

Práctica II

SQL: Lenguaje de Consultas

Grado en
Ingeniería
Informática



Bases de
Datos
2021/22



Departamento de Tecnologías de la Información
Universidad de Huelva

Objetivo

- Saber utilizar, de forma eficiente, el lenguaje SQL para consultar bases de datos relacionales

Contenidos

1 El lenguaje SQL

- 1.1 Lenguaje de definición de datos
- 1.2 Lenguaje de manipulación de datos

2 Lenguaje de consulta

- 2.1 Consultas básicas
- 2.2 Consultas a más de una tabla
- 2.3 Consultas anidadas o subconsultas
- 2.4 La función EXISTS
- 2.5 Concatenación externa
- 2.6 Funciones de agrupación
- 2.7 La división en SQL

Duración

- 5 clases

Bibliografía

- Capítulo 8 [Elmasri 02]
- Capítulo 3 [Silberschatz 06]
- Capítulo 5 [Connolly 05]

1 El Lenguaje SQL

1.1 Lenguaje de Definición de Datos

La documentación sobre el Lenguaje de Definición de Datos de SQL se encuentra en la sección "Práctica I" disponible en la página web de la asignatura.

1.2 Lenguaje de Manipulación de Datos

SQL proporciona cuatro instrucciones para la manipulación de datos:

- SELECT: seleccionar] *Extracción de datos*
- UPDATE: actualizar
- DELETE: eliminar
- INSERT: insertar] *Instrucciones de Actualización*

La parte del LMD relacionada con la extracción de los datos se suele denominar **lenguaje de consulta**

La documentación sobre las *Instrucciones de Actualización* de SQL se encuentra en la sección "Práctica I" disponible en la página web de la asignatura.

2 Lenguaje de Consulta

2.1 Consultas básicas

- Forma básica de instrucción SELECT:

```
SELECT <lista de atributos>
FROM   <lista de tablas>
WHERE  <condición>
```

- <lista de atributos>: nombres de los atributos cuyos valores va a obtener la consulta
- <lista de tablas>: nombres de las tablas requeridas para procesar la consulta
- <condición>: expresión lógica que identifica las tuplas que obtendrá la consulta

- Una consulta básica en SQL corresponde a la combinación de las operaciones **seleccionar** y **proyectar** del álgebra relacional

Ejemplos

EI-1. Obtener el código, nombre y especialidad de las asignaturas de tercero que tengan más de 4,5 créditos

SQL: Lenguaje de Consulta

- ❑ Suele ser habitual asignar **alias** a los nombres de las tablas y **cualificar** los atributos, sobre todo cuando se realizan consultas a varias tablas, con objeto de eliminar la ambigüedad y facilitar la escritura. Esto se consigue anteponiendo el nombre de la tabla o el alias a los atributos y separando los dos con un **punto**
- ❑ Pueden **renombrarse** los nombres de los atributos en el resultado, escribiendo el nuevo nombre justo detrás del atributo o bien utilizando la sentencia AS, en la cláusula SELECT
- ❑ Se puede **omitar** la cláusula WHERE para indicar una selección de tuplas incondicional
- ❑ Para obtener los valores de **todos** los atributos de las tuplas seleccionadas se utiliza el carácter asterisco (*)

EI-2. Obtener toda la información disponible de todos los profesores

- ❑ SQL trata las tablas como **multiconjuntos** ⇒ no elimina las tuplas repetidas que se obtengan en el resultado
- ❑ Hay que indicárselo explícitamente mediante la cláusula **DISTINCT**.

SQL: Lenguaje de Consulta

EI-3. Obtener los distintos tipos de ordenadores que existen en la Escuela

- ❑ Podemos **calcular** valores y mostrarlos en el resultado de la consulta, mediante operaciones aritméticas realizadas sobre los campos
- ❑ La cláusula ORDER BY y las palabras reservadas DESC y ASC permiten mostrar el resultado de la consulta de forma **ordenada** según los campos que se elijan como criterio de ordenación
- ❑ Por defecto, ordena **ascendentemente**

EI-4. Mostrar el nombre de los alumnos, el número de hermanos y el descuento que le corresponde (el descuento es de 300 € por hermano), ordenados de mayor a menor según el descuento, y en caso de igualdad, ordenados alfabéticamente por nombre

SQL: Lenguaje de Consulta

- ❑ BETWEEN: comparar un valor **entre** otros dos valores
- ❑ IN: comprueba que un valor **pertece** a un conjunto valores.

