Anexo II - Enviar correo electrónico

A través de la función de correo, las cabeceras MINE (“cabez”) son opcionales:

mail(“Dest”, “Asunto”, “Mensaje”, “Cabez”**)**

- **Dest:** Dirección del correo del destinatario.
- **Asunto:** Texto que aparecerá en el encabezado que recibirá el destinatario.
- **Mensaje**: texto que aparecerá en el *cuerpo del mensaje*.
- **Cabez**: Cabeceras MINE, pueden contener, (cada uno en una línea, sin punto y coma, sin comillas;
    - **From:** *Nombre* **<***e-mail***>**
        - **Nombre**: *nombre del remitente*, aparecerá en **De:** cuando el destinatario reciba el mensaje.
        - **e-mail** es *una dirección de correo*, debe ir entre < y >.
    - **Reply-To:** *correo*
        - **correo**: dirección a la que se enviará la respuesta si el destinatario usa **Responder**.
    - **Cc:** *correo1,correo2,...*
        - **correo1**, **correo2** direcciones a las que se envía *copia*.
    - **Bcc:** *correo1,correo2,...*
        - Lo mismo salvo que *los receptores sólo verán su dirección*.
    - **X-Mailer:PHP/".phpversion()**
        - *frivolidad* que indica que versión de PHP con que ha sido creado.
    - **Date:** *xxxxx*
        - *xxxxx:* fecha de envío, a obtener con las funciones de fecha.
    - **MIME-Version: 1.0**
        - especificará la versión MIME que ha de utilizar el cliente de correo para poder interpretar adecuadamente el contenido de los mensajes.
    - **Content-Type:** *tipo/subtipo;(ojo al punto y coma*)
        - Por defecto los mensajes se envían como **texto sin formato**, para especificar un formato hay que *incluir* este elemento, hay varios tipos de *tipo/subtipo*:
        - **text/plain** opción por defecto, señala que el contenido del mensaje es de tipo **texto** (*text*) y del subtipo **sin formato** (*plain*).

            En ambos casos tras el punto y coma se puede poner la especificación del tipo de alfabeto (**charset=**) seguida del tipo de codificación (charset=\"ISO-8859-1\ alfabeto latino).

        - **text/HTML** tipo **texto**, pero el *subtipo* es **HTML**, el mensaje se visualizará en formato **HTML** si el *cliente de correo* lo permite.
        - **image/jpeg** o **image/gif** para imágenes.
        - **audio/basic** para sonidos.
        - **video/mpeg** para video
        - **application/octet-stream** Ejecutables, comprimidos y otros ficheros adjuntos, después del *punto y coma*, **name=** seguido del *nombre y extensión* del fichero name=\"apachito.zip\
        - **multipart/alternative** Indica que el mensaje tiene *varias partes* (*multiplart*) de las que el destinatario *ha de ver una sola* (*alternative*).

- **multipart/mixed** permite *adjuntar ficheros* al mensaje.

En un multipart las partes de un mensaje deben ir separadas por el separador boundary.

boundary=*cadena*

1.1 El cuerpo del mensaje

Cuando se trata de un *multiparte* el mensaje ha de contener: los *separadores* de las diferentes partes, los *encabezados* de cada una de las partes y sus respectivos *contenidos*.

La secuencia habría de ser de este tipo:

- Separador
- Content-type
- Content-Transfer-Encoding
- **Content-Disposition
- **Lectura del fichero
- **Codificación
- Inserción en cuerpo
- Separador
- .....
- otra parte
- ...
- Separador final

Los apartados ** sólo se incluirían si junto con el mensaje se adjuntan ficheros.

- **Content-Transfer-Encoding** puede especificar una de los siguientes codificaciones:

  7BIT **8BIT** **BASE64** **BINARY** **QUOTED-PRINTABLE** (ver libro).

- **Content-Disposition**: Permite dos opciones: **inline** o **attachment**. **Inline** permite que los contenidos se visualicen junto con el cuerpo del mensaje mientras que con **attachment** sólo aparecerían como ficheros adjuntos.
  Este elemento del encabezado lleva -*separada por punto y coma*- una segunda parte. **filename=**, donde se puede especificar entre comillas un nombre y una extensión (igual o distinta de la original) con la que se denominará al fichero en el mensaje recibido.
- **Lectura del fichero**
  Cuando se trata de insertar un *fichero* el proceso es el típico de lectura de ficheros, es decir: Hay que crear el identificador de recurso del fichero en modo **lectura**. Recoger en una variable el *buffer* de lectura. Cerrar el fichero.
- **Codificación**
  Una vez recogido en el fichero a transmitir en una variable, el paso siguiente es *codificar* esa variable. La codificación más habitual y flexible –**base64**– requerirá el uso de dos nuevas funciones PHP:

  base64_encode

  chunk_split

- Mediante la primera se realiza la codificación propiamente dicha mientras que la segunda organiza el fichero codificado en trozos, de 76 caracteres cada uno, insertando detrás de cada uno un salto de línea.

- **Inserción en el cuerpo**
  La fase final del proceso es la de *agrupar* los diferentes *trozos* en una sola variable, que será la que se insertará como parámetro texto en la función e-mail.
- La inserción de ficheros adjuntos requiere que éstos estén disponibles en el servidor por lo que, antes de enviarlos, habrá que *subirlos* al servidor utilizando un proceso como el que hemos analizado cuando hablábamos de *Transferencia de ficheros*.