullInSource	Collation
department_id	int	no	4	10	0	no	(n/a)		(n/a)		NULL
department_name	varchar	no	30			yes	no		yes		Modern_Spanish_CI_AS
manager_id	int	no	4	10	0	yes	(n/a)		(n/a)		NULL
location_id	int	no	4	10	0	yes	(n/a)		(n/a)		NULL

  

Identity	Seed	Increment	Not For Replication
No identity column defined	NULL	NULL	NULL

  

RowGuidCol
No rowguidcol column defined

  

Data_located_on_filegroup
PRIMARY

  

index_name	index_description	index_keys
dept_id_pk	clustered, unique, primary key located on PRIMARY	department_id

  

constraint_type	constraint_name	delete_action	update_action	status_enabled	status_for_replication	constraint_keys
PRIMARY KEY (clustered)	dept_id_pk	(n/a)	(n/a)	(n/a)	(n/a)	department_id
FOREIGN KEY	dept_loc_fk	No Action	No Action	Enabled	Is_For_Replication	location_id

  

Table is referenced by foreign key
Human_Resources.dbo.employees: emp_dept_fk
Human_Resources.dbo.job_history: jhist_dept_fk

2. El departamento de Recursos Humanos requiere un reporte que muestre los campos: employee\_id, last\_name y job\_id, asicomo el campo hire\_date con el alias StartDate.

```
-- 2
SELECT emp.employee_id,
       emp.last_name,
       emp.job_id,
       emp.hire_date AS StartDate
FROM employees AS emp;
```

employee_id	last_name	job_id	StartDate
100	King	AD_PRES	2003-06-17
101	Kochhar	AD_VP	2005-09-21
102	De Haan	AD_VP	2001-01-13
103	Hunold	IT_PROG	2006-01-03
104	Ernst	IT_PROG	2007-05-21
105	Austin	IT_PROG	2005-06-25
106	Pataballa	IT_PROG	2006-02-05
107	Lorentz	IT_PROG	2007-02-07
108	Greenberg	FI_MGR	2002-08-17
109	Faviet	FI_ACCOUNT	2002-08-16

3. Finalmente el departamento de Recursos Humanos requiere un listado de todos valores del campo JOB\_ID de la tabla EMPLOYEES pero que se muestren de forma única y no repetida.

```
-- 3
SELECT DISTINCT job_id FROM employees;
```

job_id
AC_ACCOUNT
AC_MGR
AD_ASST
AD_PRES
AD_VP
FI_ACCOUNT
FI_MGR
HR_REP
IT_PROG
MK_MAN

### 3. Actividad No 03 – Consultas Básicas

1. El departamento de Recursos Humanos requiere ampliar el reporte anterior (4.2.2) para hacerlo más comprensible, por lo que se requiere que los encabezados de las columnas sean: Emp No, Empleado, Puesto y Fecha Contratación.

```
-- //////////////////////////////////////////////////
-- 1
SELECT emp.employee_id AS 'Emp N°',
       emp.last_name AS Empleado,
       emp.job_id AS Puesto,
       emp.hire_date AS 'Fecha de contratación'
FROM employees AS emp;
```

	Emp N°	Empleado	Puesto	Fecha de contratación
1	100	King	AD_PRES	2003-06-17
2	101	Kochhar	AD_VP	2005-09-21
3	102	De Haan	AD_VP	2001-01-13
4	103	Hunold	IT_PROG	2006-01-03
5	104	Ernst	IT_PROG	2007-05-21
6	105	Austin	IT_PROG	2005-06-25
7	106	Pataballa	IT_PROG	2006-02-05
8	107	Lorentz	IT_PROG	2007-02-07
9	108	Greenberg	FI_MGR	2002-08-17
10	109	Faviet	FI_ACCOUNT	2002-08-16

2. Adicionalmente el departamento de Recursos Humanos requiere un reporte más sencillo, en el que se muestre los campos: last\_name y job.id en una sola y única columna (los datos deben estar separados por una coma) que tenga como alias Empleado y Puesto.

```
-- //////////////////////////////////////////////////
-- 2
SELECT CONCAT(emp.last_name,',',emp.job_id) AS 'Empleado y Puesto'
FROM employees AS emp;
```

