



ENTREGA SQL CONSULTAS I

MARIO JIMÉNEZ MARSET

ÍNDICE

1. ENUNCIADO - OBJETIVOS.....	3
2. DESARROLLO – PROCEDIMIENTOS.....	3

1. ENUNCIADO - OBJETIVOS

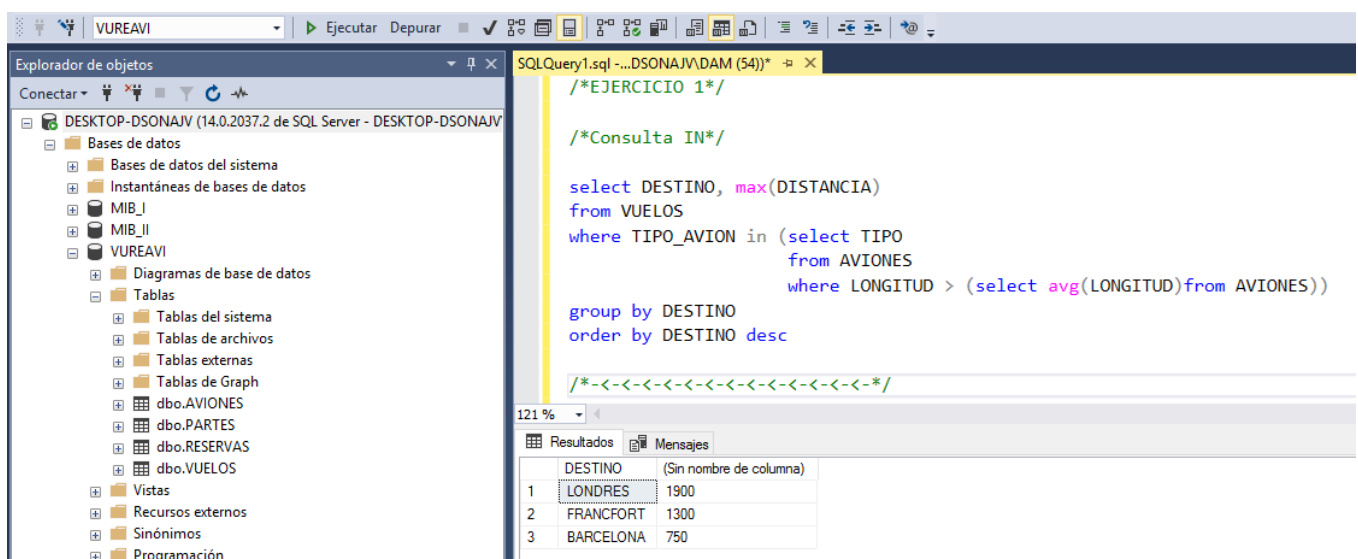
En esta práctica se pedía realizar diez consultas en SQL de tres formas diferentes. La primera forma era utilizando “IN”, la segunda con “EXISTS” y la tercera con “JOIN”.

Cada consulta se ha resuelto sacando el mismo resultado en las tres consultas de cada ejercicio.

2. DESARROLLO – PROCEDIMIENTOS

Se muestran capturas de pantalla de las consultas y su resultado.

