

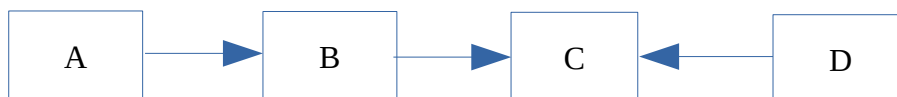
PRÁCTICA 6 de HTML: Final

Fecha tope entrega	Miércoles, 9 de marzo.	Fecha tope defensa	Viernes, 11 de marzo.
Tipo	Individual	Asunto en email	HLC06
Formato fichero	Acceso a un repositorio privado (colaborador) vuestro en Git		

Requisitos:

1.- Partiendo de la base de datos de la práctica 6 de Java que, de ser no haber sido aportada, debe desarrollarse y tener el visto bueno del profesor. El sistema de BDs será Derby y tendrá las siguientes características:

1.1.- Crea 4 tablas con una relación $1 \rightarrow N$ y otra $1 \rightarrow N \leftarrow 1$, como las del siguiente ejemplo :



1.2.- La tabla A sería algo como socio, cliente, usuario, etc. Para validarse en la aplicación con los campos usuario y contraseña. Tendrá obligatoriamente: NIF, foto y fecha.

* 1.3.- A, B y D tendrán sus PK tipo entero. En C la PK estará formada por 3 campos: las dos FKs procedentes de B y D y un contador relativo a la FK de B o D.

1.4.- Habrá un campo calculado en C que dependerá de valores de B y/o D.

1.5.- Existirán campos de tipo entero, real, alfanumérico y fecha.

2.- Desarrolla una aplicación Java Web con las siguientes características:

2.1.- Página de inicio: mostrará información general (autor, nombre, etc) y permitirá el inicio de sesión, que consistirá en validar una fila de la tabla A (socio, cliente, etc.) mediante el nombre y la contraseña. Una vez logeado se crea una sesión y permitirá el acceso a las dos páginas siguientes:

2.2.- Ver datos de B relacionados con el elemento de A validado.

2.2.1.- Desde cada elemento de B habrá un enlace a la siguiente página, siempre que tenga elementos de la tabla C relacionados.

2.3.- Ver datos de C relacionados con un elemento de B.

2.4.- Modo de visualización de las dos páginas anteriores:

2.4.1.- La tabla que tenga imagen se mostrará todos sus datos con article

2.4.2.- La otra tabla se mostrarán todos sus datos con tablas:

- Tendrá encabezado (`<tr><th>`).

- Efectos CSS: responsive (`<div style="overflow-x: auto;">`), diferencia visual entre las filas pares e impares (`"tr:nth-child(even)"`) y cambio de color al pasar el ratón (`"tr:hover"`).

2.5.- Si la tabla B y C no tienen imágenes pueden modificarse o cambiar la tabla B por la D. Esto modificaría el enunciado del apartado "2.3", referendario a D y a todos sus datos (ya que no están relacionados con A), y en el apartado "2.4", las filas de C relacionadas con D.

2.6.- Se podrá cerrar sesión desde todas las páginas donde este abierta, y nos redirigirá a la página de inicio con la sesión destruida.

3.- Debe usarse JSP y Servlet. Preferiblemente JSP para la vista y Servlet para las capas modelo y controlador

4.- Para la persistencia de datos (acceso a la BDs) se recomienda usar el framework Hibernate.

5.- Habrá control de sesión: validación, acceso a datos y logout. Evitar copiar y pegar URL y el retroceso del navegador.

6.- La salida será en HTML5 y se usará JavaScript, CSS y bootstrap.

7.- Se valorará la estética y la funcionalidad.

8.- Manual en PDF de los pasos seguidos en la configuración de Hibernate en esta práctica.

9.- La base de datos puede ser en Derby u otra con Docker. En ambos casos se entregará los ficheros para su creación, así como los pasos seguidos para la puesta en marcha, especialmente en Docker.

10.- Voluntario: usar JSF.