

Módulo: Bases de Datos
UT03: Diseño Físico de Base de dato
Actividad de evaluación (Supuesto práctico – RA2)

NOMBRE:

Tipo de datos	Descripción	Rango de valores		Almacenamiento requerido
		Con signo	Sin signo	
TINYINT[(N)]	Entero muy pequeño	-128 a 127	0 a 255	1 byte
SMALLINT[(N)]	Entero pequeño	-32.768 a 32.767	0 a 65.535	2 bytes
MEDIUMINT[(N)]	Entero mediano	-8.388.608 a 8.388.607	0 a 16.777.215	3 bytes
INT[(N)] INTEGER[(N)]	Entero estándar	-2.147.683.648 a 2.147.483.647	0 a 4.294.967.295	4 bytes
BIGINT[(N)]	Entero largo	-9.223.372.036.854.775.808 a 9.223.372.036.854.775.807	0 a 18.446.744.073.709.551.615	8 bytes
FLOAT[(N,D)]	Número pequeño de simple precisión de coma flotante	Valor mínimo distinto de 0 ±1,175.494.351E-38s	Valor máximo distinto de 0 ±3,402.823.466E+38s	4 bytes
DOUBLE[(N,D)] REAL[(N,D)]	Número grande de doble de precisión de coma flotante	Valor mínimo distinto de 0 ±2,2.250.738.585.072.014 E-308	Valor máximo distinto de 0 ±1,7.976.931.348.623.157 E+308	8 bytes
DECIMAL[(N[,D))] NUMERIC[(N[,D))] DEC[(N[,D))]	Número de coma flotante, representado como una cadena	Varios, el rango depende de N y D		N+2 bytes

N : Tamaño máximo de pantalla (precisión)

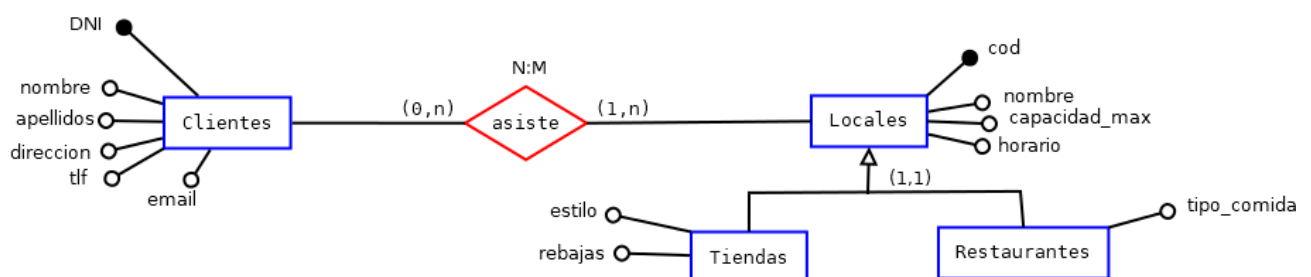
D : Número de decimales (escala)

Tipo	Descripción	Rango	Almacenamiento
CHAR(N)	Cadena de longitud fija. Se rellena con espacios.	De 1 a 255 caracteres	N bytes
VARCHAR(N)	Cadena de longitud variable.	De 1 a 255 caracteres	L+1 bytes
TINYBLOB BLOB MEDIUMBLOB LONGBLOB	Pequeño valor BLOB BLOB Normal BLOB Medio BLOB Grande	$L < 2^8$ $L < 2^{16}$ $L < 2^{24}$ $L < 2^{32}$	L+1 bytes L+2 bytes L+3 bytes L+4 bytes
TINYTEXT TEXT MEDIUMTEXT LONGTEXT	Pequeño valor TEXT TEXT Normal TEXT Medio TEXT Grande	$L < 2^8$ $L < 2^{16}$ $L < 2^{24}$ $L < 2^{32}$	L+1 bytes L+2 bytes L+3 bytes L+4 bytes
ENUM("valor1", "valor2",...)	Se le asigna un valor del conjunto enunciado	Número máximo de valores: 65.535	1 ó 2 bytes
SET("valor1","valor2",...)	Se le asigna 0 o más valores del conjunto	Número máximo de valores: 64	1,2,3,4, u 8 bytes

