

## Operaciones numéricas

Usando la base de datos ciclistas (1 y 2) o miscelanea (3).

1. Si la dificultad es el cociente de la altura por la pendiente, mostrar para cada puerto con número de etapa par su dificultad con un solo decimal, mostrando primero los de mayor dificultad. (9 filas)

nombrepuerto	dificultad
Sierra Nevada	416.6
Coll de Ordino	396.0
Puerto de Navalmoral	380.2
Arcalis	318.5
Puerto de Pedro Bernardo	312.5
Puerto de Mijares	305.0
Coll de la Comella	204.0
Lagos de Covadonga	162.0
Alto del Naranco	80.7

```
mysql> create table Puertos2 (nombrePuerto varchar(30), etapa_con_puerto int, altura int, pendiente int);
```

```
mysql> use Ciclistas;
Database changed
mysql> create table Puertos2 (nombrePuerto varchar(30), etapa_con_puerto int, altura int, pendiente int);
Query OK, 0 rows affected (0,08 sec)
```

```
mysql> insert into Puertos2 values ('Puerto de la Morcuera',19,1760,92);
```

```
mysql> insert into Puertos2 values ('Puerto de la Morcuera',19,1760,92);
Query OK, 1 row affected (0,02 sec)
```

```
mysql> insert into Puertos2 values ('Coll de Ordino', 15, 1980, 73);
```

```
mysql> insert into Puertos2 values ('Coll de Ordino', 15, 1980, 73);
Query OK, 1 row affected (0,03 sec)
```

```
mysql> insert into Puertos2 values ('Puerto de Mijares', 2,1525,79);
```

```
mysql> insert into Puertos2 values ('Puerto de Mijares', 2,1525,79);  
Query OK, 1 row affected (0,01 sec)
```

```
mysql> insert into Puertos2 values ('Navacerrada',11, 1860, 80);
```

```
mysql> insert into Puertos2 values ('Navacerrada',11, 1860, 80);  
Query OK, 1 row affected (0,01 sec)
```

```
mysql>insert into Puertos2 values ('Cruz de la Demanda', 16, 1850, 90);
```

```
mysql> insert into Puertos2 values ('Cruz de la Demanda', 16, 1850, 90);  
Query OK, 1 row affected (0,01 sec)
```

```
mysql> insert into Puertos2 values ('Coll de la Comella', 10, 1632, 95);
```

```
mysql> insert into Puertos2 values ('Coll de la Comella', 10, 1632, 95);  
Query OK, 1 row affected (0,01 sec)
```

```
mysql>insert into Puertos2 values ('Cerler-Circo de Ampriu', 18, 2500, 84);
```

```
mysql> insert into Puertos2 values ('Cerler-Circo de Ampriu', 18, 2500, 84);  
Query OK, 1 row affected (0,04 sec)
```

```
mysql>insert into Puertos2 values ('Arcalis', 1, 2230, 81);
```

```
mysql> insert into Puertos2 values ('Arcalis', 1, 2230, 81);  
Query OK, 1 row affected (0,01 sec)
```

```
mysql>insert into Puertos2 values ('Sierra Nevada', 8, 2500, 78);
```

```
mysql> insert into Puertos2 values ('Sierra Nevada', 8, 2500, 78);  
Query OK, 1 row affected (0,04 sec)
```

```
mysql> select * from Puertos2;
```

```
mysql> select * from Puertos2;
```

nombrePuerto	etapa_con_puerto	altura	pendiente
Coll de Ordino	15	1980	73
Puerto de Mijares	2	1525	79
Navacerrada	11	1860	80
Cruz de la Demanda	16	1850	90
Coll de la Comella	10	1632	95
Cerler-Circo de Ampriu	18	2500	84
Arcalis	1	2230	81
Sierra Nevada	8	2500	78
Puerto de la Morcuera	19	1760	92