EI-5. Obtener el dni y nombre de los alumnos nacidos entre 1970 y 1974, y cuya localidad de nacimiento sea Huelva o Cádiz

- ❑ En atributos de tipo cadena, la selección se puede realizar utilizando un patrón de búsqueda
- ❑ El operador de comparación es LIKE (**como**)
- ❑ Para la descripción de **patrones** se utilizan dos caracteres especiales:
 - % (tanto por ciento): sustituye a un número arbitrario de caracteres
 - _ (subrayado): sustituye a un solo carácter
- ❑ NOT delante del operador LIKE: **complementario** de la búsqueda

SQL: Lenguaje de Consulta

EI-6. Obtener el nombre de todos los alumnos cuyo nombre empiece por la letra M y hayan nacido en una ciudad cuyo nombre tenga, al menos, 6 caracteres pero no comience por la letra P

- ❑ conjunto de funciones predefinidas para manipulación de fechas
 - **CURRENT_DATE**: devuelve la fecha actual
 - **SYSDATE**: devuelve la fecha y hora actuales
 - **EXTRACT({ YEAR | MONTH | DAY } FROM fecha)**: extrae la parte de la fecha indicada

EI-7. Obtener un listado de los años de nacimiento de los alumnos, ordenados crecientemente

SQL: Lenguaje de Consulta

El esquema completo de la base de datos “*MoviFacturas*” es el siguiente:

COMPAÑIA (cif, nombre, web) CP: cif Único: nombre VNN: nombre	TARIFA (tarifa,compañia,descripcion,coste) CP: (tarifa,compañia) CAj: compañía → COMPAÑIA (cif) VNN: coste
LLAMADA (tf_origen,tf_destino,fecha_hora, duracion) CP: (tf_origen,fecha_hora) CAj: tf_origen → TELEFONO(numero) CAj: tf_destino → TELEFONO(numero) VNN: duracion VNN: tf_destino,duracion Unico: (tf_destino, fecha_hora)	TELEFONO (numero,f_contrato,tipo,puntos, compañía,tarifa,cliente) CP: numero CAj: (tarifa,compañia) → TARIFA CAj: compañía → COMPAÑIA (cif) CAj: cliente → CLIENTE (dni) VNN: compañía, tarifa, cliente
CLIENTE (dni, nombre, f_nac, direccion, cp, ciudad provincia) CP: dni VNN: nombre	

Este esquema está creado en la base de datos “*rabida*” y su propietario es el usuario “*mf*”. Todos vosotros tenéis permiso de lectura (SELECT) sobre las tablas del esquema. Puesto que el propietario es otro usuario, en las consultas debéis anteponer el nombre de las tablas al nombre del propietario.

Ejemplo: `SELECT *
FROM mf.cliente
WHERE provincia = 'Huelva';`

Sobre la BD de MoviFacturas, resolver las siguientes consultas:

MF-1. Obtener el nombre de las compañías cuya dirección web contenga la cadena 'et' y acabe en 'com'

nombre

Kietostar
Petafón

MF-2. Obtener el nombre y dirección de los clientes nacidos en 1973 o 1985 y cuyo código postal comience por 15, ordenado ascendentemente por el nombre y, en caso de igualdad, descendentemente por la dirección

nombre	direccion
Ricardo Reyes Neruda	Carretera de Isla Negra Km. 20
Rita Martínez de Castro	C/ Sar nº 84

Sobre la BD de MoviFacturas, resolver las siguientes consultas:

MF-3. Obtener el teléfono de destino de las llamadas realizadas desde el número “666010101”, en el año 2006

```
tf_destino
-----
666789789
666456456
654234234
```

MF-4. Obtener los números de teléfono que han llamado alguna vez al “666010101”, entre las 10:00 y las 12:00

```
tf_origen
-----
666789789
654012012
```

Consultas a más de una tabla

- Antes de la versión SQL-92 la concatenación de tablas se conseguía mediante un producto cartesiano en el que se indicaba la condición de combinación en la cláusula WHERE
- El producto cartesiano se construye poniendo en la cláusula FROM las tablas que queremos componer separadas por comas

EI-8. Obtener los nombres de las asignaturas junto con el nombre del profesor responsable

- Se puede obtener el mismo resultado utilizando la cláusula INNER JOIN (introducido en SQL2) e indicando explícitamente la condición de relación entre las tablas

