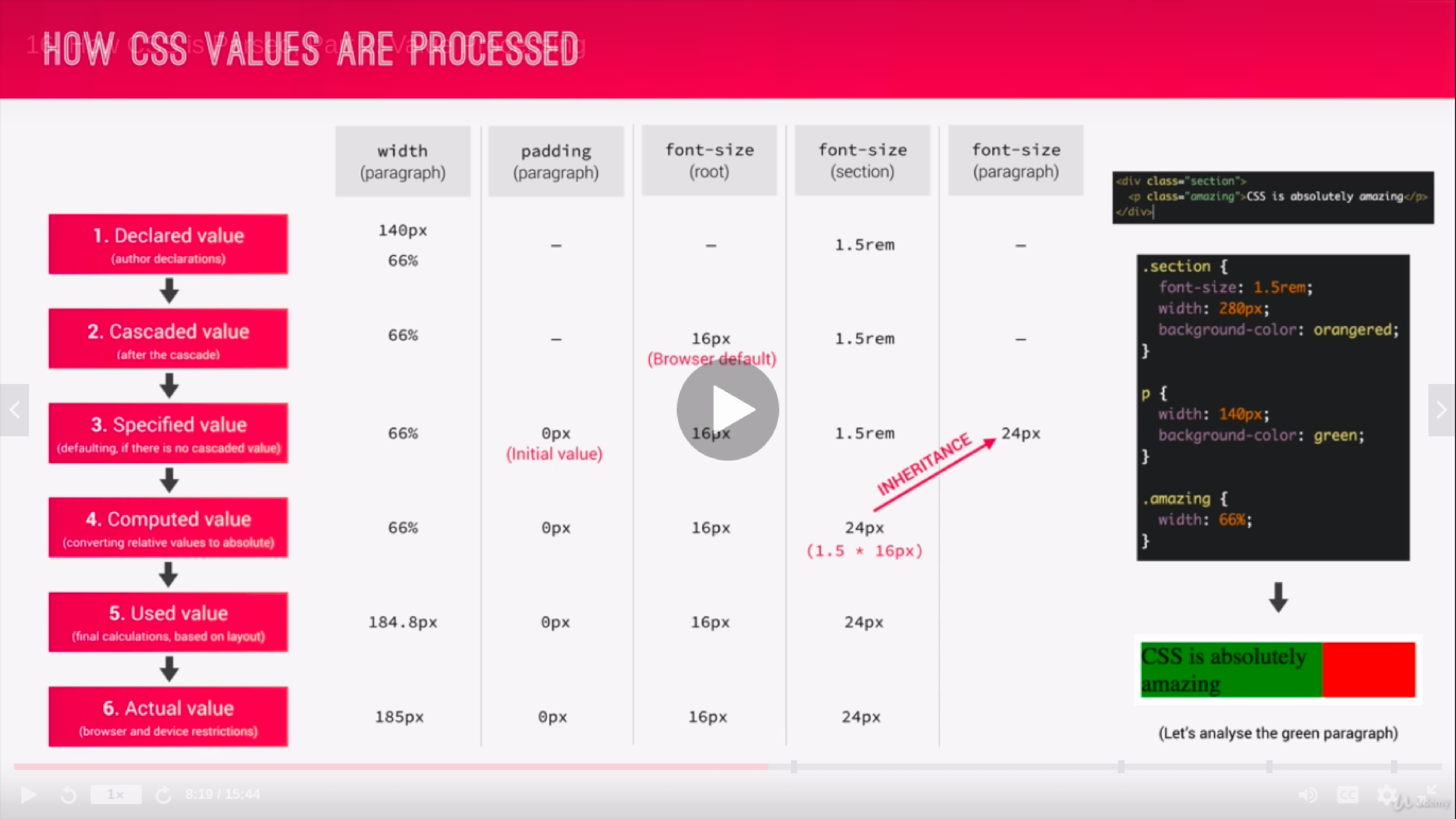

HOW CSS VALUES ARE PROCESSED

	width (paragraph)	padding (paragraph)	font-size (root)	font-size (section)	font-size (paragraph)
1. Declared value (author declarations)	140px 66%	–	–	1.5rem	–
2. Cascaded value (after the cascade)	66%	–	16px (Browser default)	1.5rem	–
3. Specified value (defaulting, if there is no cascaded value)	66%	0px (Initial value)	16px	1.5rem	24px
4. Computed value (converting relative values to absolute)	66%	0px	16px	24px (1.5 * 16px)	
5. Used value (final calculations, based on layout)	184.8px	0px	16px	24px	
6. Actual value (browser and device restrictions)	185px	0px	16px	24px	

INHERITANCE

```
<div class="section">
  <p class="amazing">CSS is absolutely amazing</p>
</div>
```

```
.section {
  font-size: 1.5rem;
  width: 280px;
  background-color: orangered;
}

p {
  width: 140px;
  background-color: green;
}

.amazing {
  width: 66%;
}
```

CSS is absolutely amazing

(Let's analyse the green paragraph)

HOW UNITS ARE CONVERTED FROM RELATIVE TO ABSOLUTE (PX)

		Example (x)	How to convert to pixels	Result in pixels
	% (fonts)	150%	x% * parent's computed font-size	24px
	% (lengths)	10%	x% * parent's computed **width**	100px
Font-based	em (font)	3em	x * **parent** computed font-size	72px (3 * 24)
	em (lengths)	2em	x * **current element** computed font-size	48px
	rem	10rem	x * **root** computed font-size	160px
Viewport-based	vh	90vh	x * 1% of viewport height	90% of the current viewport height
	vw	80vw	x * 1% of viewport width	80% of the current viewport width

```
html, body {
  font-size: 16px;
  width: 80vw;
}

header {
  font-size: 150%;
  padding: 2em;
  margin-bottom: 10rem;
  height: 90vh;
  width: 1000px;
}

.header-child {
  font-size: 3em;
  padding: 10%;
}
```