

PUBLICACIÓN DE PÁGINAS WEB.

XAMPP.

- Paquete de software libre WAMP/LAMP/MAMP
- (Windows|Linux|Multiplataforma Apache MySQL PHP)
- Herramienta de desarrollo que permite probar nuestro trabajo (páginas web, programas, etc.), a nivel local sin necesidad de acceso a Internet.
- Incluye:
 - Servidor web Apache.
 - SGBD MariaDB que sustituye a MySQL.
 - phpMyAdmin para gestionar bases de datos.
 - PHP:
 - Preprocesador de hipertexto - Hypertext PreProcessor.
 - Lenguaje de scripts para desarrollo web del lado servidor.
 - El código PHP se ejecuta en el servidor.
 - Servidores FTP como Filezilla Server.
 - Webalizer.
 - Software de analítica web.
 - Otras aplicaciones:
 - Apache Tomcat, OpenSSL, XML, PERL.
 - Carpeta htdocs.
 - Sirve para almacenar los sitios web a los que luego se accederá mediante un navegador.
 - Rutas de acceso al sitio web:
 - <https://localhost/ruta de acceso/página de inicio.html>
 - Ejemplo:
 - <https://localhost/sitios/prueba/> (Se puede obviar la página web si el archivo es index.html)
 - <https://localhost/sitios/prueba/primeraPágina.html>
 - Si sólo se escribe localhost se accede al panel de control o interfaz del servidor. (dashboard).

LOCALHOST.

- Dispositivo local, servidor local o anfitrión local.
- Se refiere al ordenador o dispositivo local que se está usando, es decir, al propio equipo.
- **Acceso al servidor local:**
 - Con nombre:
 - localhost
 - Con dirección IP:
 - IPv4 → 127.0.0.1
 - IPv6 → ::1 o 0000:0000:0000:0000:0000:0000:0000:0001

CONCEPTOS DE FTP Y PUBLICACIÓN WEB.

PROTOCOLOS PARA TRANSFERIR ARCHIVOS.

FTP. (pág. 37-44)

- Protocolo de Transferencia de Archivos – File Transfer Protocol.
- Método más habitual para transferir archivos entre ordenadores.
- Puertos:
 - 20:

- Conexión de datos.
 - Transferencia de archivos y datos.
 - 21
 - Conexión de control.
 - Conexión remota al servidor.
 - Negociación de la seguridad y autenticación.
- Poco seguro.
 - Transmite los datos en modo texto (ASCII) sin cifrar.
 - Puertos desde el lado cliente superiores al 1024.
- Lista de puerto lógicos para red.
 - https://es.wikipedia.org/wiki/Anexo:Puertos_de_red

Modos de conexión. (pág. 55-56)

- **Modo activo.**
 - Nativo de FTP.
 - Cliente inicia conexión control.
 - Servidor inicia conexión datos.
- **Modo pasivo.**
 - Cliente inicia conexión control.
 - Cliente inicia conexión datos.

SFTP - Secure Shell FTP. (pág. 38-41)

- Protocolo seguro de transferencia de archivos.
- Seguridad garantizada mediante cifrado usando claves basadas en SSH.
- Utiliza el puerto 22.
- Transmite los datos en binario cifrados.

Tipos de transferencias de datos. (pág. 56)

ASCII.

- Transmite byte a byte.
- Ideal para transmitir archivos de texto, HTML, etc.

Modo binario.

- Transmite bit a bit.
- Ideal para transferir archivos no de texto como los multimedia, ejecutables, etc.

Modo automático.

- Detecta el tipo de archivo y establece en consecuencia el tipo de transferencia.

FTPS – FTP IMPLICITO. (pág. 42).

- El cliente establece una conexión de control con el puerto 21 del servidor.
- Se establece una conexión SSL/TLS.
- Si el servidor no soporta FTPS se cierra la conexión.
- Todas las conexiones son cifradas (control y datos).
- El cliente y el servidor no negocian.

FTPES -FTP EXPLICITO.

- El cliente establece una conexión de control con el puerto 21 del servidor y se solicita explícitamente una conexión segura.

- Envío de los comandos AUTH SSL y AUTH TLS y si el servidor los soporta establece la conexión SSL o TLS.
- Si el servidor no soporta FTPES ofrece al cliente la posibilidad de usar FTP no seguro.
- Se negocia que partes de la comunicación, control y datos, serán cifradas.

PROTOCOLOS DE SEGURIDAD – CIFRADO WEB.

SSH – Secure Shell. (pág. 38-41)

- Protocolo de acceso remoto a un servidor mediante un canal seguro en el que la información está cifrada.
- Permite:
 - Copia de datos segura.
 - Gestionar claves de tipo RSA.
 - Transfiere datos por un canal seguro tunelizado.
 - Ejecutar comandos remotos.
- Usa el puerto 22.
- Ofrece autenticación, confidencialidad e integridad.
- Se autentican los 2 extremos de la conexión.
- Se cifran los datos transmitidos:
 - Información, nombres de usuario, contraseñas, etc...

SSL – Secure Socket Layer. 1995

- Capa de conexiones seguras.
- Protocolo que utiliza certificados digitales para establecer comunicaciones seguras vía internet.
- Sustituido por TLS.
- Encripta los datos.
- Usado en bancos, tiendas virtuales y servicios que requieren envío de datos personales y contraseñas.
- Ofrece autenticación, confidencialidad e integridad.
- Última versión 3.0.

TLS – Transporte Layer Security. 1999.

- Seguridad en las capas de transporte.
- Protocolo que utiliza certificados digitales y algoritmos criptográficos para establecer comunicaciones seguras vía internet.
- Versión actual TLS 1.3 (2018).
- Es más seguro, flexible y eficiente que SSL.
- Más seguridad en el cifrado.
- Se lleva a cabo la comunicación mediante un canal seguro y cifrado en el que se negocia la criptografía de mensaje, se autentican las claves de cifrado y se realiza una transmisión segura.
- Sustituye a SSL 2.0 y 3.0.
- Como SSL sigue siendo muy conocido, muchos proveedores usan el término SSL o SSL/TLS, pero todos se refieren a la versión actual de TLS, la 1.3.

FTP MODO COMANDO.

- Windows, Linux, Mac OS-X, etc., disponen de la opción para conectarnos a un servidor ftp mediante consola.
- Uso desde Windows.
 - Hay que abrir la consola CMD o PowerShell de Windows como administrador:
 - Inicio/Ejecutar/CMD o PowerShell.

- Inicio/Buscar/CMD o PowerShell.
- Buscar/CMD o PowerShell
- Inicio/Sistema de Windows/Símbolo del Sistema o PowerShell.
- Hay que ejecutar CMD o PowerShell como administrador para ir a C:\Windows\System32.
- Si no se abre como administrador, hay que ir a System32, si no está establecida la ruta al ejecutable de ftp ([ftp.exe](#)) en las variables de entorno de Windows.

