Guía de Ejercicios - Programación III (PHP) - 2016

Profesores: Neiner Maximiliano, Villegas Octavio

**Parte 7** - Ejercicios con Archivos

**Aplicación Nº 34 (Contar letras)**

Se quiere realizar una aplicación que lea un archivo (../misArchivos/palabras.txt) y ofrezca estadísticas sobre cuantas palabras de 1, 2, 3, 4 y más de 4 letras hay en el texto. No tener en cuenta los espacios en blanco ni saltos de líneas como palabras.

Los resultados mostrarlos en una tabla.

**Aplicación Nº 35 (Copiar archivos)**

Generar una aplicación que sea capaz de copiar un archivo de texto (su ubicación se ingresará por la página) hacia otro archivo que será creado y alojado en ***./misArchivos/yyyy\_mm\_dd\_hh\_ii\_ss.txt***, dónde yyyy corresponde al año en curso, mm al mes, dd al día, hh hora, ii minutos y ss segundos.

**Aplicación Nº 36 (Copiar archivos invirtiendo su contenido)**

Modificar el ejercicio anterior para que el contenido del archivo se copie de manera invertida, es decir, si el archivo origen posee ‘Hola mundo’ en el archivo destino quede ‘odnum aloH’.

**Aplicación Nº 37 (Encriptar / Desencriptar archivos)**

Crear una página web que permita encriptar mensajes y que se guarden en un archivo de texto y que sólo si se lee el archivo desde la página se podrá acceder a su texto claro, es decir se desencriptará.

Crear la clase **Enigma**, la cual tendrá la funcionalidad de encriptar/desencriptar los mensajes.

Su método estático ***Encriptar***, recibirá un mensaje y a cada número del código ASCII de cada carácter del string le sumará 200. Una vez que cambie todos los caracteres, lo guardará en un archivo de texto (el path también se recibirá por parámetro). Retornará TRUE si pudo guardar correctamente el archivo encriptado, FALSE, caso contrario.

El método estático ***Desencriptar***, recibirá sólo el path de dónde se leerá el archivo. Realizar el proceso inverso para restarle a cada número del código ASCII de cada carácter leído 200, para poder retornar el mensaje y ser mostrado desesncriptado.

**Aplicación Nº 38 (Generar Tabla de Imágenes)**

Hacer un programa que muestre en una tabla de 4 columnas todas las imágenes del directorio "fotos". Para ello consulte en la web (en concreto la referencia de funciones de directorios). En el directorio sólo existirán fotos. Utilizar el tag <img> para la visualización.

**Aplicación Nº 39 (Generar Tabla de Imágnes II)**

Ídem al anterior, pero que muestre las fotos en un tamaño de 100x100 píxeles y que al pulsar se muestre la foto en su tamaño original en una página distinta (agregarle un link para poder volver a la página de inicio).

**Aplicación Nº 40 (Galería de Imágenes)**

Amplíe el ejercicio de la galería de fotos realizada anteriormente y permita al usuario añadir nuevas fotos. Para ello hay que poner el atributo enc\_type=”multipart/form-data” en el FORM y usar la variable $\_FILES['foto'].

**Aplicación Nº 41 (Files UpLoad)**

Se necesita crear una página que le permita al usuario subir al servidor web cualquier tipo de archivo. Sólo se restringirá el tamaño de cada archivo según el tipo de extensión que posea.

Para archivos con extensión **.doc** o **.docx** el tamaño máximo será de **60 Kb**.

Archivos con extensión **.jpg**, **jpeg** o **gif** el valor máximo será de **300 kb**.

Para el resto de las extensiones el máximo permitido será de **500 kb**.

Dichos archivos se almacenarán en una carpeta llamada **‘Uploads’** que se ubicará en el directorio raíz del servidor web.

Se deberá informar si se logró subir el archivo o no. Si se pudo, informar el nombre del archivo, su extensión y que tamaño posee.