Boletín II de exercicios sobre Deseño Físico

1. Cambiar o xogo de carácteres para a base de datos practicas1. O novo xogo de carácteres será latin1, co sistema de colación ‘general’.

alter database practicas1

character set latin1

collate latin1\_general\_ci;

1. Realizar modificacións en táboas empregando a sentenza ALTER en varios casos:

* Facer os seguintes cambios na táboa fabricante da base de datos practicas1:
* Engadir a columna de tamaño variable email, asignándolle como valor por defecto o valor nulo.
* Engadir a columna de tamaño fixo cif, que non admite valores nulos, e colocala na segunda posición, despois da columna idFabricante.
* Engadir un índice asociado á columna cif.
* Cambiar o motor de almacenamento que ten actualmente, por Innodb.

/\*cambios na táboa de fabricante\*/

alter table fabricante

add column email varchar(150) default null,

add column cif char(9) not null after idFabricante,

add index idx\_fabicante\_cif (cif),

engine = Innodb;

* Facer os seguintes cambios na táboa fabricante da base de datos practicas1:
* Cambiar a clave primaria da táboa fabricante. A nova clave primaria pasa a ser a columna cif.
* Eliminar a columna idFabricante.
* Eliminar o índice asociado á columna nome.

/\*cambios na táboa de fabricante\*/

alter table fabricante

add primary key (cif),

drop column idFabricante,

drop index idx\_fabricante\_nome;

* Facer os seguintes cambios na táboa grupo da base de datos practicas1:
* Cambiarlle o nome a columna cache. O novo nome é custo.
* Asignarlle á columna dataFormación como valor por defecto o valor cero.

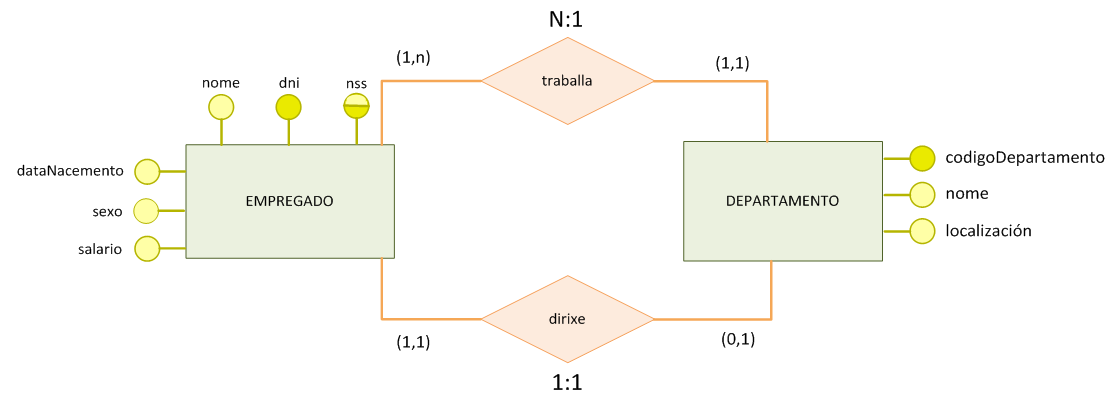
/\*cambios na táboa grupo\*/

**alter** **table** grupo

**change** cache custo **mediumint** unsigned**,**

**alter** dataFormacion **set** **default** 0**;**

1. Crear na base de datos practicas1, as táboas empregado e departamento que están relacionadas como se indica no seguinte diagrama E/R:



Baixo os seguintes supostos:

* Crear as táboas empregado só sentenzas CREATE TABLE. Teñen que conter as restricións de clave foranea.
* Crear primeiro as dúas táboas, sen claves foráneas, e despois engadir as cla-ves foránea que establecen a relación entre elas coa axuda da sentenza ALTER TABLE. Ten en conta que debes borrar primeiro as táboas, se xa existiran do exercicio anterior.
* Desactivar a verificación de claves foráneas, coa variable foreign\_key\_checks, e despois facer a creación das táboas como se pide no primeiro suposto deste exercicio. Borrar primeiro as táboas, se xa existiran. Ó finalizar volve activar a variable de verificación.

