

Examen Desarrollo Web en Entorno Servidor

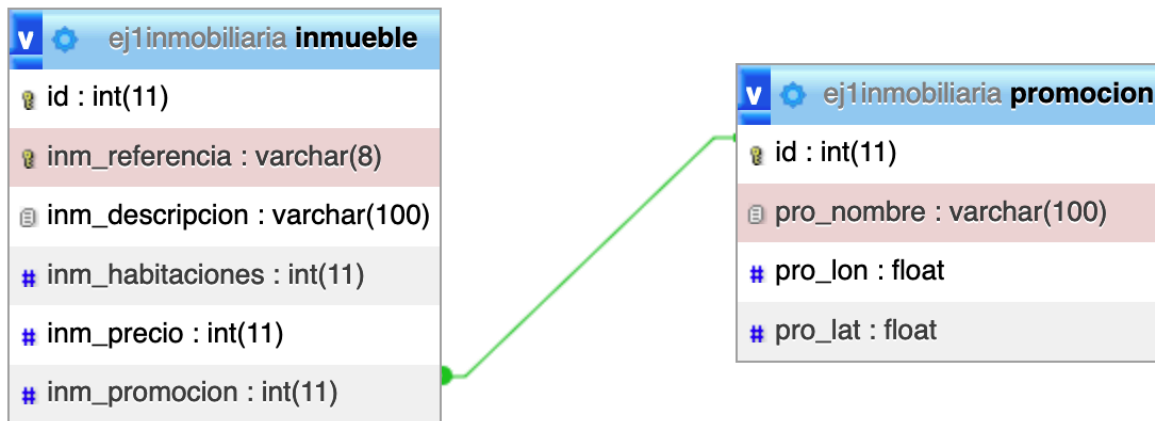
Curso	2º Desarrollo de Aplicaciones Web
Periodo	2ª Evaluación
Contenido	UT05 (Parte II), UT06
Fecha	16/02/2024

Consideraciones generales

- Para la realización de los ejercicios prácticos puedes utilizar los apuntes del módulo y los ejercicios prácticos que encontrarás en el Aula Virtual.
- También puedes recurrir a blogs o páginas de Internet para encontrar la forma de resolver los ejercicios, pero **no se pueden usar herramientas de IA** como ChatGPT, etc.
- Lo que **no está permitido** es **recibir ayuda de algún compañero o de cualquier otra persona** durante la realización del examen. Por lo tanto, no se podrá hacer uso de aplicaciones de mensajería instantánea, móviles, tablets o cualquier otro dispositivo que permita dicha comunicación.
- Durante la realización del examen se puede requerir la conexión por parte del alumnado a una **reunión telemática en la que debe compartir su pantalla** para que el profesor pueda supervisar la realización del examen.
- Debes reservar los **últimos 5 minutos** del examen para “**empaquetar**” y **subir tus soluciones** a los ejercicios al aula virtual, recuerda que pasada la hora de entrega no se podrá entregar la tarea, dándose el examen como **no presentado**.
- A la hora de empaquetar tus ejercicios crea una carpeta por cada ejercicio (Ejercicio1, Ejercicio2, etc.) sin espacios en el nombre y dentro incluye la solución (código fuente, proyecto de Netbeans, proyecto de VisualStudio, carpeta de Visual Studio Code, etc.). Luego comprime todas las carpetas en un fichero ZIP con el siguiente nombre: **Apellido1_Apellido2_Nombre.zip**. Es muy importante que el nombre del fichero ZIP no tenga acentos, eñes, caracteres especiales ni espacios.
- A la hora de evaluar los ejercicios, además del funcionamiento correcto de los mismos, se tendrán en cuenta otros factores como son:
 - **Utilización adecuada de los conceptos** vistos en clase.
 - **Legibilidad** del código (tabulaciones, espaciado de caracteres, nombres de variables y subprocesos, uso de **comentarios**, etc.).
- El profesor se reserva el derecho a **realizar una entrevista** (presencial o telemática) en caso de tener dudas sobre la autoría del examen por parte del alumnado.

Ejercicio 1. Symfony [2 puntos]

Partiendo del siguiente esquema de base de datos (ej1inmobiliaria.sql), debes crear un proyecto en Symfony (no es necesaria autenticación de usuarios) para realizar el CRUD de las tablas.