1. Visualice, para cada destino, la mayor distancia recorrida hacia él, por vuelos realizados por aviones con longitud mayor que la media, ordenados alfabéticamente.



```

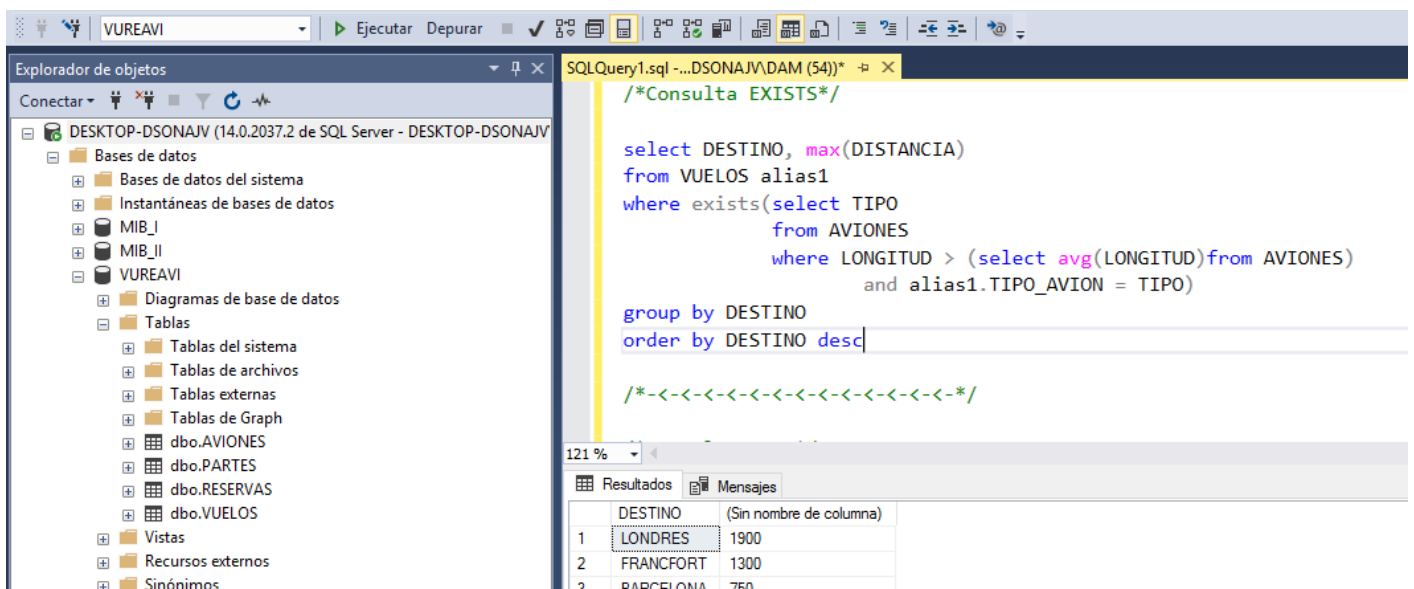
/*EJERCICIO 1*/

/*Consulta IN*/

select DESTINO, max(DISTANCIA)
from VUELOS
where TIPO_AVION in (select TIPO
                    from AVIONES
                    where LONGITUD > (select avg(LONGITUD) from AVIONES))

group by DESTINO
order by DESTINO desc
  
```

	DESTINO	(Sin nombre de columna)
1	LONDRES	1900
2	FRANCFORT	1300
3	BARCELONA	750



```

/*Consulta EXISTS*/

select DESTINO, max(DISTANCIA)
from VUELOS alias1
where exists(select TIPO
            from AVIONES
            where LONGITUD > (select avg(LONGITUD) from AVIONES)
            and alias1.TIPO_AVION = TIPO)

group by DESTINO
order by DESTINO desc
  
```

	DESTINO	(Sin nombre de columna)
1	LONDRES	1900
2	FRANCFORT	1300
3	BARCELONA	750

The screenshot shows the SQL Server Enterprise Manager interface. On the left, the 'Explorador de objetos' (Object Explorer) displays the database structure for 'DESKTOP-DSONAJV (14.0.2037.2 de SQL Server - DESKTOP-DSONAJV)'. The 'Bases de datos' (Databases) folder is expanded, showing 'Bases de datos del sistema', 'Instantáneas de bases de datos', 'MIB_I', 'MIB_II', and 'VUREAVI'. Under 'VUREAVI', the 'Tablas' (Tables) folder is expanded, showing 'Tablas del sistema', 'Tablas de archivos', 'Tablas externas', 'Tablas de Graph', and several tables: 'dbo.AVIONES', 'dbo.PARTES', 'dbo.RESERVAS', and 'dbo.VUELOS'. On the right, the 'SQLQuery1.sql' window shows a query:

```

/*<<<<<<<<<<<<<<<<<<<*/

/*Consulta JOIN*/

select DESTINO, max(DISTANCIA)
from VUELOS, AVIONES
where TIPO_AVION = TIPO and
      LONGITUD > (select avg(LONGITUD) from AVIONES)

group by DESTINO
order by DESTINO desc
  