	Empleado y Puesto
1	King,AD_PRES
2	Kochhar,AD_VP
3	De Haan,AD_VP
4	Hunold,IT_PROG
5	Ernst,IT_PROG
6	Austin,IT_PROG
7	Pataballa,IT_PROG
8	Lorentz,IT_PROG
9	Greenberg,FI_MGR
10	Faviet,FI_ACCOUNT

3. Finalmente a modo de práctica, realizar una consulta que muestre todos los campos de la tabla EMPLOYEES, en una sola y única columna, los datos deben estar separados por una coma y la columna debe tener como encabezado Los Empleados

```
-- //////////////////////////////////////////////////
-- 3
SELECT CONCAT(emp.employee_id,',',
              emp.first_name,',',
              emp.last_name,',',
              emp.email,',',
              emp.phone_number,',',
              emp.hire_date,',',
              emp.job_id,',',
              emp.salary,',',
              emp.commission_pct,',',
              emp.manager_id,',',
              emp.department_id) AS 'Los empleados'
FROM employees AS emp;
```

	Los empleados
1	100,Steven,King,SKING,515.123.4567,2003-06-17,AD_P...
2	101,Neena,Kochhar,NKOCHHAR,515.123.4568,2005-09-...
3	102,Lex,De Haan,LDEHAAN,515.123.4569,2001-01-13,A...
4	103,Alexander,Hunold,AHUNOLD,590.423.4567,2006-01-...
5	104,Bruce,Ernst,BERNST,590.423.4568,2007-05-21,IT_P...
6	105,David,Austin,DAUSTIN,590.423.4569,2005-06-25,IT_...
7	106,Valli,Pataballa,VPATABAL,590.423.4560,2006-02-05,I...
8	107,Diana,Lorentz,DLORENTZ,590.423.5567,2007-02-07,...
9	108,Nancy,Greenberg,NGREENBE,515.124.4569,2002-08-...
10	109,Daniel,Faviet,DFAVIET,515.124.4169,2002-08-16,FI_...

## 4. Actividad No 04 – Restricción y Ordenamiento

1. Debido a problemas con el presupuesto, el departamento de Recursos Humanos requiere un reporte que muestre los apellidos (`last_name`) y salarios (`salary`) de todos los empleados que ganen más de \$ 12,000.
2. Asimismo se requiere realizar una consulta que muestre los apellidos (`last_name`) y el número de departamento (`department_id`) para los empleados que tengan numero (`employee_id`) 176.
3. El departamento de Recursos Humanos necesita determinar los mayores y menores sueldos, modificar la consulta del ítem 4.1. para mostrar el apellido y salario de cada empleado cuyo sueldo no esté en el rango de \$ 5,000 a \$ 12,000.
4. Crear un reporte que muestre los apellidos (`last_name`), puesto (`job_id`) y fecha de contratación (`hire_date`), de los empleados que apellidan 'Matos' y 'Taylor', asimismo presentar el reporte ordenado ascendentemente por fecha de contratación.
5. Mostrar los apellidos (`last_name`) y número de departamento (`departamento_id`) de todos los empleados que pertenezcan a los departamentos 20 o 50 en orden alfabético ascendente por el apellido.
6. Modificar el reporte del ítem 4.1. para mostrar los apellidos y salarios de los empleados que tengan un salario entre los \$ 5,000 a \$ 12,000 y pertenezcan a los números de departamento 20 o 50. Asimismo etiquetar las cabeceras de los resultados con los alias Empleado y Salario Mensual respectivamente.
7. El departamento de Recursos Humanos necesita un listado de apellidos (`last_name`) y fecha de contratación (`hire_date`) de todos los empleados que fueron contratados el año 1994.
8. Crear un reporte que muestre los apellidos (`last_name`) y puesto (`job_id`) de todos los empleados que no tengan un administrador (`manager`).
9. Crear un reporte para mostrar los apellidos (`last_name`), salario (`salary`) y % de comisión (`commission_pct`). Ordenar los datos por salario y comisión de manera descendente, utilizar la opción numérica de la cláusula ORDER BY.
10. El personal del departamento de Recursos Humanos desea tener mayor flexibilidad con los reportes hechos. Por ejemplo se requiere un reporte de los apellidos (`last_name`) y salarios (`salary`) de todos los empleados que tengan un salario mayor a un monto que el personal de Recursos Humanos ingresará. Probar con el valor \$ 12,000
11. El departamento de Recursos Humanos requiere extraer reporte basados en el Administrador (`manager_id`). Se requiere crear una consulta que pregunte al usuario por el Administrador (`manager_id`) y genere un reporte con los números de empleado (`employee_id`), apellidos (`last_name`), salarios (`salary`) y numero de departamento de los empleados que este Administrador tiene a su cargo. Adicionalmente también se desea tener la habilidad de ordenar este reporte en base a una determinada columna. Probar con los siguientes valores:  
Administrador (`manager_id`) = 103, ordenado por Apellido (`last_name`)  
Administrador (`manager_id`) = 201, ordenado por Salario (`salary`)  
Administrador (`manager_id`) = 124, ordenado por No de Empleado (`employee_id`)