**use** practicas1**;**

**drop** **table** **if** **exists** empregado**;**

**drop** **table** **if** **exists** departamento**;**

/\*creación das táboas\*/

**create** **table** empregado

**(**

dni **char(**9**)** **not** **null,**

nss **char(**10**)** **not** **null,**

nome **varchar(**80**)** **not** **null,**

dataNacemento **date,**

sexo enum**(**'h'**,**'m'**)** **comment** 'h = home, m = muller'**,**

salario **decimal(**11**,**2**)** unsigned**,**

codigoDepartamento **smallint** unsigned**,**

**primary** **key** **(**dni**),**

**index** idx\_empregado\_nss **(**nss**)**

**);**

**create** **table** departamento

**(**

codigoDepartamento **smallint** unsigned auto\_increment **not** **null,**

nome **varchar(**60**)** **not** **null,**

localizacion **varchar(**40**),**

dniXefe **char(**9**),**

**primary** **key** **(**codigoDepartamento**)**

**);**

/\*engadir as restriccións de clave foránea par reprentar as relacións \*/

**alter** **table** empregado

**add** **constraint** fk\_empregado\_departamento **foreign** **key** **(**codigoDepartamento**)**

**references** departamento**(**codigoDepartamento**)**

**on** **delete** **set** **null**

**on** **update** **cascade;**

**alter** **table** departamento

**add** **constraint** fk\_departamento\_empregado **foreign** **key** **(**dniXefe**)**

**references** empregado**(**dni**)**

**on** **delete** **set** **null**

**on** **update** **cascade**

--------

/\*desactivar a verificación de claves foráneas \*/

**set** FOREIGN\_KEY\_CHECKS **=** 0**;**

**use** practicas1**;**

**drop** **table** **if** **exists** empregado**;**

**drop** **table** **if** **exists** departamento**;**

**create** **table** empregado

**(**

dni **char(**9**)** **not** **null,**

nss **char(**10**)** **not** **null,**

nome **varchar(**80**)** **not** **null,**

dataNacemento **date,**

sexo enum**(**'h'**,**'m'**)** **comment** 'h = home, m = muller'**,**

salario **decimal(**11**,**2**)** unsigned**,**

**primary** **key** **(**dni**),**

**index** idx\_empregado\_nss **(**nss**),**

codigoDepartamento **smallint** unsigned**,**

**constraint** fk\_empregado\_departamento **foreign** **key** **(**codigoDepartamento**)**

**references** departamento**(**codigoDepartamento**)**

**on** **delete** **set** **null**

**on** **update** **cascade**

**);**

**create** **table** departamento

**(**

codigoDepartamento **smallint** unsigned auto\_increment **not** **null,**

nome **varchar(**60**)** **not** **null,**

localizacion **varchar(**40**),**

dniXefe **char(**9**),**

**primary** **key** **(**codigoDepartamento**),**

**constraint** fk\_departamento\_empregado **foreign** **key** **(**dniXefe**)**

**references** empregado**(**dni**)**

**on** **delete** **set** **null**

**on** **update** **cascade**

**);**

/\*activar a verificación de claves foráneas\*/

**set** FOREIGN\_KEY\_CHECKS **=** 1**;**

1. Crear o esquema dunha base de datos creando primeiro as táboas e definindo despois as relacións entre eles en dous supostos.
   1. Crear a base de datos eleccionModulos e crear nela as táboas profesor, grupo, modulo e imparte, segundo a seguinte descrición:

Táboa profesor:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Nome | Tipo | Null | Clave | Observacións |
| idProfesor | char(15) | Non | Primary | Nome curto do profesor |
| corpo | enum | Non |  | Corpo ao que pertence. Valores permitidos:  **S** = Profesorado Secundaria **T** = Profesorado Técnico |
| nrp | char(20) | Non | Única | Número de Rexistro Persoal |
| nome | varchar(30) | Non |  |  |
| apelidos | varchar(60) | Non | Indice |  |
| horasLectivas | smallint (sen signo) |  |  | Nº de horas de clase que imparte o profesor. Por defecto: 18 |

Táboa grupo:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Nome | Tipo | Null | Clave | Observacións |
| idGrupo | char(15) | Non | Primary | Clave para identificar o grupo |
| descricion | varchar(100) | Non |  | Descrición do nome do grupo |
| horasTitoria | smallint (sen signo) |  |  | Nº de horas de redución para o profesor titor |
| idProfesor | char(15) |  |  | Identificador do profesor que é titor do grupo |