COMANDOS DE MSDOS O WINDOWS. (pág. 22 a 25 incluidos comandos de Linux)

- **cd.**
 - Permite abrir un directorio o carpeta, o cerrarlo y subir a un nivel superior.
 - Sintaxis:
 - cd nombre de directorio (Abre una carpeta)
 - cd ruta de acceso/nombre de directorio (Abre una carpeta especificando la ruta para llegar a ella).
 - cd.. (Cierra la carpeta o directorio actual).
 - Ejemplos:
 - cd Windows
 - cd C:\Windows\System32
 - cd..
- **dir.**
 - Muestra el contenido de un directorio.
 - Si la carpeta tiene mucho contenido y no se ve todo, se pueden hacer pausas usando el parámetro /p:
 - Ejemplo:
 - C:\Windows\System32>dir /p
- **md.**
 - Crea una carpeta en la ubicación local.
 - Primero hay que ubicarse en la carpeta donde se va a crear la nueva usando el comando cd.
 - Sintaxis:
 - md nombre de la carpeta nueva.
 - Ejemplo:
 - md nueva
- **rd.**
 - Borra una carpeta en la ubicación local.
 - Primero hay que ubicarse en la carpeta donde está la carpeta a borrar usando el comando cd.
 - Sintaxis:
 - rd nombre de la carpeta a borrar.
 - Ejemplo:
 - rd nueva
- **del.**
 - Permite borrar un archivo local.
 - Hay que estar en la carpeta que contiene el archivo a borrar, para ello se usa cd.
 - Se pueden usar comodines (* y ?).
 - Sintaxis:
 - del nombre del archivo a borrar.
 - Ejemplo uso *:
 - del uno.html
 - del *.jpg

- del pagina.*
 - del *.*
- Ejemplo uso ?:
 - Hay que usar tantas interrogaciones como caracteres lleve el patrón.
 - l????.jpg borraría luisa.jpg, lucia.jpg, lucas.jpg, pero no luis.jpg que tiene un carácter menos que el patrón.
- **exit.**
 - Cierra CMD o Powershell.

COMANDOS FTP. (pag. 45,46,50,51)

- **ftp.**
 - Iniciar el cliente ftp.
 - Conexión sólo al cliente:
 - Uso del comando ftp.
 - Ejemplo:
 - C:\Windows\System32>ftp
 - ftp>
 - Conexión a cliente y servidor:
 - ftp nombre servidor
 - Ejemplos:
 - ftp 127.0.0.1
 - ftp localhost
 - ftp ftp.udc.es
 - Ejemplo:
 - C:\Windows\System32>ftp
 - ftp>
- **close.**
 - Cierra la conexión al servidor, pero no desconecta el cliente ftp.
- **quit.**
 - Cierra la conexión al servidor y cierra el cliente ftp.
- **bye.**
 - Cierra la conexión al servidor y cierra el cliente ftp.
- **open.**
 - Permite conectarse a un servidor cuando ya se está conectado al cliente ftp.
 - Sintaxis:
 - open nombre del servidor número de puerto
 - ejemplos:
 - open 127.0.0.1 21
 - open localhost
 - open ftp.uma.es
 - identificación tras conectarse:
 - Especificar usuario (puede ser anonymous).
 - Escribir su contraseña si la tiene.
- **user.**
 - Conectar un usuario al servidor.
 - El servidor debe estar ya conectado.
 - Pide a continuación la contraseña.
 - Sintaxis:
 - user nombre de usuario.
 - Ejemplo:

- user ángel
 - Se puede especificar usuario y contraseña a la vez:
 - Sintaxis:
 - user nombre de usuario contraseña
 - Ejemplo:
 - user angel 1234.
- **help.**
 - Muestra el listado de comandos que se pueden utilizar o información sobre el comando especificado.
 - Sintaxis:
 - help
 - help nombre de comando
 - Ejemplos:
 - help (Muestra listado comandos).
 - help mkdir (Muestra información sobre el comando mkdir).
- **?.**
 - Muestra el listado de comandos que se pueden utilizar o información sobre el comando especificado.
 - Sintaxis:
 - ?
 - ? nombre de comando
 - Ejemplos:
 - ? (Muestra listado comandos).
 - ? mkdir (Muestra información sobre el comando mkdir).
- **verbose.**
 - Activa o desactiva el modo detallado, que muestra más o menos información sobre las operaciones que se están realizando.
 - Sintaxis:
 - verbose (para activar/desactivar).
- **bell.**
 - Activa un sonido que se produce tras cada instrucción de transferencia completada.
 - Por defecto no activo.
 - Sintaxis:
 - bell (para activar/desactivar).
- **type.**
 - Muestra el modo de transferencia de archivos activo.
 - Por defecto ASCII.
 - Sintaxis:
 - type (para activar/desactivar).
- **ascii.**
 - Cambia al modo ascii de transferencia de archivos.
 - Tipos de archivos: txt, html, json, xml, css...
 - Sintaxis:
 - ascii.
- **binary.**
 - Cambia al modo binario de transferencia de archivos.
 - Tipos de archivos: mp3, avi, mp4, jpg, docx, xlsx, wav...
 - Sintaxis:
 - binary.

- **pwd.**
 - Muestra el directorio en el que nos encontramos (actual) en el equipo remoto.
 - Sintaxis:
 - pwd.
- **cd.**
 - Permite cambiar a un directorio determinado en el ordenador remoto.
 - Sintaxis:
 - cd nombre del directorio.
 - cd ruta de acceso/directorio.
 - Ejemplos:
 - cd nuevo (cambio al directorio nuevo desde cualquier lugar).
 - cd / (vuelta al directorio raíz).
 - cd nuevo/otro (cambio al directorio “otro” que está dentro del directorio “nuevo” que está dentro del raíz).
 - cd /nuevo y no cd nuevo si queremos salir de la carpeta otro y volver a la de nivel superior.
- **ls.**
 - Muestra un listado con todos los archivos y carpetas que haya en la carpeta actual remota.
 - Sólo muestra los nombres de los archivos y las carpetas.
 - Sintaxis:
 - ls
- **dir.**
 - Muestra un listado con todos los archivos y carpetas que haya en la carpeta actual remota incluyendo:
 - Permisos, tamaño archivos y carpetas, canal (ftp), fecha/hora y nombre de archivos y carpetas.
 - También se puede mostrar el contenido de una carpeta sobre la cual no estemos posicionados especificando la ruta hasta ella.
 - Sintaxis:
 - dir
 - dir /ruta de acceso a carpeta cuyo contenido se quiere mostrar.
 - Ejemplos:
 - dir (muestra el contenido de la carpeta actual).
 - dir /carpeta1/carpeta1A (muestra el contenido de la carpeta carpeta1A).
- **!dir.**
 - Muestra un listado con todos los archivos y carpetas que haya en la carpeta local actual incluyendo:
 - Tamaño archivos y carpetas, fecha/hora y nombre de archivos y carpetas.
 - Sintaxis:
 - !dir
- **lcd.**
 - Cambia el directorio de trabajo a nivel local estando en ftp.
 - El directorio queda listo para desde él subir cosas al remoto, o a él descargar desde este último.
 - Sintaxis:
 - lcd nombre del directorio.
 - lcd ruta de acceso/directorio.
 - Ejemplos:
 - lcd nuevo (cambio al directorio nuevo desde la carpeta previa).

