



Bases de datos

Actividades del tema 5.

YEHOR BURLACHENKO



Actividades.

Ej 1. Las funciones de grupo funcionan en varias filas para producir un resultado por grupo.

Respuesta: Verdadero/Falso

Ej 2. Las funciones de grupo incluyen valores nulos en los cálculos.

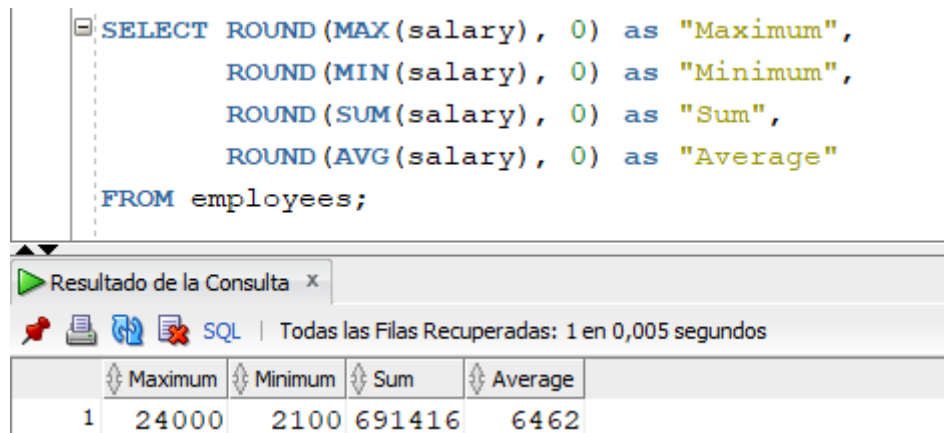
Respuesta: Verdadero/Falso

Ej 3. La cláusula WHERE restringe las filas antes de incluirlas en un cálculo de grupo.

Respuesta: Verdadero/Falso

Ej 4. Encontrar el valor más alto, el valor más bajo, la suma y la media del salario de todos los empleados. Etiquete las columnas como Maximum, Minimum, Sum y Average, respectivamente. Redondee los resultados al número entero más cercano. Guarde la sentencia SQL como lab_05_04.sql. Ejecute la consulta.

Respuesta:



```
SELECT ROUND(MAX(salary), 0) as "Maximum",  
        ROUND(MIN(salary), 0) as "Minimum",  
        ROUND(SUM(salary), 0) as "Sum",  
        ROUND(AVG(salary), 0) as "Average"  
FROM employees;
```

	Maximum	Minimum	Sum	Average
1	24000	2100	691416	6462

Ej 5. Modifique la consulta del archivo lab_05_04.sql para mostrar el valor mínimo, el valor máximo, la suma y la media del salario de cada tipo de cargo. Vuelva a guardar lab_05_05.sql y ejecute la sentencia en el archivo.

Respuesta:

<pre> SELECT ROUND(MAX(salary), 0) as "Maximum", ROUND(MIN(salary), 0) as "Minimum", ROUND(SUM(salary), 0) as "Sum", ROUND(AVG(salary), 0) as "Average" FROM employees; </pre>					
<div> <div>Resultado de la Consulta x</div> <div> <div> <div></div> <div></div> <div></div> <div></div> </div> <div>SQL</div> <div>Todas las Filas Recuperadas: 19 en 0,008 segundos</div> </div> </div>					
JOB_ID	Maximum	Minimum	Sum	Average	
1 IT_PROG	9000	4200	28800	5760	
2 AC_MGR	12008	12008	12008	12008	
3 AC_ACCOUNT	8300	8300	8300	8300	
4 ST_MAN	8200	5800	36400	7280	
5 PU_MAN	11000	11000	11000	11000	
6 AD_ASST	4400	4400	4400	4400	
7 AD_VP	17000	17000	34000	17000	
8 SH_CLERK	4200	2500	64300	3215	
9 FI_ACCOUNT	9000	6900	39600	7920	
10 FI_MGR	12008	12008	12008	12008	
11 PU_CLERK	3100	2500	13900	2780	
12 SA_MAN	14000	10500	61000	12200	
13 MK_MAN	13000	13000	13000	13000	
14 PR_REP	10000	10000	10000	10000	
15 AD_PRES	24000	24000	24000	24000	
16 SA_REP	11500	6100	250500	8350	
17 MK_REP	6000	6000	6000	6000	
18 ST_CLERK	3600	2100	55700	2785	
19 HR_REP	6500	6500	6500	6500	