```

Below the query, the 'Resultados' (Results) window shows the output:

	DESTINO	(Sin nombre de columna)
1	LONDRES	1900
2	FRANCFORT	1300
3	BARCELONA	750

- Visualice el número de parte menor para cada fecha que corresponda a vuelos que ni parten ni llegan a Madrid y que recorren una distancia mayor que la media de los realizados por Iberia, ordenados de menor a mayor.

The screenshot shows the SQL Server Enterprise Manager interface. On the left, the 'Explorador de objetos' (Object Explorer) displays the database structure for 'DESKTOP-DSONAJV (14.0.2037.2 de SQL Server - DESKTOP-DSONAJV)'. The 'Bases de datos' (Databases) folder is expanded, showing 'Bases de datos del sistema', 'Instantáneas de bases de datos', 'MIB_I', 'MIB_II', and 'VUREAVI'. Under 'VUREAVI', the 'Tablas' (Tables) folder is expanded, showing 'Tablas del sistema', 'Tablas de archivos', 'Tablas externas', 'Tablas de Graph', and several tables: 'dbo.AVIONES', 'dbo.PARTES', 'dbo.RESERVAS', and 'dbo.VUELOS'. On the right, the 'SQLQuery1.sql' window shows a query:

```

/*EJERCICIO 2*/

/*Consulta IN*/

select FECHA, min(NUM_PARTE) as NUM_PARTE
from PARTES
where NUM_VUELO in (select NUM_VUELO
                    from VUELOS
                    where ORIGEN not like 'MADRID' and DESTINO not like 'MADRID' and DISTANCIA > (select avg(DISTANCIA)
                                                                                               from VUELOS
                                                                                               where NUM_VUELO like 'IB%'))

group by FECHA
order by 2
  
```

Below the query, the 'Resultados' (Results) window shows the output:

	FECHA	NUM_PARTE
1	27.09.20	8
2	28.09.20	9

SQLQuery1.sql -...DSONAJV\DM (54)*

```
/*Consulta EXISTS*/  
  
select FECHA, min(NUM_PARTE) as NUM_PARTE  
from PARTES alias2  
where exists (select NUM_VUELO  
              from VUELOS  
              where ORIGEN not like 'MADRID' and DESTINO not like 'MADRID' and DISTANCIA > (select avg(DISTANCIA)  
                                                                                          from VUELOS  
                                                                                          where NUM_VUELO like 'IB%')  
              and alias2.NUM_VUELO = NUM_VUELO)  
  
group by FECHA  
order by 2  
  
/*-----*/
```

121 %

Resultados Mensajes

	FECHA	NUM_PARTE
1	27.09.20	8
2	28.09.20	9

SQLQuery1.sql -...DSONAJV\DM (54)*

```
/*-----*/  
  
/*Consulta JOIN*/  
  
select FECHA, min(NUM_PARTE) as NUM_PARTE  
from PARTES, VUELOS  
where PARTES.NUM_VUELO = VUELOS.NUM_VUELO and ORIGEN not like 'MADRID' and DESTINO not like 'MADRID'  
      and DISTANCIA > (select avg(DISTANCIA) from VUELOS where NUM_VUELO like 'IB%')  
  
group by FECHA  
order by 2
```

121 %

Resultados Mensajes

	FECHA	NUM_PARTE
1	27.09.20	8
2	28.09.20	9

SQLQuery1.sql -...DSOAJV\BAM (54))

```

/*-<-<-<-<-<-<-<-<-<-<-<-<-*/

/*Consulta JOIN*/

select distinct TIPO, ENVERGADURA*2 as DOBLE_ENVERGADURA, power(LONGITUD,2) as CUADRADO_LONGITUD
from VUELOS vu, AVIONES av
where TIPO_AVION = TIPO and (ORIGEN LIKE 'M%' or DESTINO LIKE 'M%')
and LONGITUD < (select avg(LONGITUD)from AVIONES)

order by ENVERGADURA*2 desc

