BITS: 0 o 1

ENCODING: codificación de letras (cada letra pertenece a un encoding)

Origen del uso binario en los ordenadores (máquinas de lleno o vacío)

Origen término bug: (bicho) error de computación

4 generación: microprocesadores

Partes ordenador: disco, ram, placa base, gpu, cpu, buses (En el máster 1º parte ram y cpu), 2º parte gpu)

5º Generación: más capacidad de almacenamiento y diversos elementos gráficos DVD…

6º Generación (actual): crisis de microprocesadores, (ya no caben más) computación cuántica (más de 0 y 1, qbit) (Web IBM programación en cuántica)

Relación cliente/servidor: (cliente es la persona o entidad que pide información y servidor quien la otorga) Servidores en físico o en cloud.

Lenguajes entre cliente y servidor, HTTP es el “traductor”,

API: lista de cosas que puede hacer un servidor (la manera de consumir estas APIS se denomina “REST” (invocar al resultado))

Código de respuesta: 200 (ok), no 200 (mal)

Lenguajes de programación:

Compilado: un compilador traduce el código (C++)

Interpretado: el código se ejecuta directamente, sin la necesidad de un compilador

Intermedio: el código se compila cada vez que lo usas

WEB INTERESANTE “CODESIGNAL”

Descargar puttygen e instalar

Caché: memoria de acceso rápido para acelerar un proceso

Log: manera que tiene un ordenador de decirnos lo que está pasando

Encriptación: clave pública o clave privada (la clave privada nunca se puede compartir) Símil de las llaves, una abre todo directamente (privada) y la pública

Protocolos: http web, y resto

Para Fichero sftp

Para comandos SSH Putty

Códigos de puertos de los protocolos HTTP 80, FTP 22 Y SSH 22 SFTP 443

CPU / GPU / TPU

Sistema operativo: sistema capaz de gestionar la comunicación entre el resto de elementos del ordenador y el usuario

Tipos de archivos: CSV (fichero separado por comas), Excel, TXT,

SOFTWARE PARA INSTALAR

GitKraken: para guardar las prácticas 1 carpeta por práctica, portfolio para guardar tus cosas y poder compartirlas

Typora: para hacer notas

Putty

WinSCP no hace falta para MAC

Java JDK

Doker

UNIX

En el terminal

LS comando -L argumento Hola parámetro

Ls-> listado

Ls-l -> listado en modo de lista

-rw… permisos: si pone una d delante es directorio

-root: tienes todo el poder en todas las carpetas (administrador)

-tamaño de fichero

-fecha de modificación

-nombre de fichero

Otros comandos:

-t: ordenar por tiempo

-a: mostrar ficheros ocultos

\*se pueden poner todos los comandos juntos por ejemplo -lta

History mostral el historial de comandos que has ejecutado por sesión

Pwd: comando para saber donde estás dentro de la jerarquía

Cd .. comando para subir de jerarquía ( el / es el nodo rey)

En el nodo rey encontraremos las carpetas básicas: entre ellas las más importante son bin: comandos ejecutables de linux, etc los logs(ficheros donde se registra lo que hace una aplicación), lib librerías, home carpeta donde arrancas, root carpeta del usuario

Colores: verde fichero y azul: carpeta

Para bajar CD espacio y la carpeta que queramos entrar

Para dar un salto CD / y carpeta p.e. cd/root

Dos posibles rutas: absolutas con /root o relativas subiendo y bajando

CREAR UN FICHERO: TOUCH FICHERO PEDRO.TXT

BORRAR UN FICHERO: RM PEDRO.TXT (si lo borras, olvídate de el)

CREAR UNA CARPETA: MKDIR to

BORRAR CARPETA: rm -rf edemt

Para crear directamente un fichero en una carpeta tienes que poner touch edem/fichero.txt

Copiar archivo entre carpetas con cp a.txt ../destino/a.txt

Mover archivo mv

Comando echo escribir por pantalla

Para escribir en fichero echo > fichero.txt

Para mostrar el contenido cat fichero.txt

Para mostrar el contenido de un fichero reducido tail fichero.txt tail -N te muestra el fichero – N líneasec

Para dar permiso chmod 777…

CLOUD

IAAS: infraestructura como servicio

PAAS: plataforma como servicio

SAAS: software como serviciod

GIT

ESQUEMA DE DIAPOSITIVA CVS

Si te equivocas en un fichero local, podrás hacer un REVERT

RAMAS

Rama principal, rama máster (nunca se le hacen cambios commits)

Procedimiento: se crea una rama, se hacen cambios sobre ella “push”(versiones 1.0.1, 1.0.2…), una vez hechos todos los cambios, hacemos un pull request (máster hace los cambios y los pone en su archivo), tenemos la versión 1.1

Cosas que hay que hacer: pequeños cambios o commits

Hacer ramas separadas por cada cambio

Intentar escribir mensajes de commit comprendibles

Cosas que no hacer: subir código comentado para debugear o no útil

No mezclar las commits, una función una rama

y