9 rows in set (0,00 sec)

```
mysql> SELECT nombrePuerto, ROUND(altura / pendiente, 1) AS dificultad from Puertos2  
WHERE Etapa_con_puerto % 2 = 0 ORDER BY dificultad DESC LIMIT 9;
```

```
mysql> SELECT nombrePuerto, ROUND(altura / pendiente, 1) AS dificultad from Puertos2 WHERE Etapa_con_puerto % 2 = 0 ORDER BY dificultad DESC LIMIT 9;
```

nombrePuerto	dificultad
Sierra Nevada	32.1
Cerler-Circo de Ampriu	29.8
Cruz de la Demanda	20.6
Puerto de Mijares	19.3
Coll de la Comella	17.2

```
5 rows in set (0,00 sec)
```

2. Mostrar la salida de cada etapa y cuánto hay entre el número del dorsal del ganador y el número de la etapa que ganó, así como los kilómetros, de las etapas cuya raíz cuadrada de la distancia sea menor que 13. (7 filas)

salida	diferencia	kms
Valladolid	0	9
Granada	46	150
Benidorm	7	40
Benidorm	26	150
Santander	11	160
Cangas de Ons	13	140
Segovia	18	52

```
mysql> create table etapas (salida VARCHAR(100), dorsal_ganador INT, num_etapa INT, kms INT);
```

```
mysql> create table etapas (salida VARCHAR(100), dorsal_ganador INT, num_etapa INT, kms INT);  
Query OK, 0 rows affected (0,08 sec)
```

```
mysql> insert into etapas values ('Valladolid', 1, 1, 9);
```

```
mysql> insert into etapas values ('Valladolid', 1, 1, 9);  
Query OK, 1 row affected (0,01 sec)
```

```
mysql> insert into etapas values ('Granada', 50, 4, 150);
```

```
mysql> insert into etapas values ('Granada', 50, 4, 150);  
Query OK, 1 row affected (0,02 sec)
```

```
mysql> insert into etapas values('Benidorm', 10, 3, 40);
```

```
mysql> insert into etapas values('Benidorm', 10, 3, 40);  
Query OK, 1 row affected (0,01 sec)
```

```
mysql> insert into etapas values ('Benidorm', 30, 4, 150);
```

```
mysql> insert into etapas values ('Benidorm', 30, 4, 150);  
Query OK, 1 row affected (0,01 sec)
```

```
mysql> insert into etapas values ('Santander', 15, 4, 160);
```

```
mysql> insert into etapas values ('Santander', 15, 4, 160);  
Query OK, 1 row affected (0,01 sec)
```

```
mysql> insert into etapas values ('Cangas de Ons', 20, 7, 140);
```

```
mysql> insert into etapas values ('Cangas de Ons', 20, 7, 140);  
Query OK, 1 row affected (0,02 sec)
```

```
mysql> insert into etapas values ('Segovia', 25, 7, 52);
```

```
mysql> insert into etapas values ('Segovia', 25, 7, 52);  
Query OK, 1 row affected (0,01 sec)
```

```
mysql> SELECT salida, (dorsal_ganador - num_etapa) AS diferencia, kms FROM etapas  
WHERE SQRT(kms) < 13 LIMIT 7;
```

```
mysql> SELECT salida, (dorsal_ganador - num_etapa) AS diferencia, kms FROM etapas WHERE SQRT(kms) < 13 LIMIT 7;  
+-----+-----+-----+  
| salida | diferencia | kms |  
+-----+-----+-----+  
| Valladolid | 0 | 9 |  
| Granada | 46 | 150 |  
| Benidorm | 7 | 40 |  
| Benidorm | 26 | 150 |  
| Santander | 11 | 160 |  
| Cangas de Ons | 13 | 140 |  
| Segovia | 18 | 52 |  
+-----+-----+-----+  
7 rows in set (0,00 sec)
```

3.

De la tabla FACTURAS visualizar la fecha de factura, el producto, las unidades, el número de paquetes y los huecos del último paquete de aquellas facturas en que las unidades sean inferiores a 36, ordenados descendientemente por la columna de fecha de factura. Se ha de tener en cuenta que en un paquete caben 12 unidades.