- lcd c:/nuevo/otro (cambio al directorio “otro” que está dentro del directorio “nuevo” que está dentro del disco duro en Windows).
 - lcd c:/nuevo (estando en la carpeta “otro”, cambio al directorio “nuevo”, es decir, se sale de una carpeta para ir a la previa en el disco duro en Windows).
 - lcd /nuevo/otro (cambio al directorio “otro” que está dentro del directorio “nuevo” que está dentro del directorio raíz en Linux).
- **!.**
 - Permite salirse temporalmente al equipo local para ejecutar comandos de Windows o de otro sistema operativo.
 - No desconecta el servidor ni cierra el cliente ftp.
 - Para volver al servidor ftp se usa exit.
- **mkdir.**
 - Crea una carpeta en la ubicación remota.
 - Primero hay que ubicarse en la carpeta donde se va a crear la nueva usando el comando cd.
 - También se puede especificar una ruta a la carpeta dentro de la cual se creará la nueva.
 - Si se quiere crear a nivel local hay que salirse temporalmente al equipo local (!) y usar el comando md (explicado en comandos de Windows).
 - Sintaxis:
 - mkdir nombre de la carpeta nueva.
 - Ejemplo:
 - mkdir nueva
 - mkdir /carpeta1/carpeta2/nueva
- **rmdir.**
 - Borra una carpeta en la ubicación remota.
 - Primero hay que ubicarse en la carpeta donde está la carpeta a borrar usando el comando cd.
 - También se puede especificar una ruta a la carpeta que se va a borrar.
 - Para borrar una carpeta debe estar vacía y hay que salirse de ella al nivel superior.
 - Si se quiere borrar a nivel local hay que salirse temporalmente al equipo local (!) y usar el comando rd (explicado en comandos de Windows).
 - Sintaxis:
 - rmdir nombre de la carpeta a borrar.
 - Ejemplo:
 - rmdir nueva
 - rmdir /carpeta1/carpeta2/nueva
- **prompt.**
 - Activa/desactiva el modo interactivo.
 - Pide/Deja de pedir confirmación.
 - Sintaxis:
 - prompt.
- **hash.**
 - Cada bloque de datos transferido se alterna con el signo #.
- **status.**
 - Muestra el estado de activación/desactivación de los comandos:
 - verbose, prompt, type, hash, glob y bell.
- **put.**
 - Permite subir o copiar un archivo al servidor.
 - Hay que estar en la carpeta que contiene el archivo a subir/copiar, para ello se usa lcd.
 - Con cd seleccionar la carpeta remota donde se van a copiar los archivos.

- No pide confirmación.
- Si el archivo tiene un nombre compuesto con espacios en blanco, hay que especificarlo entre comillas dobles.
- Sintaxis:
 - put nombre de archivo.extensión
 - put “nombre de archivo compuesto.extensión”
- Ejemplos:
 - put paisaje.jpg
 - put index.html
 - put “primera página web.ntml”
- **mput.**
 - Permite subir o copiar varios archivos al servidor.
 - Hay que estar en la carpeta que contiene el archivo a subir/copiar, para ello se usa lcd.
 - Con cd seleccionar la carpeta remota donde se van a copiar los archivos.
 - Pide confirmación:
 - Respuesta Si → y o yes. (Si no se especifica, valor por defecto).
 - Respuesta No → n o no.
 - Admite comodines:
 - Asterisco (*)
 - Representa un conjunto de caracteres distintos.
 - Sólo se incluye en el archivo lo que es común a todos.
 - Ejemplo:
 - *.png (dibujo.png, foto.png, playa.png)
 - pagina.* (pagina.html, pagina.docx)
 - pagina*.* (pagina1.html, paginanueva.docx, pagina24.html)
 - pagina*.html (pagina1.html, pagina24.html)
 - *.* (Todos los archivos con todas las extensiones).
 - Interrogación (?).
 - Representa un único carácter.
 - Se utilizan tantas interrogaciones como caracteres a sustituir.
 - Las interrogaciones representan los caracteres distintos.
 - Ejemplos:
 - L???? Mostraría Lucia Laura Lucas y no Luis o Lorena.
 - L???a Mostraría Lucia Laura pero no Lucas.
 - Si el archivo tiene un nombre compuesto con espacios en blanco, hay que especificarlo entre comillas dobles.
 - Sintaxis:
 - mput nombre de archivo1.extensión nombre de archivo2.extensión nombre de archivoN.extensión
 - mput “nombre de archivo compuesto1.extensión” “nombre de archivo compuesto2.extensión” “nombre de archivo compuestoN.extensión”
 - Ejemplos:
 - mput dibujo.png foto.png playa.png
 - mput *.png
 - mput “pagina principal.html” index.html
 - mput pagina.html , pagina.docx
 - mput pagina.*
 - mput *.*
 - mput pa?????1.* (pagina1.html pagina1.jpg partes1.png)
- **get.**

- Permite descargar o copiar un archivo al equipo local.
 - Hay que estar en la carpeta que contiene el archivo a descargar/copiar, para ello se usa `cd`.
 - El destino local también tiene que estar seleccionado con `lcd`.
 - No pide confirmación.
 - Si el archivo tiene un nombre compuesto con espacios en blanco, hay que especificarlo entre comillas dobles.
 - Sintaxis:
 - `get nombre de archivo.extensión`
 - `get "nombre de archivo compuesto.extensión"`
 - Ejemplos:
 - `get paisaje.jpg`
 - `get index.html`
 - `get "primera página web.ntml"`
- **mget.**
- Permite descargar o copiar varios archivos al equipo local.
 - Hay que estar en la carpeta que contiene el archivo a descargar/copiar, para ello se usa `cd`.
 - El destino local también tiene que estar seleccionado con `lcd`.
 - Pide confirmación:
 - Respuesta Si → y o yes. (Si no se especifica, valor por defecto).
 - Respuesta No → n o no.
 - Admite comodines:
 - Asterisco (*)
 - Representa un conjunto de caracteres distintos.
 - Sólo se incluye en el archivo lo que es común a todos.
 - Ejemplo:
 - `*.png` (dibujo.png, foto.png, playa.png)
 - `pagina.*` (pagina.html, pagina.docx)
 - `*.*` (Todos los archivos con todas las extensiones).
 - Interrogación (?).
 - Representa un único carácter.
 - Se utilizan tantas interrogaciones como caracteres a sustituir.
 - Las interrogaciones representan los caracteres distintos.
 - Ejemplos:
 - `L????` Mostraría Lucia Laura Lucas y no Luis o Lorena.
 - `L???a` Mostraría Lucia Laura pero no Lucas.
 - Si el archivo tiene un nombre compuesto con espacios en blanco, hay que especificarlo entre comillas dobles.
 - Sintaxis:
 - `mget nombre de archivo1.extensión nombre de archivo2.extensión nombre de archivoN.extensión`
 - `mget "nombre de archivo compuesto1.extensión" "nombre de archivo compuesto2.extensión" "nombre de archivo compuestoN.extensión"`
 - Ejemplos:
 - `mget dibujo.png foto.png playa.png`
 - `mget *.png`
 - `mget "pagina principal.html" index.html`
 - `mget pagina.html, pagina.docx`
 - `mget pagina.*`

- mget *.*
 - mget pa????1.* (pagina1.html pagina1.jpg partes1.png)
- **delete.**
 - Permite borrar un archivo remoto.
 - Hay que estar en la carpeta que contiene el archivo a borrar, para ello se usa cd.
 - No pide confirmación.
 - Si el archivo tiene un nombre compuesto con espacios en blanco, hay que especificarlo entre comillas dobles.
 - **del** comando equivalente en Windows.
 - Sintaxis:
 - delete nombre de archivo.extensión
 - delete “nombre de archivo compuesto.extensión”
 - Ejemplos:
 - delete paisaje.jpg
 - delete index.html
 - delete “primera página web.ntml”
- **mdelete.**
 - Permite borrar varios archivos remotos.
 - Hay que estar en la carpeta que contiene los archivos a borrar, para ello se usa cd.
 - **del** comando equivalente en Windows.
 - Pide confirmación:
 - Respuesta Si → y o yes. (Si no se especifica, valor por defecto).
 - Respuesta No → n o no.
 - Admite comodines:
 - Asterisco (*)
 - Representa un conjunto de caracteres distintos.
 - Sólo se incluye en el archivo lo que es común a todos.
 - Ejemplo:
 - *.png (dibujo.png, foto.png, playa.png)
 - pagina.* (pagina.html, pagina.docx)
 - *.* (Todos los archivos con todas las extensiones).
 - Interrogación (?).
 - Representa un único carácter.
 - Se utilizan tantas interrogaciones como caracteres a sustituir.
 - Las interrogaciones representan los caracteres distintos.
 - Ejemplos:
 - L???? Mostraría Lucia Laura Lucas y no Luis o Lorena.
 - L????a Mostraría Lucia Laura pero no Lucas.
 - Si el archivo tiene un nombre compuesto con espacios en blanco, hay que especificarlo entre comillas dobles.
 - Sintaxis:
 - mdelete nombre de archivo1.extensión nombre de archivo2.extensión nombre de archivoN.extensión
 - mdelete “nombre de archivo compuesto1.extensión” “nombre de archivo compuesto2.extensión” “nombre de archivo compuestoN.extensión”
 - Ejemplos:
 - mdelete dibujo.png foto.png playa.png
 - mdelete *.png
 - mdelete “pagina principal.html” index.html
 - mdelete pagina.html, pagina.docx

