





**TIL**

was die Stunde von CSS geschlagen hat

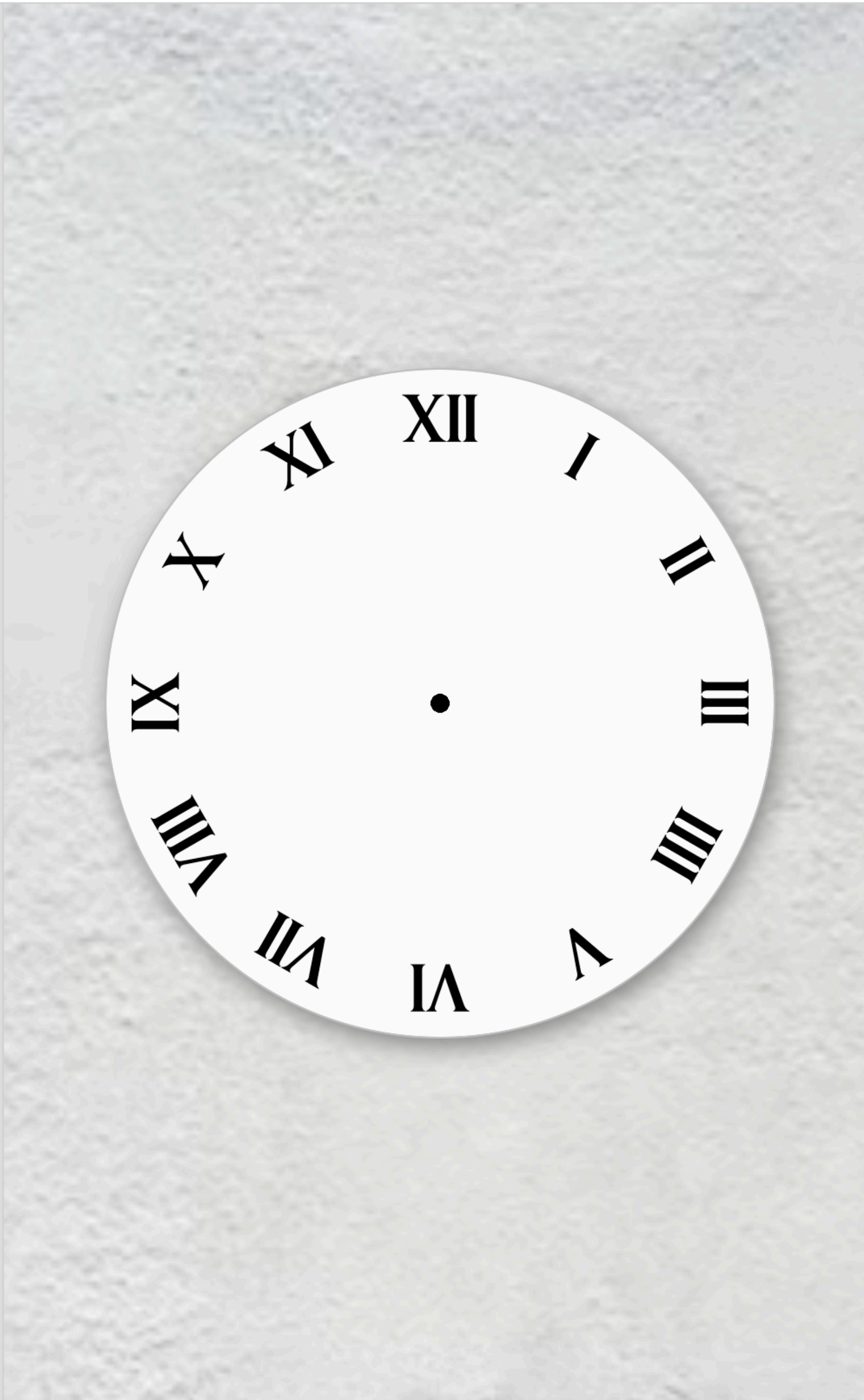
HTML

```
1 <ol class="clock" role="image">
2   <li></li>
3   <li></li>
4   <li></li>
5   <li></li>
6   <li></li>
7   <li></li>
8   <li></li>
9   <li></li>
10  <li></li>
11  <li></li>
12  <li></li>
13  <li></li>
14 </ol>
```

\* CSS (SCSS)

JS

```
1 // Look Ma, no JavaScript!
```



HTML

```
1 <main lang="de">
2 <h1>Marathon, Olympische Spiele Moskau 1980</h1>
3 <ol>
4 <li>Waldemar Cierpinski <span>(DDR)</span></li>
5 <li>Gerard Nijboer <span>(IOC Niederlande)</span></li>
6 <li>Satymkul Dschumanasarow <span>(Sowjetunion)</span></li>
7 <li>Uladsimir Kotau <span>(Sowjetunion)</span></li>
```