SQL: Lenguaje de Consulta

- Si coincide el nombre del atributo en ambas tablas, se puede utilizar la cláusula **USING**, indicando el atributo de combinación
- La cláusula **NATURAL JOIN** se utiliza para combinar tablas según todos los atributos cuyos nombre coincidan en ambas tablas

EI-9. Obtener los números de los alumnos que se han matriculado en Bases de Datos I en el curso 2002-03

EI-10. Obtener los nombres de los alumnos que han aprobado la asignatura Algoritmos y Estructuras de Datos I en la convocatoria de febrero_junio de 2001

SQL: Lenguaje de Consulta

EI-11. Obtener un listado con el número de despacho, pero sólo de aquellos donde hay al menos 2 profesores

- Operaciones de la **teoría de conjuntos**:
 - unión (UNION)
 - intersección (INTERSECT)
 - diferencia (MINUS)
- Las tres operaciones tratan a las tablas como conjuntos, es decir, **eliminan** las tuplas repetidas del resultado
- Las tablas deben ser **compatibles**: mismo número de atributos y mismos tipos de datos

SQL: Lenguaje de Consulta

EI-12. Obtener una lista con todas las asignaturas de las que es responsable o docente la profesora Dolores Toscano Barriga

SQL: Lenguaje de Consulta

- Para preguntar por atributos con valores nulos, SQL utiliza las cláusulas **IS NULL** (es nulo) e **IS NOT NULL** (no es nulo) en lugar de usar los operadores = y ≠

EI-13. Obtener los nombres de los alumnos que no se han presentado a ‘Bases de Datos I’ en diciembre de 2002 por haberla aprobado en una convocatoria anterior del mismo año

Sobre la BD de MoviFacturas, resolver las siguientes consultas:

MF-5. Mostrar el código y coste de las tarifas junto con el nombre de la compañía que las ofrecen, de aquellas tarifas cuya descripción indique que otras personas deben estar también en la misma compañía

TARIFA	COSTE	NOMBRE
dúo	0,2	Kietostar
familiar	0,15	Kietostar
dúo	0,15	Aotra
empresa	0,19	Petafón

MF-6. Nombre y número de teléfonos de aquellos abonados con contrato que tienen tarifas inferiores a 0,20 €.

NUMERO	NOMBRE
654789789	José García Márquez
678234234	Rita Martínez de Castro
666456456	Juan Jiménez Platero

Sobre la BD de MoviFacturas, resolver las siguientes consultas:

MF-7. Obtener el código de las tarifas, el nombre de las compañías, los números de teléfono y los puntos, de aquellos teléfonos que se contrataron en el año 2006 y que hayan obtenido más de 200 puntos.

TARIFA	NOMBRE	NUMERO	PUNTOS
familiar	Kietostar	654789789	24590
joven	Kietostar	654234234	250
autónomos	Aotra	678234234	15700

MF-8. Obtener los números de teléfono (origen y destino), así como el tipo de contrato, de los clientes que alguna vez hablaron por teléfono entre las 8 y las 10 de la mañana

ORIGEN	T	DESTINO	T
678234234	C	666010101	T
654123321	C	654345345	C

Sobre la BD de MoviFacturas, resolver las siguientes consultas:

MF-9. Interesa conocer los nombres y números de teléfono de los clientes (origen y destino) que, perteneciendo a compañías distintas, mantuvieron llamadas que superaron los 15 minutos. Se desea conocer, también, la fecha y la hora de dichas llamadas así como la duración de esas llamadas

Cliente origen	TF_ORIGEN	Cliente destino	TF_DESTIN	FECHA_HORA	DURACION
Rita Martínez de Castro	678234234	Ramón Martínez Sabina	654123321	01/10/06 13:12:11	2045
José García Márquez	654789789	Ricardo Reyes Neruda	678111222	08/10/06 10:55:15	1345

Consultas anidadas o subconsultas

- A veces resulta necesario seleccionar valores de la base de datos para que actúen en la condición de comparación
- La forma más simple de realizar esta operación es mediante las **consultas anidadas** o **subconsultas**: consultas SELECT completas dentro de la cláusula WHERE de otra consulta (**consulta exterior**).
- Podemos utilizar los operadores de comparación (=, >, >=, <, <= y <>) para comparar un valor de la tupla que se está examinando con un único valor producido por la subconsulta
- Cuando se utilicen operadores de comparación, la subconsulta debe devolver un **único** valor

EI-14. Obtener una lista de los alumnos que hayan nacido en la misma ciudad que Samuel Toscano Villegas y tengan el mismo número de hermanos que Beatriz Rico Vázquez