- mdelete pagina.*
 - mdelete *.*
 - mdelete pa?????1.* (pagina1.html pagina1.jpg partes1.png)
- **rename.**
 - Cambia el nombre de un archivo en el servidor.
 - Hay que estar ubicado en la carpeta donde esté el archivo al que se le va a cambiar el nombre.
 - Sintaxis:
 - rename nombre antiguo.extension nombre nuevo.extension.
 - Ejemplo:
 - rename a.txt b.txt
- **glob.**
 - Permite el uso de comodines para subir, bajar o eliminar archivos.
 - Por defecto están activados.
 - Sintaxis:
 - glob (para activar/desactivar)

UBICACIÓN DE RED A UN SERVIDOR FTP EN WINDOWS.

- **Creación.**
 1. Abrir o seleccionar “este equipo” en Windows.
 2. Menú contextual/Agregar una ubicación de red o Agregar una ubicación de red en la barra de herramientas.
 3. Siguiente.
 4. Elegir una ubicación personalizada.
 5. Siguiente.
 6. Escribir una dirección a un sitio web, ftp o a una carpeta compartida. (las direcciones externas hay que escribirlas con protocolo).
 - a. Ejemplos:
 - i. ftp://localhost.
 - ii. ftp://127.0.0.1
 - iii. ftp://ftp.uv.es
 - iv. http://www.misitio.es
 7. Siguiente.
 8. Especificar usuario que se conectara a la ubicación de red.
 - a. Usuario determinado poniendo su nombre.
 - b. Usuario anonymous.
 9. Siguiente.
 10. Dar nombre a la ubicación para identificarla mejor (opcional)
 11. Siguiente.
 12. Finalizar (Se puede o no abrir ahora la ubicación de red).
- **Cambio de usuario en una misma ubicación de red.**
 - **Dentro del sitio web abierto:**
 - Menú contextual/Iniciar sesión como
 - Especificar nombre de usuario y contraseña.
 - Iniciar sesión.
- **Eliminación de una ubicación de red.**
 - **Sobre ubicación de red a eliminar:**
 - Menú contextual / eliminar o pulsar la tecla de suprimir.

BUSCADORES.

Concepto.

- Sistema informático que busca información en la web en distintos formatos (textos, imágenes, noticias, etc, ...).
- La información encontrada está relacionada con una consulta realizada usando palabras clave o arboles jerárquicos temáticos (directorios o categorías).

Tipos.

- **Motores de búsqueda.**

- Buscan por contenido.
- Se emplean palabras o frases clave que definen los criterios de búsqueda.
- Buscan en sus propias bases de datos porque antes sus robots han etiquetado, catalogado o indexado las páginas web con su contenido.
- La indexación o catalogación se hace automáticamente.
- Los robots o arañas que recorren Internet rastreando las páginas, cuando las encuentran guardan sus datos y luego se dirigen a las distintas páginas que están enlazada a ella tanto externas como internas, de modo, que no es necesario registrar más que la página inicial del sitio.
- Adicionalmente, volverán rastrear las páginas para comprobar cambios o actualizaciones en ellas para así mantener la información de las bases de datos actualizadas.
- Tipos:

- **Generalistas.**

- Buscadores de ámbito general que abarcan todo tipo de contenido.
 - La información obtenida puede ser muy genérica y poco fiable o rigurosa.
 - Motores de búsqueda generalistas:
 - Google, Bing, Baidu, Yahoo Search, Yandex, Ask, Altavista, etc...

- **Temáticos, especializados o verticales.**

- Buscadores de ámbito específico que centran su actividad en materias concretas.
 - Creación y mantenimiento de los contenidos se realiza por personas expertas, por lo que la información suele ser más fiable y precisa.
 - No solo muestran información contenida en páginas web, si no también listas de distribución, archivos ftp, bases de datos específicas, grupos de noticias, canales IRC, etc.
 - Buscadores temáticos:
 - Google académico, ARQ, Eco-Portal, ...

- **Índices, directorios, catálogos o buscadores por categorías.**

- Mantienen una organización de las páginas incluidas en su base de datos por categorías, es decir, tiene un directorio navegable por temas.
- Dentro de cada directorio hay páginas relacionadas con un tema.
- Se avanza o navega desde los temas más generales hasta los más específicos a través de directorios temáticos.
- La creación, el mantenimiento y la organización de las categorías y las páginas en los directorios corre a cargo de personas que visitan las páginas y vigilan que todas estén clasificadas en el lugar correcto.
- Si todas o varias páginas de un sitio se quieren que estén en el buscador hay registrarlas una a una.
- La mayoría han caído en desuso han desaparecido.
- Buscadores por categorías:
 - DMOZ, Yahoo Search,...

- **Metabuscadore.**

- Buscadores en otros buscadores.
- No tienen bases de datos propias, sino que usan la de otros para extraer la mejor información de cada uno de ellos.
- Presentan resultados de múltiples fuentes.
- Ventaja:
 - Amplían notablemente el ámbito de búsqueda.
- Suelen ser del tipo motor de búsqueda, aunque no tienen base de datos propia.
- Pueden ser generalistas o temáticos.
- Metabuscadores más conocidos:
 - Tripadvisor (Hoteles).
 - Booking (Hoteles).
 - Jooble (Empleo).
 - Indeed(Empleo)
 - Zapmeta(Generalista).
 - Zoo (páginas amarillas y blancas).
 - DuckDuckGo.

Criterios para Búsquedas mediante palabras clave.

- Usar las palabras clave correctas que definan el criterio de búsqueda.
 - Si se buscan *martillos*, no usar el termino genérico *herramientas*.
- Usar el mínimo número de palabras que definan el criterio de búsqueda.
 - *El hotel de Cullera que tenía una tienda de ropa, en el hacían paellas muy buenas y te regalaban un llavero.*
 - *Hoteles Cullera con tienda.*
- Añadir ubicación al termino de búsqueda si es necesario.
 - *Hoteles Jaén*, mejor que *Hoteles* solo si se va a ir a Jaén.
- Para una frase explícita, encerrar ésta entre comillas.
 - *Ciencia Ficción* buscará páginas de Ciencia Ficción, Ciencia solo y Ficción solo, mientras que *"Ciencia Ficción"* encontrará únicamente páginas de ese tema.
- Utilizar filtros temáticos según el tipo de resultado a encontrar, (todo, video, noticias, imágenes, etc.).
- Para resultados con todas las palabras usar el operador lógico *AND* o el signo más (+).
 - *Aviones and Papel*, buscará aviones de papel y no de otros tipos.
- Para incluir uno u otro de los términos elegidos, se usa el operador lógico *OR*.
 - *"Motos OR Bicicletas"*, buscará páginas en las que hablen de motos y bicis, y también, páginas que hablen solo de motos o solo de bicis.
- Para excluir un término se usa un guion (-).
 - *"Vehículos -Coches"*, mostrará todo tipo de vehículos excepto coches.
- Se pueden crear patrones usando el comodín asterisco (*), que representa a un conjunto de caracteres.
 - *"* de terror"* buscará películas de terror, animes de terror, novelas de terror, etc.
- Para intervalos numéricos se separan los 2 números con 2 puntos (..). Primero se escribe el límite inferior y después el superior.
 - *"móviles de 200 .. 500 euros"*.
- Para obtener en los resultados contenidos con un número aproximado se utiliza el comando *around* (nº).
 - *"hoteles around (30) habitaciones"*.
- Para resultados que incluyan en su contenido un determinado texto, se usan los comandos *intext* o *allintext*.