Tipo	Descripcion	Rango	Almacenamiento
DATE	Representa una fecha en formato 'AAAA-MM-DD'	"1000-01-01" a "9999-12-31"	3 bytes
TIME	Representa una hora en formato: 'HH:MM:SS'	"-838:59:59" a "838:59:59"	3 bytes
DATETIME	Representa una combinación de fecha y hora en formato: 'AAAA-MM-DD HH:MM:SS'	"1000-01-01 00:00:00" a "9999-12-31 23:59:59"	8 bytes
TIMESTAMP	Representa una combinación de fecha y hora en formato: 'AAAAMMDD HHMMSS'	19700101000000 a cualquier fecha del año 2037	4 bytes
YEAR[(2 4)]	Representa un año en formato: AAAA	1901 a 2155	1 byte

ENUNCIADO

Los Hermanos Domínguez quieren construir otro centro comercial. Para desarrollar su web y una aplicación móvil se requiere diseñar y crear su base de datos. Alguien muy amable nos ha facilitado el diagrama E-R y su diseño lógico:



Clientes(DNI, nombre, apellidos, direccion, tlf, email);
 Locales(cod, nombre, capacidad_max, horario);
 Tiendas(cod*, estilo, rebajas);
 cod hace referencia a Locales.cod, no admite nulo
 Restaurantes(cod*, tipo_comida);
 cod hace referencia a Locales.cod, no admite nulo
 Clientes_locales(DNI*, cod*);
 cod hace referencia a Locales.cod, no admite nulo
 DNI hace referencia a Clientes.DNI, no admite nulo

NOTA: En cada ejercicio se debe adjuntar la sentencia que resuelve el ejercicio y una captura de pantalla una vez ejecutada dicha sentencia.

DADAS LAS SIGUIENTES TABLAS:

TABLA Clientes:

Nombre de campo	¿requerido? (no admite null)	Valor por defecto	Clave	Descripción
DNI	SÍ	-----	Primaria	Siempre 9 caracteres
nombre	SÍ	-----		Máximo 20 caracteres
apellidos	NO	-----		Máximo 20 caracteres
direccion	NO	‘c/ La felicidad se encuentra entre nosotros, 1’		Máximo 100 caracteres
tlf	NO	-----		Numero positivo menor a 700000000. (no incluir el carácter ‘+’)
email	SÍ	-----		Máximo 70 caracteres

TABLA Locales:

Nombre de campo	¿requerido? (no admite null)	Valor por defecto	Clave	Descripción
cod	SÍ	-----	Primaria	Auto_incremental
nombre	SÍ	-----		Máximo 20 caracteres
capacidad_max	SÍ	10		Debe ser un número positivo y menor a 150
horario		‘Lunes-Viernes de 09:00 a 22:00’		Máximo 50 caracteres

**** Si el código tiene menos de 2 dígitos se rellenará con ceros sólo hasta el 2º dígito**

TABLA Tiendas:

Nombre de campo	¿requerido? (no admite null)	Valor por defecto	Clave	Descripción
cod	SÍ	-----	Primaria/ Foránea	Auto_incremental
estilo	SÍ	-----		Tendrá que tener uno o varios entre el siguiente conjunto: Deportivo, Casual, Urbano, Punk, Étnico y Gótico.
rebajas	SÍ	0		Debe marcar si hay rebajas o no

TABLA Restaurantes:

Nombre de campo	¿requerido? (no admite null)	Valor por defecto	Clave	Descripción
cod	SÍ	-----	Primaria/ Foránea	Auto_incremental
tipo_comida	SÍ	-----		Tendrá que tener uno entre el siguiente conjunto: Canaria, Italiana, India, China, Japonesa, Mexicana, Vegana

REALIZAR:

Adjunta la sentencia para resolver el ejercicio y una captura de pantalla una vez realizada la misma.