12. Generar un listado de apellidos (`last_name`) de todos los empleados que tengan la letra 'a' en la tercera letra de su apellido.
13. Mostrar los apellidos (`last_name`) de todos los empleados que tengan tanto la letra 'a' como la letra 'e' en su apellido.
14. Mostrar los apellidos (`last_name`), puestos (`job_id`) y salario (`salary`) de todos los empleados que sean Representantes de Ventas (`SA_REP`) o Responsables de Inventario (`ST_CLERK`) y cuyos salarios no sean iguales a \$ 2,500, \$ 3,500 o \$ 7,000.
15. Modificar el reporte del ítem 4.6 y mostrar adicionalmente los datos de comisión (`commission_pct`) de todos los empleados que solamente el 20 % de comisión.

## 5. Actividad No 05 – Funciones

1. Se requiere realizar una consulta que visualice la fecha del sistema.
2. El departamento de Recursos Humanos necesita un reporte de todos los empleados que muestre el No de Empleado, Apellidos, Salario y una columna más con el cálculo del salario incrementado en 15.5 % (expresado solo en enteros) esta columna debe etiquetarse Nuevo Salario
3. Modificar la consulta anterior y adicionar una columna que muestre el resultado de la resta entre el antiguo salario y el nuevo salario. Etiquetar esta columna como Incremento.
4. Crear un reporte que muestre los Apellidos (con la primera letra en Mayúsculas y las demás en Minúsculas) y la longitud de los apellidos (colocar alias Longitud), para todos aquellos empleados quienes sus apellidos empiecen con las letras ‘J’, ‘A’ y ‘M’. Ordenar los resultados por la columna Apellido.
5. Modificar la consulta anterior a fin de que consulte primero al usuario con que letra empieza el apellido a buscar. Considerar que no importa si la letra esta mayúscula o minúscula de igual manera debe mostrar los resultados.
6. El departamento de Recursos Humanos la duración o tiempo de permanencia de cada empleado, mostrar el Apellido y el calculo del número de meses entre la fecha de hoy y la fecha en que fue contratado el empleado, Etiquetar la columna como Meses Trabajados, ordenar los resultados por el resultado de los números de meses, Redondear el número de meses al entero más cercano.
7. Crear una consulta que devuelva los Apellidos y Salarios de todos los empleados, Formatear la columna salario para que muestre 15 caracteres, completar con el símbolo ‘\$’ los espacios previos al valor de la columna salario, ejemplo: \$\$\$\$\$\$\$\$\$10000. Etiquetar esta columna como Salario.
8. Crear una consulta que muestre en una única columna los primeros 8 caracteres del apellido de los empleados e indique sus salarios representados por asteriscos (\*), cada asterisco representa el valor 1000. Ordenar el listado por el salario de los empleados. Asimismo Etiquetar la columna como ‘Empleados y sus Salarios’.
9. Finalmente crear una consulta que muestre los Apellidos de los empleados y el No de Semanas Empleado hasta la actualidad para todos los empleados del departamento No 90, truncar el número de semanas a sin decimales. Ordenar el resultado por el No de Semanas y etiquetar la columna como tenencia.



