

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> 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, 53);
```

```
mysql> insert into Puertos2 values('Puerto de la Morcuera', 19, 1760, 53);  
Query OK, 1 row affected (0,01 sec)
```

```
mysql> INSERT INTO Puertos2 VALUES('Coll de Ordino', 15, 1980, 37);
```

```
mysql> insert into Puertos2 values('Coll de Ordino', 15, 1980, 37);  
Query OK, 1 row affected (0,01 sec)
```

```
mysql> INSERT INTO Puertos2 VALUES('Puerto de Mijares', 2, 1525, 47);
```

```
mysql> insert into Puertos2 values('Puerto de Mijares', 2, 1525, 47);  
Query OK, 1 row affected (0,01 sec)
```

```
mysql> INSERT INTO Puertos2 VALUES('Navacerrada', 11, 1860, 29);
```

```
mysql> insert into Puertos2 values('Navacerrada', 11, 1860, 29);  
Query OK, 1 row affected (0,02 sec)
```

```
mysql> INSERT INTO Puertos2 VALUES('Cruz de la Demanda', 16, 1850, 21);
```

```
mysql> insert into Puertos2 values('Cruz de la Demanda', 16, 1850, 21);  
Query OK, 1 row affected (0,01 sec)
```

```
mysql> INSERT INTO Puertos2 VALUES('Coll de la Comella', 11, 1860, 29);
```

```
mysql> insert into Puertos2 values('Coll de la Comella', 11, 1860, 29);  
Query OK, 1 row affected (0,01 sec)
```

```
mysql> INSERT INTO Puertos2 VALUES('Cerler-Circo de Ampriu', 18, 2500, 81);
```

```
mysql> insert into Puertos2 values('Cerler-Circo de Ampriu', 18, 2500, 81);  
Query OK, 1 row affected (0,01 sec)
```

```
mysql> INSERT INTO Puertos2 VALUES('Arcalis', 1, 2230, 75);
```

```
mysql> insert into Puertos2 values('Arcalis', 1, 2230, 75);  
Query OK, 1 row affected (0,01 sec)
```

```
mysql> INSERT INTO Puertos2 VALUES('Sierra Nevada', 8, 2500, 73);
```

```
mysql> insert into Puertos2 values('Sierra Nevada', 8, 2500, 73);  
Query OK, 1 row affected (0,01 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 |  
+-----+-----+  
| Cruz de la Demanda | 88.1 |  
| Sierra Nevada | 34.2 |  
| Puerto de Mijares | 32.4 |  
| Cerler-Circo de Ampriu | 30.9 |  
+-----+-----+  
4 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> SELECT lugar_salida, (ganador_dorsal - id_etapa) AS diferencia_dorsal_etapa, distancia_km FROM etapas WHERE SQRT(distancia_km) < 13 LIMIT 7;

```
mysql> SELECT lugar_salida, (ganador_dorsal - id_etapa) AS diferencia_dorsal_etapa, distancia_km FROM etapas WHERE SQRT(distancia_km) < 13 LIMIT 7;
+-----+-----+-----+
| lugar_salida | diferencia_dorsal_etapa | distancia_km |
+-----+-----+-----+
| Madrid      | 4 | 100.00 |
| Barcelona   | 6 | 120.00 |
| Valencia    | 9 | 130.00 |
| Sevilla     | 6 | 110.00 |
| Bilbao      | 4 | 85.00  |
| Zaragoza    | 9 | 135.00 |
| Granada     | -1 | 95.00  |
+-----+-----+-----+
7 rows in set (0,01 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 20 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 limón	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> SELECT fecha_factura, producto, unidades, CEIL(unidades / 12) AS
numero_paquetes, 12 - (unidades % 12) AS huecos_ultimo_paquete FROM FACTURAS
WHERE unidades < 36 ORDER BY fecha_factura DESC LIMIT 20;
```

```
mysql> SELECT fecha_factura, producto, unidades, CEIL(unidades / 12) AS numero_paquetes, 12 - (unidades % 12) AS huecos_ultimo_paquete FROM FACTURAS WHERE unidades < 36 ORDER BY fecha_factura DESC LIMIT 20;
```