```

121 %

Resultados Mensajes

	TIPO	DOBLE_ENVERGADURA	CUADRADO_LONGITUD
1	D9S	57	1466,88994155884
2	72S	50,4	1310,44005523682

4. Visualice, para cada origen, la menor distancia recorrida desde el por vuelos realizados por aviones con capacidad menor que la media ordenados alfabéticamente.

SQLQuery1.sql -...DSOAJV\BAM (54))

```

/*EJERCICIO 4*/

/*Consulta IN*/

select ORIGEN, min(DISTANCIA)
from VUELOS
where TIPO_AVION in (select TIPO
from AVIONES
where CAPACIDAD < (select avg(CAPACIDAD)from AVIONES))

group by ORIGEN
order by ORIGEN

```

121 %

Resultados Mensajes

	ORIGEN	(Sin nombre de columna)
1	BARCELONA	950
2	DUBLIN	1200
3	MADRID	600
4	MALAGA	900
5	SEVILLA	600

SQLQuery1.sql -...DSONAJV\DAM (54)*

```

/*-<-<-<-<-<-<-<-<-<-<-<-*/

/*Consulta EXISTS*/

select ORIGEN, min(DISTANCIA)
from VUELOS alias4
where exists (select TIPO
              from AVIONES
              where CAPACIDAD < (select avg(CAPACIDAD)from AVIONES)
              and alias4.TIPO_AVION = TIPO)

group by ORIGEN
order by ORIGEN

```

121 %

Resultados Mensajes

	ORIGEN	(Sin nombre de columna)
1	BARCELONA	950
2	DUBLIN	1200
3	MADRID	600
4	MALAGA	900
5	SEVILLA	600

SQLQuery1.sql -...DSONAJV\DAM (54)*

```

/*-<-<-<-<-<-<-<-<-<-<-<-*/

/*Consulta JOIN*/

select ORIGEN, min(DISTANCIA)
from VUELOS, AVIONES
where TIPO_AVION = TIPO and CAPACIDAD < (select avg(CAPACIDAD)from AVIONES)

group by ORIGEN
order by ORIGEN

```

121 %

Resultados Mensajes

	ORIGEN	(Sin nombre de columna)
1	BARCELONA	950
2	DUBLIN	1200
3	MADRID	600
4	MALAGA	900
5	SEVILLA	600


```

SQLQuery1.sql -...DSONAJV\DAM (54))*
/*-<-<-<-<-<-<-<-<-<-<-*/

/*Consulta JOIN*/

select substring(ORIGEN,1,3) as LETRAS_ORIGEN, substring(DESTINO,1,3) as LETRAS_DESTINO
from VUELOS, AVIONES
where TIPO_AVION = TIPO and LONGITUD > (select avg(LONGITUD)from AVIONES)
and ENVERGADURA < 2/3*(select max(ENVERGADURA)from AVIONES)

order by DESTINO

```

121 %

Resultados Mensajes

LETRAS_ORIGEN	LETRAS_DESTINO
---------------	----------------

6. Visualice los diferentes dos primeros caracteres de los num_vuelo y el origen de los vuelos a que corresponden partes con número de parte entre 2 y 8 y que recorren distancias mayores que la media, ordenándolos alfabéticamente por origen.

```

SQLQuery1.sql -...DSONAJV\DAM (54))*
/*EJERCICIO 6*/

/*Consulta IN*/

select substring(NUM_VUELO,1,2), ORIGEN
from VUELOS
where DISTANCIA > (select avg(DISTANCIA)from VUELOS) and NUM_VUELO in (select NUM_VUELO
from PARTES
where NUM PARTE >=2 and NUM PARTE <=8)

order by ORIGEN

