

## Boletín 2. Gestión de proyectos

BDSW Tema 2 –Bases de datos relacionales

11 de febrero de 2015

## Índice

<b>1</b>	<b>Introducción</b>	<b>1</b>
<b>2</b>	<b>Descripción del problema</b>	<b>1</b>
2.1	Entidades y relaciones . . . . .	1
<b>3</b>	<b>La base de datos</b>	<b>2</b>
3.1	Creación de la base de datos . . . . .	2
3.2	Tablas y relaciones . . . . .	3
3.3	Datos . . . . .	5
<b>4</b>	<b>Actividad</b>	<b>5</b>

## Índice de figuras

1	Diagrama de clases UML que representa el sistema de gestión de empleados, departamentos y proyectos.	2
---	--	---

## Índice de listados

1	Script de creación de la base de datos empresa utilizada en este ejercicio. . . . .	2
2	Script de creación de las tablas, relaciones y restricciones de la base de datos empresa utilizada en este ejercicio. . . . .	3

## 1 Introducción

Este ejemplo está basado en una actividad empresarial simple: gestión de empleados, departamentos y proyectos. En la elaboración del ejemplo utilizamos una aproximación relacional:

- Se implementa el esquema utilizando sólo los tipos de datos intrínsecos de MySQL.
- Mediante esta aproximación, se crean tablas que contienen los datos de la aplicación.
- Se utilizan técnicas bien conocidas para implementar las relaciones entre entidades.

## 2 Descripción del problema

Nuestra empresa se dedica al desarrollo de proyectos tecnológicos, que son desarrollados por los departamentos. Algunas de las características de la organización son:

- La empresa se organiza alrededor del departamento. El departamento agrupa un conjunto de empleados y se encarga del desarrollo de proyectos destinando recursos (tiempo de los empleados) a estos.
- Cada departamento es administrado por un departamento administrador, cuenta con un conjunto de empleados adscritos y un gestor.
- Los proyectos están adscritos a un único departamento y cuentan con un responsable único. Los empleados participan en los proyectos durante un cierto periodo de tiempo decidido por la empresa.
- Los proyectos se organizan en una serie de subproyectos, cada uno de los cuales se adscribe a un departamento y cuenta con un responsable

## 2.1 Entidades y relaciones

En el modelo se distinguen las siguientes entidades:

- Los empleados.
- Los departamentos.
- Los proyectos.

y podemos identificar las siguientes relaciones:

- Los departamentos tienen una relación uno-a-muchos con los empleados que los constituyen.

- Cada departamento cuenta con un único departamento administrador y un departamento puede administrar a ninguno, uno o muchos departamentos.
- Cada departamento tiene uno o ningún *manager*.
- Los departamentos tienen una relación uno a muchos con los proyectos que gestionan.
- Los proyectos mantiene una relación muchos-a-muchos con los empleados que participan en el proyectos. Esta relación tiene una duración en el tiempo que es la participación del empleado con el proyecto.
- Cada proyecto cuenta con un único responsable mientras que un empleado puede ser responsable de varios proyectos.
- Cada proyecto puede pertenecer a un proyecto de mayor entidad y a su vez contar con cierto número de subproyectos.

En la figura 1 se muestra el diagrama de clases UML que describe el sistema descrito.

