

PHP

1. **header.php.** Realiza un script PHP que utilice la función `header` para realizar enviar una cabecera HTTP al navegador del usuario y hacerlo que navegue por ejemplo a otra página de nombre *saludo.php*.
Reflexiona sobre el uso que puede tener la función `header` en un formulario de inicio de sesión.
2. **config.php. / include.php** Es muy común en PHP que tengamos un fichero con las variables y constantes que utiliza nuestra aplicación web para trabajar: directorios de subida de imágenes, directorios donde se almacenan PDFs, carpetas temporales, etc. Por otro lado, es posible también utilizar variables definidas en otras páginas... Implementa un fichero **config.php** con la definición de una serie de constantes en PHP que utilicen **define** para realizar estas definiciones de constantes..Por otro lado, en el mismo archivo define varias variables.
Crea un archivo, de nombre **include.php**, que ilustre el funcionamiento descrito anteriormente incluyendo el fichero **config.php** y presentando por pantalla las constantes y variables definidas en el fichero de configuración.
Reflexiona qué utilidad puede tener esto en nuestras aplicaciones web.
3. **filegetcontents.php.** Ilustra el funcionamiento de la función `file_get_contents` de PHP descargando el contenido remoto de un sitio web en una cadena y mostrándolo por pantalla.
4. **webscrapping.php.** En base al ejercicio anterior, realiza un script en PHP que dada una URL recibida en un campo de tipo texto de un formulario realice la descarga de su código fuente HTML y obtenga las direcciones de correo electrónico.
5. **pruebasession.php.** Realiza un script en PHP que ilustre el funcionamiento de las sesiones en PHP. Confirma que las sesiones se mantienen en el sitio web incluso aún cambiando de página web dentro del mismo servidor. Prueba a cerrar el navegador y acceder de nuevo a la página/s para verificar que si la sesión persiste o no.
6. **cookies.php.** Crea diferentes cookies y envíalas al cliente, muéstralas por pantalla y verifica su persistencia aún cerrando el navegador. Familiarízate con la manera de establecer las cookies mediante la función `time()` y piensa cómo podemos forzar al navegador del usuario para “renovar” sus cookies.
7. En base al ejercicio anterior, modifica las cookies desde el navegador web con alguna extensión propia del navegador que estés utilizando; por ejemplo, la popular extensión Web Developer de chrispederick.com. ¿Qué problemas pueden existir si el “lado cliente” puede “modificar” las cookies de sesión? ¿Cómo podemos evitar este riesgo en PHP?
Realiza un script de nombre **cookieSegura.php** que impida que el “lado cliente” modifique las cookies que envía el servidor.