

## Funciones para trabajar con horas

En el documento anterior vimos algunas funciones para manejar fechas, pues ahora ha llegado el momento de conocer otras para manipular las horas. Os recuerdo los enlaces que se pueden consultar para tener más información sobre estos tipos de datos.

Hay distintos tipos en MySQL para manejar las horas (TIME, DATETIME, TIMESTAMP, YEAR), algunos de los cuales también almacenan la fecha y que se comportan de manera distinta cada uno de ellos. Para saber sus peculiaridades podemos acudir a la documentación oficial sobre MySQL disponible en español o más actualizada en inglés, en los enlaces siguientes y en el orden citado:

- <http://download.nust.na/pub6/mysql/doc/refman/5.0/es/time.html>  
<https://dev.mysql.com/doc/refman/8.0/en/time.html>

En este enlace: <http://mysql.conclase.net/curso/?sqlfun=MINUTE> podemos ver la explicación y el uso de muchas de estas funciones pero no vamos a necesitar todas ahora, y puede que nunca hagamos uso de algunas de ellas, así que vamos a centrarnos en unas cuantas de esas funciones que nos pueden ser de utilidad para abordar consultas en las que estén implicadas horas.

Al igual que con las fechas manejadas por MySQL se cita los periodos de tiempo de mayor a menor (año>mes>día), con las horas funciona el mismo principio de referirse primero al periodo mayor y en último lugar al más pequeño y se ponen horas:minutos:segundos; esto nos resulta más natural que la manera de almacenar las fechas porque en España también se escriben las horas así y no tenemos que recordar usar un formato distinto del habitual, a diferencia de lo que ocurre con las fechas.

**HOUR():** devuelve la hora de una expresión *TIME*.

The screenshot shows two examples of the `HOUR()` function in a MySQL query editor. The first example, labeled 131, shows a query `SELECT HOUR('15:10:37');` with a result grid showing the value 15. The second example, labeled 132, shows a query `SELECT HOUR(HoraInicio) FROM Alquileres WHERE ID_Alquiler='1018';` with a result grid showing the value 8.

```
131 • SELECT HOUR('15:10:37');
```

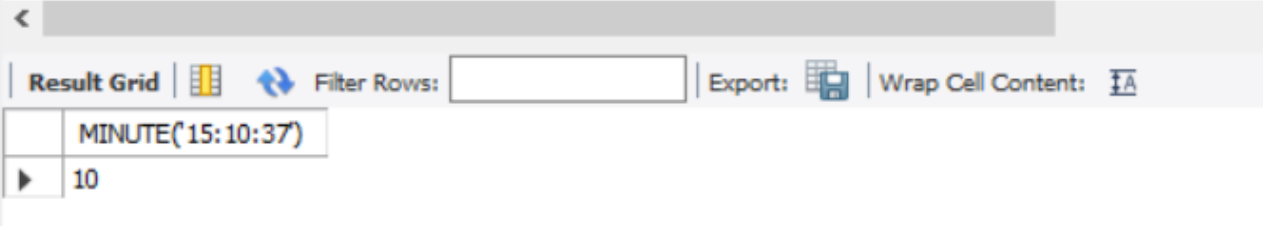
	HOUR('15:10:37')
▶	15

```
132 • SELECT HOUR(HoraInicio) FROM Alquileres WHERE ID_Alquiler='1018';
```

	HOUR(HoraInicio)
▶	8

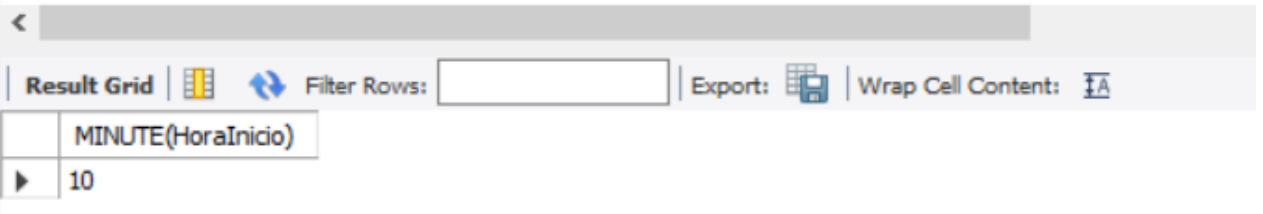
**MINUTE()**: funcionamiento análogo a HOUR() pero con los minutos:

```
134      -- MINUTE(): devuelve los minutos de la hora pasada por parámetro
135 •    SELECT MINUTE('15:10:37');
```



The screenshot shows a SQL query result grid. The header row contains the expression `MINUTE('15:10:37')`. The first data row shows the result `10`. The interface includes a 'Result Grid' tab, a 'Filter Rows' input field, and buttons for 'Export' and 'Wrap Cell Content'.