- Sintaxis:
 - *allintex: "texto"*.
 - *intext: "texto"*.
- Ejemplos:
 - *intext: "en los próximos 10 días"*.
 - *allintext: "ordenadores baratos"*.
- Para resultados que incluyan en su título un determinado texto, se usan los comandos *intitle* o *allintitle*.
 - Sintaxis:
 - *allintitle: "texto"*.
 - *intitle: "texto"*.
 - Ejemplos:
 - *allintitle: "cuenca"*.
 - *intitle: "operador"*.
- Para resultados que incluyan en sus enlaces un determinado texto, se usan los comandos *inurl* o *allinurl*.
 - Sintaxis:
 - *allinurl: "texto"*.
 - *inurl: "texto"*.
 - Ejemplos:
 - *allinurl: "cuenca"*.
 - *inurl: "operador"*.
- Para buscar dentro de una web concreta que, o no tiene buscador interno, o el que tiene es poco preciso, se usa el comando *site*.
 - Sintaxis:
 - *site: página web*.
 - Ejemplo:
 - *site:www.mediamarkt.es*.
- Para obtener resultados con información sobre una página concreta se usa el comando *info*.
 - Sintaxis:
 - *info: página web*.
 - Ejemplo:
 - *info: www.filmaffinity.com*.
- Para buscar páginas relacionadas con una determinada se usa comando *related*.
 - Sintaxis:
 - *related: página web*.
 - Ejemplo:
 - *related: www.filmaffinity.com*.
- Se pueden combinar varios comandos u operadores usando paréntesis ().
 - Ejemplo:
 - *("Alojamiento en Cuenca" or "Dormir en Cuenca") -Hotel*
- Conversión de unidades y monedas:
 - Incluir el comando *EN* entre las unidades a convertir.
 - Ejemplos:
 - *"200 euros EN yenes"*.
 - *"17 libras EN euros"*.
 - *"23 kilómetros EN millas"*.
- Para buscar definiciones de un término concreto se usa el comando *define*.
 - Sintaxis:
 - *define: término para el que se busca la definición*.

- Ejemplo:
 - *define:Ciencia ficción*
- Par buscar mapas se usan los comandos *map* o *mapa*.
 - Sintaxis:
 - *map: Ubicación.*
 - *mapa: Ubicación.*
 - Ejemplos:
 - *Map:Tatooine*
 - *Mapa:Rocadragon.*
- Para buscar una página web guardada en la cache si se cae la conexión y no se puede acceder al sitio en un determinado momento, se usa el comando *cache*.
 - Sintaxis:
 - *cache: página web.*
 - Ejemplo:
 - *cache:www.renfe.com*
- Para buscar documentos de un tipo se usa el comando *filetype*.
 - Sintaxis:
 - *Filetype:tipo de archivo.*
 - Ejemplos:
 - *Filetype:pdf html.*
 - *Filetype:docx manual de Word.*
- Para ver que tiempo meteorológico hace en una ciudad determinada, se usar el comando *weather*.
 - En caso de que haya ciudades con el mismo nombre en distintos países, hay que incluir también el código del país de la ciudad que se quiere consultar.
 - Sintaxis:
 - *weather:localidad.*
 - *weather:localidad, código de país.*
 - Ejemplos:
 - *Weather:Honolulu.*
 - *Weather:Córdoba, ar.*
- Para ver qué hora es en una ciudad concreta, se usar el comando *time*.
 - En caso de que haya ciudades con el mismo nombre en distintos países, hay que incluir también el código del país de la ciudad que se quiere consultar.
 - Sintaxis:
 - *time:localidad.*
 - *time:localidad, código de país.*
 - Ejemplos:
 - *time:Honolulu.*
 - *time:Córdoba, ar.*

ROBOTS O BOTS DE BÚSQUEDA. (PAG. 87-88)

- Programa que recorre la web de forma automática y basándose en determinados algoritmos analiza las páginas web y las clasifica, cataloga o indexa.
- Indexación:
 - Proceso por el cual la información de las páginas se almacena de forma ordenadas y clasificada en una base de datos.
 - Cuando un usuario hace una consulta con un buscador lo hace a la base de datos de éste.
 - Los robots más comunes son las arañas o crawlers que son los que acceden a los sitios web.
- Frecuencia de paso:

- Los robots pasan por los sitios web con una frecuencia determinada por la popularidad del sitio.
- A mayor número de visitas diarias, mayor frecuencia de paso.

Robots más habituales:

User-Agent	Motor de búsqueda	Campo
baiduspider	Baidu	General
baiduspider-image	Baidu	Imágenes
baiduspider-mobile	Baidu	Móvil
baiduspider-news	Baidu	Noticias
baiduspider-video	Baidu	Video
baiduspider-favo	Baidu	Favoritos o Marcadores
baiduspider-ads	Baidu	Publicidad
bingbot	Bing	General
msnbot	Bing	General
msnbot-media	Bing	Imágenes y video
adidxbot	Bing	Publicidad
Googlebot	Google	General
Googlebot-Image	Google	Imágenes
Googlebot-Mobile	Google	Móvil
Googlebot-News	Google	Noticias
Googlebot-Video	Google	Video
Storebot-Google	Google	Compras
Mediapartners-Google	Google	AdSense
AdsBot-Google	Google	Google Ads
slurp	Yahoo!	General
yandex	Yandex	General

- Enlaces a las páginas de robots de los buscadores más comunes.
 - Google:

<https://developers.google.com/search/docs/advanced/crawling/overview-google-crawlers?hl=es-419>

ARCHIVO ROBOTS.TXT

- Archivo de texto plano ubicado en la raíz de un sitio web que indica a los rastreadores de los motores de búsqueda (robots) a que partes pueden o no acceder del sitio.
- Así se impide que determinados contenidos sean indexados o mostrados en los resultados de búsqueda.

A tener en cuenta.

- **El robots.txt no es vinculante** para los rastreadores y puede ser ignorado en ocasiones. Aunque por lo general, los rastreadores siguen estas reglas.

- **El robots.txt no impide la indexación.** El rastreador puede llegar a la URL desde vínculos externos o enlaces internos y añadirlos al índice.
- **Si bloqueamos una página que ya estaba incluida en el índice, Google la seguirá mostrando en el resultado.** Para asegurarnos de que Google no indexa la URL, debemos permitir que vuelva a rastrear la página y acto seguido indicarle que no la indexe. ¿Cómo lo hacemos? Para ello, es preferible utilizar meta etiquetas dentro de la página web.
- Para indicarle al robot de Google que revise nuestro sitio web:
 - http://www.google.es/intl/es/submit_content.html

Creación:

- Hay que guardarlo en la raíz del sitio web.
 - <https://www.mitio.es/robots.txt>
- Tiene que ser un archivo de texto plano sin formato.
- Se pueden incluir una o varias reglas para bloquear o permitir el acceso de un determinado agente de usuario, rastreador o robot a un directorio, subdirectorio o archivo de un sitio web.
- Obligatorio debe llamarse robots.txt.
- Solo puede haber un archivo robots.txt por sitio.
- Se pueden aplicar a:
 - Subdominios:
 - <https://otrositio.mitio.es/robots.txt>
 - Puertos.
 - <https://www.mitio.es:8181/robots.txt>

Comandos para los archivos de robots.

