

Empleados

Pepe Calderón, Fernando Sola, Daniel Ayala, Inma Hernández, Margarita Cruz, Carlos Arévalo, David Ruiz



Escuela Técnica Superior de
Ingeniería Informática

Índice

1. Requisitos	2
2. Requisitos de información (RI)	3
2.1. RI-01: Departamentos	3
2.2. RI-02: Empleados	3
3. Reglas de negocio (RN)	4
3.1. RN-01: Departamentos	4
3.2. RN-02: Empleados	4
4. Modelo Conceptual	5
5. Diagrama de clases	6
6. Intensión	7
7. Extensión	8
8. Álgebra relacional	9
9. Modelo Tecnológico (MariaDB)	11
10. Script SQL para crear la base de datos	12
11. Script SQL para la carga inicial de datos	13
12. Consultas	14
13. SQL Avanzado	15
13.1. Procedimientos almacenados	15
13.2. Funciones	16
13.3. Cursores	17
13.4. Triggers (disparadores)	17

Empleados

Requisitos

Catálogo de Requisitos Se pretende realizar un pequeño sistema de información para gestionar los empleados de los departamentos de una empresa. Cada empleado pertenece a un departamento y puede tener un jefe. Cada departamento pertenece a una localidad. Cada empleado tiene un salario y una comisión, además se almacena la fecha de inicio y de finalización de su contrato.

Requisitos de información (RI)

2.1. RI-01: Departamentos

- **Como:** Profesor de la asignatura
- **Quiero:** Poder almacenar la siguiente información de los departamentos: nombre del departamento(obligatorio) y localidad(opcional)
- **Para:** Que el alumno practique con un sistema de información sencillo

2.2. RI-02: Empleados

- **Como:** Profesor de la asignatura
- **Quiero:** Poder almacenar la siguiente información de los empleados: nombre (obligatorio y único), salario, comisión, fecha de inicio y finalización del contrato (opcionales), jefe (opcional) y departamento al que pertenece (obligatorio)
- **Para:** Que el alumno practique con un sistema de información sencillo

Reglas de negocio (RN)

3.1. RN-01: Departamentos

- **Como:** Profesor de la asignatura
- **Quiero:** Un departamento no puede tener más de cinco empleados y la combinación nombre y localidad no puede repetirse
- **Para:** Que el alumno practique con restricciones simples

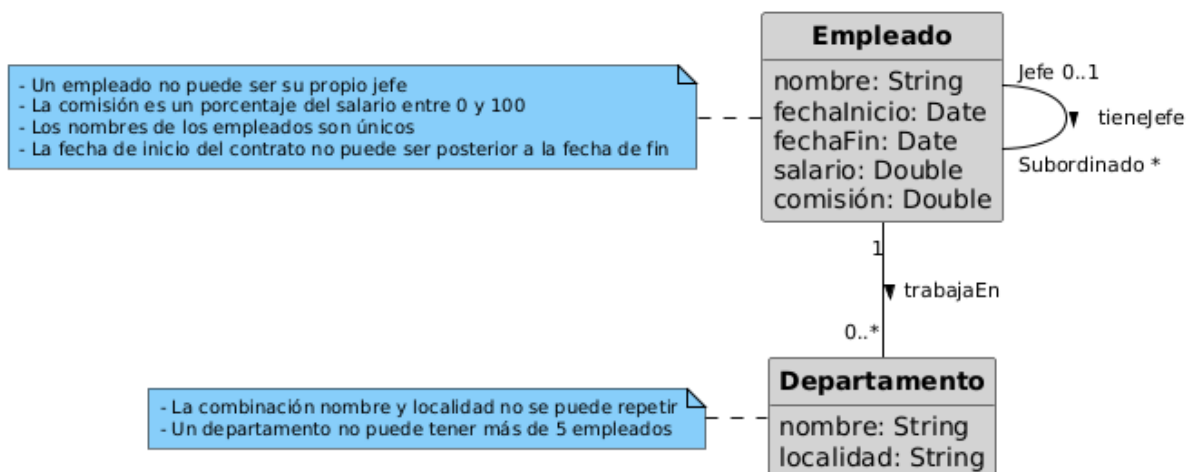
3.2. RN-02: Empleados

- **Como:** Profesor de la asignatura
- **Quiero:** Un empleado no puede ser jefe de si mismo, la comisión es un porcentaje del salario y no puede ser negativa, ni modificarse en más de un 20% de golpe
- **Para:** Que el alumno practique con restricciones simples

Modelo Conceptual

Modelo conceptual

Diagrama de clases



Modelo relacional

Intensión

```
Departamentos(departamentoId, nombre, localidad)
  PK(departamentoId)
  AK(nombre, localidad)
Empleados(empleadoId, departamentoId, jefeId, nombre, salario, fechaInicio,
fechaFin, comision)
  PK(empleadoId)
  FK(departamentoId) / Departamentos
  FK(jefeId) / Empleados
  AK(nombre)
```

Extensión

```
Departamentos = {
  (d1, 'Arte', 'Cádiz'),
  (d2, 'Historia', null),
  (d3, 'Informática', 'Sevilla')
}
Empleados = {
  (e1, d1, NULL, 'Pedro', 2300.00, '2017-09-15', NULL, 0.2),
  (e2, d1, NULL, 'Jose', 2500.00, '2018-08-15', NULL, 0.5),
  (e3, d2, NULL, 'Lola', 2300.00, '2018-08-15', NULL, 0.3),
  (e4, d1, e1, 'Luis', 1300.00, '2018-08-15', '2018-11-15', 0),
  (e5, d1, e1, 'Ana', 1300.00, '2018-08-15', '2018-11-15', 0)
}
```