SQL: Lenguaje de Consulta

- ❑ El operador IN se utiliza para comparar la **pertenencia** de un valor a un conjunto de valores
- ❑ La **no pertenencia** de un valor a un conjunto se expresa mediante NOT IN
- ❑ El resultado de la subconsulta puede contener más de un valor

EI-15. Obtener una lista con los alumnos matriculados en BDII y no en BDI

- ❑ Operador (=ANY): devuelve verdadero si el valor que se está comparando es igual a **algún** valor del conjunto (equivalente a IN)
- ❑ El operador = se puede sustituir por cualquier otro operador de comparación (>, >=, <, <=, <>)

EI-16. Obtener una lista de los profesores que tienen más antigüedad que alguno de los profesores del despacho FC-7366 (sin contar los del propio despacho FC-7366)

- Operador (= ALL): devuelve verdadero si el valor que se está comparando es igual a **todos** los valores devueltos por la subconsulta
- El operador = se puede sustituir por cualquier otro operador de comparación (>, >=, <, <=, <>)

SQL: Lenguaje de Consulta

EI-17. Obtener una lista con los nombres de los alumnos que hayan sacado una nota más alta en septiembre que la más alta de las notas de febrero/junio, en el año 2002 y en la asignatura Bases de Datos I

Sobre la BD de MoviFacturas, resolver las siguientes consultas:

MF-10. Obtener la fecha (día-mes-año) en la que se realizó la llamada de mayor duración

Fecha

01/10/2006

MF-11. Obtener el nombre de los abonados de la compañía ‘Aotra’ con el mismo tipo de tarifa que la del telefono “654123321”

nombre

Ricardo Reyes Neruda

Sobre la BD de MoviFacturas, resolver las siguientes consultas:

MF-12. Mostrar, utilizando para ello una subconsulta, el número de teléfono, fecha de contrato y tipo de los abonados que han llamado a teléfonos de clientes de fuera de la provincia de La Coruña durante el mes de octubre de 2006.

NUMERO	F_CONTRA	T
654345345	10/09/06	C
654789789	30/05/06	C
666010101	09/10/05	T
678111222	02/02/02	T
678234234	30/08/06	C

Sobre la BD de MoviFacturas, resolver las siguientes consultas:

MF-13. Se necesita conocer el nombre de los clientes que tienen teléfonos con tarifa “dúo” pero no “autónomos”. Utiliza subconsultas para obtener la solución.

NOMBRE

Ramón Martínez Sabina
Ricardo Reyes Neruda

Sobre la BD de MoviFacturas, resolver las siguientes consultas:

MF-14. Obtener mediante subconsultas los nombres de clientes y números de teléfono de aquellos que hicieron llamadas a teléfonos de la compañía Petafón pero no Aotra

NOMBRE	NUMERO
Ramón Martínez Sabina	666010101
Maria Machado Ruiz	666789789
Rita Martinez de Castro	678234234

Sobre la BD de MoviFacturas, resolver las siguientes consultas:

MF-15. Nombre de los clientes de la compañía Kietostar que hicieron las llamadas de mayor duración en septiembre de 2006

NOMBRE
Ramón Martínez Sabina

MF-16. Se necesita conocer el nombre de los clientes que tienen teléfonos con fecha de contratación anterior a alguno de los teléfonos de Ramón Martínez Sabina, excluido, claro, el propio Ramón Martínez Sabina.

NOMBRE
Ricardo Reyes Neruda
María Machado Ruiz
Jesús Ríos Lorca
Juan Jiménez Platero

La función EXISTS

- Cuando una consulta anidada o subconsulta puede existir por sí sola se dice que es una subconsulta independiente
- Subconsultas correlacionadas:** subconsultas cuyos valores de ejecución dependen de la tupla que está siendo examinada en la consulta exterior
- La consulta anidada se evalúa una sola vez para cada tupla de la consulta externa
- La función **EXISTS** se utiliza para comprobar si el resultado de una consulta correlacionada está o no vacío (contiene o no tuplas)
 - Devuelve *Verdad* si la consulta anidada contiene alguna tupla, y
 - *Falso* en caso contrario
- La función **NOT EXISTS** realiza la operación contraria

EI-18. Listado de los despachos donde hay profesores que no son responsables de ninguna asignatura

SQL: Lenguaje de Consulta

□ Funcionamiento:

