**1. Prioritize Content Delivery Network (CDN) Usage**

A **CDN (Content Delivery Network)** stores copies of your website's content on servers around the world.  
👉 **Why?**  
It makes your website faster by delivering files from the server closest to the user’s location, reducing delays.  
**Example:** Cloudflare or AWS CloudFront.

**2. Optimize Images and Media Files**

Large image and video files slow down your website.  
👉 **Solution:**  
Use tools to **compress images and videos** without losing quality. This reduces file size and speeds up loading.  
**Example:** TinyPNG for images or HandBrake for videos.

**3. Minimize HTTP Requests**

Every element on your web page (like images, CSS, or JavaScript files) creates a separate **HTTP request** to the server.  
👉 **Why minimize?**  
Fewer requests mean faster page loading. Combine files and reduce unnecessary elements on the page.

**4. Leverage Browser Caching**

Browser caching stores **static files (images, CSS, JavaScript)** on the user’s device after their first visit.  
👉 **Benefit:**  
When they return to the site, it loads faster because the browser already has the files saved.  
**How?** Configure caching settings on your server.

**5. Reduce Server Response Time**

Server response time is the time your server takes to respond to a user’s request.  
👉 **How to improve it?**

* Use **fast hosting services**
* **Optimize backend code**
* Reduce unnecessary **database queries**

**6. Minify CSS and JavaScript Files**

Minification means **removing unnecessary spaces, comments, and characters** from code.  
👉 **Why?**  
It makes your **CSS and JavaScript files smaller**, helping the website load faster.  
**Tools:** CSSNano or UglifyJS.

**7. Implement Lazy Loading**

**Lazy loading** means images or videos are only loaded when the user **scrolls** to them.  
👉 **Why?**  
This reduces the initial page load time and saves bandwidth.  
**Example:** Add loading="lazy" to image or video tags.

**8. Use Asynchronous Loading for JavaScript**

If JavaScript files load **synchronously**, they can block the page from rendering until they finish loading.  
👉 **Solution:**  
Use **asynchronous (async)** or **deferred (defer) loading** to load JavaScript without blocking the page.

**9. Audit and Optimize Third-Party Scripts**

Third-party scripts (like ads, analytics, or widgets) can slow down your website.  
👉 **What to do?**  
Regularly review them and **remove unnecessary scripts** or use lightweight alternatives.

**10. Regularly Test Website Speed**

Use tools like **Google PageSpeed Insights**, **GTmetrix**, or **Lighthouse** to monitor your website’s speed.  
👉 **Why?**  
They provide **detailed reports** on performance issues and suggestions for improvement.