- **Comentarios:**
 - Son de una línea y cada una debe empezar por una almohadilla (#).
 - Sintaxis:
 - # Comentario 1.
 - # Comentario 2.
 - Ejemplos:
 - # Estos robots no pueden acceder al sitio.
- **user-agent.**
 - Se emplea para especificar los robots.
 - Sintaxis:
 - User-agent: nombre del robot
 - User-agent: * (todos los robots)
 - Ejemplos:
 - User-agent: Googlebot
 - User-agent: Yandex
 - User-agent: *
- **Disallow.**
 - Rechazar o denegar acceso.
 - Sintaxis:
 - Disallow: ruta de acceso, nombre carpeta o directorio, archivo a bloquear.
 - / representa al directorio raíz.
 - Ejemplos:
 - User-agent: Yandex
 - Disallow: /
 - Disallow: /imágenes/

- Disallow: /scripts/uno.js
 - Allow: /documentos
 - User-agent: slurp
 - Disallow: /multimedia/
- **Allow.**
 - Permitir acceso.
 - Sintaxis:
 - Allow: ruta de acceso, nombre carpeta o directorio, archivo a permitir acceso.
 - Ejemplo:
 - User-agent: *
 - Disallow: /
 - Allow: /productos/
 - User-agent: slurp
 - Allow: /imágenes/
- **Sitemap.**
 - Permite especificar el nombre del archivo que contiene el mapa del sitio (extensión xml), es decir, el archivo que incluye todos los enlaces páginas, elementos multimedia u otros archivos.
 - Se pueden incluir varios archivos.
 - Se escribe la dirección absoluta al archivo xml que incluye el mapa del sitio.
 - La ruta de acceso al archivo con el mapa del sitio debe ser absoluta.
 - Sintaxis:
 - Sitemap: https://dominio/ruta de acceso/archivo sitemap.xml
 - Ejemplo:
 - Sitemap: https://www.misitio.com/mapas/sitemap.xml
 - Sitemap: https://www.misitio.com/mapas/multimedia/sitemap_imagenes.xml

CREACIÓN DE UN ARCHIVO SITEMAP.

- Los sitemap se rigen mediante unos protocolos marcados por:
 - <https://www.sitemaps.org/es/>

- Según estas directrices, un sitemap básico puede estructurarse así:

```
<?xml version="1.0" encoding="UTF-8"?>

<urlset xmlns="http://www.sitemaps.org/schemas/sitemap/0.9">

  <url>
    .....

    <loc>http://www.example.com/</loc>
    .....

    <lastmod>2005-01-01</lastmod>
    .....

    <changefreq>monthly</changefreq>
    .....

    <priority>0.8</priority>
    .....

  </url>

</urlset>
```

- **Contenido.**
 - La primera línea de código es obligatoria y define la codificación del archivo.
 - Etiquetas:
 - **<urlset>**
 - Establece el listado de direcciones URL.
 - **<url>**
 - Anidada a <urlset>.
 - Incluye el sitio web a mostrar en el archivo sitemap.xml.
 - Puede haber varias, para distintos sitios web.
 - Incluye anidadas las siguientes etiquetas informativas:
 - **<loc>**
 - Indica cual es la dirección web o URL
 - Sólo se especifica una por etiqueta.
 - Etiqueta obligatoria.
 - **<lastmod>**
 - Indica la fecha de la última actualización o modificación de contenido o programación de la URL.
 - Etiqueta opcional.
 - **<changefreq>**
 - Indica con qué frecuencia se actualiza o modifica el sitio web.
 - Etiqueta opcional.
 - **<priority>**
 - Importancia que el creador da a una página del sitio web respecto del conjunto de páginas del sitio.
 - Se indica con un valor numérico con decimales comprendido entre 0 y 1.
 - Etiqueta opcional.

Comodines o reglas de concordancia.

Asterisco (*).

- Permite crear un patrón representando a conjunto de caracteres.

- Ejemplo:
 - Disallow: /imágenes/*.jpg → (Todos los archivos jpg).
 - Disallow: /imágenes/logo.* → (Todos los archivos cuyo nombre es logo).
 - Disallow: /imágenes/foto*.* → (Todos los archivos cuyo nombre empiece por foto y tengan cualquier extensión).

Dólar (\$).

- Especifica que un archivo deber terminar por los caracteres que le preceden.
- Ejemplo:
 - Disallow: /imágenes/logo*.jpg → (excluiría a los archivos marcados en verde).
 - Logotipo.html
 - Logot.jpg.html
 - Logo1.jpg
 - Disallow: /imágenes/logo*.jpg\$ → (excluiría sólo al archivo marcado en verde, que es el único que termina en jpg).
 - Logotipo.html
 - Logot.jpg.html
 - Logo1.jpg
- **Reglas útiles, comunes o comandos más usados:**
 - Indicaciones a un bot concreto:
 - User-agent: baiduspider.
 - Indicaciones a todos los bots:
 - User-agent: *.
 - Bloqueo de un sitio web entero:
 - Disallow: /
 - Bloqueo de un directorio y su contenido:
 - Disallow: /css/
 - Disallow: /multimedia/audios/
 - Bloqueo de una página web concreta:
 - Disallow: /index.html
 - Bloquear un archivo dentro de una ruta concreta:
 - Disallow: /imágenes/png/logo.png
 - Bloqueo de todas las imágenes para un bot concreto:
 - User-agent: slurp
 - Disallow: /imágenes/*.*
 - Disallow: /imágenes/
 - User-agent: baiduspider-image
 - User-agent: googlebot-image
 - Disallow: /
 - Disallow: /imágenes/
 - Bloqueo de un tipo de archivos concreto:
 - Disallow: /imágenes/*.jpg
 - Disallow: /estilos/css/*.css
 - Disallow: /estilos/css/*.css\$
 - Bloqueo de archivos con el mismo nombre:
 - Disallow: /imágenes/paisajes*.jpg
 - Disallow: /imágenes/paisajes*.*

- Disallow: /imágenes/paisajes.*
- Bloqueo de todos los archivos:
 - Disallow: /paginas/*.*
- Acceso completo a todos los bots:
 - User-agent: *
 - Disallow:
 - Allow:/
 - No usar un archivo robots.txt
 - Usar un archivo robots.txt vacío.
- Permitir rastreo a un bot concreto:
 - User-agent: *
 - Disallow:/
 - Disallow: /imágenes/
 - User-agent: Bingbot
 - Allow: /imágenes/*.png
 - Allow: /paginas/
- Excluir todos los archivos menos uno (o varios).
 - Disallow: /multimedia/videos/
 - Allow: /multimedia/videos/película.mp4

METAETIQUETAS. (pag. 85-96)

- Permiten especificar metadatos para un documento.
- Metadato:
 - Metainformación.
 - Datos que describen datos.
- Se incluyen en la cabecera <head> del documento Web.
- Aportan información técnica a los motores de búsqueda y mejoran el posicionamiento SEO (Search Engine Optimization – Optimización para Motores de Búsqueda).
- Es una forma de comunicarse con los motores de búsqueda para:
 - Informales de cómo deben interpretar nuestra web.
 - Qué deben mostrar en los **SERP** (resultados de una búsqueda).
- **SERP** (Search Engine Results Page).
 - Página de resultados de un motor de búsqueda.
- Sintaxis:


```
<head>
  <meta atributo 1 = "valor" atributo 2 ="valor" .... atributo N = "valor"/>
</head>
```
- Ejemplo:


```
<meta charset = "utf-8"/>
```
- **Atributos**.
 - **charset**.
 - Especifica el conjunto de caracteres usado en el documento.
 - Sintaxis:
 - <meta charset =" juego de caracteres">
 - **name**.
 - Especifica nombre del metadato a utilizar.
 - Especifica la meta-etiqueta a usar.

