

# 1. EXPLICACIÓN

He creado una sencilla interfaz basada en dos contenedores:

- Uno a la izquierda donde aparece un menú con 4 opciones. Cada opción representa cada uno de los ejercicios propuestos, excepto el último que incluye juntos los tres últimos ejercicios.
- El otro contenedor es usado para ir cargando cada uno de los datos según pulsemos sobre cada opción del menú.

Para cargar los datos en dicho contenedor he usado "Jquery" con el fin de evitar cargar la página en su totalidad y solo cargar dicho contenido. Otra ventaja de usar dicha tecnología es que, tras realizar peticiones http con jquery, los datos no se pasan por la url quedando visibles para el usuario.

## 2. OPCIONES DEL MENÚ

### 2.1. EJERCICIO 1

Si seleccionamos en dicha opción del menú se nos desplegará una tabla con la solución propuesta en el ejercicio 1. En este ejercicio se nos pedía que desplegaráramos un listado de empleados con los siguientes campos: (id, nombre, apellidos, departamento, cargo, salario y fecha de contratación). La lista está ordenada por fecha de contratación, limitada a 50 filas y puesto que existen varios salarios para una misma persona con diferentes fechas, nos quedamos con el último salario.

La consulta queda de la siguiente forma:

```
SELECT employees.emp_no, employees.first_name,  
employees.last_name, employees.hire_date, titles.title,  
departments.dept_name, MAX(salary)  
FROM salaries, employees, titles, departments, dept_emp  
WHERE departments.dept_no=dept_emp.dept_no AND  
dept_emp.emp_no=employees.emp_no AND  
titles.emp_no=employees.emp_no AND  
employees.emp_no=salaries.emp_no group by employees.hire_date  
LIMIT 50
```

## 2.2. EJERCICIO 2

Esta pestaña del menú despliega un listado de empleados con la solución propuesta para el ejercicio 2. Este listado contiene los siguientes campos: (id, nombre, apellidos, género, departamento, cargo, salario, fecha de contratación y fecha de nacimiento). El listado está ordenado por la fecha de contratación y limitado a 50 filas.

La consulta queda de la siguiente forma:

```
SELECT employees.emp_no, employees.first_name,  
employees.last_name, employees.gender, employees.birth_date,  
employees.hire_date, titles.title, departments.dept_name, MAX(salary)  
FROM salaries, employees, titles, departments, dept_emp  
WHERE departments.dept_no=dept_emp.dept_no AND  
dept_emp.emp_no=employees.emp_no AND  
titles.emp_no=employees.emp_no AND  
employees.emp_no=salaries.emp_no group by employees.hire_date  
LIMIT 50
```

## 2.3. EJERCICIO 3

En esta opción del menú he implementado un formulario para creación de un nuevo empleado. Los campos a insertar son los siguientes: (nombre, apellidos, fecha de nacimiento, género, departamento, cargo y salario). El identificador asociado a cada nuevo empleado es un campo oculto que contiene el siguiente identificador libre. He tenido que crear este campo puesto que en la base de datos para empleado no ha sido marcada la casilla de “autoincremento” para la tabla “employees” para que lo haga de forma automática.

Cada uno de los campos a insertar son obligatorios y en caso de dejar alguno vacío, se muestra un mensaje de error pidiendo que se rellene, a la vez que el formulario no es procesado si hay errores.

Estas comprobaciones han sido implementadas haciendo uso de “JQuery” y además existe una función para comprobar si la fecha insertada es una fecha válida.

Cuando todos los datos insertados son válidos, se procesa la información y se redirige nuevamente al formulario de inserción de un nuevo empleado.

## 2.4. EJERCICIO 4

Esta pestaña reúne las soluciones a los últimos tres ejercicios propuestos. Si hacemos “click” sobre ella aparecen dos pestañas desplegables. La primera nos permite mostrar un listado de empleados por departamento. Este listado

está ordenado por fecha de contratación y está limitado a 50 filas. El segundo, nos mostrará un listado de empleados por cargo. También está ordenado por fecha de contratación y limitado a 50 filas.

En el ejercicio 5 se nos pedía que incorporáramos a los listados la opción de hacer “click” en cada identificador y que nos mostrara información sobre el empleado seleccionado. Esta información se mostrará justo debajo del listado con los datos en color rojo.

Por último y como requisito adicional, se nos pedía que implementáramos un botón para insertar un nuevo empleado. Este botón lo he insertado solo en la tabla que muestra el listado de empleados por departamento. Cuando hacemos “click” sobre dicho botón, aparece un formulario (el mismo formulario que el pedido en el ejercicio 3). Este formulario también comprueba que los campos no estén vacíos, y en caso de estarlos lo indica con un mensaje. Cuando todos los campos son correctos, procesamos la información para insertarlo.

### **3. HERRAMIENTAS UTILIZADAS**

El editor usado para programar el ejercicio ha sido sublime text.

Para probar su funcionamiento lo he alojado en un servidor local usando WAMP.

La base de datos la he gestionado MySql WorkBench.

### **4. ANOTACIONES**

Dentro del proyecto “APIRest”, existe un fichero llamado “conexión.php”. Este fichero es el encargado de realizar la conexión con la base de datos. Para ser probado es necesario modificar los campos “DB\_DSN” con el nombre de la base de datos y “DB\_USER” con el usuario. En caso de disponer de contraseña en la BD, sería necesario añadir una nueva variable “DB\_PASS” con la clave de la BD.