

Tarea 030 – API completa – formularios Cliente

Los resultados de todas las tareas incorporarán, además del código fuente, los comentarios precisos y necesarios para su fácil comprensión. No escatimes esfuerzos en comentar el código, es una buena práctica para aprender, además de ser muy útil para modificaciones o reutilizaciones futuras.

Sería muy buena práctica añadir el título y el enunciado del ejercicio (como comentarios) al principio del código fuente.

Añade documentos en formato Word con capturas de la salida por pantalla (al ejecutar la página) si consideras que queda más clara la resolución del ejercicio.

Deberás entregar dos versiones del resultado:

1. **Un único documento en Word sin comprimir:** que contenga todas las líneas de código, y por orden, primero el código **html**, luego el código **css** y por último el código **js**. Este archivo servirá para que el profesor haga anotaciones y/o correcciones a los alumnos.
2. **Los archivos de código fuente:** en un único archivo, ya sea de extensión: **html**, **js**, o **css**. Si precisas entregar varios archivos, comprímelos en un único **zip**. (No admito rar).

El nombre del archivo entregado comenzará por tu nombre seguido por TareaXXX. Ejemplo: **federicoTarea014.zip**

Tarea 030 – API completa – formularios Cliente

Sobre las API desarrolladas en la tarea anterior:

1. Haz que la ruta raíz del servidor express cargue la pagina **html** con el **css** adjuntos. En el anexo “Pasos para crear una API” se explica como servir archivos estáticos.
2. Utilizando los archivos html y css adjuntos, se pide poner un menú para ofrecer al usuario las tareas especificadas en la tarea anterior del lado cliente. Este menú estará en la sección 1.
3. Al archivo .html sólo se le permite incorporar archivos de scripts o de css a este fichero HTML. Nunca añadir elementos HTML
4. Cuando el usuario especifique una de las opciones del menú anterior, si es necesario añadir un formulario en la sección 2. Los resultados se verán en las secciones 3 y 4 (a tu voluntad).
5. Ten muy presente la validación desde el lado cliente. OJO con las id de clientes (que no son automáticas, deben ser únicas y las especifica el usuario). Piensa qué se necesita del lado cliente y del lado servidor. ¿Se puede hacer con lo que hay en la API de servidor?
6. Procura que la interfaz de usuario sea clara, vaya borrando lo que no se necesite, etc.