Ej 6. Escriba una consulta para mostrar el número de personas con el mismo cargo. Generalice la consulta para preguntar al usuario del departamento de recursos humanos cuál es su puesto. Guarde el script en un archivo denominado lab_05_06.sql. Ejecute la consulta. Introduzca IT_PROG cuando se le solicite.

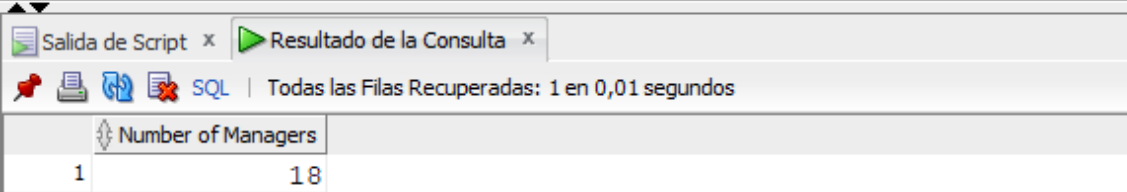
<pre> SELECT job_id, COUNT(*) FROM employees WHERE job_id = '&job_id' GROUP BY job_id; </pre>		
<div> <div>Resultado de la Consulta x</div> <div> <div> <div></div> <div></div> <div></div> <div></div> </div> <div>SQL</div> <div>Todas las Filas Recuperadas: 1 en 1</div> </div> </div>		
JOB_ID	COUNT(*)	
1 IT_PROG	5	

Respuesta:

Ej 7. Determine el número de gestores sin enumerarlos en una lista. Etiquete la columna como Number of Managers. Indicación: utilice la columna MANAGER_ID para determinar el número de gestores.

Respuesta:

```
SELECT COUNT(DISTINCT manager_id) as "Number of Managers"
FROM employees;
```

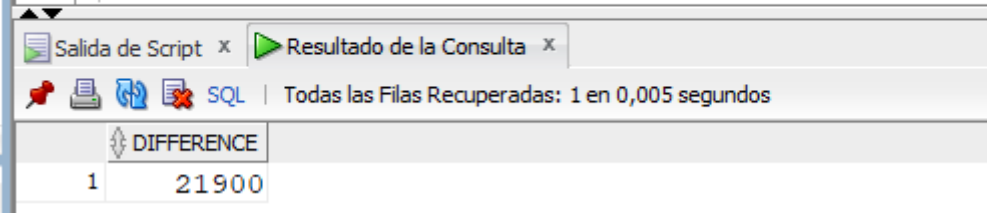


	Number of Managers
1	18

Ej 8. Busque la diferencia entre los salarios más altos y más bajos. Etiquete la columna como DIFFERENCE.

Respuesta:

```
SELECT MAX(salary) - MIN(salary) as difference
FROM employees;
```

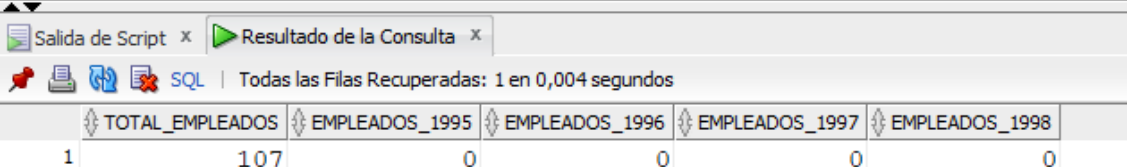


	DIFFERENCE
1	21900

Ej 9. Cree un informe para mostrar el numero de gestor y el salario del empleado con mejor sueldo de ese gestor. Excluya a cualquier trabajador del que desconozca su gestor. Excluya cualquier grupo en el que salario minimo sea 6000 dolares o menos. Ordene la salida en orden descendente de salarios.

