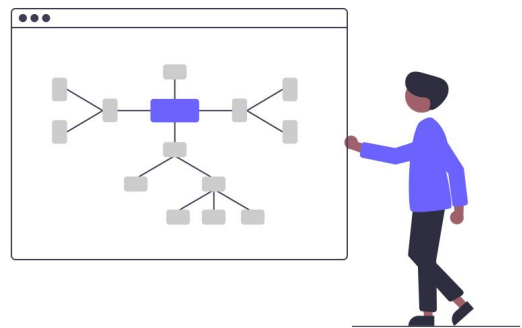


Cosas que debes saber, antes de animar elementos en CSS



- Las animaciones requieren recursos del sistema.
- Para cuidar los recursos del sistema se debe conocer la renderización de elementos.
- Procura usar (Transform y Opacity) Ocupan muchos menos recursos.



Proceso de renderización:

Javascript/CSS

Layout

Paint

Composite

Composite

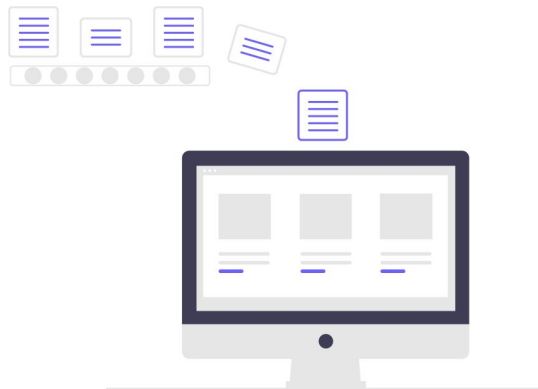
Se encarga de dar el orden adecuado a las capas del navegador, hace referencia a elementos que no necesitan layout o paint. (Transform)

- Da el orden a las capas en el navegador.
- Combina y dibuja las capas en pantalla.



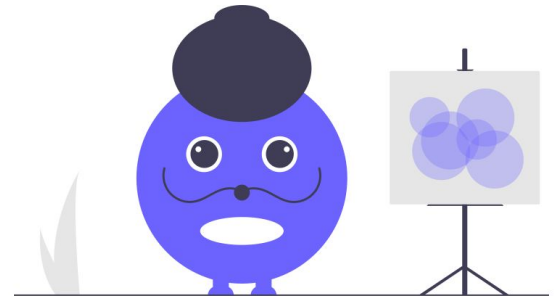
Composite

Si modificas una propiedad en la que no se requiere diseño ni pintura, el navegador la omite y continúa con la composición. Esta versión final es la más económica y la ideal para los puntos de presión alta del ciclo de vida de la app, como las animaciones o el desplazamiento.



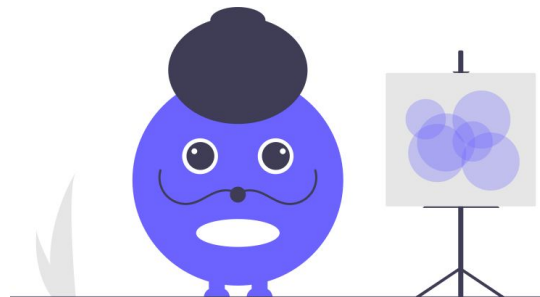
Paint

La pintura es el proceso de rellenar los píxeles. Esto implica dibujar texto, colores, imágenes, bordes y sombras. básicamente todas las partes visuales de los elementos.



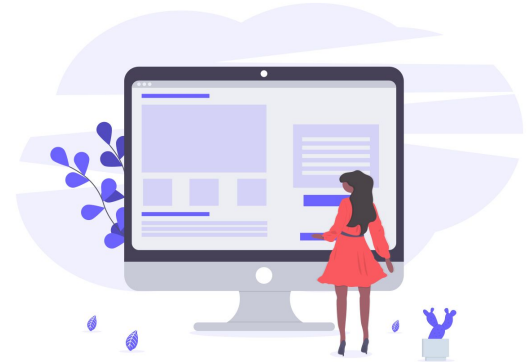
Paint

Si cambiaste una propiedad de "paint", como una imagen de fondo, color de texto o sombras (es decir, una que no afecta el diseño de la página) el navegador omite el diseño, pero realiza la pintura de todos modos.



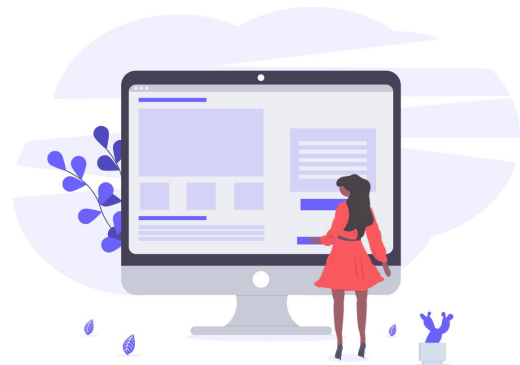
Layout

Es el proceso a través del cual el navegador descifra la información geométrica de los elementos: su tamaño y ubicación en la página. (Width, Height).



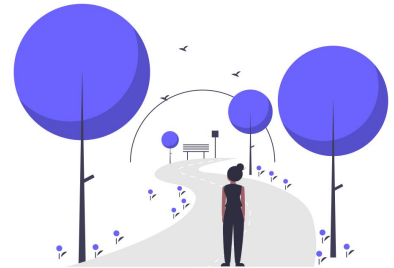
Layout

Si modificas una propiedad de “layout”, dicha propiedad modifica la geometría de un elemento (como su ancho, su altura o su posición a la izquierda o en la parte superior), y el navegador deberá verificar todos los otros elementos y “redistribuir” la página. Las áreas afectadas se deberán volver a pintar, y los elementos pintados finales se deberán volver a componer en conjunto.



Recomendaciones:

- Procura usar Composite.
- Evita usar animaciones simultáneas.
- Procura el uso de Transform y Opacity.



Referencias:

<https://developers.google.com/web/fundamentals/performance/rendering/stick-to-compositor-only-properties-and-manage-layer-count?hl=es>

<https://developers.google.com/web/fundamentals/performance/rendering/>

<https://developers.google.com/web/fundamentals/performance/rendering/avoid-large-complex-layouts-and-layout-thrashing>

<https://developers.google.com/web/fundamentals/performance/rendering/simplify-paint-complexity-and-reduce-paint-areas>

<https://csstriggers.com/>