Audience

- Who is the web.dev audience?
  - Metrics-initiated problem solver (Alice)
  - Goal-driven problem solver (Damir)
  - Topic deep-diver (Andrew)
  - Demo builder (Emily)
  - New developer (Jaimie)
- Writing to the audience
  - Value propositions

When you're writing for web.dev, it's important to consider the persona or personas you're writing for. Keeping your audience in mind helps you make good writing decisions, like how much background information to include or whether you need to define a key term. Those considerations help ensure that your readers can learn or accomplish something.

Who is the web.dev audience?

web.dev readers come with a variety of needs and preferences based on several characteristics:

- Prior knowledge about the domains of web development (for example, accessibility or performance)
- Role (for example, hobbyist, student, data scientist, frontend engineer)
- Organization (for example, independent contractor, startup, large company in a non-tech industry)
- Focus on a business vertical (for example, e-commerce)
- Preferences for frameworks and tools (for example, React, Gulp, webpack)
- Purpose for reading (for example, fix a Lighthouse audit, solve a specific problem, improve general skills)

Trying to address all permutations of those characteristics is unmanageable, so we've captured the most common clusters in several personas.

Metrics-initiated problem solver (Alice)

Alice is a developer working in a team for a large e-commerce website. The site's homepage received a low Lighthouse score for performance. Alice has been assigned to resolve the top three opportunities for improving the performance in the next two weeks, with the first solution due later today.

- **Prior knowledge:** Alice is familiar with DevTools and understands the Lighthouse metrics well enough to tell whether a resource is relevant to her problem. Alice also has enough experience to be able to apply a concept from a generic sample code to her specific problem.
- **Entry point:** A "Learn more" link in a Lighthouse performance audit takes Alice to the related post on web.dev.
- **Goal:** Find a solution to a low Lighthouse score that is directly applicable to her existing project as quickly as possible.

Goal-driven problem solver (Damir)

Damir is a developer at a small fintech startup. His boss has asked him to help turn their company's site into a PWA. After trying to enable some offline functionality, Damir realizes through a Stack Overflow post that the best way to solve his problem is to create a service worker. He needs to learn how to create a service worker that works with his company's existing site and the dynamic data that it handles.

- **Prior knowledge:** Damir is a more advanced developer. He can quickly skim documentation to find the code snippets and examples relevant to his use case.
- **Entry point:** Damir searches for "service worker for PWA", and one of the first search results is a post in the web.dev Installable collection.
- **Goal:** Find a specific, implementable solution to an issue as quickly as possible. (Unlike Alice, Damir is trying to fix a problem identified by his team or manager rather than by Lighthouse.)

Topic deep-diver (Andrew)

Andrew is a frontend developer at a startup whose main product is a peer-to-peer resource sharing web app. He wants to become the go-to accessibility person on the engineering team since accessibility is an issue he feels is important but knows very little about. Andrew has signed up to give a 30-minute accessibility introductory talk to his team in two weeks.

- **Prior knowledge:** Andrew has heard of accessibility and knows that it's about helping users with special needs, but he doesn't have a fully accurate sense of who those users are or how they interact with his product.

- **Entry point:** Andrew's searches for accessibility, and one of the first search results is the landing page for the Accessible to all collection.
- **Goal:** Master the basics of web accessibility and find high-quality resources to learn from and share with fellow engineers.

Demo builder (Emily)

Emily works as a software engineer. In her free time, she likes experimenting with the most recent web development technologies and tools. Emily has no experience with portals and wants to try adding seamless page transitions to a prototype on her personal portfolio.

- **Prior knowledge:** Emily has extensive web development experience and is aware of the technology she wants to learn (portals in this case) but doesn't know any details about how to use it.
- **Entry point:** Emily has used web.dev before, so she visits the site homepage and searches for "portals." She finds a relevant post on the blog.
- **Goal:** Create a working prototype of a specific type of app or site (one using portals in this case) as quickly as possible.

New developer (Jaimie)

Jaimie first took a web development class at their local community college and is now a computer science student at a California state university. They are applying for internships to gain more hands-on experience, and they want to learn about the latest web development technologies to become a more competitive candidate.

- **Prior knowledge:** Jaimie has taken several college-level computer science courses, but they've only covered the basics of HTML, CSS, and vanilla JavaScript.
- **Entry point:** Whether directed to the page through searching or at the recommendation of a friend or instructor, Jaimie arrives at the "Getting Started" collection (forthcoming) on web.dev.
- **Goal:** Learn about the most recent web development tools and best practices to create a personal site from scratch to apply for industry jobs.

Writing to the audience

That said, we want our content to address as much of our audience as we can. How do you do that in a given piece, especially when the needs of different readers compete? One way to address multiple audience segments is to provide multiple paths for a given topic. For example, you might write:

- A post using vanilla JavaScript, with associated codelabs using common frameworks to solve the same problem
- A post about optimizing images in general, with associated codelabs showing how to integrate image optimization into different build processes
- A post about how to address an issue most web developers encounter, with sections at the end focused on solving the problem as it appears in particular business verticals

Be mindful of your voice. You want readers to feel they can trust the guidance you're giving. They should feel neither talked down to nor overwhelmed. Check out Instruction for adult learners in the Writing effective instruction post.

Value propositions

A value proposition is a promise to readers that a piece of content will be useful and provide value to them. Create a value proposition for everything you write by crafting titles, subtitles, and descriptions that tell your target audience why your post or codelab is worth reading. Here are some common strategies:

- State what readers will learn or understand by reading the piece
- Explain what a piece helps readers do
- Describe how a technique or tool improves the user experience

---

□□□ #2
□ □□□ □□□ 2026-10-31 22:38:48 UTC
□ □□□ □□□ 2026-10-31 22:38:49 UTC