

# 2 RECUPERATORIO - PRIMER PARCIAL – PROGRAMACION III – 1 cuat. 2021

## Aclaración:

Se debe crear un archivo por cada entidad de PHP. Todos los métodos deben estar declarados dentro de clases.

PDO es requerido para interactuar con la base de datos.

Se deben respetar los nombres de los archivos, de las clases, de los métodos y de los parámetros.

Realizando todos los puntos del parcial se podrá obtener una nota máxima de 6 (seis).

**Ciudad.php.** Crear, en **./clases**, la clase **Ciudad** con atributos públicos (id, nombre, poblacion, pais y pathFoto), constructor (con todos sus parámetros opcionales), un método de instancia ToJSON(), que retornará los datos de la instancia (en una cadena con formato JSON).

Agregar los siguientes métodos:

- **Agregar():** agrega, a partir de la instancia actual, un nuevo registro en la tabla **ciudades** (id, nombre, poblacion, pais, foto), de la base de datos **ciudades\_bd**. Retorna **true**, si se pudo agregar, **false**, caso contrario.
- **Traer():** retorna un array de objetos de tipo Ciudad, recuperados de la base de datos.
- **Existe(array):** retorna true, si la instancia actual está en el array de objetos de tipo Ciudad que recibe como parámetro (comparar por nombre y pais). Caso contrario retorna false.
- **Eliminar():** elimina de la base de datos el registro coincidente con la instancia actual (comparar por nombre y pais). Retorna **true**, si se pudo eliminar, **false**, caso contrario.
- **Modificar():** Modifica en la base de datos el registro coincidente con la instancia actual (comparar por id). Retorna **true**, si se pudo modificar, **false**, caso contrario.
- **GuardarEnArchivo(string):** escribirá (**./archivos**) en un archivo json (*ciudades\_modificadas.json* o *ciudades\_borradas.json*, según lo recibido por parámetro) toda la información de la ciudad, de la instancia actual, más la nueva ubicación de la foto (modificada o borrada).

**VerificarCiudad.php:** Se recibe por POST el parámetro **ciudad**, que será una cadena **JSON** (nombre y pais), si coincide con algún registro de la base de datos se retornará los datos del objeto. Caso contrario informar: si no coincide el nombre o el pais o ambos.

**AgregarCiudad.php:** Se recibirán por POST todos los valores: **nombre**, **poblacion**, **pais** y **foto** para registrar una ciudad en la base de datos.

Verificar la previa existencia de la ciudad. Si la ciudad ya existe en la base de datos, se retornará true (en éxito) y mostrará un mensaje que indique lo acontecido.

Si la ciudad no existe, se agregará a la base. La imagen guardarla en **“./ciudades/imagenes/”**, con el nombre formado por el **nombre** punto **pais** punto hora, minutos y segundos del alta (**Ejemplo: bernal.argentina.105905.jpg**).

Se retornará un **JSON** que contendrá: éxito(bool) y mensaje(string) indicando lo acontecido.

**ListadoCiudades.php:** (GET, sin parámetros) Se mostrará el listado **completo** de las ciudades (obtenidas de la base de datos) en una tabla (HTML con cabecera).

**Nota:** preparar la tabla (HTML) con una columna extra para que muestre la imagen de la foto.

Si es pasado el parámetro **dato** (con valor **“json”**), se retornará un **JSON** que contendrá: {dato:[]}, que será el array de objetos de tipo ciudad, recuperado de la base de datos.

**ModificarCiudad.php:** Se recibirán por POST los siguientes valores: **ciudad\_json** (id, nombre, poblacion, pais y pathFoto, en formato de cadena JSON) y **foto**, para modificar una ciudad en la base de datos. Invocar al método Modificar.

Si se pudo modificar en la base de datos, la foto modificada se moverá al subdirectorio `“./ciudadesModificadas/”`, con el nombre formado por el **id** punto **nombre** punto **pais** punto **'modificado'** (**Ejemplo: 10.liverpool.inglaterra.modificado.jpg**). Invocar al método `GuardarEnArchivo(“ciudades_modificadas.json”)`. Si recibe por GET (sin parámetros), se mostrarán en una tabla (HTML) la información de todas las ciudades modificadas y sus respectivas imagenes. Se retornará un **JSON** que contendrá: éxito(bool), mensaje(string) indicando lo acontecido y dato(array).

**Nota:** El valor del id, será el id de la ciudad 'original', mientras que el resto de los valores serán los de la ciudad modificada.

**EliminarCiudad.php:** Si recibe un **nombre** por GET, retorna si esa ciudad está en la base o no (mostrar mensaje). Si recibe por GET (sin parámetros), se mostrarán en una tabla (HTML) la información de todas las ciudades borradas y sus respectivas imagenes. Si recibe el parámetro **ciudad\_json** (id, nombre, y pais, en formato de cadena JSON) por POST, más el parámetro **accion** con valor **"borrar"**, se deberá borrar la ciudad de la base de datos. Si se pudo borrar, invocar al método `GuardarEnArchivo(“ciudades_borradas.json”)`. La foto se moverá al subdirectorio `“./ciudadesBorradas/”`, con el nombre formado por el **id** punto **nombre** punto **'borrado'** (**Ejemplo: 91218.madrid.borrado.jpg**). Retornar un JSON que contendrá: éxito(bool), mensaje(string) indicando lo acontecido y dato(array).