## 6. Actividad No 06 – Funciones de Conversión

1. Crear un reporte que muestre lo siguiente por cada empleado.  
(Apellido del empleado) gana (Salario) pero quisiera (3 veces Salario).  
Etiquetar la columna como Sueldos Soñados.
2. Realizar una consulta que muestre el Apellido del empleado, fecha de contratación y la Fecha de Revisión del Salario, la cual es el primer Lunes después de cada seis meses de servicio, etiquetar la columna como Revisión, asimismo el formato de esta fecha debe ser similar al siguiente:

Lunes, el veintiuno de julio, 2003

3. Mostrar un reporte que tenga los Apellidos, Fecha de Contratación y el Día de Inicio de cada empleado (Lunes, Martes, etc. . . ), etiquetar la última columna como Día. Ordenar los resultados por el Día de Inicio empezando por Lunes.
4. Crear un listado que muestre los Apellidos de los empleados y sus Montos de Comisión, en caso no tenga comisión deberá mostrar el texto ‘Sin Comisión’, etiquetar esta ultima columna como Comisión.
5. Utilizando la función DECODE, crear un reporte que muestre los apellidos, los puestos y los grados de los empleados basados en sus puestos, utilizando la siguiente información:

Puesto	Grado
AD_PRES	A
ST_MAN	B
IT_PROG	C
SA_REP	D
ST_CLERK	E
Ninguno de los Anteriores	0

6. Rescribir la consulta anterior utilizando la función CASE.

## 7. Actividad No 07 – Funciones de Agrupación

1. El departamento de Recursos Humanos requiere un reporte que muestre el máximo, el mínimo, la suma y el promedio de los salarios de todos los empleados, Etiquetar esta columnas como Máximo, Mínimo, Suma y Promedio respectivamente, Redondear estos valores a enteros sin decimales.
2. Modificar la consulta anterior para mostrar el máximo, mínimo, suma y promedio de los salarios por cada Puesto de trabajo.
3. Realizar un reporte que muestre la cantidad de empleados por Puesto de trabajo. Con la opción de que el usuario pueda ingresar todos los puestos o uno solo.
4. Determinar el número de Administradores o Supervisores utilizar la columna `manager_id` para esto. Etiquetar la columna como No de Administradores
5. Encontrar la diferencia entre el máximo y mínimo salario de los empleados. Etiquetar la columna como Diferencia
6. Crear un reporte que muestre los No de Administradores (`manager_id`) y el salario de su empleado peor pagado. Excluir a los empleados cuyo Administrador no se conozca. Excluir asimismo cualquier grupo cuyo salario mínimo sea \$6000 o menos. Ordenar los resultados por el mínimo salario en forma descendente.
7. Crear una consulta que muestre el número total de empleados, así como el número total de empleados contratados en los años 1995, 1996, 1997 y 1998, etiquetar las columnas apropiadamente.
8. Crear una consulta matriz que muestre el puesto, el salario por cada puesto basado en el No de Departamento del empleado y el total del salario para cada puesto para los departamento 20, 50, 80 y 90, colocar un nombre apropiado a cada columna.

## 8. Actividad No 08 – Enlaces

1. El departamento de Recursos Humanos requiere un reporte que muestre las direcciones de todos los departamentos. Utilizar las tablas LOCATIONS y COUNTRIES. Mostrar el ID de la Ubicación (location\_id), dirección (street\_address), ciudad (city), estado o provincia (state\_province) y país (country\_name).

```
select l.location_id , l.street_address , l.city , l.state_province , c.country_name
from locations as l
join countries as c
on l.country_id = c.country_id
```

