

Examen 1ºEvaluación AED DAM2 2021-2022

Ejecicio 1) (6ptos)

Se debe desarrollar una aplicación web que use la base de datos empresa sin modificarla(se incluye la ddbb)

Se pasa también una carpeta css con los estilos y tres páginas jsp: el índice, buscarempleado y crearempleado

Se debe desarrollar el DAO para la base de datos (únicamente tiene una tabla emleado)

Es importante darse cuenta que la tabla tiene una clave foránea a sí misma que sirve para identificar el jefe de cada empleado. Si un empleado tiene null en el campo es que no tiene jefe

Se muestran capturas de pantalla del funcionamiento y las salidas

Lo siguiente es el json que devuelve si se buscan empleados sin poner ningún filtro (se dejan todas las casillas en blanco) Observar como ejemplo que primero ha listado al empleado 10 y luego muestra todos los datos de su jefe (empleado 25) como se muestran todos los datos del empleado 25 también se muestran los datos del jefe de ese empleado 25 que en este caso es el empleado 22. Finalmente el empleado 22 no tiene ningún jefe así que lo que muestra en ese campo es: null

```
{
  "id" : 10,
  "nombre" : "Francisco",
  "apellidos" : "Álvarez Herrera",
  "fecha_contrato" : "2018-11-10",
  "numero" : 100,
  "calle" : "Iriarte",
  "municipio" : "Puerto de la Cruz",
  "jefe" : {
    "id" : 25,
    "nombre" : "Armindia",
    "apellidos" : "García Herrera",
    "fecha_contrato" : "2020-09-09",
    "numero" : 25,
    "calle" : "San Antonio",
    "municipio" : "La Laguna",
    "jefe" : {
      "id" : 22,
      "nombre" : "Ana",
      "apellidos" : "Pérez Rico",
      "fecha_contrato" : "2019-03-12",
      "numero" : 5,
      "calle" : "San Antonio",
      "municipio" : "Puerto de la Cruz",
      "jefe" : null
    }
  }
}
```

```
{
  "id" : 14,
  "nombre" : "Luis",
  "apellidos" : "González Ruíz",
  "fecha_contrato" : "2016-09-10",
  "numero" : 100,
  "calle" : "Iriarte",
  "municipio" : "Puerto de la Cruz",
  "jefe" : {
    "id" : 25,
    "nombre" : "Armindá",
    "apellidos" : "García Herrera",
    "fecha_contrato" : "2020-09-09",
    "numero" : 25,
    "calle" : "San Antonio",
    "municipio" : "La Laguna",
    "jefe" : {
      "id" : 22,
      "nombre" : "Ana",
      "apellidos" : "Pérez Rico",
      "fecha_contrato" : "2019-03-12",
      "numero" : 5,
      "calle" : "San Antonio",
      "municipio" : "Puerto de la Cruz",
      "jefe" : null
    }
  }
}
```

```
{
  "id" : 15,
  "nombre" : "Marta",
  "apellidos" : "Díaz Díaz",
  "fecha_contrato" : "2019-07-12",
  "numero" : 31,
  "calle" : "Trinidad",
  "municipio" : "La Laguna",
  "jefe" : {
    "id" : 25,
    "nombre" : "Armindá",
    "apellidos" : "García Herrera",
    "fecha_contrato" : "2020-09-09",
    "numero" : 25,
    "calle" : "San Antonio",
    "municipio" : "La Laguna",
    "jefe" : {
      "id" : 22,
      "nombre" : "Ana",
      "apellidos" : "Pérez Rico",
      "fecha_contrato" : "2019-03-12",
      "numero" : 5,

```

```
    "calle" : "San Antonio",
    "municipio" : "Puerto de la Cruz",
    "jefe" : null
  }
}
```

```
{
  "id" : 22,
  "nombre" : "Ana",
  "apellidos" : "Pérez Rico",
  "fecha_contrato" : "2019-03-12",
  "numero" : 5,
  "calle" : "San Antonio",
  "municipio" : "Puerto de la Cruz",
  "jefe" : null
}
```

```
{
  "id" : 25,
  "nombre" : "Arminda",
  "apellidos" : "García Herrera",
  "fecha_contrato" : "2020-09-09",
  "numero" : 25,
  "calle" : "San Antonio",
  "municipio" : "La Laguna",
  "jefe" : {
    "id" : 22,
    "nombre" : "Ana",
    "apellidos" : "Pérez Rico",
    "fecha_contrato" : "2019-03-12",
    "numero" : 5,
    "calle" : "San Antonio",
    "municipio" : "Puerto de la Cruz",
    "jefe" : null
  }
}
```

Se debe crear un listener al comienzo de la aplicación que genere un pool de conexiones. También habrá un interfaz llamado CRUD que incorpora los métodos: save, findById, update, delete, findAll

Debe haber una clase Empleado (un java bean) con los atributos:

```
Integer id;  
String nombre;  
String apellidos;  
Date fecha_contrato;  
  
Integer numero;  
String calle;  
String municipio;  
Empleado jefe;
```