```

121 %

Resultados Mensajes

(Sin nombre de columna)	ORIGEN
1 LH	COPENHAGUE
2 IB	MADRID

```
SQLQuery1.sql -...DSONAJV\DAM (54))* -p X
/*Consulta EXISTS*/

select substring(NUM_VUELO,1,2), ORIGEN
from VUELOS alias6
where exists (select NUM_VUELO
              from PARTES
              where NUM_PARTE >=2 and NUM_PARTE <=8

              and alias6.NUM_VUELO = NUM_VUELO)

and DISTANCIA > (select avg(DISTANCIA)from VUELOS)

order by ORIGEN
```

121 %

Resultados Mensajes

	(Sin nombre de columna)	ORIGEN
1	LH	COPENHAGUE
2	IB	MADRID

```
SQLQuery1.sql -...DSONAJV\DAM (54))* -p X

/*-<-<-<-<-<-<-<-<-<-<-<-*/

/*Consulta JOIN*/

select distinct substring(VUELOS.NUM_VUELO,1,2) as LETRAS_VUELOS, ORIGEN
from VUELOS, PARTES
where VUELOS.NUM_VUELO = PARTES.NUM_VUELO and DISTANCIA > (select avg(DISTANCIA)from VUELOS)
and (NUM_PARTE >=2 and NUM_PARTE <=8)

order by ORIGEN
```

121 %

Resultados Mensajes

	(Sin nombre de columna)	ORIGEN
1	LH	COPENHAGUE
2	IB	MADRID

7. Visualice los números de vuelo, las tres primeras letras del origen y las tres primeras letras del destino para los vuelos realizados por aviones con posibilidad de almacenar más combustible que la media de todos y con longitud menor que $\frac{2}{3}$ la máxima longitud, ordenándolos por número de vuelo.

```
SQLQuery1.sql -...DSONAJV\...DAM (54)) *
/*EJERCICIO 7*/
/*Consulta IN*/

select NUM_VUELO, substring(ORIGEN,1,3) as LETRAS_ORIGEN, substring(DESTINO,1,3) as LETRAS_DESTINO
from VUELOS
where TIPO_AVION in(select TIPO
                    from AVIONES
                    where COMBUSTIBLE > (select avg(COMBUSTIBLE)from AVIONES) and LONGITUD < 2/3*(select max(LONGITUD)from AVIONES))

order by NUM_VUELO

/*-<-<-<-<-<-<-<-<-<-<-<-<-<-<-<-<-*/
```

121 %

Resultados Mensajes

NUM_VUELO	LETRAS_ORIGEN	LETRAS_DESTINO
-----------	---------------	----------------

En este ejercicio pasa lo mismo que lo explicado en el ejercicio 5.

```
SQLQuery1.sql -...DSONAJV\...DAM (54)) *
/*Consulta EXISTS*/

select NUM_VUELO, substring(ORIGEN,1,3) as LETRAS_ORIGEN, substring(DESTINO,1,3) as LETRAS_DESTINO
from VUELOS alias7
where exists (select TIPO
             from AVIONES
             where COMBUSTIBLE > (select avg(COMBUSTIBLE)from AVIONES) and LONGITUD < 2/3*(select max(LONGITUD)from AVIONES)

             and alias7.TIPO_AVION = TIPO)

order by NUM_VUELO
```

121 %

Resultados Mensajes

NUM_VUELO	LETRAS_ORIGEN	LETRAS_DESTINO
-----------	---------------	----------------

```

SQLQuery1.sql -...DSONAJV\,DAM (54))* ✕
/*-<-<-<-<-<-<-<-<-<-<-*/

/*Consulta JOIN*/

select NUM_VUELO, substring(ORIGEN,1,3) as LETRAS_ORIGEN, substring(DESTINO,1,3) as LETRAS_DESTINO
from VUELOS, AVIONES
where VUELOS.TIPO_AVION = AVIONES.TIPO and COMBUSTIBLE > (select avg(COMBUSTIBLE)from AVIONES)
and LONGITUD < 2/3*(select max(LONGITUD)from AVIONES)

order by NUM_VUELO