- Atributo asociado al atributo content.
- Sintaxis:
 - name = "valor"
- **content.**
 - Especifica el valor o contenido de un metadato concreto.
 - Atributo asociado a los atributos name y http-equiv.
 - Sintaxis:
 - content = "valor"
- **http-equiv.**
 - Especifica órdenes o directivas para el procesamiento http en el servidor.
 - Permite indicar al servidor algunas propiedades o comportamiento del documento web.
 - Atributo asociado al atributo content.
 - Sintaxis:
 - http-equiv = "metadato".
- **Sintaxis combinada name y content o http-equiv y content.**
 - <meta name="metadato" content= "valor del metadato">
 - <meta http-equiv ="metadato" content= "valor del metadato">
 - Ejemplos:
 - <meta name = "Author" content = "Ana López García">
 - <meta http-equiv ="refresh" content= "30">
- **Listado de metadatos para el atributo name.**
 - **application-name.**
 - Describe el nombre de la aplicación web que el documento representa.
 - Solo puede aparecer una vez.
 - **author.**
 - Nombre del autor del documento web.
 - Se pueden incluir varios autores, cada uno en una declaración meta distinta.
 - Se puede incluir el correo electrónico del autor.
 - **owner.**
 - Indica quien es el propietario.
 - **copyright.**
 - Especifica el nombre del propietario de los derechos de autor.
 - Indica el copyright de la página.
 - **description.**
 - Breve descripción de la página o su propósito.
 - Sólo puede haber una descripción en el documento web.
 - Consejo:
 - Entre 50 y 160 caracteres.
 - **keywords.**
 - Palabras clave en la página web.
 - Especifican un conjunto de identificadores separados por comas que son palabras clave que describen los contenidos del documento web.
 - Importante para el posicionamiento SEO.
 - **generator.**
 - Indicar el nombre programa o programas de diseño utilizados para generar el documento web.
 - Si se han usado múltiples programas hay que incluir múltiples declaraciones meta.
 - **robots.**
 - Especifica las órdenes o directivas para los robots de rastreo.

- Indica a los robots de los buscadores que información debe o no, ser tomada en cuenta.
- Indica cómo se deben rastrear los enlaces que el robot encuentre en el documento web.
- Indica a los robots si se debe o no indexar la página.
- En el atributo **name** se pueden especificar todos los robots con el valor “robots” o incluir rastreadores específicos sustituyendo “robots” por el nombre de éstos.
- Ejemplos:
 - name = “robots” (Todos los robots de todos los motores de búsqueda).
 - name = “Bingbot” (Un robot concreto).
 - name = “Bingbot, Yandex, Baiduspider” (Varios robots).
- Directivas:
 - **all.**
 - Sin restricciones de indexación o publicación.
 - Valor predeterminado.
 - Equivale a index y follow juntos.
 - **Index / noindex.**
 - La página web debe / no debe ser indexada en la base de datos del motor de búsqueda.
 - Por lo tanto, la página se muestra/no se muestra en los resultados de una búsqueda.
 - **follow / nofollow.**
 - Permite / Impide seguir los enlaces de una página web cuando ésta es rastreada por un robot.
 - **none.**
 - Equivale a noindex y nofollow juntos.
 - **noarchive.**
 - Se usa para que el rastreador no guarde la página en la cache de búsqueda.
 - Esta cache almacena el contenido que tenía una página la última vez de que se accedió a ella, así, la siguiente vez, el robot accederá más rápido a la página.
 - En caso de que la página haya sido modificada, es mejor que se acceda al servidor para rastrear la página en su último estado, que a la cache.
 - **nosnippet.**
 - No se muestra un fragmento de la página o de la descripción en los resultados búsqueda.
 - **noimageindex.**
 - No se indexan las imágenes de la página.
- **language.**
 - Especifica en que idioma está escrita la página web.
 - Se usa un código de país de 2 caracteres: es, pt, uk, en_UK en_US.
- **revisit-after.**
 - Indica al robot cada cuanto tiempo se actualiza la página y, por tanto, cada cuanto deben revisarla.
 - Periodo temporal escrito en inglés y medido en días o meses (1 month, 15 days, ...).
- **viewport.**
 - Permite ajustar el contenido del documento web a las distintas dimensiones, resoluciones y densidades de pixeles de los dispositivos.

- Indica al navegador como controlar la dimensión de la pantalla escalando el contenido para que no se salga de los márgenes del dispositivo.
- Si no se usa, no se cargan los distintos diseños creados para las pantallas de los diferentes dispositivos.
- **Listado de metadatos del atributo http-equiv.**
 - **content-type.**
 - Especifica el tipo de contenido y juego de caracteres que usa en la página.
 - Alternativa al atributo charset = "juego de caracteres".
 - Sintaxis:
 - <meta http-equiv = "content-type" content="tipo MIME; charset = juego de caracteres">
 - **refresh.**
 - Se emplea para actualizar o redireccionar una página web.
 - Actualizar:
 - Permite indicar el tiempo de refresco que debe pasar entre cada actualización.
 - Sintaxis:
 - <meta http-equiv = "refresh" content= "nº segundos">
 - Redireccionar:
 - Tras pasar el tiempo especificado, el servidor envía directamente a otra página web.
 - Sintaxis:
 - <meta http-equiv = "refresh" content= "nº segundos; URL del destino">
 - Ejemplo:
 - <meta http-equiv = "refresh" content= "30; URL = http://www.unsitio.es/inicio.html"> (Externa)
 - <meta http-equiv = "refresh" content= "30; URL = redirecciones/inicio.html"> (Interna)
 - **expires.**
 - Especifica el día y hora exactos en el cual expira la página guardada en la cache del navegador, por lo que ésta debe descargarse de nuevo desde el servidor para poder visualizarse.
 - Sintaxis:
 - <meta http-equiv = "expires" content= "día de la semana, dd mm aaaa hh:mm:ss GMT">
 - <meta http-equiv = "expires" content= "0">
 - Ejemplo:
 - <meta http-equiv = "expires" content= "Sun, 20 07 2020 12:00:00 GMT">
 - **cache-control.**
 - Indica a los navegadores si se deben o no guardar en cache (archivos temporales de Internet), las páginas y archivos web que se cargan.
 - Sintaxis:
 - <meta http-equiv="Cache-control" content="valores">
 - Ejemplo:
 - <meta http-equiv="Cache-control" content="no-cache">

Terminología:

- **SEO (Search Engine Optimization).**
 - Optimización para motores de búsqueda.
 - También se denomina posicionamiento orgánico o posicionamiento en buscadores.
 - **Concepto:**
 - Conjunto de técnicas para aumentar la visibilidad de un sitio web en los resultados orgánicos de una búsqueda.
 - **Objetivo:**
 - Aparecer en las primeras posiciones de los resultados de una búsqueda al usar determinadas palabras clave.
- **SEM (Search Engine Marketing).**
 - Marketing en buscadores.
 - Tiene que ver con el lugar que ocupa un sitio web en los resultados de pago.
 - Disciplina que incluye la contratación, gestión y análisis de los enlaces patrocinados en un buscador.
 - *Google Ads es la plataforma de anuncios de Google.*
- **SERP (Search Engine Page Result).**
 - Página que un buscador muestra con los resultados de una búsqueda.
 - **Tipos de resultados:**
 - **Naturales u orgánicos.**
 - Se basan en un algoritmo que determina que resultados son más relevantes para una búsqueda determinada.
 - No se puede comprar aparecer en los resultados orgánicos, ni tampoco en qué posición aparecer.
 - **Patrocinados.**
 - Resultados de pago.
 - Publicidad comprada mediante sistemas como Google Ads.
- **Snippet.**
 - Unidad formada por el título y la descripción de una página mostrada en los resultados de una búsqueda.
 - Un buen título mejora el posicionamiento y aumenta también la tasa de clics o CTR (Click-Through Rate).
 - Uso etiqueta <title>. Entre 55 y 60 caracteres.
 - Importancia del análisis de palabras clave.

