Entornos de desarrollo integrados (IDE): entorno donde escribir código y aplicaciones. Principal: Visual Studio (VS) Code

Documentación de VS Code: <https://code.visualstudio.com/docs/setup/windows>

Introducción a HTML

Internet funciona con solicitud / respuesta

La solicitud lleva a que verifique el nombre especificado dentro del Sistema de nombres de dominio (DNS), devuelve la ruta IP.

Request en http

Respuesta http siempre viene codificada en java, html, css

Interfaz de usuario gráfica, Aplicación, Word

Descripción generada automáticamente

Web servers en donde se puede buscar en la red (Apache, Nginx, Tomcats), recibe http request y devolver la respuesta en html, java, css. Para juntar la información internamente se pueden usar otros lenguajes de programación

Interpreters: estos otros programas que unen la información a mostrar en la respuesta http. Python Java, etc.

Databases: MySQL, Orcacle, etc. De aquí se saca la información

HTML: lenguaje marcado de hypertexto

Etiquetas: <> para apertura, </> para cierre

Document objecto model (DOM): jerarquía que surge del anidamiento

La etiqueta de encabezado <head> solo muestra metadata y no aparece en la página

<title> indica el nombre que se verá en la pestaña de la página

La etiqueta <meta name=”” content=”descripción”> muestra lo que aparecerá en la página de búsqueda

<h1> encabezado. Hay 6 niveles, siendo 1 el más grande

JavaScript (.js): variable Null implica que es vacío; mientras que undefined aún no tiene asignado ningún valor

Html: comentarios con <!-->. En javascript se colocan con //

Booleanos = true or false

Console.log solo es el resultado visual; mientras que return entrega el valor y permite guardarlo como variable. Return se usa además para declarar el resultado final de una función. CON RETURN ACABA LA FUNCION Y NO LE HACE CASO A LO QUE SIGUE DESPUES

No importa si se llama a la función antes de definirla; siempre que esté definida en alguna parte del código

a-=2 es lo mismo que a=a-2

para el caso de incremento o reducción en una unidad: a-=1 == a=a-1 == a--

message=”hola”

message.lenght será igual a 4 (cuenta los caracteres). En cambio en un arreglo cuenta el número de elementos

x+y genera la unión total de caracteres, si es que son textos. También funciona entre cadenas y números

i%2 representa al resto de dividir i entre 2

Math.round(num) y Math.floor(num) redondea y devuelve el menor entero, respectivamente. Math.ceil(num) devuelve el entero superior. Math.trunc(num) devuelve el entero más cercano a cero

HTML transforma el código en lo que queremos mostrar en la página web. Los elementos que utiliza son:

* Encabezados
* Imágenes
* Vínculos
* Listas
* Tablas
* Formularios

Mediante etiquetas. El navegador interpreta el contenido de las etiquetas

La etiqueta div ocupa todo el espacio horizontal de la página, con altura igual al tamaño de lo que se escribe (utiliza todo el bloque de la pantalla – tipo blockdisplay”)

La etiqueta span también usa la misma altura, pero solo ocupa horizontalmente lo que ocupe el texto (tipo inline, solo lo que necesita)

“si no cabe a la derecha, se manda para abajo”

El estilo CSS se puede escribir en la misma página (en la cabecera) ó en un archivo aparte

Tres formas de colocar un selector:

* Nombre de etiqueta titulo { propiedad: valor}. Afecta a todas esas etiquetas!!. Para ser más específico, se puede colocar id en los elementos (incluso h1, p, etc.)
* #título { } selector de ID, solo a un elemento
* .título {} para clases

Propiedad “color” cambia color de letra. “background-color” cambia el color de fondo

Atributo para etiquetas “class” se usa como .class:”texto-verde”. Esto se debe crear en CSS como clase

Selector avandado: \* { } selecciona todos los elementos (las que tienen propiedad individual permanecen)

En CSS se comenta con /\* \*/

Span > label { } este selector modifica todas las etiquetas que se encuentren dentro de algún span

Propiedad display: block ocupa todo el ancho del padre; inline ocupa lo necesario y no puede aplicársele propiedades como ancho o padding; inline-block sí permite aplicar propiedades;

Los elementos que son bloques por default son div, h1, párrafos, tablas, form, body, html, listas

Los elementos que son inline son span, imágenes, hipervínculo (anchorlink),

Elemento inline-block: button

Estos elementos sí se pueden convertir a otro tipo de elementos con css. Por ejemplo p{display:inline-block}

Los porcentajes se usan con contenidos adaptables. En cambio pixeles se usan para que las medidas nunca cambien.