```

121 %

Resultados Mensajes

NUM_VUELO	LETRAS_ORIGEN	LETRAS_DESTINO
-----------	---------------	----------------

8. Visualice los diferentes dos primeros caracteres de los números de vuelo y el destino de los vuelos a que corresponden partes con combustibles consumidos mayores que un tercio de la media de todos los combustibles consumidos, ordenados alfabéticamente por destinos.

```

SQLQuery1.sql -...DSONAJV\,DAM (54))* ✕
/*EJERCICIO 8*/

/*Consulta IN*/

select substring(NUM_VUELO,1,2), DESTINO
from VUELOS
where NUM_VUELO in(select NUM_VUELO
from PARTES
where COMB_CONSUMIDO > 1/3*(select avg(COMB_CONSUMIDO)from PARTES))

order by DESTINO
/*-<-<-<-<-<-<-<-<-<-<-*/

```

121 %

Resultados Mensajes

	(Sin nombre de columna)	DESTINO
1	LH	FRANCFORT
2	IB	LONDRES
3	AF	PARIS
4	IB	SEVILLA
5	IB	TENERIFE

SQLQuery1.sql -...DSONAJV\1DAM (54))*

```
/*Consulta EXISTS*/  
  
select substring(NUM_VUELO,1,2), DESTINO  
from VUELOS alias8  
where exists (select NUM_VUELO  
              from PARTES  
              where COMB_CONSUMIDO > 1/3*(select avg(COMB_CONSUMIDO)from PARTES)  
  
              and alias8.NUM_VUELO = NUM_VUELO)  
  
order by DESTINO
```

121 %

Resultados Mensajes

	(Sin nombre de columna)	DESTINO
1	LH	FRANCFORT
2	IB	LONDRES
3	AF	PARIS
4	IB	SEVILLA
5	IB	TENERIFE

SQLQuery1.sql -...DSONAJV\1DAM (54))*

```
/*-<-<-<-<-<-<-<-<-<-<-<-*/  
  
/*Consulta JOIN*/  
  
select distinct substring(VUELOS.NUM_VUELO,1,2), DESTINO  
from VUELOS, PARTES  
where VUELOS.NUM_VUELO = PARTES.NUM_VUELO and COMB_CONSUMIDO > 1/3*(select avg(COMB_CONSUMIDO)from PARTES)  
  
order by DESTINO
```

121 %

Resultados Mensajes

	(Sin nombre de columna)	DESTINO
1	LH	FRANCFORT
2	IB	LONDRES
3	AF	PARIS
4	IB	SEVILLA
5	IB	TENERIFE

9. Visualice el total de plazas libres por número de vuelo para los realizados desde Madrid a Barcelona o Sevilla y que recorran una distancia mayor que la media de todos los vuelos que salen de Madrid, ordenándolos de menor a mayor.

```
SQLQuery1.sql -...DSONAJV\DAM (54))* -p X
/*EJERCICIO 9*/

/*Consulta IN*/

select PLAZAS_LIBRES
from RESERVAS
where NUM_VUELO in(select NUM_VUELO
                    from VUELOS
                    where ORIGEN LIKE 'MADRID' and (DESTINO LIKE 'BARCELONA' or DESTINO LIKE 'SEVILLA')
                    and DISTANCIA > (select avg(DISTANCIA)from VUELOS where ORIGEN LIKE 'MADRID'))

order by PLAZAS_LIBRES asc
```

121 %

Resultados Mensajes

PLAZAS_LIBRES

En este ejercicio hay una situación similar a los ejercicios 5 y 7, ya que, al ser tan específica la subconsulta, ningún NUM_VUELO cumple todas las condiciones.

```
SQLQuery1.sql -...DSONAJV\DAM (54))* -p X
/*Consulta EXISTS*/

select PLAZAS_LIBRES
from RESERVAS alias9
where exists(select NUM_VUELO
             from VUELOS
             where ORIGEN like 'MADRID' and (DESTINO like 'BARCELONA' or DESTINO like 'SEVILLA')

             and DISTANCIA > (select avg(DISTANCIA)from VUELOS where ORIGEN like 'MADRID')

             and alias9.NUM_VUELO = NUM_VUELO)

order by PLAZAS_LIBRES asc
```

