



INSTITUTO TECNOLÓGICO SUPERIOR DE JEREZ

27 de septiembre del 2019

Jerez, Zac

Ingeniería en sistemas computacionales

Semestre: 5

Unidad 2: lenguaje de manipulación de datos

Alumna: Leticia carrera venegas

Correo: Letycv25@gmail.com

Num control: S17070155

Actividad: Consultas con agregación

Taller de bases de datos

Docente: ISC Salvador Acevedo Sandoval



1. Mostrar el salario del empleado que gana mas

```
mysql> SELECT MAX(Salary) FROM Staff;
+-----+
| MAX(Salary) |
+-----+
|      30000 |
+-----+
1 row in set (0.00 sec)
```

```
db2 => SELECT MAX(Salary) FROM Staff
1
-----
      30000

1 registro(s) seleccionado(s).
```

2. Mostrar el salario del empleado que gana menos

```
mysql> SELECT MIN(Salary) FROM Staff;
+-----+
| MIN(Salary) |
+-----+
|       9000 |
+-----+
1 row in set (0.00 sec)
```

```
db2 => SELECT MIN(Salary) FROM Staff
1
-----
      9000

1 registro(s) seleccionado(s).
db2 =>
```

3. Muestre cual es el promedio del salario que perciben los trabajadores

```
mysql> SELECT AVG(Salary) as Promedio_Salario FROM Staff;
+-----+
| Promedio_Salario |
+-----+
|      17000.0000 |
+-----+
1 row in set (0.00 sec)
```

```
db2 => SELECT AVG(Salary) as Promedio_Salario FROM Staff
PROMEDIO_SALARIO
-----
      17000

  1 registro(s) seleccionado(s).
db2 => █
```

4. Crear una consulta que muestre la cantidad que gasta la empresa en salarios

```
mysql> SELECT AVG(Salary) as Promedio_Salario FROM Staff;
+-----+
| Promedio_Salario |
+-----+
|      17000.0000 |
+-----+
1 row in set (0.00 sec)
```

```
db2 => SELECT AVG(Salary) as Promedio_Salario FROM Staff
PROMEDIO_SALARIO
-----
      17000

  1 registro(s) seleccionado(s).
db2 => █
```

5. Crear una consulta que muestre la cantidad que gasta la empresa en salarios quincenales (supniendo que el dato almancenado es mensual)

```
mysql> SELECT SUM(Salary/2) AS Suma_Salarios_Quincenales FROM Staff;
+-----+
| Suma_Salarios_Quincenales |
+-----+
|          51000.0000 |
+-----+
1 row in set (0.00 sec)
```

```
db2 => SELECT SUM(Salary/2) AS Suma_Salarios_Quincenales FROM Staff
SUMA_SALARIOS_QUINCENALES
-----
                51000

    1 registro(s) seleccionado(s).
db2 =>
```

6. Mostrar cuantas propiedades en renta existen

```
mysql> SELECT COUNT(PropertyNo) AS Propiedades_EN_Renta FROM PropertyForRent;
+-----+
| Propiedades_EN_Renta |
+-----+
|          6 |
+-----+
1 row in set (0.00 sec)
```

```
db2 => SELECT COUNT(PropertyNo) AS Propiedades_EN_Renta FROM PropertyForRent
PROPIEDADES_EN_RENTA
-----
                6

    1 registro(s) seleccionado(s).
```

7. Mostrar cuantas visitas a las propiedades se han hecho

```
mysql> SELECT COUNT(FK_clientNo) AS Numero_Visitas FROM Viewing;
+-----+
| Numero_Visitas |
+-----+
|                5 |
+-----+
1 row in set (0.00 sec)
```

```
db2 => SELECT COUNT(ClientNo) AS Numero_Visitas FROM viewing
NUMERO_VISITAS
-----
                5

1 registro(s) seleccionado(s).
db2 => █
```

8. Mostar la cantidad de clientes que atiende la empresa

```
mysql> SELECT COUNT(ClientNo) AS Cantidad_Clientes FROM Client;
+-----+
| Cantidad_Clientes |
+-----+
|                  4 |
+-----+
1 row in set (0.07 sec)

mysql> █
```

```
db2 => SELECT COUNT(ClientNo) AS Cantidad_Clientes FROM Client
CANTIDAD_CLIENTES
-----
                4

1 registro(s) seleccionado(s).
```

9. Mostrar cuantas propiedades en renta que cuesten más de 350 euros existen

```
mysql> SELECT COUNT(PropertyNo) AS Propiedades_RentaMayor_350 FROM PropertyForRent WHERE rent > 350;
+-----+
| Propiedades_RentaMayor_350 |
+-----+
| 5 |
+-----+
1 row in set (0.00 sec)
```

```
PROPIEDADES_RENTASMAYORES_350
-----
5
1 registro(s) seleccionado(s).
db2 =>
```

10. Mostrar cuantas visitas a la propiedad CR56 se han hecho

```
mysql> SELECT COUNT(FK_clientNo) AS Visitas_CR56 FROM viewing WHERE FK_clientNo = 'CR56';
+-----+
| Visitas_CR56 |
+-----+
| 3 |
+-----+
1 row in set (0.00 sec)
```

```
db2 => SELECT COUNT(ClientNo) AS Visitas_CR56 FROM viewing WHERE ClientNo = 'CR56'
VISITAS_CR56
-----
3
1 registro(s) seleccionado(s).
```