1. Si existe elimina la base de datos 'centro_comercial'. Utiliza la consola

(0,25 puntos)

```
mysql> drop database if exists centro_comercial;  
Query OK, 0 rows affected, 1 warning (0.00 sec)
```

2. Crea la base de datos 'centro_comercial' estableciendo un charset=latin1 y collate por defecto. Utiliza la consola

(0,25 puntos)

```
mysql> create database centro_comercial charset=latin1;  
Query OK, 1 row affected (0.00 sec)
```

3. Muestra las tablas existentes en la base de datos 'centro_comercial' Utiliza la consola

(0,25 puntos)

```
mysql> show tables;  
Empty set (0.00 sec)
```

4. Crea la tabla 'clientes'.

(1 punto)

- a) Establece la clave primaria como restricción de columna
- b) Especificar el motor de almacenamiento InnoDB

```
• create table clientes(  
    dni char(9) primary key,  
    nombre varchar(20) not null,  
    apellidos varchar(20),  
    direccion varchar(100) default 'c/ La felicidad se encuentra en',  
    tlf int,  
    email varchar(70) not null  
) engine = InnoDB;
```

5. Crea la tabla 'locales'

(1 punto)

a) Establece la clave primaria como restricción de tabla

b) Añadir un índice único al nombre del local

```
14 • create table locales(  
15     cod int(2) zerofill auto_increment,  
16     nombre varchar(20) not null,  
17     capacidad_max tinyint default '10',  
18     horario varchar(50) default 'Lunes-Viernes de 09:00 a 22:00',  
19     primary key (cod),  
20     index indiceNombre (nombre)  
21 );
```

6. Crear la tabla 'tiendas'

(1 punto)

a) Indicar la clave primaria como restricción de columna.

b) Indica la clave foránea al crear la tabla:

- No se permitirán eliminar locales si están registrada en esta tabla
- Al actualizar un local se actualizaran las claves foráneas que tenga referenciadas (en cascada).

```
23 • create table tiendas(  
24     cod int(2) zerofill primary key,  
25     estilo set('deportivo', 'casual', 'urbano', 'punk', 'etnico', 'gotico') not null,  
26     rebajas tinyint not null,  
27     foreign key (cod) references locales(cod) on delete restrict on update cascade  
28 );
```

7. Crear la tabla 'restaurantes'

(0,75 puntos)

- Indicar la clave primaria como restricción de columna.
- No añadir la clave foránea

```
30 • create table restaurantes(  
31     cod int(2) zerofill primary key,  
32     tipo_comida enum('canaria', 'italiana', 'india', 'china', 'japonesa', 'mexicana', 'vegana')  
33 );
```

8. Crear la tabla 'clientes_locales'

(1 punto)

- Indicar la clave primaria.
- Indica la clave foránea al crear la tabla:
 - Si se eliminan los clientes de la base de datos se deberá eliminar de dicha tabla

```
• create table clientes_locales(  
    dni char(9),  
    cod int(2) zerofill,  
    primary key (dni,cod),  
    foreign key (dni) references clientes(dni) on delete cascade,  
    foreign key (cod) references locales(cod)  
);
```

9. Borrar la columna 'apellidos' de la tabla 'clientes'

(0,25 puntos)

Mostrar la descripción de la tabla

```
43 • alter table clientes drop apellidos;  
44 • describe clientes;
```

Result Grid							Filter Rows: <input type="text"/>		Export:	Wrap Cell Content:
#	Field	Type	Null	Key	Default	Extra				
1	dni	char(9)	NO	PRI	NULL					
2	nombre	varchar(20)	NO		NULL					
3	direccion	varchar(100)	YES		c/ La felicidad se encuentra entre n...					
4	tlf	int	YES		NULL					
5	email	varchar(70)	NO		NULL					

10. Añadir la columna 'apellidos' en la misma posición que estaba. (0,25 punto)
Mostrar la descripción de la tabla

```
45 • alter table clientes add apellidos varchar(20) after nombre;
46 • describe clientes;
```