location_id	street_address	city	state_province	country_name
1	1237 Via Cole di Rie	Roma	NULL	Italy
2	1100 33091 Calle della Testa	Venice	NULL	Italy
3	2017 Shinsyoku-cho	Tokyo	Tokyo Prefecture	Japan
4	9450 Kameya-cho	Hiroshima	NULL	Japan
5	2014 Jabbarovskiy Rd	Southern	Texas	United States of America
6	2011 Interoce Blvd	South San Francisco	California	United States of America
7	2007 Zappia St	South Brunswick	New Jersey	United States of America
8	2004 Charade Rd	Seattle	Washington	United States of America
9	147 Spadina Ave	Toronto	Ontario	Canada
10	6502 Blaisdell St	Whitehorse	Yukon	Canada
11	40-5-12 Langjiagang	Beijing	NULL	China
12	1296 Wazirpur Rd	Bombay	Maharashtra	India
13	12-98 Victoria Street	Sydney	New South Wales	Australia
14	198 Clementi North	Singapore	NULL	Singapore
15	5204 Arthur St	London	NULL	United Kingdom
16	Magdalen Centre, The Oxford Science Park	Oxford	Oxford	United Kingdom
17	5702 Chester Road	Sheffield	Manchester	United Kingdom
18	Schwanthalerstr. 7021	Munich	Bavaria	Germany
19	Rua Frei Caneca 1360	Sao Paulo	Sao Paulo	Brazil
20	20 Rue des Corps Saints	Geneve	Geneve	Switzerland
21	Munsterstrasse 321	Bern	BE	Switzerland
22	Peter Bleuhofstrad 437	Utrecht	Utrecht	Netherlands
23	Marconi Escobedo 5951	Mexico City	Distrito Federal	Mexico
24	2014 america city	brasil	brasil	america

2. El departamento de Recursos Humanos necesita un reporte de todos empleados, que muestres los apellidos de empleado (last\_name), el No de departamento (department\_id) y el nombre del departamento (deperment\_date) al cual pertenece.

	last_name	department_id	department_name
1	Grant	NULL	NULL
2	Higgins	110	Accounting
3	Gietz	110	Accounting
4	Whalen	10	Administration
5	King	90	Executive
6	Kochhar	90	Executive
7	De Haan	90	Executive
8	Greenberg	100	Finance
9	Faviet	100	Finance
10	Chen	100	Finance
11	Sciarra	100	Finance
12	Uman	100	Finance
13	Popp	100	Finance
14	Mavris	40	Human Resources
15	Hunold	60	IT
16	Ernst	60	IT
17	Austin	60	IT
18	Pataballa	60	IT
19	Lorentz	60	IT
20	Hartstein	20	Marketing
21	Fay	20	Marketing
22	Baer	70	Public Relations
23	Raphaely	30	Purchasing
24	Khoo	30	Purchasing
25	Baida	30	Purchasing
26	Tobias	30	Purchasing
27	Himuro	30	Purchasing
28	Colmenares	30	Purchasing
29	Russell	80	Sales
30	Partners	80	Sales
31	Errazuriz	80	Sales
32	Cambrault	80	Sales
33	Zlotkey	80	Sales
34	Tucker	80	Sales
35	Bernstein	80	Sales
36	Hall	80	Sales
37	Olsen	80	Sales
38	Cambrault	80	Sales
39	Tivvauilt	80	Sales

3. El departamento de Recursos Humanos necesita un reporte de los empleados de la ciudad de Toronto. Mostrar los Apellidos, Puesto, No de Departamento y Nombre de Departamento de todos los empleados que trabajan en Toronto.

	last_name	department_id	job_title	department_name	city
1	Hartstein	20	Marketing Manager	Marketing	Toronto
2	Fay	20	Marketing Representative	Marketing	Toronto

4. Crear un reporte que muestre los Apellidos y No de Identificación de los empleados, asimismo también debe mostrarse el Apellido y No de Identificación de su Administrador.

	ID_Empleado	Empleado	ID_Manager	Manager
1	101	Kochhar	100	King
2	102	De Haan	100	King
3	114	Raphaely	100	King
4	120	Weiss	100	King
5	121	Fripp	100	King
6	122	Kaufling	100	King
7	123	Vollman	100	King
8	124	Mourgos	100	King
9	145	Russell	100	King
10	146	Partners	100	King
11	147	Errazuriz	100	King
12	148	Cambrault	100	King
13	149	Zlotkey	100	King
14	201	Hartstein	100	King
15	200	Whalen	101	Kochhar
16	203	Mavris	101	Kochhar
17	204	Baer	101	Kochhar
18	205	Higgins	101	Kochhar
19	108	Greenberg	101	Kochhar
20	103	Hunold	102	De Haan
21	104	Emst	103	Hunold
22	105	Austin	103	Hunold
23	106	Pataballa	103	Hunold
24	107	Lorentz	103	Hunold
25	109	Faviet	108	Greenb...
26	110	Chen	108	Greenb...
27	111	Sciarra	108	Greenb...
28	112	Uman	108	Greenb...
29	113	Popp	108	Greenb...
30	115	Khoo	114	Rapha...
31	116	Baida	114	Rapha...
32	117	Tobias	114	Rapha...
33	118	Himuro	114	Rapha...
34	119	Colmena...	114	Rapha...
35	125	Nayer	120	Weiss
36	126	Mikkilineni	120	Weiss