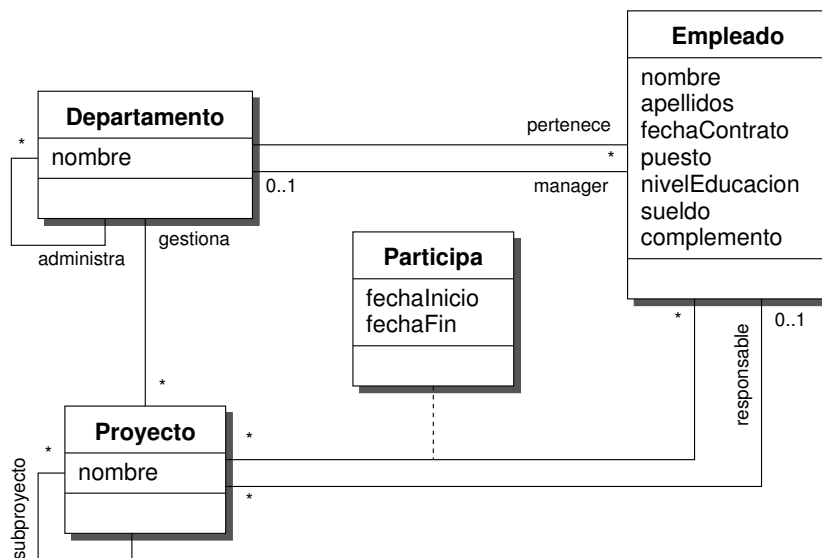


Figura 1: Diagrama de clases UML que representa el sistema de gestión de empleados, departamentos y proyectos.

## 3 La base de datos

### 3.1 Creación de la base de datos

En el listado 1 se muestra el código SQL que crea la base de datos, el usuario de base de datos (`00_database.sql`). Para ejecutar el código en MySQL utilizamos la interfaz en modo línea a la base de datos:

```

1  unix_$ mysql -u root -p
2  Enter password:
3  mysql> \. 00_database.sql
4  mysql> quit
5  Bye
6  unix_$

```

Listado 1: Script de creación de la base de datos empresa utilizada en este ejercicio.

```

1  -- *****
2  -- **
3  -- ** GESTION DE EMPLEADOS DEPARTAMENTOS Y PROYECTOS.
4  -- **
5  -- ** Base de datos: empresa
6  -- ** Usuario: usuario
7  -- ** Password: pusuario
8  -- **
9  -- ** Script de creacion de la base de datos y del usuario.
10 -- **
11 -- *****
12 CREATE DATABASE empresa;
13 GRANT ALL ON empresa.* TO usuario@'%' IDENTIFIED BY 'usupw'

```

```

14      WITH GRANT OPTION;
15      GRANT ALL ON empresa.* TO usuario@localhost IDENTIFIED BY 'usupw'
16      WITH GRANT OPTION;

```

## 3.2 Tablas y relaciones

Las tablas, relaciones y restricciones de la base de datos se muestran en el listado 2. Ejecute el *script* SQL como se ha explicado en el punto anterior.

Listado 2: *Script* de creación de las tablas, relaciones y restricciones de la base de datos empresa utilizada en este ejercicio.

```

1  -- *****
2  -- **
3  -- ** GESTION DE EMPLEADOS DEPARTAMENTOS Y PROYECTOS.
4  -- **
5  -- ** Base de datos: empresa
6  -- ** Usuario: usuario
7  -- ** Password: pusuario
8  -- **
9  -- ** Script de creacion de las tablas.
10 -- **
11 -- *****
12 USE empresa;
13 -- *****
14 -- **
15 -- ** DEPARTAMENTOS
16 -- **
17 -- *****
18 CREATE TABLE Departamentos
19 (
20     idDepartamento CHAR(3) NOT NULL,
21     nombre VARCHAR(36) NOT NULL,
22     manager CHAR(6) ,
23     administrador CHAR(3)
24 );
25
26 CREATE UNIQUE INDEX Deptos_idDepto_IDX
27 ON Departamentos (idDepartamento);
28
29 ALTER TABLE Departamentos
30 ADD CONSTRAINT Deptos_idDepto_PK
31 PRIMARY KEY (idDepartamento);
32
33 -- *****
34 -- **
35 -- ** EMPLEADOS
36 -- **
37 -- *****
38 CREATE TABLE Empleados
39 (
40     idEmpleado CHAR(6) NOT NULL,
41     nombre VARCHAR(20) NOT NULL,
42     apellidos VARCHAR(50) NOT NULL,
43     departamento CHAR(3) ,
44     fechaContrato DATE ,
45     puesto CHAR(8) ,
46     nivelEducacion SMALLINT ,
47     sueldo DECIMAL(9,2) ,
48     complemento DECIMAL(9,2)
49 );
50 CREATE UNIQUE INDEX Emp_idEmp_IDX
51 ON Empleados (idEmpleado);
52
53 ALTER TABLE Empleados
54 ADD CONSTRAINT Emp_idEmp_PK