Result Grid							Filter Rows:	Export:	Wrap Cell Content:
#	Field	Type	Null	Key	Default	Extra			
1	dni	char(9)	NO	PRI	NULL				
2	nombre	varchar(20)	NO		NULL				
3	apellidos	varchar(20)	YES		NULL				
4	direccion	varchar(100)	YES		c/ La felicidad se encuentra entre n...				
5	tlf	int	YES		NULL				

11. Añadir la clave foránea de la tabla 'restaurantes' (1 punto)
 a) Al eliminar un local la clave foránea se quedará con el valor null.
 b) Al actualizar un local se actualizarán las claves foráneas

```
47 • alter table restaurantes add foreign key (cod) references locales(cod) on delete set null on update cascade;
```

Action Output				Message
#	Time	Action		
144	16:46:02	create table clientes_local...	0 row(s) affected, 2 warning(s):	1681 The ZEROFILL attribute is deprecated and will be removed in a future release. Use the LPAD function to zero-pad numbers, or store the for...
145	16:46:02	alter table clientes drop a...	0 row(s) affected	Records: 0 Duplicates: 0 Warnings: 0
146	16:46:02	alter table clientes add ap...	0 row(s) affected	Records: 0 Duplicates: 0 Warnings: 0
147	16:46:02	alter table restaurantes a...	Error Code: 1830. Column 'cod' cannot be NOT NULL; needed in a foreign key constraint 'restaurantes_ibfk_1' SET NULL	

Da error porque no puedes eliminar un local y que en restaurantes se ponga su respectiva entrada a null, debido a que es una clave foranea no puede ser null. Yo lo arreglaria quitando el set null en el on delete de la sentencia, quedando así:

```
• alter table restaurantes add foreign key (cod) references locales(cod) on update cascade;
```

