Prefers-reduced-motion

What is it for?

Some people struggle with motion on websites. This can be due to a number of reasons, including (but not limited to):

- ADHD
- Motion sickness
- Vestibular disorders

While it's best not to use any motion that autoplays, sometimes it's unavoidable. So that this doesn't create barriers for users, you can use the prefers-reduced-motion media query to stop the motion from happening for those who choose not to see it.

Reducing motion can have other benefits too, such as saving battery life and data on mobile devices.

Many new operating systems allow you to choose to reduce the motion that you see on websites (at least, on websites where they've used the prefers-reduced-motion media query).

If you're on a Mac, you can toggle it on or off in your settings here - `System Settings` -> `Accessibility` -> `Display` -> `Reduce motion`

If you're on Windows, you can toggle it on or off in your settings here - `Settings` -> `Accessibility` -> `Visual effects` -> `Animation effects`

Take a look at Can I use... prefers-reduced-motion for information about which browsers support this feature

How to support it

It's like any other media query (such as screen size). The two options available are `reduce` or `no-preference`. You can use it in CSS to turn off an element's motion.

If you're providing moving media to developers, also provide a non-moving variant that they can use in the media query.

If you're a developer, there are two ways you can use it. Either you can reduce the motion if the user has set a preference to reduce motion, like this:

```
.moving-element {
    animation: bounce 500ms;
}
@media (prefers-reduced-motion: reduce) {
    .moving-element {
        animation: none;
    }
}
```

Or you can set the motion to happen if the user hasn't set a preference. Here's an example of using it like this:

```
@media (prefers-reduced-motion: no-preference) {
    .moving-element {
        animation: bounce 500ms;
    }
}
```

This is the recommended way, because browsers that don't support the prefers-reduced-motion media query will ignore this code, so your element will have no motion as standard. It's also less code. People who haven't turned on the setting will still see the motion, in the same way as the first method.

Useful links

- Web Content Accessibility Guidelines (WCAG) 2.2 Techniques - Using the CSS prefers-reduced-motion query to prevent motion
- Respecting users' motion preferences
- MDN - prefers-reduced-motion