\* CSS

```
1 h1 {
2   font: 400 2rem/1.3 'Fira Sans', sans-serif;
3 }
4
5 li {
6   font: 400 1.4rem/1.3 'Fira Sans', sans-serif;
7   margin-block: 0.7em;
8
9   &:nth-of-type(-n+3) {
10    list-style: medals;
11  }
12 }
13
14 @counter-style medals {
15   system: fixed;
16   symbols: '🥇' '🥈' '🥉';
17   suffix: ' ';
18 }
```

JS

# Marathon, Olympische Spiele Moskau 1980

- 🥇 Waldemar Cierpinski (DDR)
- 🥈 Gerard Nijboer (IOC Niederlande)
- 🥉 Satymkul Dschumanasarow (Sowjetunion)
- Uladsimir Kotau (Sowjetunion)
- Leonid Mossejew (Sowjetunion)
- Rodolfo Gómez (Mexiko)
- Dereje Nedi (Äthiopien)
- Massimo Magnani (IOC Italien)
- Karel Lismont (IOC Belgien)
- Robert de Castella (IOC Australien)

Heinz Ferdinand Oetzel

1927 - 2023

Hannelore Oetzel

1938



HTML

CSS

```
18 @counter-style upper-roman {
19   system: additive;
20   range: 1 3999;
21   additive-symbols:
22     1000 M,
23     900 CM,
24     500 D,
25     400 CD,
26     100 C,
27     90 XC,
28     50 L,
29     40 XL,
30     10 X,
31     9 IX,
32     5 V,
33     4 IV,
34     1 I;
35 }
36
37 @counter-style iiii-roman {
38   range: 1 4;
39   system: symbolic;
40   symbols: 'I';
41   fallback: upper-roman;
42 }
```