En clase EmpleadoDAO **NO HACE FALTA CREAR EL CÓDIGO DE TODOS LOS MÉTODOS DAO** Esta prueba precisa del método: **save**(Empleado dao) y los métodos que sean precisos para realizar las **búsquedas** solicitadas (**findById**, etc)

En la base de datos **NO SE HARÁ GUARDADOS EN CASCADA** Esto es: si guardamos un empleado que tiene un jefe **NO SE CREA ESE JEFE** en la base de datos. Ya tiene que existir previamente el empleado jefe para poder guardar el nuevo empleado que es su subordinado

Nota: **No hace falta filtrar por nombre. No se va a puntuar**

Puntuaciones del Ejercicio 1):

a) Que funcione correctamente el guardado de un nuevo empleado: 2ptos
observar que debe ser correcto el método: **save**(Empleado dao) y que lo que reciba ese método sea un Empleado con los atributos que antes se han establecido: Eso significa que El jefe es otro objeto Empleado NO es un id (para esto hay dos soluciones: 1 buscar primero el empleado jefe en la base de datos o 2 “inventarse” un empleado jefe con datos falso a excepción del id del jefe que da el usuario como parámetro que SÍ debe ser real para que no haya problemas en la inserción)

Nota: Esta parte puede puntuar la mitad: 1pto si en lugar de guardar correctamente el empleado con los datos de su jefe de la forma solicitada lo que se hace es guardar null en el campo jefe

b) Que funcione correctamente la búsqueda de un empleado por id: 2ptos
(de igual forma que en el apartado a) si la búsqueda por id no consigue mostrar los datos del jefe del empleado sino un null entonces puntúa la mitad: 1pto). Si se consigue mostrar los datos de su jefe pero no los del jefe de su jefe entonces el ejercicio puntúa: 1.5ptos

Nota: en los siguientes dos apartados no se va a penalizar no mostrar los datos del jefe de cada empleado mostrado. Puede mostrarse en ese campo jefe: null y se seguirá considerando bien realizado

c) Que funcione correctamente la búsqueda de los empleados por fecha: 1ptos
(0.5 si se filtra correctamente por la fecha que esté establecida estando la otra vacía. 0.5 si se filtra correctamente por las dos fechas estando ambas establecidas)

d) Que funcione correctamente la búsqueda de todos los empleados: 1pto
(esto ocurre cuando se dejan todas las casillas del formulario vacías y se pulsa el botón de buscar)

Imágenes de la aplicación:

Salida en pantalla cuando se pulsa en el botón buscar sin introducir nada en los campos:

Gestionar Empleados

Buscar Empleados

Por ID Empleado:

Por Nombre:

Inició contrato después de:

Inició contrato antes de:

Buscar

Nota: si se busca por id no se busca por nada más. Si se busca por nombre no se busca por fechas. Se puede buscar por las dos fechas o por una (si se deja en blanco la otra)

```
{
  "id" : 10,
  "nombre" : "Francisco",
  "apellidos" : "Álvarez Herrera",
  "fecha_contrato" : "2018-11-10",
  "numero" : 100,
  "calle" : "Iriarte",
  "municipio" : "Puerto de la Cruz",
  "jefe" : {
    "id" : 25,
    "nombre" : "Arminda",
    "apellidos" : "García Herrera",
    "fecha_contrato" : "2020-09-09",
    "numero" : 25,
    "calle" : "San Antonio",
    "municipio" : "La Laguna",
    "jefe" : {
```

Salida en pantalla si se busca por empleado 25 (el número aparece en el campo: ID empleado **pero se hace únicamente para ilustrar donde se hace la búsqueda**, Al regreso de la consulta POST pueden aparecer todos los campos vacíos lo que es importante es que sea correcta la salida en el textarea)

Gestionar Empleados

Buscar Empleados

Por ID Empleado:

Por Nombre:

Inició contrato después de:

Inició contrato antes de:

Buscar

Nota: si se busca por id no se busca por nada más. Si se busca por nombre no se busca por fechas. Se puede buscar por las dos fechas o por una (si se deja en blanco la otra)

```
{
  "id": 25,
  "nombre": "Arminda",
  "apellidos": "García Herrera",
  "fecha_contrato": "2020-09-09",
  "numero": 25,
  "calle": "San Antonio",
  "municipio": "La Laguna",
  "jefe": {
    "id": 22,
    "nombre": "Ana",
    "apellidos": "Pérez Rico",
    "fecha_contrato": "2019-03-12",
    "numero": 5,
    "calle": "San Antonio",
    "municipio": "Puerto de la Cruz",
    "jefe": null
  }
}
```

Ejemplo de un empleado que se crea en la base de datos sin tener jefe:

Gestionar Empleados

Crear Empleado

ID Empleado:

421

Nombre:

Armengol

Apellidos:

Marti

Fecha de contrato:

2009-04-03

ID del jefe (en blanco si no tiene):

Dirección:

Municipio: Adeje Calle: San Martín Número: 42

Crear