Total 87 filas:

fecha_fac	producto	unidades	num_paquetes	hueco_ultimo_paquete
2005-06-30	Y. fresa semi	8	1	4
2005-06-30	Leche calcio	12	1	0
2005-06-30	Yogurt natural	12	1	0

2005-06-30	Y. fresa semi	12	1	0
2005-06-30	Flan	5	1	7
2005-06-30	Cuajada	7	1	5
2005-06-30	Leche calcio	24	2	0
2005-06-30	Leche entera	24	2	0
2005-06-30	Leche desnatada	24	2	0
2005-06-30	Cuajada	6	1	6
2005-06-30	Queso fresco	3	1	9
2005-06-30	Yogurt limón	12	1	0
2005-06-30	Yogurt fresa	16	2	8
2005-06-30	Y. natural semi	20	2	4
...	...	...	...	...
2005-06-01	Crema catalana	10	1	2
2005-06-01	Leche semi	24	2	0
2005-06-01	Leche desnatada	24	2	0
2005-06-01	Yogurt natural	12	1	0

2005-06-01	Yogurt lim�n	8	1	4
2005-06-01	Y. natural semi	8	1	4
2005-06-01	Y. fresa semi	12	1	0
2005-06-01	Leche entera	24	2	0
2005-06-01	Leche semi	24	2	0
2005-06-01	Leche calcio	12	1	0
2005-06-01	Crema catalana	6	1	6
2005-06-01	Cuajada	10	1	2

```
mysql>CREATE TABLE FACTURAS (
    fecha_fac DATE,
    producto VARCHAR(50),
    unidades INT
);
```

```
mysql> CREATE TABLE FACTURAS (
->     fecha_fac DATE,
->     producto VARCHAR(50),
->     unidades INT
-> );
Query OK, 0 rows affected (0,09 sec)
```

```
mysql>INSERT INTO FACTURAS (fecha_fac, producto, unidades) VALUES
('2005-06-30', 'Y. fresa semi', 8),
('2005-06-30', 'Leche calcio', 12),
('2005-06-30', 'Yogurt natural', 12),
('2005-06-30', 'Y. fresa semi', 12),
('2005-06-30', 'Flan', 5),
('2005-06-30', 'Cuajada', 7),
('2005-06-30', 'Leche calcio', 24),
('2005-06-30', 'Leche entera', 24),
('2005-06-30', 'Leche desnatada', 24),
('2005-06-30', 'Cuajada', 6),
```

```

('2005-06-30', 'Queso fresco', 3),
('2005-06-30', 'Yogurt limçn', 12),
('2005-06-30', 'Yogurt fresa', 16),
('2005-06-30', 'Y. natural semi', 20),
('2005-06-01', 'Crema catalana', 10),
('2005-06-01', 'Leche semi', 24),
('2005-06-01', 'Leche desnatada', 24),
('2005-06-01', 'Yogurt natural', 12),
('2005-06-01', 'Yogurt limçn', 8),
('2005-06-01', 'Y. natural semi', 8),
('2005-06-01', 'Y. fresa semi', 12),
('2005-06-01', 'Leche entera', 24),
('2005-06-01', 'Leche semi', 24),
('2005-06-01', 'Leche calcio', 12),
('2005-06-01', 'Crema catalana', 6),
('2005-06-01', 'Cuajada', 10);