JS

I  
II  
III  
IIII  
V  
VI  
VII  
VIII  
IX  
X  
XI  
XII  
XIII  
XIV  
XV



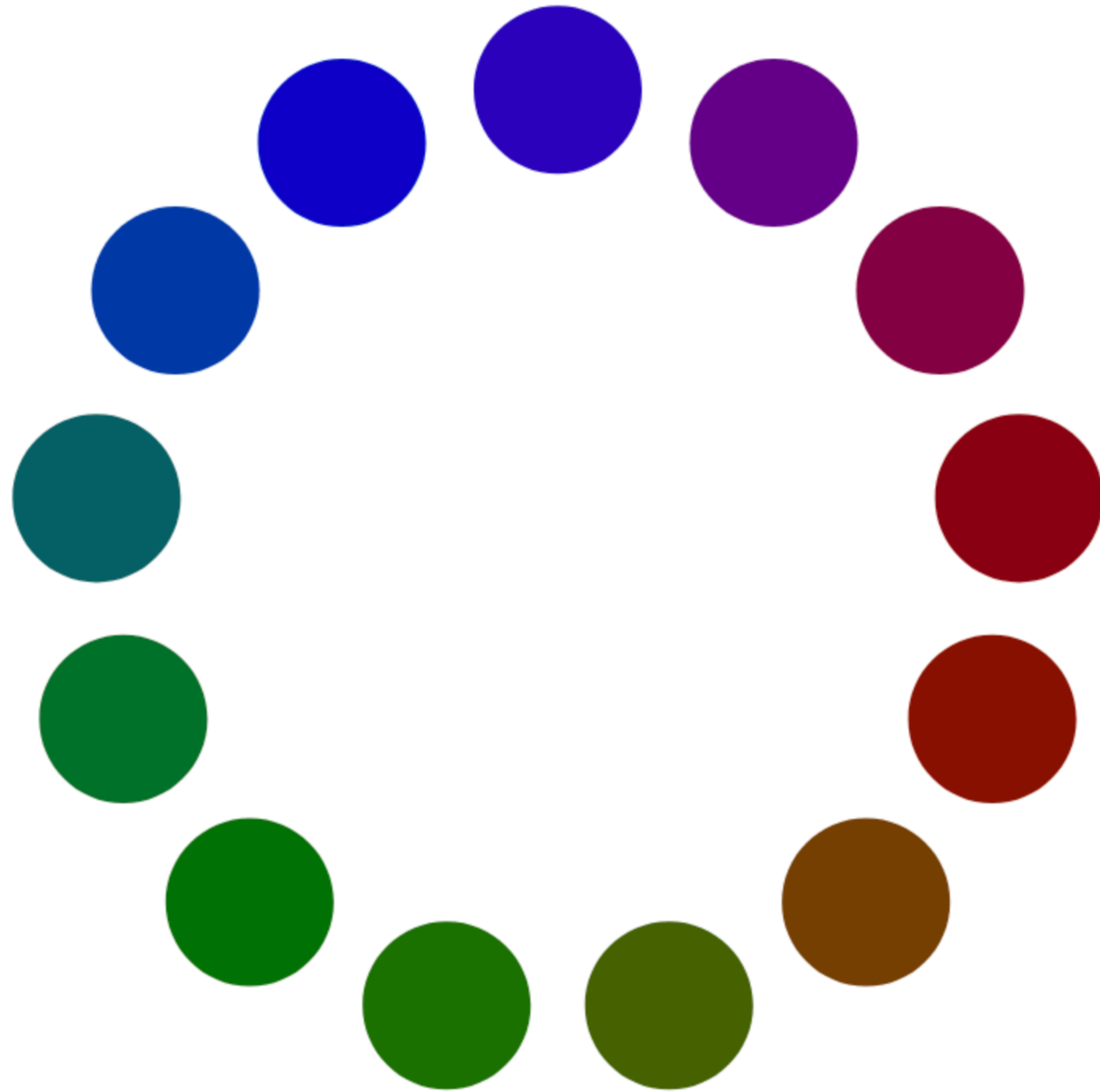


SHEPHERD PATENTEE

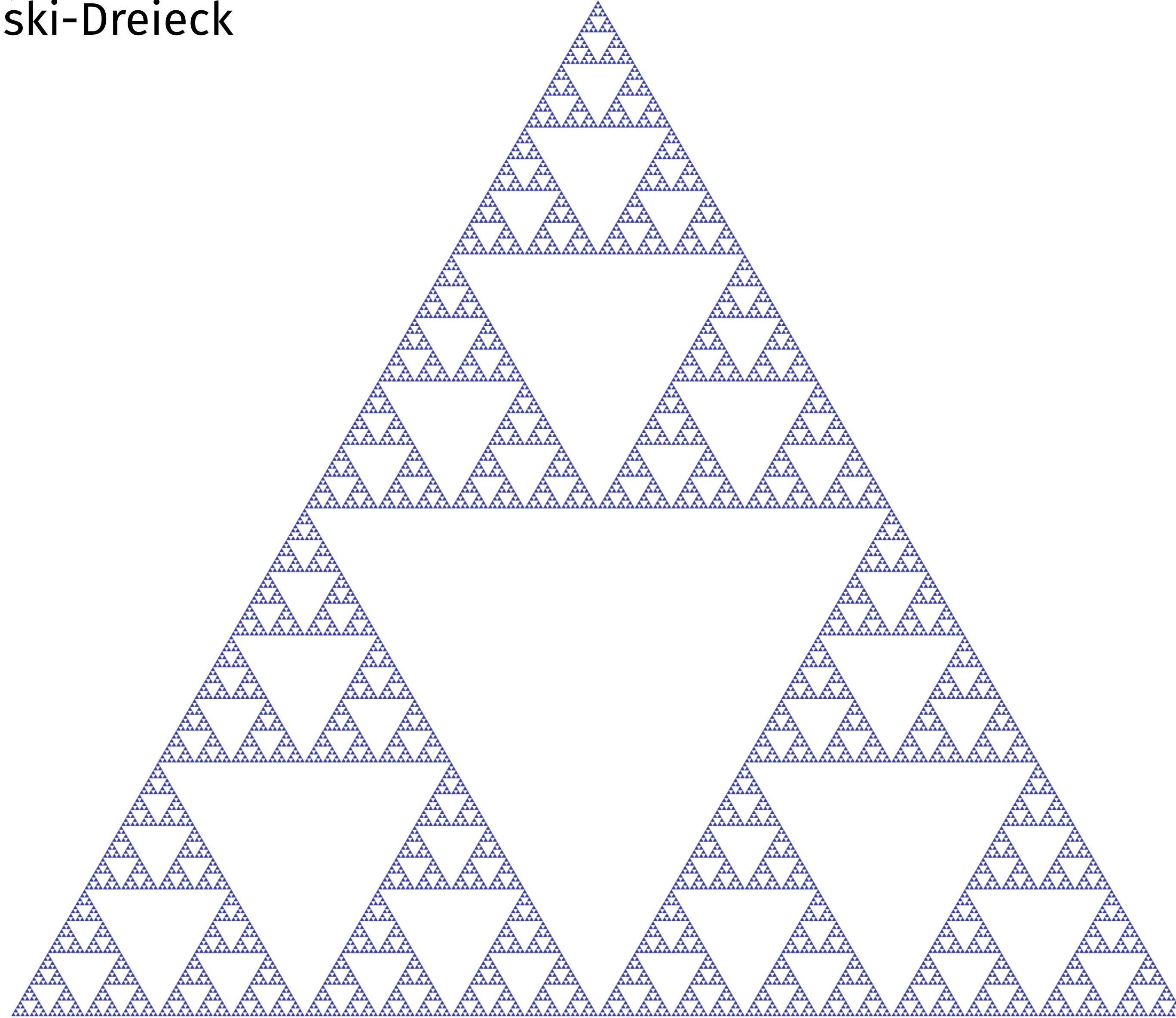
53 LEADENHALL ST LONDON

CALVA O-MAGNETIC CLOCK





# Sierpiński-Dreieck

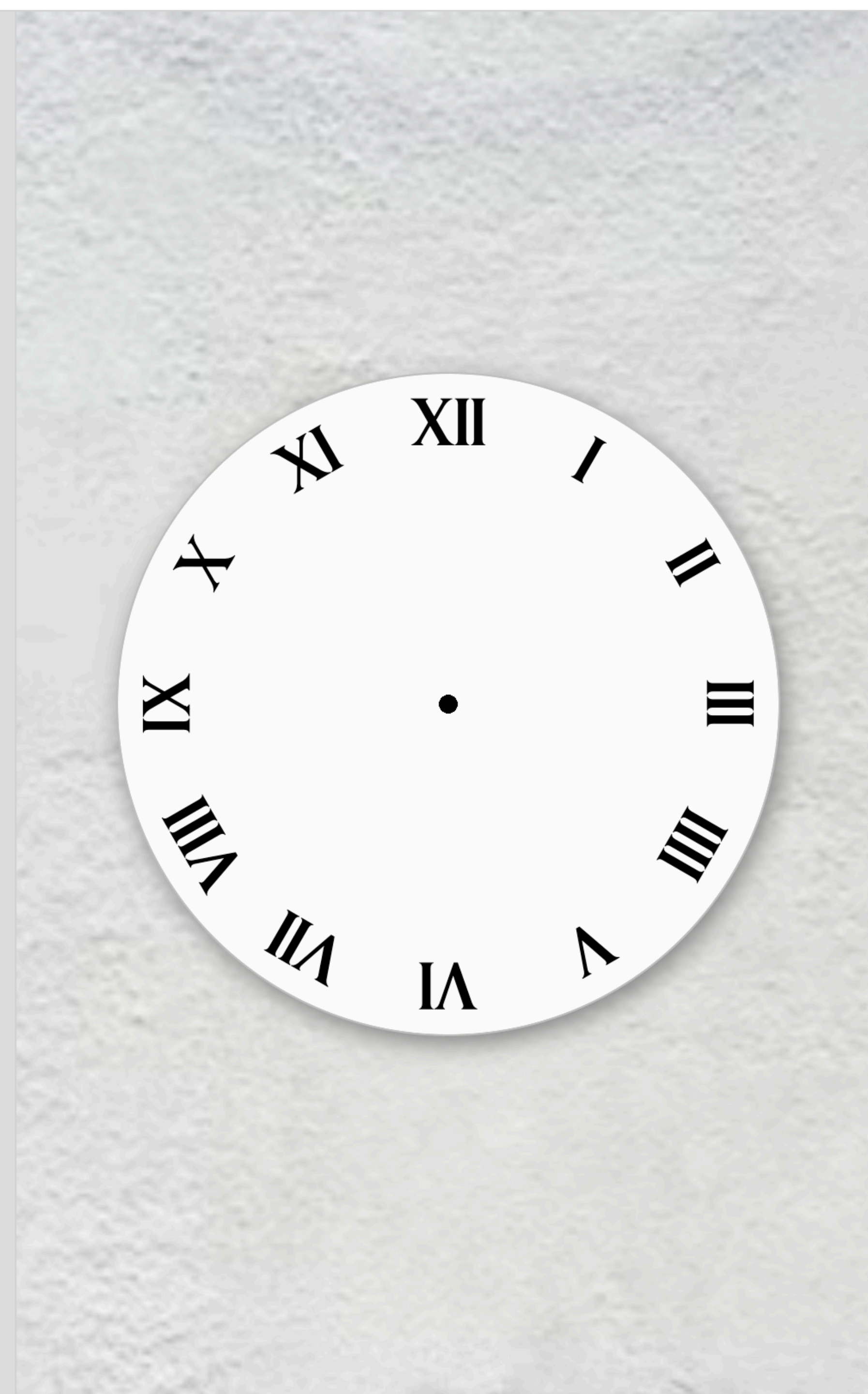


HTML

\* CSS (SCSS)

```
39 li:nth-of-type(2) { --index: 2 }
40 li:nth-of-type(3) { --index: 3 }
41 and so on...
42
43 preprocessor for convenience:
44 */
45 @for $i from 1 through 12 {
46   &:has(li:nth-of-type(#{ $i })) { --count: #{ $i } }
47   & li:nth-of-type(#{ $i }) { --index: #{ $i } }
48 }
49
50 & li {
51   font: 10cqw/1 'Alchemist', serif;
52   font-variant-numeric: proportional-nums lining-nums;
53   grid-column: 1;
54   grid-row: 1;
55   place-self: center;
56   text-align: center;
57   height: 100cqw;
58   padding: 3cqw;
59   --angle: calc(var(--index) / var(--count) * 1turn);
60   rotate: var(--angle);
61
62 &::marker {
63   content: counter(list-item, var(--style));
```