Interfaz de usuario gráfica, Sitio web

Descripción generada automáticamente

Se recomienda usar el modelo de caja

El borde tiene fuera de él al margen; mientras que dentro de él a padding. Así, margin aleja al bloque de lo que lo rodea; mientras que padding aleja al borde del contenido

\* Outline: enmarca todos los elementos

Posición natural es tal cual aparece sin cambios en la pantalla

Posición absoluta coloca al objeto dentro del plano del objeto padre. El objeto PIERDE su posición natural. Por lo que otros objetos pasan a tomar dicha posición natural.

Posición relativa cambia con respecto a la natural. NO AFECTA LA NATURAL DE OTROS OBJETOS

Terminal de github

Pwd: indica la carpeta donde nos encontramos actualmente

Ls: lista todo lo disponible

Ls -l: lista más ordenada

Cd dev: (change directory) permite entrar a la carpeta dev DESDE DONDE ESTOY ACTUALMENTE

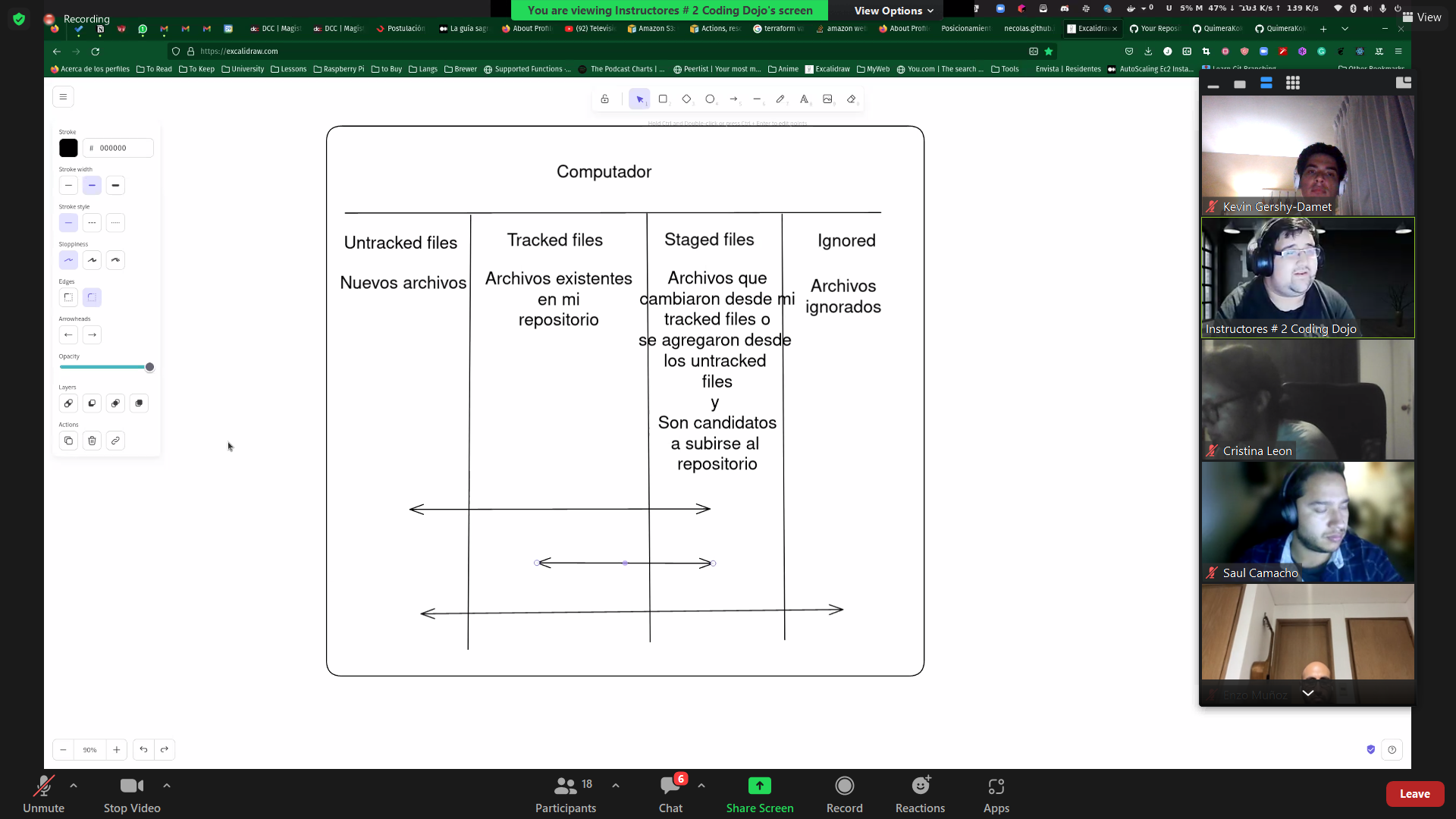
Ls-la: muestra carpetas y además añade carpeta . y .. – La carpeta . es la corriente; y la .. es la carpeta padre