```

```

mysql> INSERT INTO FACTURAS (fecha_fac, producto, unidades) VALUES
-> ('2005-06-30', 'Y. fresa semi', 8),
-> ('2005-06-30', 'Leche calcio', 12),
-> ('2005-06-30', 'Yogurt natural', 12),
-> ('2005-06-30', 'Y. fresa semi', 12),
-> ('2005-06-30', 'Flan', 5),
-> ('2005-06-30', 'Cuajada', 7),
-> ('2005-06-30', 'Leche calcio', 24),
-> ('2005-06-30', 'Leche entera', 24),
-> ('2005-06-30', 'Leche desnatada', 24),
-> ('2005-06-30', 'Cuajada', 6),
-> ('2005-06-30', 'Queso fresco', 3),
-> ('2005-06-30', 'Yogurt limçn', 12),
-> ('2005-06-30', 'Yogurt fresa', 16),
-> ('2005-06-30', 'Y. natural semi', 20),
-> ('2005-06-01', 'Crema catalana', 10),
-> ('2005-06-01', 'Leche semi', 24),
-> ('2005-06-01', 'Leche desnatada', 24),
-> ('2005-06-01', 'Yogurt natural', 12),
-> ('2005-06-01', 'Yogurt limçn', 8),
-> ('2005-06-01', 'Y. natural semi', 8),
-> ('2005-06-01', 'Y. fresa semi', 12),
-> ('2005-06-01', 'Leche entera', 24),
-> ('2005-06-01', 'Leche semi', 24),
-> ('2005-06-01', 'Leche calcio', 12),
-> ('2005-06-01', 'Crema catalana', 6),
-> ('2005-06-01', 'Cuajada', 10);
Query OK, 26 rows affected (0,02 sec)
Records: 26  Duplicates: 0  Warnings: 0

```



```
mysql>SELECT
    fecha_fac,
    producto,
    unidades,
    CEIL(unidades / 12) AS num_paquetes,
    (CEIL(unidades / 12) * 12) - unidades AS hueco_ultimo_paquete
FROM
    FACTURAS
WHERE
    unidades < 36
ORDER BY
    fecha_fac DESC;
```

```
mysql> SELECT
->     fecha_fac,
->     producto,
->     unidades,
->     CEIL(unidades / 12) AS num_paquetes,
->     (CEIL(unidades / 12) * 12) - unidades AS hueco_ultimo_paquete
-> FROM
->     FACTURAS
-> WHERE
->     unidades < 36
-> ORDER BY
->     fecha_fac DESC;
```

fecha_fac	producto	unidades	num_paquetes	hueco_ultimo_paquete
2005-06-30	Leche calcio	12	1	0
2005-06-30	Y. natural semi	20	2	4
2005-06-30	Yogurt fresa	16	2	8
2005-06-30	Yogurt limón	12	1	0
2005-06-30	Queso fresco	3	1	9
2005-06-30	Cuajada	6	1	6
2005-06-30	Leche desnatada	24	2	0
2005-06-30	Leche calcio	24	2	0
2005-06-30	Cuajada	7	1	5
2005-06-30	Flan	5	1	7
2005-06-30	Y. fresa semi	12	1	0
2005-06-30	Yogurt natural	12	1	0
2005-06-30	Y. fresa semi	8	1	4
2005-06-30	Leche entera	24	2	0
2005-06-01	Cuajada	10	1	2
2005-06-01	Crema catalana	6	1	6
2005-06-01	Leche calcio	12	1	0
2005-06-01	Leche semi	24	2	0
2005-06-01	Leche entera	24	2	0
2005-06-01	Y. fresa semi	12	1	0
2005-06-01	Y. natural semi	8	1	4
2005-06-01	Yogurt limón	8	1	4
2005-06-01	Yogurt natural	12	1	0
2005-06-01	Leche desnatada	24	2	0
2005-06-01	Leche semi	24	2	0
2005-06-01	Crema catalana	10	1	2

26 rows in set (0,00 sec)

# Funciones de fecha

En las bases de datos world y bebés:

1. Muestra el nombre, código y número de centenarios cumplidos por todos los países desde su fundación (independencia), que tengan más de tres centenarios, ordenados de mayor a menor número de centenarios del país. Ha de funcionar siempre, por lo que la fecha actual se ha de obtener automáticamente, no escribirse. Ordenar por centenarios. (14 filas)