JS



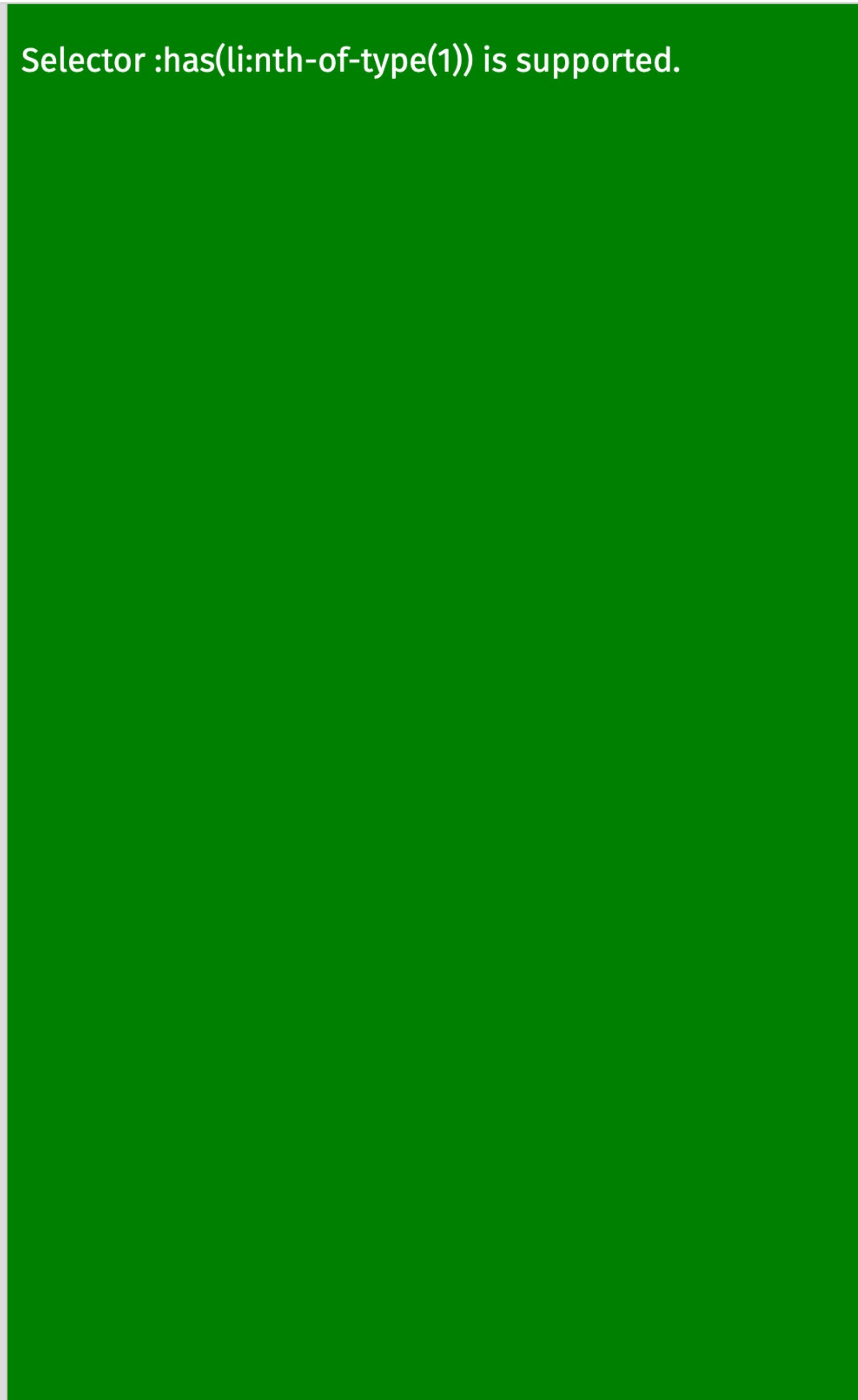
HTML

\* CSS

```
1 @supports (selector(:has(li:nth-of-type(1)))) {
2   html { background: green }
3   #not { display: none }
4 }
5
6 @supports not (selector(:has(li:nth-of-type(1)))) {
7   html { background: red }
8 }
9
10 html {
11   color: white;
12   font: 1.2em/1.4 Fira Sans;
13 }
```