- Por cada tupla de la consulta externa (en este caso por cada profesor), se evalúa la consulta interna con dicha tupla
- En la consulta anidada se seleccionan las tuplas de la tabla ASIGNATURA que cumplen la condición de la cláusula WHERE del profesor que se esté evaluando
- Si el resultado de la consulta correlacionada no devuelve ninguna tupla, entonces se selecciona el profesor

EI-19. Listado de los alumnos que se han matriculado de alguna asignatura en el año 2000 o 2002, y de ninguna asignatura en el año 2001

SQL: Lenguaje de Consulta

- ❑ Aunque en la mayoría de las ocasiones las **consultas correlacionadas** van asociadas a las cláusulas EXISTS y NOT EXISTS, éstas se pueden utilizar para resolver otros tipos de consultas

EI-20. Obtener el nombre de los profesores y los números de sus despachos de aquellos profesores que no comparten despacho

Sobre la BD de MoviFacturas, resolver las siguientes consultas:

MF-17. Utilizando consultas correlacionadas, mostrar el nombre de los abonados que han llamado el día '16/10/06'

NOMBRE

Ramón Martínez Sabina
Rita Martínez de Castro

MF-18. Utilizando consultas correlacionadas, obtener el nombre de los abonados que han realizado llamadas de menos de 1 minuto y medio

NOMBRE

Ramón Martínez Sabina
José García Márquez
Maria Machado Ruiz
Rita Martínez de Castro

Sobre la BD de MoviFacturas, resolver las siguientes consultas:

MF-19. Utilizando consultas correlacionadas, obtener el nombre de los abonados de la compañía 'KietoStar' que no hicieron ninguna llamada el mes de septiembre

NUMERO	NOMBRE
654789789	José García Márquez
654234234	Antonio Hierro Santander
654345345	Rita Martínez de Castro

MF-20. Utilizando consultas correlacionadas, mostrar todos los datos de los teléfonos que hayan llamado al número 654234234 pero no al 666789789

NUMERO	F_CONTRA	T	PUNTOS	COMPAÑIA	TARIFA	CLIENTE
678111222	02/02/02	T	1010	B00000002	dúo	59000003T
654345345	10/09/06	C	170	A00000001	dúo	89000002Z

Sobre la BD de MoviFacturas, resolver las siguientes consultas:

MF-21. Utilizando consultas correlacionadas, obtener el nombre y número de teléfono de los clientes de la compañía Kietostar que no han hecho llamadas a otros teléfonos de la misma compañía

NOMBRE	NUMERO
-----	-----
Antonio Hierro Santander Jesús Ríos Lorca	654234234 654012012

Concatenación externa

- SQL, al igual que el álgebra relacional, permite la agrupación de tablas incluso si los campos de unión no son coincidentes o uno de sus valores es nulo (concatenación externa)

EI-21. Obtener los nombres de **todos** los profesores junto con las asignaturas de las que son responsables

- Izquierda: LEFT OUTER JOIN
- Derecha: RIGHT OUTER JOIN
- Ambos: FULL OUTER JOIN

Funciones de agrupación

- Funciones más habituales:
 - **COUNT**: cuenta. COUNT(*) devuelve el número de filas en el resultado de la consulta
 - **SUM**: suma
 - **MAX**: máximo
 - **MIN**: mínimo
 - **AVG**: promedio

SQL: Lenguaje de Consulta

- Se puede utilizar la función COUNT para contar los valores (no nulos) de una columna en vez de las tuplas. Necesitaremos utilizar la cláusula DISTINCT para eliminar los duplicados

EI-22. Se desea saber cuántos despachos ocupan los profesores de la escuela

EI-23. Obtener el número de matriculados, la nota máxima, la mínima y la nota media de la asignatura 'Bases de Datos I' en la convocatoria de septiembre de 2002

- Algunas veces resulta necesario utilizar estas funciones para seleccionar tuplas que cumplen alguna condición
- En este caso, se especifica una consulta anidada (correlacionada) con la función deseada y se usará dicha consulta en la cláusula WHERE de una consulta exterior

SQL: Lenguaje de Consulta

EI-24. Obtener, mediante una consulta correlacionada, los nombres de las asignaturas que tienen recomendadas 2 o más asignaturas.

EI-25. Obtener, mediante una consulta correlacionada, los nombres de los profesores que tienen más antigüedad que, al menos, otros cinco profesores.