5. Modificar la consulta anterior para que incluya también a los empleados quienes no tienen Administrador asignado.

	ID_Empleado	Empleado	ID_Manager	Manager
1	100	King	NULL	NULL
2	101	Kochhar	100	King
3	102	De Haan	100	King
4	114	Raphaely	100	King
5	120	Weiss	100	King
6	121	Fripp	100	King
7	122	Kaufing	100	King
8	123	Vollman	100	King
9	124	Mourgos	100	King
10	145	Russell	100	King
11	146	Partners	100	King
12	147	Errazuriz	100	King
13	148	Cambraut	100	King
14	149	Zlotkey	100	King
15	201	Hartstein	100	King
16	200	Whalen	101	Kochhar
17	203	Mavris	101	Kochhar
18	204	Baer	101	Kochhar
19	205	Higgins	101	Kochhar
20	108	Green...	101	Kochhar
21	103	Hunold	102	De Haan
22	104	Ernst	103	Hunold
23	105	Austin	103	Hunold
24	106	Pataballa	103	Hunold
25	107	Lorentz	103	Hunold
26	109	Faviet	108	Green...
27	110	Chen	108	Green...
28	111	Sciarra	108	Green...
29	112	Urman	108	Green...
30	113	Popp	108	Green...
31	115	Khoo	114	Rapha...
32	116	Baida	114	Rapha...
33	117	Tobias	114	Rapha...
34	118	Himuro	114	Rapha...
35	119	Colmen...	114	Rapha...
36	125	Nayer	120	Weiss
37	126	Mikkil...	120	Weiss
38	127	Landry	120	Weiss
39	128	Markle	120	Weiss

6. Crear un reporte que muestre los No de Departamento y Apellidos de todos los empleados, asimismo adicionar una columna con los Apellidos de todos empleados que trabajan en el mismo departamento. Etiquetar esta columna como Colega.

```

-- EJERCICIO 6
SELECT e.department_id "DEPARTAMENTO", e.last_name "EMPLEADO",
       f.last_name "COLEGA"
FROM employees e
JOIN employees f
ON (e.department_id = f.department_id)

```

DEPARTAMENTO	EMPLEADO	COLEGA
10	Raphaely	Colmen...
10	Rhoo	Raphaely
10	Rhoo	Baia
10	Rhoo	Tobias
10	Rhoo	Hunold
10	Rhoo	Colmen...
10	Baia	Raphaely
10	Baia	Rhoo
10	Baia	Tobias
10	Baia	Hunold
10	Baia	Colmen...
10	Tobias	Raphaely
10	Tobias	Rhoo
10	Tobias	Baia
10	Tobias	Hunold
10	Tobias	Colmen...
10	Hunold	Raphaely
10	Hunold	Rhoo

7. El departamento de Recursos Humanos requiere un reporte de todo el personal que fue contratado después del empleado apellidado 'Davies'. Crear un reporte que muestre el apellidos y fecha de contratación de todo los empleados contratado después de 'Davies'.
8. El departamento de Recursos Humanos requiere de un reporte que el apellido del empleado, fecha de contratación del empleado, apellido del administrador, fecha de contratación del administrador. Para todos aquellos empleados que fueron contratados antes que sus Administradores.