11. Mostar la cantidad de clientes que puedan pagar una renta mayor a 500 euros atiende la empresa

```
mysql> SELECT COUNT(ClientNo) AS Clientes_RentaMayor_500 FROM Client WHERE maxRent > 500;
+-----+
| Clientes_RentaMayor_500 |
+-----+
| 2 |
+-----+
1 row in set (0.00 sec)
```

```
db2 => SELECT COUNT(ClientNo) AS Clientes_RentaMayor_500 FROM Client WHERE maxRent > 500
CLIENTES_RENTAMAYOR_500
-----
2
1 registro(s) seleccionado(s).
db2 =>
```

12. Calcular el promedio de la renta que pueden pagar los clientes

```
mysql> SELECT AVG(maxRent) AS Promedio_Maximo_Rentas FROM Client;
+-----+
| Promedio_Maximo_Rentas |
+-----+
|          531.2500 |
+-----+
1 row in set (0.00 sec)
```

```
db2 => SELECT AVG(maxRent) AS Promedio_Maximo_Rentas FROM Client
PROMEDIO_MAXIMO_RENTAS
-----
          531

1 registro(s) seleccionado(s).
db2 =>
```

13. Mostrar el total de rentas recaudadas al mes

```
mysql> SELECT SUM(rent) AS TOTAL_RENTAS FROM PropertyForRent;
+-----+
| TOTAL_RENTAS |
+-----+
|          2825 |
+-----+
1 row in set (0.00 sec)
```

```
db2 => SELECT SUM(rent) AS TOTAL_RENTAS FROM PropertyForRent
TOTAL_RENTAS
-----
          2825

1 registro(s) seleccionado(s).
```

14. Mostrar cual es la renta más cara pagada y cuál es la más barata

```
mysql> SELECT MAX(rent) AS Renta_Mas_Cara FROM PropertyForRent;
+-----+
| Renta_Mas_Cara |
+-----+
|           650 |
+-----+
1 row in set (0.00 sec)

mysql> SELECT MIN(rent) AS Renta_Mas_Barata FROM PropertyForRent;
+-----+
| Renta_Mas_Barata |
+-----+
|           350 |
+-----+
1 row in set (0.00 sec)
```

```
db2 => SELECT MAX(rent) AS Renta_Mas_Cara FROM PropertyForRent
RENTA_MAS_CARA
-----
          650

1 registro(s) seleccionado(s).

db2 => SELECT MIN(rent) AS Renta_Mas_Barata FROM PropertyForRent
RENTA_MAS_BARATA
-----
          350

1 registro(s) seleccionado(s).
```

15. Calcular el promedio de la renta que recibe la empresa

```
mysql> SELECT AVG(rent) AS Promedio_Renta FROM PropertyForRent;
+-----+
| Promedio_Renta |
+-----+
|       470.8333 |
+-----+
1 row in set (0.00 sec)
```



```
db2 => SELECT AVG(rent) AS Promedio_Renta FROM PropertyForRent
```

| PROMEDIO_RENTA |
|----------------|
| 470 |

```
1 registro(s) seleccionado(s).
db2 =>
```

16. Mostrar el total de rentas que pueden pagar los clientes al mes

```
mysql> SELECT SUM(maxRent) AS Total_Rentas_Clientes FROM Client;
```

| Total_Rentas_Clientes |
|-----------------------|
| 2125 |

```
1 row in set (0.00 sec)
```

```
db2 => SELECT SUM(maxRent) AS Total_Rentas_Mes FROM Client
```

| TOTAL_RENTAS_MES |
|------------------|
| 2125 |

```
1 registro(s) seleccionado(s).
```

17. Mostrar el total de rentas recaudadas por rentar CASAS

```
mysql> SELECT SUM(rent) AS Total_Rentas_Casas FROM PropertyForRent WHERE type = 'House';
```

| Total_Rentas_Casas |
|--------------------|
| 1250 |

```
1 row in set (0.00 sec)
```

```
db2 => SELECT SUM(rent) AS Total_Rentas_Casas FROM PropertyForRent WHERE type = 'House'
```

| TOTAL_RENTAS_CASAS |
|--------------------|
| 1250 |

```
1 registro(s) seleccionado(s).
```

18. EXPLICAR COMO FUNCIONA max y min UTILIZADO EN CAMPOS VARCHAR

Ordena los campos y en MAX muestra el valor último ordenado de A hasta Z, y en MIN muestra el primer campo ordenado.

```
mysql> SELECT MAX(City) FROM PropertyForRent;
+-----+
| MAX(City) |
+-----+
| London    |
+-----+
1 row in set (0.02 sec)

mysql> SELECT MIN(City) FROM PropertyForRent;
+-----+
| MIN(City) |
+-----+
| Aberdeen  |
+-----+
1 row in set (0.00 sec)

mysql> 
```

```
db2 => SELECT MAX(City) FROM PropertyForRent
1
-----
London

1 registro(s) seleccionado(s).

db2 => SELECT MIN(City) FROM PropertyForRent
1
-----
Aberdeen

1 registro(s) seleccionado(s).
```