

DEFINICIÓN Y MANIPULACIÓN

CREATE [TEMPORARY] TABLE [IF NOT EXISTS] <i>nomtabla</i> (columnas, restricciones) [engine=innodb]	tipos	char(<i>n</i>), varchar(<i>n</i>), int, decimal(<i>d</i> , <i>c</i>), date, datetime
	columna	nombre tipo [restricción]
	restricciones	[not] null primary key (<i>x</i>) foreign key (<i>x</i>) references T(<i>y</i>) [on {update delete} {set null cascade}] unique (<i>x</i>)

DROP [TEMPORARY] TABLE [IF EXISTS] *tabla*
INSERT INTO *tabla* [(listaColumnas)] VALUES (listaValores)
INSERT INTO *tabla* [(listaColumnas)] ordenSelect
DELETE FROM *tabla* [WHERE condición]
UPDATE *tabla* SET asignaciones [WHERE condición]

SELECT [DISTINCT] listaColumnas
FROM listaTablas
[[LEFT | RIGHT | NATURAL] JOIN *tabla* ON condición]
[WHERE condición]
[GROUP BY listaColumnas [HAVING condición]]
[ORDER BY columna [ASC | DESC][, columna [ASC | DESC]]

RESUMEN DE OPERADORES

Operadores y comparaciones:

=	AND	BETWEEN	LIKE {%,_}	UNION	op ALL	+, -, *, /
>	OR	IN		UNION ALL	op SOME	DIV
<	NOT	EXISTS			op ANY	%, MOD
>=					(op=comparación)	
<=						
<>, !=						
IS NULL						

columna IS [NOT] NULL
expresión [NOT] BETWEEN *expresión* AND *expresión*
expresión [NOT] IN { (listaValores) | consulta }
columna [NOT] LIKE *patrón*

Patrón es una cadena de caracteres que usa los comodines % y _ , cualquier cadena de caracteres, y cualquier carácter.

FUNCIONES DE AGREGACIÓN

COUNT(*)	cantidad de filas
COUNT([DISTINCT] <i>expr</i>)	cantidad de valores en la columna <i>expr</i>
SUM([DISTINCT] <i>expr</i>)	suma de todos los valores en <i>expr</i>
AVG([DISTINCT] <i>expr</i>)	promedio de todos los valores en <i>expr</i>
MIN(<i>expr</i>)	el más pequeño de todos los valores en <i>expr</i>
MAX(<i>expr</i>)	el mayor de todos los valores en <i>expr</i>
STDDEV_POP(<i>expr</i>)	desviación estándar

FUNCIONES DE FECHA

NOW()	la fecha del sistema
DATE_FORMAT(<i>d</i> , <i>f</i>)	convierte la fecha <i>d</i> en un string con el formato <i>f</i>
STR_TO_DATE(<i>d</i> , <i>f</i>)	convierte el string <i>d</i> con formato <i>f</i> al tipo fecha
DAY(<i>d</i>), DAYOFWEEK(<i>d</i>), DAYNAME(<i>d</i>), MONTH(<i>d</i>), YEAR(<i>d</i>), HOUR(<i>d</i>), MINUTE(<i>d</i>), SECOND(<i>d</i>)	devuelven partes de una fecha <i>d</i>
DATEDIFF(<i>d1</i> , <i>d2</i>)	número de días entre las fechas <i>d1</i> y <i>d2</i>
DATE_ADD(<i>d</i> ,INTERVAL <i>x</i> intrl)	añade a la fecha <i>d</i> la cantidad <i>x</i> de años, meses, etc. (intrl = YEAR, MONTH, WEEK, DAY, HOUR, MINUTE, SECOND) select date_add ('2019-01-01', INTERVAL 3 DAY)

FORMATOS DE FECHA

Especificador	Descripción		
%a	Día de semana abreviado (Sun..Sat)	%l	Hora (1..12)
%b	Mes abreviado (Jan..Dec)	%M	Nombre mes (January..December)
%c	Mes, numérico (0..12)	%m	Mes, numérico (00..12)
%d	Día del mes numérico (00..31)	%s	Segundos (00..59)
%H	Hora (00..23)	%u	Semana (00..53), lunes primer día de la semana
%h	Hora (01..12)	%W	Nombre día semana (Sunday..Saturday)
%I	Hora (01..12)	%w	Día de la semana (0=Sunday..6=Saturday)
%i	Minutos, numérico (00..59)	%Y	Año, numérico, cuatro dígitos
%k	Hora (0..23)	%y	Año, numérico (dos dígitos)

OTRAS FUNCIONES

ROUND(<i>expr</i> , <i>n</i>), TRUNCATE(<i>expr</i> , <i>n</i>)	redondea/trunca <i>expr</i> a <i>n</i> decimales
ABS(<i>expr</i>)	valor absoluto de <i>expr</i>
MOD(<i>N</i> , <i>M</i>)	residuo de <i>N</i> / <i>M</i>
CONCAT(str1,str2,...)	concatena cadenas de caracteres

EJEMPLOS

create table T (c1 int, c2 varchar(25) not null,
primary key (c1), foreign key (c2) references
V(x) on update cascade;
drop table T;
insert into T (c1, c2) values (1,'Hello');
insert into T values (1,'Hello');
insert into T (c1, c2) (select x, y from V);
delete from T where c1 = 1;
update T set c1=2, c2='Goodbye' where c3>30;
select x from T where z like 'V_Z%';
select x,y from T order by x DESC,y DESC;
select x,y from T order by 2;
select col1 alias1,col2 alias3 from table t;
select 'Hello',x, 'goodbye',y from T;

select t1.x, y, **date_format(now()),'%d/%m/%Y'**
from tabla1 t1, tabla2 t2 where t1.x=t2.x
select x, count(*) from T where y=a
group by x having sum(z) = 100;
select x from T where **z = select max(y) from T2;**
select x from T **union** select y from T2;

select x, funcion(b) from T group by x
having sum(z) >= **ALL (select ...);**
select x, funcion(b) from T where **exists** (select ...);

select x from T **left join** S on (T.a = S.b)