## 9. Actividad No 09 – SubConsultas

1. El departamento de Recursos Humanos requiere una consulta que pregunte al usuario por el Apellido del empleado, Luego la consulta deberá mostrar los Apellidos y Fecha de Contratación de todos los empleados del mismo departamento excluyendo o con excepción del empleado el cual ha sido proporcionado su apellido reporte que muestre las direcciones de todos los departamentos.

```
-- para el desarrollo se cambio apellido por id del empleado
DECLARE @apellido INT;
DECLARE @empid INT;

-- leyendo id de empleado
SET @empid = 100;
-- obteniendo id de departamento de empleado
SET @deptid = (SELECT emp.department_id
FROM employees AS emp
WHERE emp.employee_id=@empid);

-- todos los empleados del mismo departamento excluyendo al empleado ingresado anteriormente
SELECT emp.employee_id,
emp.last_name,
emp.hire_date,
emp.department_id
FROM employees AS emp
WHERE emp.department_id=@deptid
AND emp.employee_id <> @empid;
```

	employee_id	last_name	hire_date	department_id
1	100	Greenberg	2002-08-17	100
2	109	Faviet	2002-08-16	100
3	111	Sciarra	2005-05-30	100
4	112	Uman	2005-03-07	100
5	113	Popp	2007-12-07	100

2. Crear un reporte que muestre el No del Empleado, Apellidos y Salarios de todos los empleados que tienen un salario superior al promedio de salarios de todos los empleados. Ordenar los resultados por el Salario de forma ascendente.

```
-- Se considera "N° de empleado" como "id de empleado"
-- Obteniendo promedio de salario
DECLARE @prom DECIMAL(8,2); --Variable promedio
SET @prom = (SELECT AVG(salary) FROM employees);

-- Todos los empleados con sueldo superior al promedio
SELECT emp.employee_id,
emp.last_name,
emp.salary
FROM employees AS emp
WHERE emp.salary > @prom;
```

	employee_id	last_name	salary
1	100	King	24000.00
2	101	Kochhar	17000.00
3	102	De Haan	17000.00
4	103	Hunold	9000.00
5	108	Greenberg	12008.00
6	109	Faviet	9000.00
7	110	Chen	8200.00
8	111	Sciarra	7700.00
9	112	Uman	7800.00
10	113	Popp	6900.00

3. Realizar un reporte que muestre el No de Empleado y Apellidos de todos los empleados quienes trabajan

```
-- Se considera "N° de empleado" como "id de empleado"
-- Obtener los id de departamentos de los empleados que contienen "u" en su apellido
SELECT DISTINCT department_id
FROM employees
WHERE last_name LIKE '%u%';

-- Obtener todos los empleados que laboran en alguno de los departamentos hallados anteriormente
SELECT emp.employee_id,
emp.last_name,
emp.department_id
FROM employees AS emp
JOIN (SELECT DISTINCT department_id
FROM employees
WHERE last_name LIKE '%u%') AS deptid
ON emp.department_id=deptid.department_id;
```

	employee_id	last_name	department_id
1	114	Rafaely	30
2	115	Whean	30
3	116	Bella	30
4	117	Talson	30
5	118	Herman	30
6	119	Colmenares	30
7	120	Taylor	50
8	121	Recur	50
9	122	Sullivan	50
10	123	Geech	50

en el departamento de cualquier empleado que su apellido contenga la letra 'u'.

4. El departamento de Recursos Humanos requiere un reporte que muestre los Apellidos, No de Departamento y Puestos de los empleados cuya locación de departamento es 1700.

```
-- 4
SELECT emp.last_name,
       emp.department_id,
       dep.location_id
FROM employees as emp
JOIN departments as dep
ON emp.department_id=dep.department_id
WHERE dep.location_id=1700;
```

	last_name	department_id	location_id
1	King	90	1700
2	Kochhar	90	1700
3	De Haan	90	1700
4	Greenberg	100	1700
5	Faviet	100	1700
6	Chen	100	1700
7	Sciarra	100	1700
8	Uman	100	1700
9	Popp	100	1700
10	Raphaely	30	1700

5. Modificar la consulta anterior de forma que el usuario pueda introducir el No de locación.

```
-- 5
DECLARE @locid INT;
SET @locid = 1700;
SELECT emp.last_name,
       emp.department_id,
       dep.location_id
FROM employees as emp
JOIN departments as dep
ON emp.department_id=dep.department_id
WHERE dep.location_id=@locid;
```

	last_name	department_id	location_id
1	King	90	1700
2	Kochhar	90	1700
3	De Haan	90	1700
4	Greenberg	100	1700
5	Faviet	100	1700
6	Chen	100	1700
7	Sciarra	100	1700
8	Uman	100	1700
9	Popp	100	1700
10	Raphaely	30	1700