Respuesta:

```
SELECT manager_id, MIN(salary)
FROM employees
GROUP BY manager_id
HAVING MIN(salary) > 6000
ORDER BY MIN(salary) DESC;
```



	TOTAL_EMPLEADOS	EMPLEADOS_1995	EMPLEADOS_1996	EMPLEADOS_1997	EMPLEADOS_1998
1	107	0	0	0	0

Ej 10. Cree una consulta para mostrar el numero total de empleados y, de ese total, el numero de empleados contratados en 1995,1996,1997 y 1998. Cree las cabeceras de columna adecuadas.

Respuesta:

```
SELECT
    COUNT(*) AS total_empleados,
    SUM(CASE WHEN EXTRACT(YEAR FROM hire_date) = 1995 THEN 1 ELSE 0 END) AS empleados_1995,
    SUM(CASE WHEN EXTRACT(YEAR FROM hire_date) = 1996 THEN 1 ELSE 0 END) AS empleados_1996,
    SUM(CASE WHEN EXTRACT(YEAR FROM hire_date) = 1997 THEN 1 ELSE 0 END) AS empleados_1997,
    SUM(CASE WHEN EXTRACT(YEAR FROM hire_date) = 1998 THEN 1 ELSE 0 END) AS empleados_1998
FROM employees;
```

Salida de Script x Resultado de la Consulta x

Todas las Filas Recuperadas: 1 en 0,005 segundos

	TOTAL_EMPLEADOS	EMPLEADOS_1995	EMPLEADOS_1996	EMPLEADOS_1997	EMPLEADOS_1998
1	107	0	0	0	0

Ej 11. Cree una consulta de matriz para mostrar el cargo, el salario de ese cargo según el numero de departamento y el salario total cargo para los departamentos 20, 50, 80 y 90, proporcionando a cada columna una cabecera adecuada.

Respuesta:

```
SELECT
    job_id AS "Job",
    SUM(CASE WHEN department_id = 20 THEN salary ELSE NULL END) AS "Dept 20",
    SUM(CASE WHEN department_id = 50 THEN salary ELSE NULL END) AS "Dept 50",
    SUM(CASE WHEN department_id = 80 THEN salary ELSE NULL END) AS "Dept 80",
    SUM(CASE WHEN department_id = 90 THEN salary ELSE NULL END) AS "Dept 90",
    SUM(salary) AS "Total"
FROM employees
GROUP BY job_id;
```

Salida de Script x Resultado de la Consulta x

Todas las Filas Recuperadas: 19 en 0,024 segundos

	Job	Dept 20	Dept 50	Dept 80	Dept 90	Total
1	IT_PROG	(null)	(null)	(null)	(null)	28800
2	AC_MGR	(null)	(null)	(null)	(null)	12008
3	AC_ACCOUNT	(null)	(null)	(null)	(null)	8300
4	ST_MAN	(null)	36400	(null)	(null)	36400
5	PU_MAN	(null)	(null)	(null)	(null)	11000
6	AD_ASST	(null)	(null)	(null)	(null)	4400
7	AD_VP	(null)	(null)	(null)	34000	34000
8	SH_CLERK	(null)	64300	(null)	(null)	64300
9	FI_ACCOUNT	(null)	(null)	(null)	(null)	39600
10	FI_MGR	(null)	(null)	(null)	(null)	12008
11	PU_CLERK	(null)	(null)	(null)	(null)	13900
12	SA_MAN	(null)	(null)	61000	(null)	61000
13	MK_MAN	13000	(null)	(null)	(null)	13000
14	PR_REP	(null)	(null)	(null)	(null)	10000
15	AD_PRES	(null)	(null)	(null)	24000	24000
16	SA_REP	(null)	(null)	243500	(null)	250500
17	MK_REP	6000	(null)	(null)	(null)	6000