Debes tener en cuenta lo siguiente:

- Revisa el script de creación de las tablas para ver en los comentarios el significado de cada campo.
- El campo “id” de cada tabla es AUTO_INCREMENT.
- Debes crear un proyecto en Symfony (versión 7) que implemente...
 - **[0,5 puntos]** CRUD de la tabla “promocion”.
 - **[1 punto]** CRUD de la tabla “inmueble”
 - En los listados de inmuebles debe aparecer el nombre (no el id) de la promoción.
 - En los formularios para crear y editar inmuebles, la promoción se seleccionará de un desplegable en el que aparezca el nombre de la misma.
 - **[0,25 puntos]** Debes hacer que cuando se ponga en la URL la raíz del proyecto (https://localhost:8000) se muestre el listado de promociones.
- Adjunta al proyecto un documento en el que indiques los comandos de Symfony que has utilizado y qué hace cada uno de ellos.

Ejercicio 2. Cliente API [2,5 puntos]

Haciendo uso de la API REST gratuita, <https://www.themealdb.com/api.php>, la cual permite acceder a una base de datos de comidas, clasificadas por categorías, áreas e ingredientes.

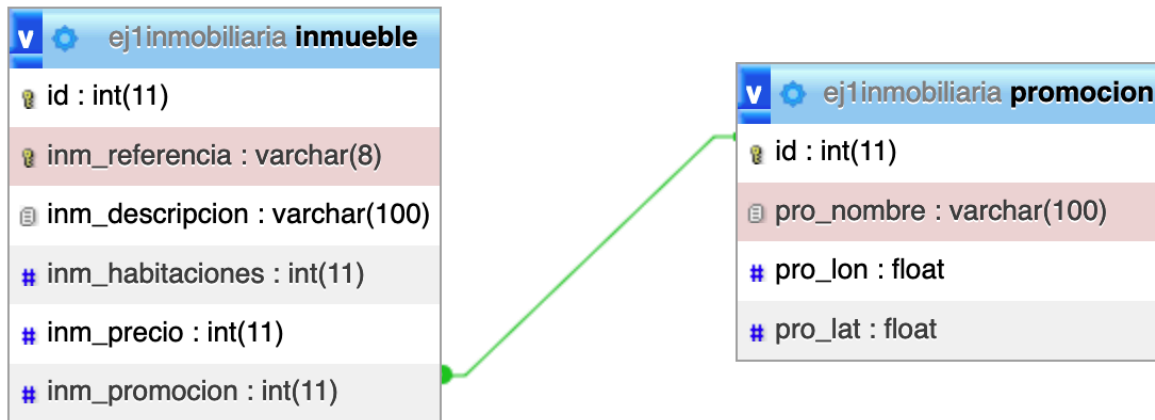
Debes implementar un script PHP con la siguiente funcionalidad (puedes usar varios scripts si lo deseas, pero teniendo en cuenta la portabilidad del código):

- **[0,5 puntos]** Inicialmente, se mostrarán las categorías de comidas disponibles para que el usuario seleccione una.
- **[1 punto]** Una vez seleccionada la categoría, pulsando un botón “Mostrar comidas” se mostrará la imagen de la categoría (strCategoryThumb) y un listado de las comidas de dicha categoría.
 - En el listado aparecerá únicamente el nombre de la comida (strMeal), que será un enlace para mostrar los datos de dicha comida.
- **[0,5 puntos]** Pulsando en el nombre de la comida se mostrará en pantalla la foto de la misma (strMealThumb), el nombre (strMeal) y las instrucciones (strInstructions) para su preparación.
- **[0,5 puntos]** Debes implementar algún mecanismo que permita que el usuario pueda volver a seleccionar otra categoría en cualquier momento y se repita el proceso.

*NOTA. La ausencia de comentarios en el código añadido por el/la alumno/a supondrá una penalización de **0,5 puntos** en la nota del ejercicio.*

Ejercicio 3. API REST [5,5 puntos]

Para este ejercicio partiremos del mismo esquema de base de datos del primer ejercicio:



Pero ahora debes implementar una API REST con autenticación mediante token, siguiendo el esquema visto en clase, y que siga la siguiente especificación...

- [1 punto] Endpoint para realizar el CRUD de la tabla “promocion”.
- [1 punto] Endpoint para realizar el CRUD de la tabla “inmueble”.
- [2 puntos] Endpoint que permita (mediante POST) incrementar en un % el precio de todos los inmuebles de una determinada promoción. El Endpoint recibirá el “id” de la promoción y el % de incremento.

Finalmente, debes implementar un script PHP que haga uso del endpoint anterior, de la siguiente manera...

- [0,75 puntos] El script deberá permitir al usuario autenticarse para usar la API REST.
- [0,75 puntos] Una vez autenticado, el usuario especificará la promoción y el % de incremento y pulsando un botón se actualizarán los precios de los inmuebles de esa promoción haciendo uso del endpoint.

*NOTA. La ausencia de comentarios en el código añadido por el/la alumno/a supondrá una penalización de **0,5 puntos** en la nota del ejercicio.*