Git clone ENLACE

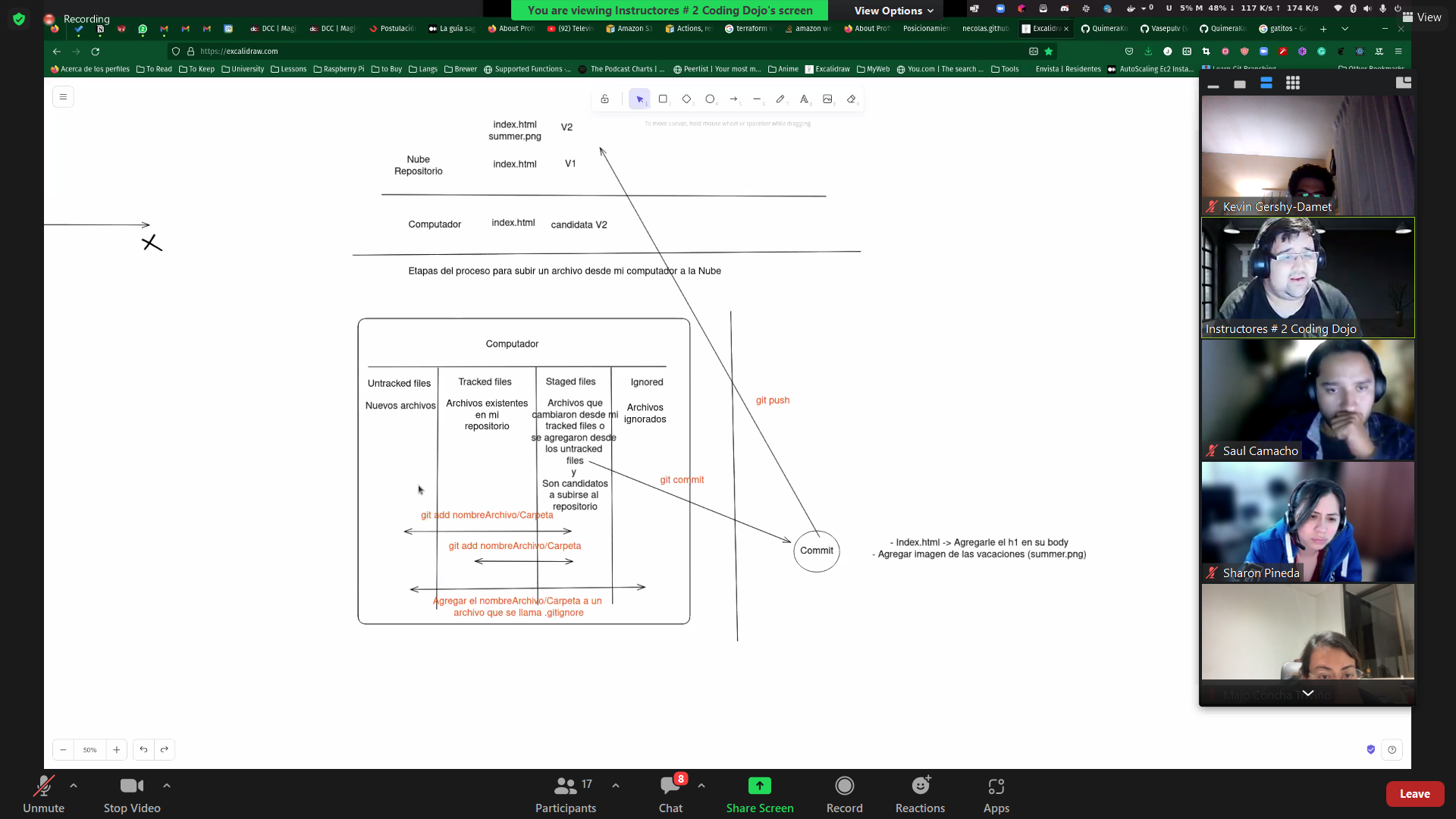
En ENLACE se copia la ruta del repositorio (en download) en github