Name	Code	centenarios	IndepYear
China	CHN	35	-1523
Ethiopia	ETH	30	-1000
Japan	JPN	26	-660
Denmark	DNK	12	800
France	FRA	11	843
Sweden	SWE	11	836
San Marino	SMR	11	885
United Kingdom	GBR	9	1066
Portugal	PRT	8	1143
Andorra	AND	7	1278
Thailand	THA	6	1350
Spain	ESP	5	1492
Switzerland	CHE	5	1499
Netherlands	NLD	4	1581

```
mysql>CREATE TABLE COUNTRY (  
    Name VARCHAR(50),  
    Code CHAR(3),  
    IndepYear INT  
);
```

```
mysql> CREATE TABLE COUNTRY (  
->     Name VARCHAR(50),  
->     Code CHAR(3),  
->     IndepYear INT  
-> );  
Query OK, 0 rows affected (0,12 sec)
```

```
mysql>INSERT INTO COUNTRY (Name, Code, IndepYear) VALUES  
( 'China', 'CHN', -1523),  
( 'Ethiopia', 'ETH', -1000),  
( 'Japan', 'JPN', -660),  
( 'Denmark', 'DNK', 800),  
( 'France', 'FRA', 843),  
( 'Sweden', 'SWE', 836),  
( 'San Marino', 'SMR', 885),  
( 'United Kingdom', 'GBR', 1066),  
( 'Portugal', 'PRT', 1143),  
( 'Andorra', 'AND', 1278),  
( 'Thailand', 'THA', 1350),  
( 'Spain', 'ESP', 1492),  
( 'Switzerland', 'CHE', 1499),  
( 'Netherlands', 'NLD', 1581);
```

```
mysql> INSERT INTO COUNTRY (Name, Code, IndepYear) VALUES  
-> ( 'China', 'CHN', -1523),  
-> ( 'Ethiopia', 'ETH', -1000),  
-> ( 'Japan', 'JPN', -660),  
-> ( 'Denmark', 'DNK', 800),  
-> ( 'France', 'FRA', 843),  
-> ( 'Sweden', 'SWE', 836),  
-> ( 'San Marino', 'SMR', 885),  
-> ( 'United Kingdom', 'GBR', 1066),  
-> ( 'Portugal', 'PRT', 1143),  
-> ( 'Andorra', 'AND', 1278),  
-> ( 'Thailand', 'THA', 1350),  
-> ( 'Spain', 'ESP', 1492),  
-> ( 'Switzerland', 'CHE', 1499),  
-> ( 'Netherlands', 'NLD', 1581);  
Query OK, 14 rows affected (0,02 sec)  
Records: 14  Duplicates: 0  Warnings: 0
```

```
mysql>SELECT
    Name,
    Code,
    FLOOR((YEAR(CURDATE()) - IndepYear) / 100) AS centenarios
FROM
    COUNTRY
WHERE
    FLOOR((YEAR(CURDATE()) - IndepYear) / 100) > 3
ORDER BY
    centenarios DESC;
```

```
mysql> SELECT
->     Name ,
->     Code ,
->     FLOOR((YEAR(CURDATE()) - IndepYear) / 100) AS centenarios
-> FROM
->     COUNTRY
-> WHERE
->     FLOOR((YEAR(CURDATE()) - IndepYear) / 100) > 3
-> ORDER BY
->     centenarios DESC;
```

Name	Code	centenarios
China	CHN	35
Ethiopia	ETH	30
Japan	JPN	26
Denmark	DNK	12
France	FRA	11
Sweden	SWE	11
San Marino	SMR	11
United Kingdom	GBR	9
Portugal	PRT	8
Andorra	AND	7
Thailand	THA	6
Spain	ESP	5
Switzerland	CHE	5
Netherlands	NLD	4

```
14 rows in set (0,00 sec)
```

2. Seleccionar de los nacidos en meses anteriores a mayo pero en días posteriores al 10 de esos meses, el nombre y apellidos y su fecha de nacimiento con el formato indicado en la tabla (7 filas).

