

Examen Desarrollo Web en Entorno Servidor - Primera evaluación

16/03/2023

1. Con la base de datos que se adjunta en la plataforma, crea un programa PHP que sea capaz de:
 - a. El primer archivo de la aplicación, *index.php*, nos mostrará la tabla de libros en una tabla HTML. (1 punto)
 - b. Al final de cada fila debe haber un botón que nos lleve a la siguiente página, *ejemplares.php*, donde se mostrará al comienzo el título del libro y debajo una lista con los ejemplares del libro seleccionado. Las líneas correspondientes a los ejemplares que no hayan sido devueltos se mostrarán con un fondo rojo. (3 puntos)
 - c. *ejemplares.php* incluirá un botón para volver a *index.php* y otro botón para añadir un ejemplar al libro seleccionado. La signature del nuevo ejemplar será una combinación de la primera letra del género del libro más dos dígitos, de tal forma que se siga el orden de las signatures, es decir, si ya hay 8 ejemplares que empiezan por P, el siguiente tendrá la signature *P09*. Al crear el ejemplar debe aparecer automáticamente en la lista. (3 puntos)
 - d. En *index.php* habrá un botón para acceder a *insertar.php*, en este archivo debe haber un formulario para insertar libros. El autor se elegirá mediante una lista desplegable donde se mostrarán los nombres de los autores. *insertar.php* incluirá un botón de aceptar y otro de cancelar. Mientras estemos aceptando seguiremos en *insertar.php*, al cancelar volveremos a *index.php* (3 puntos).

Está permitido usar tanto mysqli como PDO. Los datos de conexión a la base de datos deben ser:

- BD: biblioteca.
- Host: localhost.
- Usuario: super.
- Contraseña: 123456.
- Además debe poder ejecutarse en un servidor Linux, por lo que debéis respetar el uso de mayúsculas y minúsculas en los nombres de la base de datos, las tablas, las columnas...

Segunda evaluación

2. Utilizando POO de PHP, vamos a crear un sistema de clases para una empresa de informática:
 - a. En primer lugar creamos la clase *Trabajador*, con los atributos identificación, nombre y sueldo. (0,25 puntos)
 - b. De *Trabajador* van a aparecer dos clases heredadas: *Programador* (con atributos lenguaje y SGBD) y *Administrador* (con el atributo sistema operativo). (0,5 puntos)
 - c. La siguiente clase se llamará *Departamento*, con los atributos y métodos:
 - i. nombre (atributo).
 - ii. trabajadores (atributo, array de objetos *Programador* y *Administrador*).
 - iii. presupuesto mensual (atributo).
 - iv. inserta trabajador (método, solo se pueden insertar objetos de las clases *Programador* y *Administrador*). (0,75 puntos)
 - v. gastos (método, devuelve una suma del presupuesto del departamento y de todos los sueldos de sus trabajadores). (1 punto)
 - d. Hay que hacer una aplicación con un formulario para insertar trabajadores en un departamento. El departamento se creará directamente en el código de la aplicación, sin formulario. Si os resulta más fácil en la página puede haber dos formularios, uno para insertar programadores y otro para insertar administradores. (2,5 puntos)
 - e. Añadir el código JS o JQuery para comprobar:
 - i. Que el nombre del trabajador es una mayúscula seguida de minúsculas. (0,5 puntos)
 - ii. Que el sueldo es un número entre 1200 y 3000. (0,5 puntos)
 - iii. Que la identificación es la primera letra del nombre seguida de tres dígitos. (1 punto)
3. Con la base de datos que se adjunta en la plataforma, crea un programa con Laravel que sea capaz de:
 - a. Mostrar el contenido de la tabla Libros. Se hará en la ruta relativa “/lista”. (1 puntos)
 - b. Al final de cada libro debe aparecer un botón para llevarnos a la ruta de modificación del título del libro seleccionado. Se hará en la ruta relativa “/titulo/{isbn}”. La nueva página solo tendrá una caja de texto donde aparecerá el título actual del libro y un botón de aceptar. Al aceptar se guardarán los cambios. Podéis elegir si en vuestra aplicación al aceptar seguiremos en la misma página o volveremos a “/lista” (1,5 puntos por mostrar el formulario y 1,5 puntos guardar los cambios)