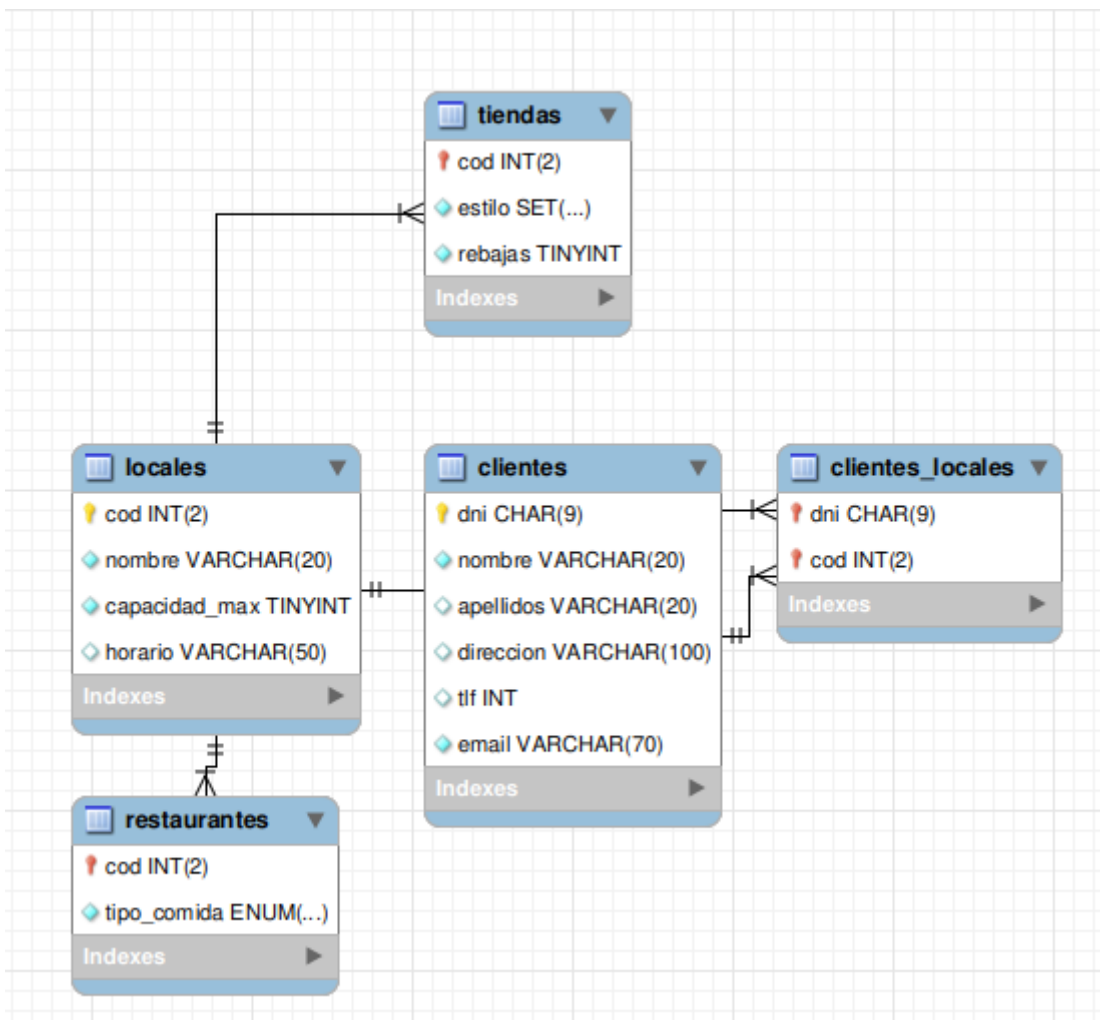

12. Añadir un comentario en la tabla 'clientes_locales' que contenga el texto:

'Esta tabla registra la relación entre clientes y locales dentro del centro comercial, indicando qué clientes han visitado qué establecimientos'

```
create table clientes_locales(  
  dni char(9),  
  cod int(2) zerofill,  
  primary key (dni,cod),  
  foreign key (dni) references clientes(dni) on delete cascade,  
  foreign key (cod) references locales(cod)  
)  
comment 'Esta tabla registra la relación entre clientes y locales dentro del centro comercial, indicando qué clientes han visitado qué establecimientos';
```

(0,5 punto)

13. Adjunta una captura del diagrama relacional resultante (0,25 puntos)



14. Adjunta el diccionario de datos

(0,25 puntos)

centro_comercial

clientes

Columna	Tipo	Nulo	Predeterminado	Enlaces a	Comentarios	Media type
dni (Primaria)	char(9)	No				
nombre	varchar(20)	No				
apellidos	varchar(20)	Sí	NULL			
direccion	varchar(100)	Sí	c/ La felicidad se encuentra entre nosotros, 1			
tlf	int	Sí	NULL			
email	varchar(70)	No				

Índices

Nombre de la clave	Tipo	Único	Empaquetado	Columna	Cardinalidad	Cotejamiento	Nulo	Comentario
PRIMARY	BTREE	Sí	No	dni	0	A	No	

clientes_locales

Comentarios de la tabla: Esta tabla registra la relación entre clientes y locales dentro del centro comercial, indicando qué clientes han visitado qué establecimientos

Columna	Tipo	Nulo	Predeterminado	Enlaces a	Comentarios	Media type
dni (Primaria)	char(9)	No		clientes -> dni		
cod (Primaria)	int(2)	No		locales -> cod		

Índices

Nombre de la clave	Tipo	Único	Empaquetado	Columna	Cardinalidad	Cotejamiento	Nulo	Comentario
PRIMARY	BTREE	Sí	No	dni	0	A	No	
				cod	0	A	No	
cod	BTREE	No	No	cod	0	A	No	

locales

Columna	Tipo	Nulo	Predeterminado	Enlaces a	Comentarios	Media type
cod (Primaria)	int(2)	No				
nombre	varchar(20)	No				
capacidad_max	tinyint	No	10			
horario	varchar(50)	Sí	Lunes-Viernes de 09:00 a 22:00			

