

Práctico Nro. 7 – Distribución en directorios. Organización de un proyecto completo. Funciones existentes en PHP. **2007**

1. Explique las diferencias entre las sentencias `include`, `require`, `include_once` y `require_once`. ¿En qué casos conviene usar cada una de ellas?

2. Si tiene dos archivos, `archivo1.php` y `archivo2.php`, ¿qué ocurre si `archivo1.php` incluye a `archivo2.php` y `archivo2.php` incluye a `archivo1.php`? ¿El resultado depende de la sentencia que se use para hacer la inclusión?

Corrobore su respuesta realizando el ejercicio en máquina y probando con las cuatro sentencias de inclusión.

3. Extraiga de las resoluciones de los prácticos anteriores todos aquellos procedimientos / funciones que se puedan reutilizar y organícelos en bibliotecas, es decir, en archivos separados con contenidos agrupados por funcionalidad. Por ejemplo, puede definir un archivo que provea manejo de strings, otro para trabajo con archivos, otro para arreglos, etc.

Una vez creadas las bibliotecas, y reorganice las soluciones de los ejercicios anteriores para que las utilicen, haciendo uso de las sentencias de inclusión que considere convenientes.

Nota: Es una práctica común utilizar la extensión `".inc.php"` para denominar los archivos de biblioteca (`".inc"` indica que es un archivo destinado a ser incluido por otros).

4. Se le entrega el código fuente de un proyecto organizado en archivos y directorios. Se le pide que cree un programa que procese todos los archivos y detecte inclusiones "circulares" entre archivos.

Sugerencia: Crear un arreglo que liste todos los archivos, usando como clave el nombre del archivo y como valor un arreglo con los archivos que incluye. La detección implica un recorrido por esta estructura.

5. Le solicitan que realice un pequeño sistema para administrar a los alumnos que asisten a las materias de una determinada carrera y los exámenes que rinden. Dicho sistema debe emplear archivos como medio de almacenamiento.

La funcionalidad requerida es la siguiente:

- Dar de alta un alumno
- Dar de baja un alumno (y todos los datos asociados a éste)
- Mostrar un listado de alumnos
- Inscribir un alumno en una materia
- Mostrar las materias en la que está inscripto un alumno
- Registrar los resultados de un examen
- Mostrar los resultados de un examen
- Mostrar todos los exámenes rendidos por un alumno (es decir, mostrar un listado de materia y calificación correspondiente)

¿Qué archivos sería necesario crear y con qué contenidos? ¿Cómo organizaría el proyecto en términos de bibliotecas?.

Realice un diagrama de dependencias, en el cual se representa con un rectángulo cada biblioteca y se dibuja una flecha de la biblioteca *A* hacia la *B* si *B* depende de *A*. *B* depende de *A* si utiliza funciones definidas en esta última. Realice también los diagramas de estructura para cada biblioteca.

Aclaración: no es necesario que implemente el sistema, sólo que plantee cómo lo estructuraría.

6. Para cada una de las siguientes funciones, indique para qué se emplea y qué parámetros puede recibir, y brinde al menos un ejemplo de uso, mostrando la salida (preferentemente, distinto a los ejemplos existentes en la ayuda de la función).

Ejemplo:

Función	Descripción	Parámetros
strlen	Función que devuelve la longitud de un string.	string \$string: el string de entrada

```
<?php
$ejemplo = array('a', 'b', 'c');
$ejemplo_str = join($ejemplo);
$long_ejemplo_str = strlen($ejemplo_str);
echo "La longitud de '$ejemplo_str' es de $long_ejemplo_str caracteres";
?>
```

La longitud de 'abc' es de 3 caracteres

- strpos
- strcmp
- strstr
- substr_count
- strtolower
- strtoupper
- trim
- sprintf
- date
- checkdate
- getdate
- number_format
- chdir
- getcwd
- opendir
- readdir
- scandir
- exit
- die
- eval
- sleep
- php_uname
- phpinfo
- phpversion
- putenv
- exec