6. Crear un reporte para el departamento de Recursos Humanos que muestre los Apellidos y Salarios de todos los empleados cuyo Administrador apellide 'King'.

```
-- 6
-- conseguir id de empleado que lleven como apellido KING
SELECT employee_id,
       last_name
FROM employees
WHERE last_name='KING';
-- conseguir id de departamentos que coincidan en manager_id con employee_id
SELECT dep.department_id
FROM departments AS dep
JOIN (SELECT employee_id,
            last_name
      FROM employees
      WHERE last_name='KING') AS marking
ON dep.manager_id=marking.employee_id;
-- FINALMENTE, apellidos y salarios de empleados que tengan como id de
-- departamento el/los id de departamentos hallados anteriormente
SELECT emp.last_name,
       emp.salary
FROM employees AS emp
JOIN (SELECT dep.department_id
      FROM departments AS dep
      JOIN (SELECT employee_id,
                  last_name
            FROM employees
            WHERE last_name='KING') AS marking
      ON dep.manager_id=marking.employee_id) AS depking
ON emp.department_id=depking.department_id;
```

	last_name	salary
1	King	24000.00
2	Kochhar	17000.00
3	De Haan	17000.00

7. Crear un reporte para el departamento de Recursos Humanos que muestre el No de Departamento, Apellidos, Puestos de todos los empleados en el departamento 'Executive'.



```
-- 7
select * from employees where department_id=90;
select * from jobs;
select * from departments where department_name='executive';

-- consiguiendo empleados con nombre de puesto
SELECT emp.department_id,
       emp.last_name,
       jobs.job_title
FROM employees AS emp
JOIN jobs
ON emp.job_id=jobs.job_id;

-- FINALMENTE, empleados con departamento Executive
SELECT empnomjob.department_id,
       empnomjob.last_name,
       empnomjob.job_title
FROM departments
JOIN (SELECT emp.department_id,
            emp.last_name,
            jobs.job_title
      FROM employees AS emp
      JOIN jobs
      ON emp.job_id=jobs.job_id) AS empnomjob
ON empnomjob.department_id=departments.department_id
WHERE department_name='executive';
```

100 %

	department_id	last_name	job_title
1	90	King	President
2	90	Kochhar	Administration Vice President
3	90	De Haan	Administration Vice President

8. Modificar la consulta del ítem 4.3 para que adicionalmente se muestro solo a los empleados que tengan un salario mayor al promedio de todos los salarios de los empleados.

## 10. Actividad No 10 – Conjuntos

1. El departamento de Recursos Humanos requiere un reporte de todos los departamentos que no contengan un empleado con el puesto ‘ST\_CLERK’. Utilizar el operador MINUS o EXCEPT para esta solicitud.
2. El departamento de Recursos Humanos requiere adicionalmente una lista de todos los países que no tengan un departamento de la empresa localizado en ellos, mostrar el código del país y el nombre. Utilizar el operador MINUS o EXCEPT para realizar esta operación.
3. Se necesita una lista de puestos de los departamentos 10, 50 y 20, en ese orden, mostrar el código del puesto y código del departamento. Utilizar el operador UNION ALL.
4. Crear un reporte que muestre que liste los códigos de los empleados y los puestos de todos aquellos empleados que tienen el mismo puesto que en el momento en el que fueron contratados por la empresa, cambiaron de puestos y luego volvieron al puesto anterior. Utilizar el operador INTERSECT.
5. El departamento de Recursos Humanos requiere un reporte que muestre lo siguiente:
  - Apellidos y códigos de departamentos de todos los registros de la tabla empleados sin importar si pertenecen a uno o ningún departamento.
  - Código de departamentos y nombres de departamentos de la tabla DEPARTAMENTOS inclusive si no existiese ningún empleado en ese departamento

Ambos requerimientos se deben mostrar en un mismo resultado. Utilizar el operador UNION ALL.