nombre	apellido1	nacimiento
Alberto	Ferro	20 de January de 2005
Carmen	Lona	14 de February de 2005
Concepción	Flores	27 de February de 2009
Mónica	Barrios	31 de March de 2005
Miguel	Bendita	17 de March de 2009
Ricardo	Fernández	28 de February de 2003
Rosa	Alba	14 de February de 2005

```
mysql>CREATE TABLE bebes (  
    nombre VARCHAR(50),  
    apellido1 VARCHAR(50),  
    nacimiento DATE  
);
```

```
mysql> CREATE TABLE bebes (  
->     nombre VARCHAR(50),  
->     apellido1 VARCHAR(50),  
->     nacimiento DATE  
-> );  
Query OK, 0 rows affected (0,12 sec)
```

```
mysql>INSERT INTO bebes (nombre, apellido1, nacimiento) VALUES
('Alberto', 'Ferro', '2005-01-20'),
('Carmen', 'Lona', '2005-02-14'),
('Concepción', 'Flores', '2009-02-27'),
('Mónica', 'Barrios', '2005-03-31'),
('Miguel', 'Bendita', '2009-03-17'),
('Ricardo', 'Fernández', '2003-02-28'),
('Rosa', 'Alba', '2005-02-14');
```

```
mysql> INSERT INTO bebes (nombre, apellido1, nacimiento) VALUES
-> ('Alberto', 'Ferro', '2005-01-20'),
-> ('Carmen', 'Lona', '2005-02-14'),
-> ('Concepción', 'Flores', '2009-02-27'),
-> ('Mónica', 'Barrios', '2005-03-31'),
-> ('Miguel', 'Bendita', '2009-03-17'),
-> ('Ricardo', 'Fernández', '2003-02-28'),
-> ('Rosa', 'Alba', '2005-02-14');
Query OK, 7 rows affected (0,02 sec)
Records: 7  Duplicates: 0  Warnings: 0
```

```
mysql>SELECT
    nombre,
    apellido1,
    DATE_FORMAT(nacimiento, '%d de %M de %Y') AS nacimiento
FROM
    bebes
WHERE
    MONTH(nacimiento) < 5
    AND DAY(nacimiento) > 10;
```

```
mysql> SELECT
->     nombre,
->     apellido1,
->     DATE_FORMAT(nacimiento, '%d de %M de %Y') AS nacimiento
-> FROM
->     bebes
-> WHERE
->     MONTH(nacimiento) < 5
->     AND DAY(nacimiento) > 10;
+-----+-----+-----+
| nombre | apellido1 | nacimiento |
+-----+-----+-----+
| Alberto | Ferro | 20 de January de 2005 |
| Carmen | Lona | 14 de February de 2005 |
| Concepción | Flores | 27 de February de 2009 |
| Mónica | Barrios | 31 de March de 2005 |
| Miguel | Bendita | 17 de March de 2009 |
| Ricardo | Fernández | 28 de February de 2003 |
| Rosa | Alba | 14 de February de 2005 |
+-----+-----+-----+
7 rows in set (0,01 sec)
```