Táboa módulo:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Nome | Tipo | Null | Clave | Observacións |
| idModulo | char(6) | Non | Primary | Clave para identificar o módulo |
| horasSemanais | tinyint (sen signo) |  |  | Descrición do nome do grupo |
| descricion | varchar(100) |  |  | Nº de horas de redución para o profesor titor |
| corpo | enum | Non |  | Corpo dos profesores que poden impartir o módulo. Valores permitidos: **S** = Profesorado Secundaria **T** = Profesorado Técnico |

Táboa imparte (clave primaria composta)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Nome | Tipo | Null | Clave | Observacións |
| idModulo | char(6) | Non | Primary | Clave para identificar o módulo |
| idGrupo | char(15) | Non | Clave para identificar o grupo |
| idProfesor | char(15) | Non | Nome curto do profesor |

/\*borrado da base de datos, se existe (só cando se están facendo probas)\*/

**drop** **database** **if** **exists** eleccionModulos**;**

/\*creación da base de datos se non existe\*/

**create** **database** **if** **not** **exists** eleccionModulos

**character** **set** utf8

**collate** utf8\_spanish\_ci**;**

/\* ou tamén \*/

**create** **schema** **if** **not** **exists** eleccionModulos

**character** **set** utf8

**collate** utf8\_spanish\_ci**;**

/\*activar vase de datos\*/

**use** eleccionModulos**;**

/\* TÁBOA profesor \*/

**create** **table** profesor **(**

idProfesor **char(**15**)** **not** **null,**

corpo enum**(**'S'**,**'T'**)** **not** **null,**

nrp **char(**20**)** **not** **null,**

nome **varchar(**30**)** **not** **null,**

apelidos **varchar(**60**)** **not** **null,**

horasLectivas **smallint** unsigned **default** 18**,**

**primary** **key** **(**idProfesor**),**

**unique** **key** **(**nrp**)**

**)engine=**InnoDB**;**

/\* TÁBOA grupo\*/

**create** **table** grupo **(**

idGrupo **char(**15**),**

descricion **varchar(**100**)** **not** **null,**

horasTitoria **smallint** unsigned**,**

idProfesor **char(**15**),**

**primary** **key** **(**idGrupo**)**

**)engine=**InnoDB**;**

/\* TÁBOA modulo\*/

**create** **table** modulo **(**

idModulo **char(**6**),**

horasSemanais **smallint** unsigned **,**

descricion **varchar(**100**)** **not** **null,**

corpo enum**(**'S'**,**'T'**),**

**primary** **key** **(**idModulo**)**

**)engine=**InnoDB**;**

/\* TÁBOA imparte \*/

**create** **table** imparte **(**

idModulo **char(**6**),**

idGrupo **char(**15**),**

idProfesor **char(**15**),**

**primary** **key** **(**idGrupo**,**idModulo**,**idProfesor**)**

**)engine=**InnoDB**;**

* 1. Definir as relacións entre esas táboas tendo en conta o seguinte grafo relacional.

u4a2_tarefa4_1_grafo.emf

/\* Definición das relacións entre as táboas: RESTRICIÓNS DE CLAVE FORÁNEA \*/

**alter** **table** grupo

**add** **constraint** fk\_grupo\_profesor **foreign** **key** **(**idprofesor**)**

**references** profesor **(**idprofesor**)**

**on** **delete** **restrict**

**on** **update** **cascade;**

**alter** **table** imparte

**add** **constraint** fk\_imparte\_modulo **foreign** **key** **(**idmodulo**)**

**references** modulo **(**idmodulo**)**

**on** **delete** **restrict**

**on** **update** **cascade,**

**add** **constraint** fk\_imparte\_profesor **foreign** **key** **(**idprofesor**)**

**references** profesor **(**idprofesor**)**

**on** **delete** **restrict**

**on** **update** **cascade,**

**add** **constraint** fk\_impate\_grupo **foreign** **key** **(**idgrupo**)**

**references** grupo **(**idgrupo**)**

**on** **delete** **restrict**

**on** **update** **cascade;**