fecha_factura	producto	unidades	numero_paquetes	huecos_ultimo_paquete
2025-04-23	Producto A	30	3	6
2025-04-20	Producto B	24	2	12
2025-04-18	Producto C	15	2	9
2025-04-15	Producto D	10	1	2
2025-04-13	Producto E	34	3	2
2025-04-10	Producto F	11	1	1
2025-04-08	Producto G	5	1	7
2025-04-05	Producto H	3	1	9
2025-04-02	Producto I	29	3	7
2025-04-01	Producto J	17	2	7
2025-03-30	Producto K	20	2	4
2025-03-28	Producto L	18	2	6
2025-03-25	Producto M	13	2	11
2025-03-22	Producto N	9	1	3
2025-03-20	Producto O	2	1	10
2025-03-18	Producto P	33	3	3
2025-03-15	Producto Q	12	1	12
2025-03-12	Producto R	22	2	2
2025-03-10	Producto S	8	1	4
2025-03-08	Producto T	28	3	8

```
20 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> SELECT nombre, codigo, FLOOR(DATEDIFF(CURDATE(),
fecha_independencia) / 365.25 / 100) AS centenarios FROM paises WHERE
FLOOR(DATEDIFF(CURDATE(), fecha_independencia) / 365.25 / 100) ORDER BY
centenarios DESC;
```

```
mysql> SELECT nombre, codigo, FLOOR(DATEDIFF(CURDATE(), fecha_independencia) / 365.25 / 100) AS centenarios FROM paises WHERE FLOOR(
DATEDIFF(CURDATE(), fecha_independencia) / 365.25 / 100) ORDER BY centenarios DESC;
+-----+-----+-----+
| nombre | codigo | centenarios |
+-----+-----+-----+
| México | MX | 2 |
| Estados Unidos | US | 2 |
| Brasil | BR | 2 |
| Argentina | AR | 2 |
| Colombia | CO | 2 |
| Chile | CL | 2 |
| Perú | PE | 2 |
| Francia | FR | 2 |
| Canadá | CA | 1 |
| Australia | AU | 1 |
| Alemania | DE | 1 |
| Italia | IT | 1 |
+-----+-----+-----+
12 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> SELECT nombre, apellidos, DATE_FORMAT(fecha_nacimiento, '%d/%m/%Y')
AS fecha_nacimiento_formateada FROM personas WHERE MONTH(fecha_nacimiento)
< 5 AND DAY(fecha_nacimiento) > 10;
```

```
mysql> SELECT nombre, apellidos, DATE_FORMAT(fecha_nacimiento, '%d/%m/%Y') AS fecha_nacimiento_formateada FROM personas WHERE MONTH(
fecha_nacimiento) < 5 AND DAY(fecha_nacimiento) > 10;
+-----+-----+-----+
| nombre | apellidos      | fecha_nacimiento_formateada |
+-----+-----+-----+
| Ana    | López García  | 15/01/1990                 |
| Carlos | Martínez Ruiz | 20/02/1985                 |
| Lucía  | Pérez Díaz    | 12/03/1992                 |
| Jorge  | Fernández Gómez | 18/04/1988                 |
| Sofía  | Ramírez Torres | 25/03/1995                 |
| David  | Sánchez Ortega | 22/01/1991                 |
| María  | González Navarro | 11/04/1993                 |
+-----+-----+-----+
7 rows in set (0,00 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> SELECT nombre, SUBSTRING_INDEX(apellidos, ' ', 1) AS primer_apellido,
DATE_ADD(fecha_nacimiento, INTERVAL 18 YEAR) AS fecha_mayoria_edad FROM
personas WHERE DAYOFWEEK(fecha_nacimiento) = 4 LIMIT 2;
```

```
mysql> SELECT nombre, SUBSTRING_INDEX(apellidos, ' ', 1) AS primer_apellido, DATE_ADD(fecha_nacimiento, INTERVAL 18 YEAR) AS fecha_m
ayoria_edad FROM personas WHERE DAYOFWEEK(fecha_nacimiento) = 4 LIMIT 2;
+-----+-----+-----+
| nombre | primer_apellido | fecha_mayoria_edad |
+-----+-----+-----+
| Carlos | Martínez        | 2003-02-20         |
+-----+-----+-----+
1 row in set (0,00 sec)
```

Funciones de varcharA

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

```
mysql> select * from bebes;
+-----+-----+-----+-----+
| nombre | apellido1 | apellido2 | fecha_nacimiento |
+-----+-----+-----+-----+
| Alejandro | Martínez | Gómez | 2024-01-15 |
| Luz | Díaz | Paz | 2024-03-08 |
| Camila | Rodríguez | López | 2024-02-20 |
| Julián | Hernández | Pardo | 2024-04-12 |
| Valentina | Suárez | Delgado | 2024-01-30 |
+-----+-----+-----+-----+
5 rows in set (0,00 sec)

mysql> SELECT * FROM bebes WHERE CHAR_LENGTH(CONCAT(nombre, apellido1, apellido2)) > 20;
+-----+-----+-----+-----+
| nombre | apellido1 | apellido2 | fecha_nacimiento |
+-----+-----+-----+-----+
| Alejandro | Martínez | Gómez | 2024-01-15 |
| Valentina | Suárez | Delgado | 2024-01-30 |
+-----+-----+-----+-----+
2 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

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

```
mysql> select * from bebes;  
+-----+-----+-----+-----+  
| nombre | apellido1 | apellido2 | fecha_nacimiento |  
+-----+-----+-----+-----+  
| Alejandro | Martínez | Gómez | 2024-01-15 |  
| Luz | Díaz | Paz | 2024-03-08 |  
| Camila | Rodríguez | López | 2024-02-20 |  
| Julián | Hernández | Pardo | 2024-04-12 |  
| Valentina | Suárez | Delgado | 2024-01-30 |  
| Bruno | Alberdi | Sánchez | 2024-02-10 |  
| Isabela | Robles | Mendoza | 2024-01-22 |  
| Gabriel | Corbalán | Reyes | 2024-03-18 |  
+-----+-----+-----+-----+  
8 rows in set (0,00 sec)  
  
mysql> SELECT CONCAT(LEFT(nombre, 1), ' ', apellido1) AS inicial_y_apellido FROM bebes WHERE LOCATE('b', apellido1) > 2 LIMIT 4;  
+-----+  
| inicial_y_apellido |  
+-----+  
| B Alberdi |  
| I Robles |  
| G Corbalán |  
+-----+  
3 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 Lçp
Manuel Robl
Miguel B
Raquel Robl
Rémulo Fernánd
Ricardo Fern nd
Roberto Barata S nch
Sandra Barata Rey

```
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
Hernánd
Rey

7 rows in set (0,00 sec)