2. Mostrar el nombre, primer apellido y día que cumplen la mayoría de edad los que nacieron un miércoles (2 filas)

nombre	apellido1	mayoriaEdad
Rémulo	Fernández	2027-08-19
Sandra	Barata	2026-05-21

```
mysql>INSERT INTO bebes (nombre, apellido1, nacimiento) VALUES
('Rémulo', 'Fernández', '2009-08-19'),
('Sandra', 'Barata', '2008-05-21'),
('Carlos', 'Ruiz', '2005-03-15'),
('Lucía', 'Gómez', '2007-01-22');
```

```
mysql> INSERT INTO bebes (nombre, apellido1, nacimiento) VALUES
-> ('Rémulo', 'Fernández', '2009-08-19'),
-> ('Sandra', 'Barata', '2008-05-21'),
-> ('Carlos', 'Ruiz', '2005-03-15'),
-> ('Lucía', 'Gómez', '2007-01-22');
Query OK, 4 rows affected (0,02 sec)
Records: 4  Duplicates: 0  Warnings: 0
```

```
mysql>SELECT
nombre,
apellido1,
DATE_ADD(nacimiento, INTERVAL 18 YEAR) AS mayoriaEdad
FROM
bebes
WHERE
DAYOFWEEK(nacimiento) = 4;
```

```
mysql> SELECT
-> nombre,
-> apellido1,
-> DATE_ADD(nacimiento, INTERVAL 18 YEAR) AS mayoriaEdad
-> FROM
-> bebes
-> WHERE
-> DAYOFWEEK(nacimiento) = 4;
+-----+-----+-----+
| nombre | apellido1 | mayoriaEdad |
+-----+-----+-----+
| Rémulo | Fernández | 2027-08-19 |
| Sandra | Barata    | 2026-05-21 |
+-----+-----+-----+
2 rows in set (0,00 sec)
```

## Funciones de varchar

Estos ejercicios se realizarán con la base de datos bebés.

1. Mostrar todas las columnas de los nacimientos cuyo nombre y apellidos juntos tengan más de 20 caracteres (4 filas).

Nombre	Apellido1	Apellido2	FechaNac	Provincia
Concepción	Flores	Cruz	2009-02-27	SA
Felipe	Fernández	Gómez	2007-11-13	S
Fernando	Fern ndez	Fern ndez	2005-01-02	AL
Rémulo	Fernández	Alto	2009-08-19	CA

```
mysql>CREATE TABLE bebes (  
    nombre VARCHAR(50),  
    apellido1 VARCHAR(50),  
    apellido2 VARCHAR(50),  
    FechaNac DATE,  
    Provincia CHAR(2)  
);
```

```
mysql> CREATE TABLE bebes (  
->     nombre VARCHAR(50),  
->     apellido1 VARCHAR(50),  
->     apellido2 VARCHAR(50),  
->     FechaNac DATE,  
->     Provincia CHAR(2)  
-> );  
Query OK, 0 rows affected (0,96 sec)
```



```
mysql>INSERT INTO bebes (nombre, apellido1, apellido2, FechaNac, Provincia) VALUES  
( 'Concepción', 'Flores', 'Cruz', '2009-02-27', 'SA'),  
( 'Felipe', 'Fernández', 'Gómez', '2007-11-13', 'S'),  
( 'Fernando', 'Fernández', 'Fernández', '2005-01-02', 'AL'),  
( 'Rémulo', 'Fernández', 'Alto', '2009-08-19', 'CA'),  
( 'Luis', 'Paz', 'Gil', '2006-04-10', 'BI');
```

```
mysql> INSERT INTO bebes (nombre, apellido1, apellido2, FechaNac, Provincia) VALUES  
-> ( 'Concepción', 'Flores', 'Cruz', '2009-02-27', 'SA'),  
-> ( 'Felipe', 'Fernández', 'Gómez', '2007-11-13', 'S'),  
-> ( 'Fernando', 'Fernández', 'Fernández', '2005-01-02', 'AL'),  
-> ( 'Rémulo', 'Fernández', 'Alto', '2009-08-19', 'CA'),  
-> ( 'Luis', 'Paz', 'Gil', '2006-04-10', 'BI');  
Query OK, 5 rows affected (0,01 sec)  
Records: 5 Duplicates: 0 Warnings: 0
```

```
mysql>SELECT *  
FROM bebes  
WHERE LENGTH(CONCAT(nombre, apellido1, apellido2)) > 20;
```

```
mysql> SELECT *  
-> FROM bebes  
-> WHERE LENGTH(CONCAT(nombre, apellido1, apellido2)) > 20;  
+-----+-----+-----+-----+-----+  
| nombre      | apellido1 | apellido2 | FechaNac | Provincia |  
+-----+-----+-----+-----+-----+  
| Concepción  | Flores    | Cruz       | 2009-02-27 | SA        |  
| Felipe      | Fernández | Gómez      | 2007-11-13 | S         |  
| Fernando    | Fernández | Fernández  | 2005-01-02 | AL        |  
| Rémulo      | Fernández | Alto       | 2009-08-19 | CA        |  
+-----+-----+-----+-----+-----+  
4 rows in set (0,00 sec)
```

2. Mostrar la inicial del nombre y el primer apellido separado por espacio de aquellos que tienen una letra "b" en su apellido pero después de la segunda posición (4 filas).