Técnicas.

ON PAGE: (Dentro de página).

- Hace referencia tanto a la optimización del contenido, como al código fuente HTML de una página.

Técnicas:

- **Crear títulos de página únicos y precisos.**
 - <title>Título</title>
 - Opciones más usuales:
 - Título para la página + nombre sitio Web.
 - Palabra clave principal, palabra clave secundaria y nombre de marca.

- Único por página, conciso, descriptivo y semánticamente valioso.
- Máximo entre 50 y 60 caracteres.
- Usar **meta etiquetas**, ya que los buscadores las usan para identificar de que va un sitio y así indexarlo correctamente en sus bases de datos.
 - Usar la meta etiqueta **“keywords”**.
 - Conjunto de palabras clave únicas para cada página.
 - Usar la meta etiqueta **“description”** para llamar la atención al posible visitante, ya que no influye en el posicionamiento, pero si en las posibilidades de atraer clics hacia el sitio.
 - Descripción única para cada página.
- **Usar Palabras Clave en el Contenido.**
 - En el texto de la página debería aparecer la palabra clave principal, variantes de la misma y otras relacionadas.
- **Dirección URL descriptiva.**
 - No usar direcciones URL que no representen el contenido de la página.
 - Que sea corta, hasta 4 o 5 palabras y que incluya la palabra clave del sitio.
- **Crear una buena estructura para organizar el contenido web**, con nombres de carpeta descriptivas del contenido.
 - Mejor una estructura horizontal poco profunda que una vertical, ya que a los robots les cuesta más entrar a niveles profundos por el tiempo limitado que tienen para rastrear una web.
- **Correcta utilización de las etiquetas de encabezado <h1> a <h6>** para los títulos, encabezados o epígrafes.
 - Estructurar el contenido de la página usando en <h1>, <h2>, <h3> y descripciones.
 - Las etiquetas de encabezado estructuran correctamente la página y facilitan el rastreo de ésta por parte de los robots.
 - Una web bien organizada resulta más útil para los visitantes y para los buscadores.
- **Facilitar la navegación por el sitio usando:**
 - Mapas de sitio.
 - Archivo sitemap.xml.
 - Barras de navegación que permita acceso al contenido con pocos clics.
 - Breadcrumbs o migas de pan.
 - Usar enlaces de texto significativos que den información a los usuarios y a los buscadores.
- **Ofrecer contenidos y servicios de calidad.**
- **Optimizar el uso de imágenes.**
 - Siempre incluir el atributo “alt”, que permite incluir información sobre la imagen.
 - Los nombres de los archivos también deben ser significativos para facilitar la indexación a los rastreadores.
- **Optimizar el sitio web para dispositivos móviles.**
 - Uso de diseño adaptable o adaptativo (Responsive Design).
- **Enlaces Externos.**
 - Muestran a los buscadores que se es una persona real, que se preocupa de enlazar a otras webs que complementan a la nuestra, lo que en algunos casos ayuda a mejorar tu posicionamiento.
 - Proporcionar un valor añadido a tus visitantes, dirigiéndoles a otros contenidos interesantes o que complementen al tuyo, lo que genera confianza entre tus visitantes y ayuda a fidelizarlos.
 - Aumentar tu relevancia temática y tu autoridad sobre el tema, ya que te estás relacionado con una o varias páginas que también tratan o desarrollan el mismo tema.

- **Optimizar el texto.**
 - Utilizar palabras clave frecuentemente dentro del contenido de la página, pero usarlas con sentido y sin ser molestas para los usuarios.
 - Pueden resaltarse con la etiqueta o .
 - Evitar texto en formato imagen.
 - Textos bien redactados, fáciles de leer y entender.
 - Atención a la ortografía y la gramática.
 - Organizar el texto en unidades lógicas organizadas en párrafos no muy extensos precedidos por encabezados.
 - Evitar textos duplicados.
- **Las palabras clave deben tener un significado semántico** y satisfacer la expectativa de la persona que busca un determinado contenido.
- **Longitud del contenido.**
 - El contenido de mayor longitud suele posicionar mejor porque contiene más palabras clave.
 - Si el contenido no es útil para los visitantes, los robots lo detectan al recibir el sitio pocas visitas.
- **Velocidad de carga de la página.**
 - Influye en el posicionamiento.
 - Hay herramientas que miden su rendimiento como **Test my Site de Google**.
 - **Test my Site.**
 - Mide el rendimiento del sitio web, tanto en móviles como en ordenadores.
 - Proporciona un resumen con los puntos respecto a la velocidad y la personalización del sitio web a corregir para mejorar el rendimiento.

OFF PAGE: (Fuera de página).

- Técnicas destinadas a conseguir enlaces a nuestro sitio web en webs de terceros (Link Building).
- **Estrategias.**
 - Evitar hacer spam.
 - Crear contenido interesante y original para conseguir enlaces entrantes.
 - Enlace cebo – Link Baiting.
 - Si la información es atractiva e interesante para los usuarios, estos harán que se difunda por todo tipo de plataformas.
 - Darse de alta en buscadores temáticos y en directorios como DMOZ.
 - Intercambiar enlaces con otras páginas relacionadas con la nuestra.
 - **Blogs.**
 - Crear un blog para incluir contenidos actualizados.
 - Incluir en la web un enlace al blog.
 - **Guest Blogging.**
 - Escribir como invitado en los blogs de otros.
 - **Redes Sociales.**
 - Incluir en la web enlaces a redes sociales bien visibles.
 - Crear perfiles sociales en foros, redes sociales, otros blogs y páginas relacionadas.
 - Usar infografías, audios, imágenes y videos, por su impacto visual y sonoro.
 - Crear videos para YouTube, Vevo, Daylimotion, etc.
 - Generar tutoriales, guías, etc.
 - Usar **RSS.**
 - Really Simple Syndication – Sindicación realmente simple.
 - Permite que los usuarios reciban notificación cuando la web se actualice.

- **Link a terceros.**
 - Dejar link a terceros también ayuda a posicionar un sitio web. Aunque la intención es todo lo contrario (recibir enlaces), esta estrategia funciona porque nos ayuda a:
 - Ofrecer un contenido de interés para los usuarios.
 - Crear relaciones con las personas a las que enlazamos, comentar sus contenidos y conseguir un enlace recíproco
 - **Transitividad semántica.**
 - Al enlazar a un blog que enlaza a una gran marca, Google interpreta que esa gran marca y el blog están relacionados con tu página.
- Analizar otras páginas Web para observar a quien enlazan y quien las enlaza, y obtener ideas sin copiar.
- Intentar conseguir enlaces de sitios prestigiosos, como empresas importantes, marcas comerciales, sitios educativos o universidades, etc.
- Buscar el feedback y la opinión de otros usuarios.