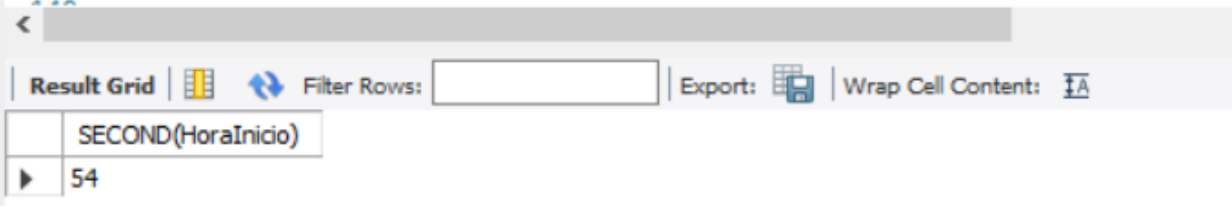
```
136 •    SELECT MINUTE(HoraInicio) FROM Alquileres WHERE ID_Alquiler='1018';
```



The screenshot shows a SQL query result grid. The header row contains the expression `MINUTE(HoraInicio)`. The first data row shows the result `10`. The interface includes a 'Result Grid' tab, a 'Filter Rows' input field, and buttons for 'Export' and 'Wrap Cell Content'.

**SECOND()**: como podíamos suponer, también nos devuelve los segundos de la hora pasada como parámetro:

```
138      -- SECOND(): devuelve los segundos de la hora pasada por parámetro
139 •    SELECT SECOND('15:10:37');
```



The screenshot shows a SQL query result grid. The header row contains the expression `SECOND(HoraInicio)`. The first data row shows the result `54`. The interface includes a 'Result Grid' tab, a 'Filter Rows' input field, and buttons for 'Export' and 'Wrap Cell Content'.

```
140 •    SELECT SECOND(HoraInicio) FROM Alquileres WHERE ID_Alquiler='1018';
```



The screenshot shows a SQL query result grid. The header row contains the expression `SECOND(HoraInicio)`. The first data row shows the result `54`. The interface includes a 'Result Grid' tab, a 'Filter Rows' input field, and buttons for 'Export' and 'Wrap Cell Content'.

**TIMEDIFF()**: esta función devuelve la diferencia entre dos valores de tiempo, estos valores pueden ser de tipo *TIME*, *TIMESTAMP* o *DATE*, pero ambos deben ser del mismo tipo:

```
141  -- TIMEDIFF(): devuelve la diferencia entre dos valores de tiempo
142  •  SELECT HoraInicio, HoraFin, TIMEDIFF(HoraFin, HoraInicio) AS TiempoAlquiler
143      FROM Alquileres WHERE ID_Alquiler='1018';
...
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content: |

	HoraInicio	HoraFin	TiempoAlquiler
▶	08:10:54	18:50:45	10:39:51

```
142  •  SELECT FechaInicio, FechaFin, TIMEDIFF(FechaFin, FechaInicio) AS TiempoAlquiler
143      FROM Alquileres WHERE ID_Alquiler='1018';
...
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content: |

	FechaInicio	FechaFin	TiempoAlquiler
▶	2020-04-03	2020-04-03	00:00:00

**TIME()**: devuelve la parte de la hora (hh:mm:ss) de una expresión de fecha y hora:

```
145  -- TIME()
146  •  SELECT TIME('2020-05-05 23:10:54');
```

Result Grid | Filter Rows: | Export: |

	TIME('2020-05-05 23:10:54')
▶	23:10:54

**NOW()**: devuelve la fecha y hora actual:

```
148  -- NOW()
149  •  SELECT NOW();
```

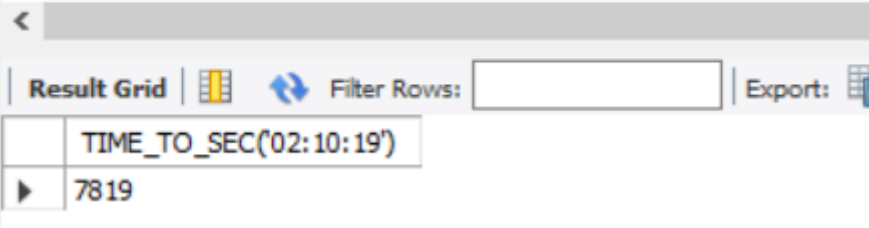
Result Grid | Filter Rows: | Export: |

	NOW()
▶	2020-02-10 12:53:04

**CURRENT\_TIMESTAMP**, **CURRENT\_TIMESTAMP()**, **LOCALTIME**, **LOCALTIME()**, **LOCALTIMESTAMP**, **LOCALTIMESTAMP()** y **SYSDATE()** son sinónimos de **NOW()** y nos darán el mismo resultado.

**TIME\_TO\_SEC()**: retorna el número de segundos que hay en el tiempo (hh:mm:ss) pasado como parámetro:

```
151      -- TIME_TO_SEC()
152 •    SELECT TIME_TO_SEC('02:10:19');
```



The screenshot shows a SQL query execution interface. At the top, the query is: `151 -- TIME_TO_SEC()` and `152 • SELECT TIME_TO_SEC('02:10:19');`. Below the query, there is a toolbar with icons for 'Result Grid', 'Filter Rows', and 'Export'. The 'Result Grid' is active, showing a table with one column and one row. The column header is `TIME_TO_SEC('02:10:19')` and the value in the row is `7819`.

TIME_TO_SEC('02:10:19')
7819

Además de estas funciones también nos van a ser de ayuda operadores ya conocidos como **BETWEEN** o los relacionales (<, >, <>, =) para tratar con este tipo de datos, al igual que sucedía con las fechas.