



25+ Years
of Experience

PROGRAMMING
ADVICES

LEARN THE
RIGHT WAY

Mohammed Abu-Hadhoud

MSA, PMOC, PMP®, PMP®, PMP-ACP®, CS, ITIL®, MCPD, MCD



لا تنسى الاشتراك في قناتنا على اليوتيوب ومشاركة القناة مع اصدقائك
لتعم الفائدة للجميع وانقاذ الاف الناس من التشتت جزاكم الله خيرا

لا تنسونا من دعائكم وادعو لوالدي بالرحمة

www.ProgrammingAdvices.com



مهم جداً

هذا الملف للمراجعة السريعة واخذ الملاحظات عليه فقط ،لانه يحتوي على اقل من 20٪ مما يتم شرحه في الفيديوهات الاستعجال والاعتماد عليه فقط سوف يجعلك تخسر كميه معلومات وخبرات كثيره

يجب عليك مشاهدة فيديو الدرس كاملا

لاتنسى عمل لايك ومشاركة القناة لتعم الفائدة للجميع
لا تنسونا من دعائكم

ProgrammingAdvices.com

Mohammed Abu-Hadhoud



لا تخن الامانة

الأسعار الخاصة بالكورسات رمزية للغاية ولا تمثل 1% من قيمتها الفعلية، ومع ذلك تتوفر كوبونات دعم لكل طالب محتاج.

عند شراء الكورس، فهو مخصص لك وحدك، ولا يحق لك القيام بما يلي:

- تحميل الفيديوهات وتوزيعها.
- مشاركة حسابك مع الآخرين.
- الاشتراك في شراء الكورس مع أصدقائك، حيث يحق لكل فرد كوبون دعم إذا كان بحاجة إليه.
- استخدام كوبون دعم دون وجه حق.

إن مخالفة هذه التعليمات قد تؤدي إلى إغلاق المنصة، مما يحرم آلاف الطلاب من فرصة التعلم. وستكون مسؤولاً أمام الله عن ذلك، وأنا لن اسامح من يسيء استخدام هذه المنصة.

تذكر أن الله لا يبارك في عمل مبني على أخذ حقوق الآخرين أو حرمانهم من التعلم. لا تبدأ حياتك بما لا يرضي الله.

مع تحياتي، د. محمد أبو هدهود



How Browsers Deal With Images?



ProgrammingAdVICES.com

HTML DEEP DIVE

Dr. Mohammed Abu-Hadhoud
DBA, MBA, PMOC, PgMP®, PMP®, PMI-RMP®, CM, ITILF, MCPD, MCSD



How Browsers Deal With Images?

- During the process of loading a webpage, the browser does indeed continue rendering the HTML content while simultaneously fetching images.
- This concurrent activity is part of the browser's optimized resource handling to ensure the webpage becomes interactive and visible to the user as quickly as possible.
- Here's how the process unfolds, with a focus on how rendering and image fetching are managed:

Concurrent Rendering and Loading

- The key to modern web browsers' performance is their ability to do many of these tasks concurrently:
 - Non-blocking Resources: Images are typically loaded in a way that doesn't block the rendering of other elements.
 - Browsers use multiple connections to handle various resources simultaneously.
 - Progressive Rendering: Browsers try to render progressively, which means they display parts of the webpage as they become ready.
 - This approach improves the user experience by making the page seem faster and responsive as soon as possible.

Detailed Process Breakdown

1- Parsing HTML:

- As the browser parses the HTML document, it constructs the DOM (Document Object Model).
- During this parsing, whenever it encounters an `` tag, it recognizes a resource that needs to be fetched. The parsing continues while these resources are being noted for fetching.

2- Sending Request:

- As soon as the browser encounters an `` tag with a `src` attribute during the HTML parsing, it sends an HTTP request for the image, even if it hasn't finished parsing the entire document.
- This is typically done asynchronously, meaning that the browser doesn't need to wait for the image to be downloaded to continue processing the HTML.

3- Downloading Image:

- Image files are downloaded in the background, allowing the browser to continue with other tasks such as parsing the rest of the HTML, loading CSS and JavaScript, and rendering content that has already been processed.
- The downloading process does not block the browser from performing these other tasks unless specifically instructed to do so (e.g., by scripts that wait for an event related to the image).

Detailed Process Breakdown

4- Rendering Image: The rendering of the image happens after the image data has been fully received and decoded.

- If the image's location in the layout is already processed and ready, the browser will display the image as soon as it's available.
- If the specified dimensions (width and height) are known before the image is fully loaded, the browser can also reserve space for the image, which helps prevent layout shifts.

5- Caching: While caching doesn't directly impact the rendering of the current page on the first visit, it significantly speeds up image loading for subsequent visits. This cache process runs in the background and is managed by the browser's cache policy.

Important Notes:

- Specifying the dimensions of an image (width and height) in the `` tag have on the page loading process : It helps prevent layout shifts during loading, improving the user experience.
- Specifying width and height helps maintain the layout stability as the browser can allocate space for images before they are fully loaded, thus preventing layout shifts.
- Specifying image dimensions is crucial because it helps the browser allocate the correct amount of space on the page for an image before the image has fully loaded. This prevents the page content from shifting unexpectedly as resources load, which can improve user experience and reduce Cumulative Layout Shift, a critical metric for user experience and SEO.
- Image optimization involves compressing the image file size without significantly degrading its quality, converting images to the most efficient format, and loading them appropriately based on the user's device and network conditions.
 - This practice is essential in web development because it reduces bandwidth usage, speeds up web page loading times, and improves the overall performance and user experience.
- It is NOT a recommended practice to use vector images for icons and simple graphics when using images in web design.
- To improve webpage load time, it's advisable to compress the size of images to reduce the amount of data transferred over the network.
- The primary reason for converting simple images such as icons into SVG format is For scalability and performance without loss of quality.

Important Notes:

- Using a Content Delivery Network (CDN) affects image loading on websites, Using a CDN can significantly improve image loading times for websites by storing copies of images on geographically distributed servers.
 - This minimizes latency by delivering images from the server closest to the user's location, reduces bandwidth costs, and can enhance the overall user experience with faster load times and reduced server load.
- HTTP/2 introduces multiplexing, which allows multiple files to be transferred simultaneously over a single connection.
 - This is a significant improvement over HTTP/1.1, where each file required a separate connection.
 - This results in more efficient use of available bandwidth and faster overall loading times for resources like images, especially on websites with many assets.

Conclusion

- Understanding that image loading and HTML rendering often occur simultaneously is crucial for web developers aiming to optimize performance and user experience.
- Efficient use of asynchronous loading, careful management of resource priorities, and understanding the impact of network conditions on resource loading are fundamental aspects of modern web development.



programmingAdvices.com
Thank You

Mohammed Abu-Hadhoud
26+ Years of Experience
MBA, PMOC, PgMP®, PMP®, PMI-RMP®, CM, ITILF, MCPD, MCSD

