

Tarefa 3.2

Axenda de contactos v4

Nome d@alumn@ *[Nome apelido1 apelido2]*

Grupo *2º DAM/DAW Ordinário*

Enunciado

Partindo do ficheiro `tarefa3.2.html` e do código implementado para a Tarefa 3.1 completa o código JS e CSS para obter o código da **Axenda de Contactos** tal e como se establece nos seguintes apartados.

Crea o ficheiro `tarefa3.2.js` a partir de `tarefa3.1.js` de xeito que a almacenaxe non se realice xa necesariamente nun *array* senón que se faga en IndexedDB, nunha base de datos chamada **AXENDA** e nun almacén de obxectos chamado **contactos**, de xeito que se conserve toda a funcionalidade definida a través das tres versións anteriores da tarefa.

IMPORTANTE: só poden modificarse os ficheiros `tarefa3.2.css` e `tarefa3.2.js`, e ningún outro.

Apartado 1: aspecto previo (0,5 puntos)

Crea o ficheiro `tarefa3.2.css` (a partir de `tarefa3.1.css`) de xeito que se manteña un estilo uniforme e homoxéneo e a funcionalidade acadadas nas tarefas anteriores **(0,5 puntos)**

Apartado 2: a funcionalidade de engadir contacto (2 puntos)

Ao engadir un novo contacto crearase o obxecto que pasará a inserirse no almacén **contactos**, preservando as características de funcionalidade e validación previas.

Apartado 2: a funcionalidade de modificar contacto (2 puntos)

Ao modificar un contacto recuperarase o obxecto da base de datos, para pasar a actualizalo no almacén **contactos**, preservando as características de funcionalidade e validación previas.

Apartado 3: a funcionalidade de buscar contacto (2 puntos)

Ao buscar un contacto recuperarase o obxecto da base de datos, para amosar os seus datos no formulario, preservando as características de funcionalidade e validación previas.

Apartado 4: a funcionalidade de eliminar contacto (2 puntos)

Ao eliminar un contacto retirarase o obxecto correspondente do almacén **contactos**, preservando as características de funcionalidade e validación previas.

Apartado 5: a funcionalidade de actualizar listaxe (1,5 puntos)

Calquera operación que implique cambios nos datos contidos na base de datos conlevará unha actualización da listaxe de contactos amosada por pantalla, de xeito que a listaxe sempre teña correspondencia cos datos actuais gardados no almacén **contactos**.

Outras condicións de avaliación

- Todo o código CSS terá que estar correctamente estruturado.
- Todo o código CSS terá que estar correctamente comentado, de xeito que os comentarios resulten explicativos e abondo detallados para entender en que consisten as regras aplicadas e a que elementos aplican.
- Todo o código JS terá que estar correctamente estruturado.
- Todo o código JS terá que estar correctamente comentado, de xeito que os comentarios resulten explicativos e abondo detallados para entender que se está a facer nos distintos bloques de código e para que serven.
- Non podes modificar o ficheiro **tarefa3.2.html**, xa que non se usará a túa versión para a corrección da tarefa.

O incumprimento dalgunha destas condicións de avaliación conlevará unha penalización máxima de -1 punto por cada unha delas.

Indicacións de entrega

Achega un ficheiro comprimido **ZIP** de xeito que só conteña:

- O ficheiro **tarefa3.2.js**.
- O ficheiro **tarefa3.2.css**.

Entrega a través desta plataforma, no xeito establecido para tal fin. O arquivo nomearase seguindo estas pautas:

apelido1_apelido2_nome – tarefa3.2.zip

Asegúrate que o nome non conteña caracteres especiais (como a letra ñ ou acentos). Así, por exemplo, unha alumna cuxo nome sexa María Begoña Sánchez Mañas debería nomear esta tarefa:

sanchez_manas_maria_begona – tarefa3.2.zip

CALQUERA TAREFA QUE NON CUMPRA AS INDICACIÓNS DE NOMENCLATURA E FORMATO ANTERIORES CONSIDERARASE COMA NON ENTREGADA.

NON SE ADMITIRÁ NINGUNHA ENTREGA FORA DE PRAZO. SE O PRAZO ESTÁ PRÓXIMO A FINALIZAR E TES ALGÚN PROBLEMA COA PLATAFORMA, FAI ENTREGA VÍA E-MAIL