```

```

55     PRIMARY KEY (idEmpleado);
56
57 ALTER TABLE Empleados
58     ADD CONSTRAINT Emp_depto_FK
59     FOREIGN KEY (departamento)
60     REFERENCES Departamentos (idDepartamento)
61     ON DELETE SET NULL;
62
63 CREATE INDEX Emp_nombre_IDX
64     ON Empleados (nombre, apellidos);
65
66 --
67 ALTER TABLE Departamentos
68     ADD CONSTRAINT Depto_manager_FK
69     FOREIGN KEY (manager)
70     REFERENCES Empleados (idEmpleado)
71     ON DELETE RESTRICT;
72
73 ALTER TABLE Departamentos
74     ADD CONSTRAINT Depto_admin_FK
75     FOREIGN KEY (administrador)
76     REFERENCES Departamentos (idDepartamento)
77     ON DELETE SET NULL;
78
79 -- *****
80 -- **
81 -- ** PROYECTOS
82 -- **
83 -- *****
84
85 CREATE TABLE Proyectos
86 (
87     idProyecto    CHAR(6)          NOT NULL,
88     nombre        VARCHAR(30)      NOT NULL,
89     departamento  CHAR(3)          ,
90     responsable   CHAR(6)          ,
91     proPrincipal  CHAR(6)
92 );
93
94 CREATE UNIQUE INDEX Proy_idProy_IDX
95     ON Proyectos (idProyecto);
96
97 ALTER TABLE Proyectos
98     ADD CONSTRAINT Proy_idProy_PK
99     PRIMARY KEY (idProyecto);
100
101 ALTER TABLE Proyectos
102     ADD CONSTRAINT Proy_depto_FK
103     FOREIGN KEY (departamento)
104     REFERENCES Departamentos (idDepartamento)
105     ON DELETE RESTRICT;
106
107 ALTER TABLE Proyectos
108     ADD CONSTRAINT Proy_respon_FK
109     FOREIGN KEY (responsable)
110     REFERENCES Empleados (idEmpleado)
111     ON DELETE RESTRICT;
112
113 ALTER TABLE Proyectos
114     ADD CONSTRAINT Proy_proPrin_FK
115     FOREIGN KEY (proPrincipal)
116     REFERENCES Proyectos (idProyecto)
117     ON DELETE SET NULL;
118
119 -- *****
120 -- **
121 -- ** ENTIDAD DE INTERSECCION EMPLEADO **
122 -- ** Y PROYECTO.
123 -- **
124 -- *****
125
126 CREATE TABLE Participa
127 (

```

```
122     idEmpleado    CHAR(6)      NOT NULL,
123     idProyecto    CHAR(6)      NOT NULL,
124     fechaInicio   DATE         NOT NULL,
125     fechaFin      DATE
126 );
127
128 CREATE UNIQUE INDEX Part_idEmp_idProy_IDX
129     ON Participa (idEmpleado, idProyecto);
130
131 ALTER TABLE Participa
132     ADD CONSTRAINT Part_idEmp_idProy_PK
133     PRIMARY KEY (idEmpleado, idProyecto);
134
135 ALTER TABLE Participa
136     ADD CONSTRAINT Part_idEmp_FK
137     FOREIGN KEY (idEmpleado)
138     REFERENCES Empleados (idEmpleado)
139     ON DELETE RESTRICT;
140
141 ALTER TABLE Participa
142     ADD CONSTRAINT Part_idProy__FK
143     FOREIGN KEY (idProyecto)
144     REFERENCES Proyectos (idProyecto)
145     ON DELETE RESTRICT;
```

### 3.3 Datos

Los datos de prueba de la base de datos se suministran en el fichero `02_data.sql`. La carga de datos se realiza del mismo modo que las operaciones anteriores.

## 4 Actividad

En el fichero `02_queries.sql` se proporcionan varias consultas SQL sobre la base de datos. Las consultas están diseñadas para mostrar la capacidad del lenguaje de consulta SQL. Ejecute las consultas y analice los resultados obtenidos.