- Suele ser habitual aplicar estas funciones integradas a subgrupos de tuplas de una relación, agrupadas por uno o varios atributos
- La cláusula que realiza la agrupación según unos determinados atributos es **GROUP BY** (agrupar por)
- En la cláusula SELECT es correcto que se especifiquen los atributos de agrupación (que aparezcan en la cláusula GROUP BY) así como funciones de agrupación, pero no atributos cuyos valores puedan variar para un mismo grupo

SQL: Lenguaje de Consulta

EI-26. Para cada asignatura y año académico, mostrar el nombre de la asignatura, el año, el número de alumnos que se han presentado y la nota media obtenida en la convocatoria de febrero_junio

- A veces ocurre que solamente queremos utilizar estas funciones con aquellos grupos de tuplas que satisfagan ciertas condiciones
- SQL proporciona la cláusula **HAVING** (que tiene), la cual aparece junto con la cláusula GROUP BY
- HAVING especifica una condición, y sólo aquellos grupos que satisfagan dicha condición, entrarán en el resultado de la consulta

EI-27. Obtener un listado con el nombre del alumno, el nombre de la asignatura y el número de veces que se ha matriculado en esa asignatura, pero sólo cuando se haya matriculado 3 o más años

SQL: Lenguaje de Consulta

EI-28. Nombre de los alumnos que hayan sacado más de un 5 de nota media en junio del 2002

EI-29. Nombre de las asignaturas y número de alumnos matriculados de las asignaturas donde se hayan matriculado más alumnos en el año 2002

- Hay que tener un especial cuidado cuando queremos restringir tuplas mediante las cláusulas WHERE y HAVING

SQL: Lenguaje de Consulta

EI- 30. Obtener el número total de alumnos que han suspendido en cada asignatura en junio de 2002, pero sólo de aquellas asignaturas en las que se hayan matriculado más de 50 alumnos

Primera tentativa:

2.7 La división en SQL

- El lenguaje SQL no proporciona un operador que implemente directamente la división
- Hay que replantear la consulta de otra manera, para lo cual nos basaremos en la división del CRT
- Por ejemplo, queremos saber los alumnos que se han matriculado de **todas** las asignaturas de las que es responsable el profesor “Santiago Baroja López”
- Nos replanteamos la pregunta de otra forma. *No existe ninguna asignatura de ese profesor de la que no está matriculado un determinado alumno*

Sobre la BD de MoviFacturas, resolver las siguientes consultas:

MF-22. Mostrar la compañía con la que se realizaron más llamadas durante el día 16/10/06

NOMBRE

Kietostar

MF-23. Obtener los números de teléfono y los nombres de los abonados que han llamado a todos los números a los que se llamó desde el 654345345 en octubre de 2006

NUMERO	NOMBRE

678234234	Rita Martínez de Castro

Sobre la BD de MoviFacturas, resolver las siguientes consultas:

MF-24. Mostrar el nombre de cada cliente junto con coste total de las llamadas que realiza con cada compañía.
El resultado debe mostrarse ordenado descendente por cliente y ascendente por compañía

CLIENTE	COMPAÑIA	COSTETOTAL
Rita Martínez de Castro	Aotra	11,964
Rita Martínez de Castro	Kietostar	4,11333333
Ricardo Reyes Neruda	Aotra	4,0225
Ramón Martínez Sabina	Kietostar	10,18
Ramón Martínez Sabina	Petafón	2,43516667
Maria Machado Ruiz	Petafón	2,14283333
José García Márquez	Kietostar	4,625
Jesús Ríos Lorca	Kietostar	3,775

MF-25. Para cada cliente residente en la provincia de 'La Coruña', mostrar la duración de todas las llamadas realizadas a clientes residentes en 'Jaén'

CLIENTE	DURACION
Rita Martínez de Castro	2753

Sobre la BD de MoviFacturas, resolver las siguientes consultas:

MF-26. Mostrar los nombres de todos los clientes que hayan realizado más de 5 llamadas

NOMBRE	TOTAL
Ramón Martínez Sabina	8
Rita Martínez de Castro	9

MF-27. Mostrar el nombre de aquellos clientes que hayan contratado con cualquier compañía, de media, una tarifa superior a la tarifa media total.

CLIENTE	MEDIA
Antonio Hierro Santander	,25
Jesús Ríos Lorca	,25
María Machado Ruiz	,23
Ramón Martínez Sabina	,195

MF-28. Mostrar los nombres de los clientes que hayan realizado llamadas a teléfonos de la compañía 'Kietostar' por un coste menor de 100€

CLIENTE
José García Márquez
Ramón Martínez Sabina
Ricardo Reyes Neruda
Rita Martínez de Castro