JS

```
1 // :has() polyfill
2 if (!CSS.supports('selector(&:has(li:nth-of-type(1)))')) {
3   const setCount = (element) => {
4     element.style.setProperty('--count', element.childElementCount);
5   };
6
7   const countObserver = new MutationObserver(setCount);
8
9   for (let element of document.querySelectorAll('.clock')) {
10     setCount(element);
11     countObserver.observe(element, {childList: true});
12   }
13 }
```



HTML CSS (SCSS) JS

```
1 // :has() polyfill
2 if (!CSS.supports('selector(&:has(li:nth-of-type(1)))')) {
3   const setCount = (element) => {
4     element.style.setProperty('--count', element.childElementCount);
5   };
6
7   const countObserver = new MutationObserver(setCount);
8
9   for (let element of document.querySelectorAll('.clock')) {
10    setCount(element);
11    countObserver.observe(element, {childList: true});
12  }
13 }
```

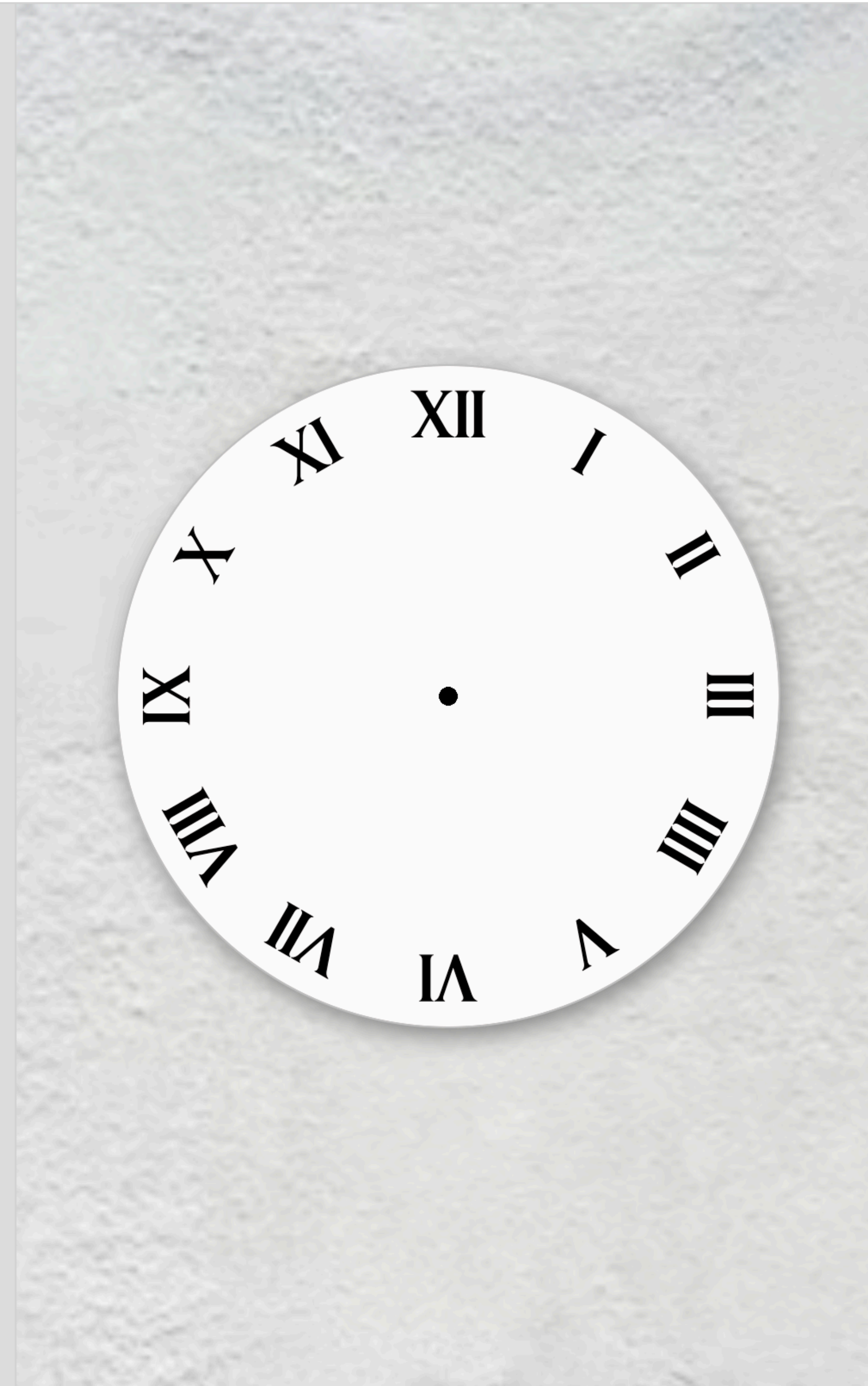






















Foto: Black Chitsulo auf Unsplash



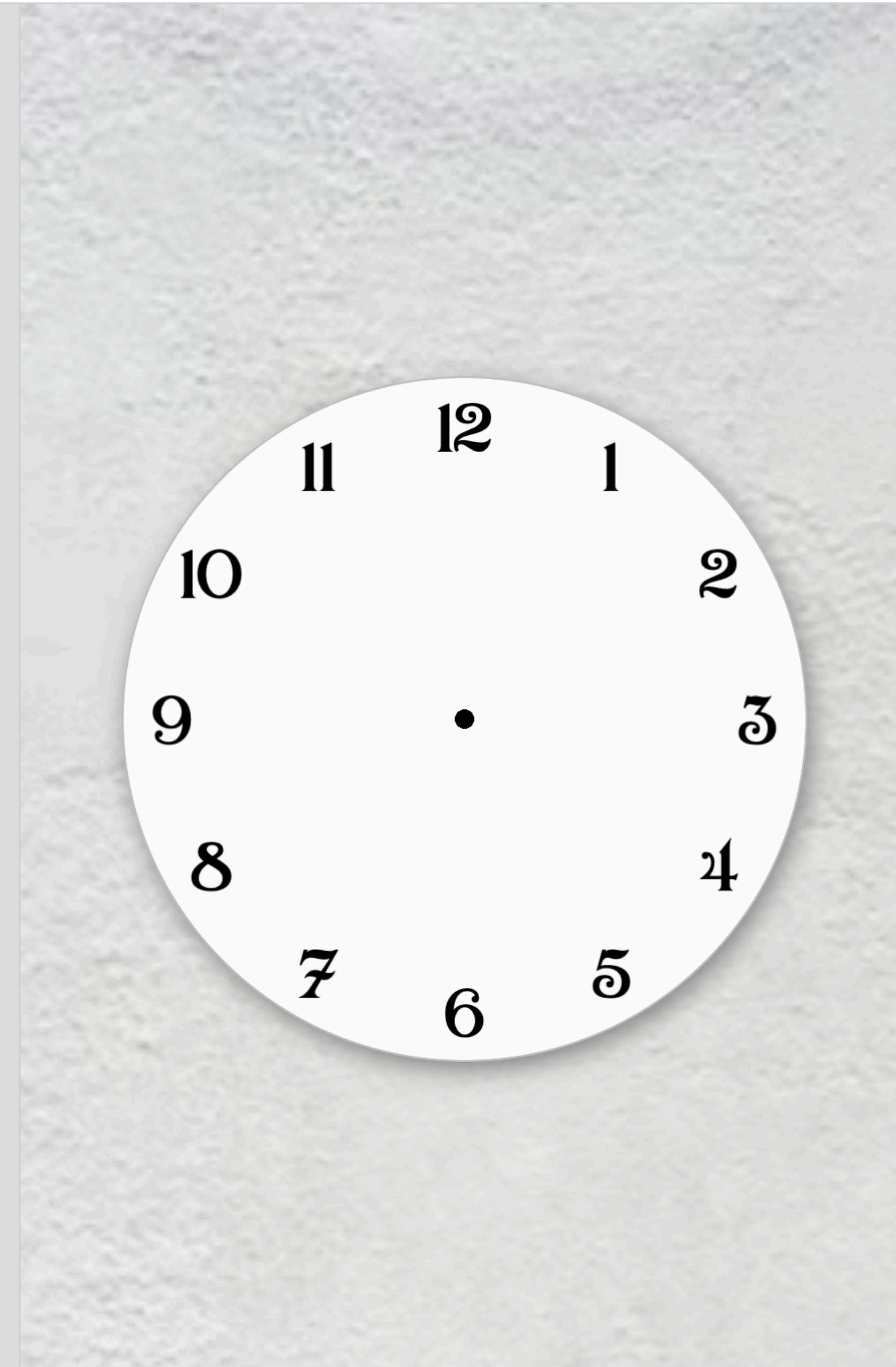
Foto: Jiyeon Park auf Unsplash

```
HTML  
* CSS (SCSS)  
29
30 & li {
31   font: 10cqw/1 'Alchemist', serif;
32   font-feature-settings: 'ss01';
33   font-variant-numeric: proportional-nums lining-nums;
34   padding: 3cqw;
35   grid-column: 1;
36   grid-row: 1;
37   place-self: center;
38   text-align: center;
39   height: 100cqw;
40   --angle: calc(var(--index) / var(--count) * 1turn);
41   rotate: var(--angle);
42
43 &::before {
44   content: counter(list-item);
45   display: block;
46 }
47
48 &:nth-of-type(n+4):nth-of-type(-n+8)::before {
49   rotate: 0.5turn;
50   margin-top: -1.8cqw; /* magic number, depending on used font */
51 }
52 }
53 }
```



