

Nombre y apellidos

Fecha

Examen práctico de DWCS (2ª evaluación)

Dado el siguiente enunciado se pide escribir en código PHP con objetos y acceso a base de datos PDO, los diferentes apartados que se desarrollan más abajo. Se deben utilizar sentencias preparadas, manejo de excepciones y transacciones. **Hacer comprobación de estricta de tipos sobre alguna propiedad o variable.**

ENUNCIADO

Un empleado es una persona que trabaja en un determinado departamento y que puede tener o no un supervisor. Disponemos también de la información de los familiares de algunos de sus empleados.

Nombre de las clases: Individuo, Empleado, Familiar, Departamento y OperacionesBBDD

Nota: El puntero PDO sólo se crea una vez.

EXAMEN

1. (2 puntos) Crear una clase base para un individuo, que reúna los atributos comunes a los empleados y familiares. Crear las clases hijas Empleado y Familiar. La clase Familiar no va a poder crear hijos. Cada empleado podrá constar de un número variable de familiares y tener o no un superior (clase Individuo). Debemos llevar la cuenta de cuantos empleados hay en nuestra empresa. Crear la clase Departamento. Las respectivas clases debe desarrollar sus constructores y destructores (si procede). Definir namespaces con **nombres completamente cualificados** y que las clases estén en una carpeta llamada Clases, y de ahí cuelguen las clases, interfaces, etc. Implementar una función de autocarga anónima que sea independiente del sistema operativo y del directorio en el cual se encuentre definido el archivo. Cuando se utilicen esas clases emplear alias.
2. Crear una clase OperacionesBBDD que aúne todas las operaciones con la BBDD. Debe tener como propiedades nombreBBDD, usuario, pwd, servidor y puntero PDO.
 - 2.1. (1 punto). Crear un método leer_configuración. Recibe dos parámetros (fichero xml y fichero xsd con el fichero de configuración y el fichero de validación respectivamente). Devuelve un array con los datos que necesito para después crear el constructor de esa clase. Escribir el destructor. Desarrollar los métodos que serializan y deserializan los datos.
 - 2.2. (2,5 punto). Crear un método borrar_familiares, que recibe el id de un empleado y borra todos los familiares asociados a ese empleado si es que tiene alguno. No puede quedar a medias.
 - 2.3. (2,5 puntos). Crear un método buscar_datos_empleado (\$id) que consulte a la BBDD, de forma que reciba el identificador del empleado, cree un objeto de tipo Empleado con los datos obtenidos de la BBDD y los saque por pantalla llamando a un método mágico. Tener en cuenta que el identificador lo va a recibir de un formulario y un empleado puede tener varios familiares y uno o ningún supervisor.
3. (2 puntos). Realizar un pequeño formulario donde se nos va a mostrar todos los empleados disponibles en la BBDD como un elemento desplegable. El procesamiento del formulario se hará en el mismo fichero, de forma que guarde en el servidor el identificador del usuario y nos redirige a la página (mostrar_datos.php) que nos va a permitir mostrar los datos de ese empleado.

Nombre y apellidos
Fecha