Álgebra relacional

- Renombrado de relaciones para acortar las expresiones en álgebra relacional:

$$\rho_{D(dId,n,l)}(Departamentos)$$

$$\rho_{E(eId,dId,jId,n,s,fI,Ff,c)}(Empleados)$$

- Empleados con sueldo < 2000:

$$\sigma_{s < 2000}(E)$$

- Fechas de alta y baja:

$$\Pi_{fI, fF}(E)$$

- Sueldo entre 2000 y 3000:

$$\sigma_{2000 < s < 3000}(E)$$

- Producto cartesiano:

$$E \times D$$

- Join Empleados–Departamentos:

$$E \bowtie D$$

- Departamentos con empleados:

$$\Pi_{dId}(E)$$

- Departamentos sin empleados:

$$\Pi_{dId}(D) \setminus \Pi_{dId}(E)$$

- Estadísticas globales de salario:

$$\text{COUNT}(*), \text{MIN}(s), \text{MAX}(s), \text{AVG}(s), \text{SUM}(s)_{\gamma}(E)$$

- Estadísticas de salario por departamento:

$$\text{COUNT}(*), \text{MIN}(s), \text{MAX}(s), \text{AVG}(s), \text{SUM}(s)_{\gamma} dId(E)$$

- Estadísticas de salarios por departamento con al menos dos empleados:

$$Dep2 \leftarrow \Pi_{dId} \left(\sigma_{\text{COUNT}(*)\gamma \geq 2} \left(\Pi_{\gamma}^{\text{COUNT}(*)} dId(E) \right) \right)$$

$$\text{COUNT}(*), \text{MIN}(s), \text{MAX}(s), \text{AVG}(s), \text{SUM}(s)_{\gamma} dId(E \bowtie Dep2)$$

Modelo Tecnológico (MariaDB)

Modelo tecnológico (MariaDB)

Script SQL para crear la base de datos

Empleados/createDB.sql

i Info

Código SQL disponible en: <https://raw.githubusercontent.com/ISSI-US/silence-db/main/Empleados/sql/createDB.sql>

Script SQL para la carga inicial de datos

Empleados/populateDB.sql

i Info

Código SQL disponible en: <https://raw.githubusercontent.com/ISSI-US/silence-db/main/Empleados/sql/populateDB.sql>

Consultas

Empleados/queries.sql

i Info

Código SQL disponible en: <https://raw.githubusercontent.com/IISSE-US/silence-db/main/Empleados/sql/queries.sql>

SQL Avanzado

13.1. Procedimientos almacenados

Realice procedimientos para insertar en las tablas Departments y Employees:

Empleados/pInsertDepartment.sql

i Info

Código SQL disponible en: <https://raw.githubusercontent.com/IISSE-US/silence-db/main/Empleados/sql/pInsertDepartment.sql>

Empleados/pInsertEmployee.sql

i Info

Código SQL disponible en: <https://raw.githubusercontent.com/IISSE-US/silence-db/main/Empleados/sql/pInsertEmployee.sql>

Realice un procedimiento para igualar las comisiones de todos los empleados al valor de la comisión media:

Empleados/pEquateFees.sql**i Info**

Código SQL disponible en: <https://raw.githubusercontent.com/IISSI-US/silence-db/main/Empleados/sql/pEquateFees.sql>

Implemente un procedimiento que aplique un aumento a la comisión de un empleado en particular:

Empleados/pRaiseFee.sql**i Info**

Código SQL disponible en: <https://raw.githubusercontent.com/IISSI-US/silence-db/main/Empleados/sql/pRaiseFee.sql>

13.2. Funciones

Implemente una función que devuelva el número de empleados de una localidad concreta:

Empleados/fNumEmployees.sql**i Info**

Código SQL disponible en: <https://raw.githubusercontent.com/IISSI-US/silence-db/main/Empleados/sql/fNumEmployees.sql>

Implemente una función que calcule la media de las comisiones de los empleados y use esa función dentro de un procedimiento almacenado para igualar las comisiones de todos los empleados al valor de la comisión media:

Empleados/fAvgFee.sql**i Info**

Código SQL disponible en: <https://raw.githubusercontent.com/IISSI-US/silence-db/main/Empleados/sql/fAvgFee.sql>

13.3. Cursores

Utilice un cursor para recorrer todos los empleados y calcular el valor acumulado de los salarios:

Empleados/fSumSalaries.sql

i Info

Código SQL disponible en: <https://raw.githubusercontent.com/IISSI-US/silence-db/main/Empleados/sql/fSumSalaries.sql>

13.4. Triggers (disparadores)

Implemente un disparador para evitar que un empleado sea su propio jefe:

Empleados/tSelfBoss.sql

i Info

Código SQL disponible en: <https://raw.githubusercontent.com/IISSI-US/silence-db/main/Empleados/sql/tSelfBoss.sql>

Implemente un disparador que evite que modifique la comisión de un empleado en más de un 20%:

Empleados/tChangeFee.sql

i Info

Código SQL disponible en: <https://raw.githubusercontent.com/IISSI-US/silence-db/main/Empleados/sql/tChangeFee.sql>

Implemente un disparador que evite que un departamento tenga más de cinco empleados:

Empleados/tMaxEmployeesDepartment.sql

i Info

Código SQL disponible en: <https://raw.githubusercontent.com/IISSE-US/silence-db/main/Empleados/sql/tMaxEmployeesDepartment.sql>

Implemente un disparador que en caso de insertar un empleado sin fecha de inicio, le ponga como fecha de inicio la fecha actual:

Empleados/tDefaultStartDate.sql**i Info**

Código SQL disponible en: <https://raw.githubusercontent.com/IISSE-US/silence-db/main/Empleados/sql/tDefaultStartDate.sql>

i Info

Versión PDF disponible