Índices

Nombre de la clave	Tipo	Único	Empaquetado	Columna	Cardinalidad	Cotejamiento	Nulo	Comentario
PRIMARY	BTREE	Sí	No	cod	0	A	No	
indiceNombre	BTREE	No	No	nombre	0	A	No	

restaurantes

Columna	Tipo	Nulo	Predeterminado	Enlaces a	Comentarios	Media type
cod (Primaria)	int(2)	No		locales -> cod		
tipo_comida	enum('canaria', 'italiana', 'india', 'china', 'japonesa', 'mexicana', 'vegana')	No				

Índices

Nombre de la clave	Tipo	Único	Empaquetado	Columna	Cardinalidad	Cotejamiento	Nulo	Comentario
PRIMARY	BTREE	Sí	No	cod	0	A	No	

tiendas

Columna	Tipo	Nulo	Predeterminado	Enlaces a	Comentarios	Media type
cod (Primaria)	int(2)	No		locales -> cod		
estilo	set('deportivo', 'casual', 'urbano', 'punk', 'etnico', 'gotico')	No				
rebajas	tinyint	No				

Índices

Nombre de la clave	Tipo	Único	Empaquetado	Columna	Cardinalidad	Cotejamiento	Nulo	Comentario
PRIMARY	BTREE	Sí	No	cod	0	A	No	

Imprimir

15. Crea un índice en el campo 'tlf' de la tabla 'clientes'. A este índice se le llamará *telefonoCliente*. (0,25 puntos)

```
create index telefonoCliente on clientes(tlf);
```

16. Eliminar la tabla 'clientes_locales'. Utiliza una sentencia que se pueda ejecutar sin que muestre error (exista o no la tabla). (0,25 puntos)

- ```
drop table if exists clientes_locales;
```

17. Crea una tabla temporal llamada *clientes\_temporal*, será una copia exacta de la tabla *clientes*. ¿Puedes mostrar la tabla con un *show...*? ¿Cómo podrías comprobar que se ha creado? ¿Podrías comentar algo acerca de su "tiempo de vida"?

*En esta actividad se valorará la explicación al igual que las sentencias utilizadas, si no argumentas ni respondes a las preguntas no conseguirás la máxima puntuación en este ejercicio.*

```
--
54 • create temporary table if not exists clientes_temporal as select * from clientes;
55 • show tables;
56
```

| # | Tables_in_centro_comerci |
|---|--------------------------|
| 1 | clientes                 |
| 2 | locales                  |
| 3 | restaurantes             |
| 4 | tiendas                  |

*La tabla no se muestra con un *show*, se podría observar con un *describe table clientes\_temporal*.*

```
56 • describe table clientes_temporal;
57
```

| # | id | select_type | table             | partition: | type | possible_key | key  | key_len | ref  | rows |
|---|----|-------------|-------------------|------------|------|--------------|------|---------|------|------|
| 1 | 1  | SIMPLE      | clientes_temporal | NULL       | ALL  | NULL         | NULL | NULL    | NULL | 1    |

*Esta tabla se mantendrá con vida mientras la sesión este activa, en el momento en que se cierre, la tabla desaparecerá.*

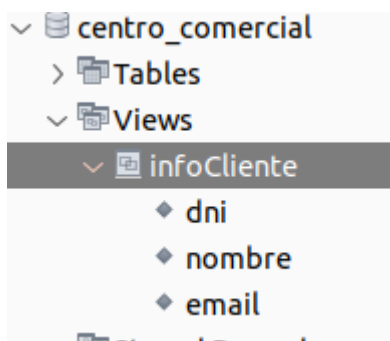
*(1 punto)*

**18. Crea una vista con el nombre 'infoCliente' utilizando la consulta:**

**select DNI, nombre, email from clientes**

*(0,25 puntos)*

**create view infoCliente as select dni, nombre, email from clientes;**



**19. En el caso en el que te hubieses creado un script llamado: 'bd\_centro\_comercial.sql'**

**¿Con qué sentencia lo hubieses ejecutado desde la consola?**

*(No realizar captura solo especificar la sentencia)*

*(0,25 puntos)*

*source bd\_centro\_comercial.sql;*