Nacido
I Alba
M Robles
R Robles
R Alba

```
mysql>INSERT INTO bebes (nombre, apellido1) VALUES  
('Isabel', 'Alba'),  
('Marcos', 'Robles'),  
('Ramón', 'Robles'),  
('Rosa', 'Alba'),  
('Luis', 'Bravo');
```

```
mysql> INSERT INTO bebes (nombre, apellido1) VALUES  
-> ('Isabel', 'Alba'),  
-> ('Marcos', 'Robles'),  
-> ('Ramón', 'Robles'),  
-> ('Rosa', 'Alba'),  
-> ('Luis', 'Bravo');  
Query OK, 5 rows affected (0,88 sec)  
Records: 5 Duplicates: 0 Warnings: 0
```

```
mysql>SELECT CONCAT(LEFT(nombre, 1), ' ', apellido1) AS inicial_y_apellido FROM  
bebes WHERE LOCATE('b', apellido1) > 2 LIMIT 4;
```

```
mysql> SELECT CONCAT(LEFT(nombre, 1), ' ', apellido1) AS inicial_y_apellido FROM bebes WHERE LOCATE('b', apellido1) > 2 LIMIT 4;  
+-----+  
| inicial_y_apellido |  
+-----+  
| I Alba             |  
| M Robles           |  
| R Robles           |  
| R Alba             |  
+-----+  
4 rows in set (0,00 sec)
```

3. Mostrar nombre y apellidos separados por espacios, pero solo el texto anterior a la segunda "e" en los tres campos juntos, de aquellos que en nombre y apellidos tiene al menos dos "e" (14 filas).

nacido
Alberto F
Concepción Flor
Daniel Lóp
Elisa Alonso López
Felip
Fernando F
Isabel Alonso Lóp
Manuel Robl
Miguel B
Raquel Robl
Rémulo Fernánd
Ricardo Fern nd
Roberto Barata S nch
Sandra Barata Rey

```
mysql>SELECT CONCAT(LEFT(nombre, LOCATE('e', nombre, LOCATE('e', nombre) + 1) - 1), ' ',
LEFT(apellido1, LOCATE('e', apellido1, LOCATE('e', apellido1) + 1) - 1), ' ', LEFT(apellido2, LOCATE('e', apellido2,
LOCATE('e', apellido2) + 1) - 1)) AS texto_antes_de_2e FROM bebes WHERE
LENGTH(REPLACE(CONCAT(nombre, apellido1, apellido2), 'e', '')) < LENGTH(CONCAT(nombre, apellido1,
apellido2)) - 1 LIMIT 14;
```

```
mysql> SELECT CONCAT(LEFT(nombre, LOCATE('e', nombre, LOCATE('e', nombre) + 1) - 1), ' ', LEFT(apellido1, LOCATE('e', apellido1, LOCATE('e', apellido1) + 1) - 1), ' ', LEFT(apellido2, LOCATE('e', apellido2, LOCATE('e', apellido2) + 1) - 1)) AS texto_antes_de_2e FROM bebes WHERE LENGTH(REPLACE(CONCAT(nombre, apellido1, apellido2), 'e', '')) < LENGTH(CONCAT(nombre, apellido1, apellido2)) - 1 LIMIT 14;
+-----+
| texto_antes_de_2e |
+-----+
| Felip Fernánd    |
| Fernánd Fernánd  |
| Fernánd          |
+-----+
4 rows in set (0,00 sec)
```