Una vez que se hacen los cambios requeridos offline, 4 ambientes (son marcas – pasan de una a otra solo cuando lo forzamos/decidimos):

1. Untracked files (archivos nuevos que se subirán al repositorio)
2. tracked files (en el repositorio y pueden ser modificados)
3. Staged files (archivos que cambiaron desde mis tracked files o se agregaron desde los untracked files – son candidatos a subirse al repositorio)
4. Ignored



Commit: paquete de cambios (Agregar nuevos archivos, modificar archivos vigentes, etc). Vienen del campo “staged”.



Git stage: verifica lo que está por subirse

Resulta de gran relevancia reset/normalizar CSS para homogeneizar contenido entre todos los navegadores. De la mano con validar los códigos. La forma más fácil de normalizar es con márgenes y padding:

\* {

padding: 0;

margin: 0;

}

Width, height: inherit (COGE EL DEL PADRE)

Value un elemento SIEMPRE es de tipo texto (javascript)

Función parseInt toma texto y lo transforma a tipo numérico. LOS NÚMEROS APARECEN EN VERDE EN LA CONSOLA

Propiedad en html: href=<mailto:correo@gmail.com>

Git remote add NOMBRE(repositorio) url

Git push NOMBRE\_REMOTO RAMA (solo sube a la rama específica) – esto sube todos los commits de una rama

Git merge - -help muestra ayuda

Programación de bajo nivel: interactúa con componentes duros del ordenador, genera movimiento de mouse, cómo se traduce el teclado en pantalla

Los objetos pueden tener propiedades con nombre de número (o letras separadas por espacios). En ese caso no se pone objeto.propiedad, sino objeto[propiedad]

Windows + R abre símbolo. Cmd permite entrar a la terminal para realizar comandos

Comandos de terminal:

Cd ruta (sin comillas) cambia la ruta

Cat archivo.js muestra el contenido

Pwd (present working directory) = cd. Muestra dónde estamos actualmente.

Ls = muestra lista de archivos disponibles

Ls -l = lista más detallada

Ls -a = lista con todos los archivos, incluido el . y el ..

El . es la misma carpeta; el .. es la carpeta que la contiene

Cd .

Cd .. regresa a la carpeta que contiene al directorio actual

Cd / raíz

mkdir new\_directory\_name CREA CARPETA

rmdir elimina carpeta vacía (si no está vacía, rmdir -rf)

touch archivo.ext crea archivo

rm archivo.ext lo elimina

mv archivo.ext ../nombre\_nuevo.ext (cambia de nombre y de ubicación)

cp archivo.ext nombre\_nuevo.ext (duplica el archivo)

cp -rf copia el directorio (con nombre y nombre nuevo)

start . (abre directorio) start archivo abre el archivo

agregar sudo si es que se quiere hacer esto en la nube

git config --global user.name "John Doe"

git config --global user.email johndoe@example.com

git config --global --list

Captura de pantalla de computadora

Descripción generada automáticamente