121 %

Resultados Mensajes

PLAZAS_LIBRES

```
SQLQuery1.sql -...DSONAJV\DAM (54)) * -> X
/*-<-<-<-<-<-<-<-<-<-<-<-<-*/
/*Consulta JOIN*/
select PLAZAS_LIBRES
from RESERVAS, VUELOS
where RESERVAS.NUM_VUELO = VUELOS.NUM_VUELO and ORIGEN LIKE 'MADRID'
      and DESTINO LIKE 'BARCELONA' or DESTINO LIKE 'SEVILLA'
      and DISTANCIA > (select avg(DISTANCIA) from VUELOS where ORIGEN LIKE 'MADRID')

order by PLAZAS_LIBRES asc
```

121 %

Resultados Mensajes

PLAZAS_LIBRES

10. Visualice la media de plazas libres por número de vuelo para los realizados desde Barcelona o Sevilla a Madrid y que recorren una distancia menor que la media de todos los vuelos que llegan a Madrid, ordenados de menor a mayor.

```
SQLQuery1.sql -...DSONAJV\DAM (54)) * -> X
/*EJERCICIO 10*/
/*Consulta IN*/
select NUM_VUELO, avg(PLAZAS_LIBRES) as MEDIA_VUELOS
from RESERVAS
where NUM_VUELO in(select NUM_VUELO
                    from VUELOS
                    where ORIGEN LIKE 'BARCELONA' or ORIGEN LIKE 'SEVILLA' and DESTINO LIKE 'MADRID'
                      and DISTANCIA < (select avg(DISTANCIA) from VUELOS where DESTINO LIKE 'MADRID'))

group by NUM_VUELO
order by NUM_VUELO asc
```

121 %

Resultados Mensajes

NUM_VUELO	MEDIA_VUELOS
-----------	--------------

Al igual que en los ejercicios 5,7 y 9, la subconsulta es demasiado específica y no hay ningún resultado, por ello el resultado sale vacío.

```
SQLQuery1.sql -...DSONAJV\DAM (54)) * - X
```

```
/*Consulta EXISTS*/  
  
select NUM_VUELO, avg(PLAZAS_LIBRES) as MEDIA_VUELOS  
from RESERVAS alias10  
where exists(select NUM_VUELO  
            from VUELOS  
            where (ORIGEN = 'BARCELONA' or ORIGEN = 'SEVILLA') and DESTINO = 'MADRID'  
  
            and DISTANCIA < (select avg(DISTANCIA) from VUELOS where DESTINO = 'MADRID')  
  
            and alias10.NUM_VUELO = NUM_VUELO)  
  
group by NUM_VUELO  
order by NUM_VUELO asc
```

121 %

Resultados Mensajes

NUM_VUELO	MEDIA_VUELOS
-----------	--------------

```
SQLQuery1.sql -...DSONAJV\DAM (54)) * - X
```

```
/*Consulta JOIN*/  
  
select RESERVAS.NUM_VUELO, avg(PLAZAS_LIBRES) as MEDIA_VUELOS  
from RESERVAS, VUELOS  
where RESERVAS.NUM_VUELO = VUELOS.NUM_VUELO and ORIGEN LIKE 'BARCELONA' or ORIGEN LIKE 'SEVILLA' and DESTINO LIKE 'MADRID'  
      and DISTANCIA < (select avg(DISTANCIA) from VUELOS where DESTINO LIKE 'MADRID')  
  
group by RESERVAS.NUM_VUELO  
order by NUM_VUELO asc
```

121 %

Resultados Mensajes

NUM_VUELO	MEDIA_VUELOS
-----------	--------------