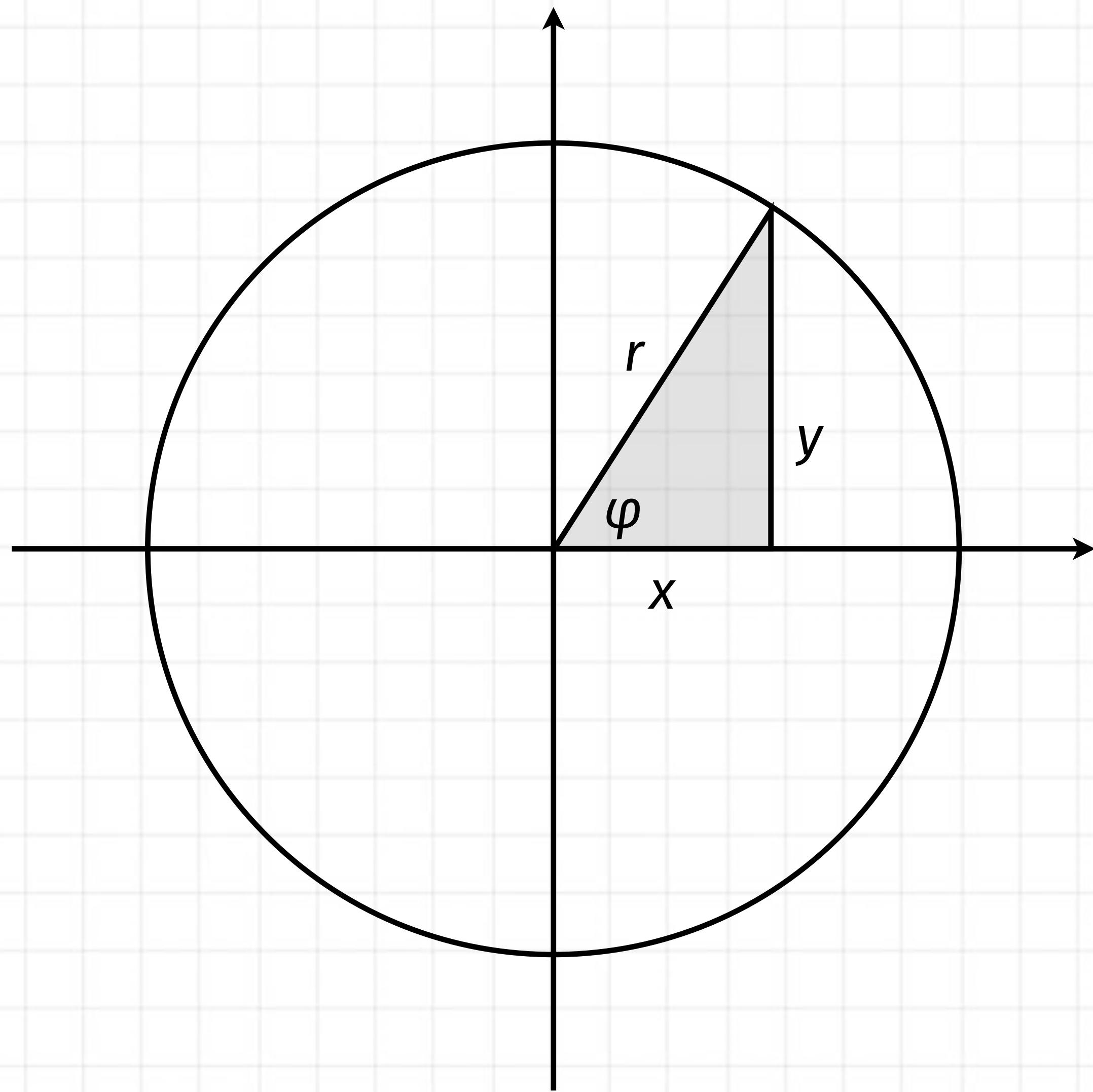
```
JS  
```

```
HTML  
* CSS (SCSS)  
29
30 & li {
31   font: 10cqw/1 'Alchemist', serif;
32   font-feature-settings: 'ss01';
33   font-variant-numeric: proportional-nums lining-nums;
34   grid-column: 1;
35   grid-row: 1;
36   place-self: center;
37   text-align: center;
38   height: 100cqw;
39   --angle: calc(var(--index) / var(--count) * 1turn);
40   rotate: var(--angle);
41
42 &::before {
43   content: counter(list-item);
44   display: block;
45   rotate: calc(-1 * var(--angle));
46   padding-block: 3cqw 1.2cqw; /* magic number, depending on used font */
47 }
48 }
49 }
```



```
JS  
```





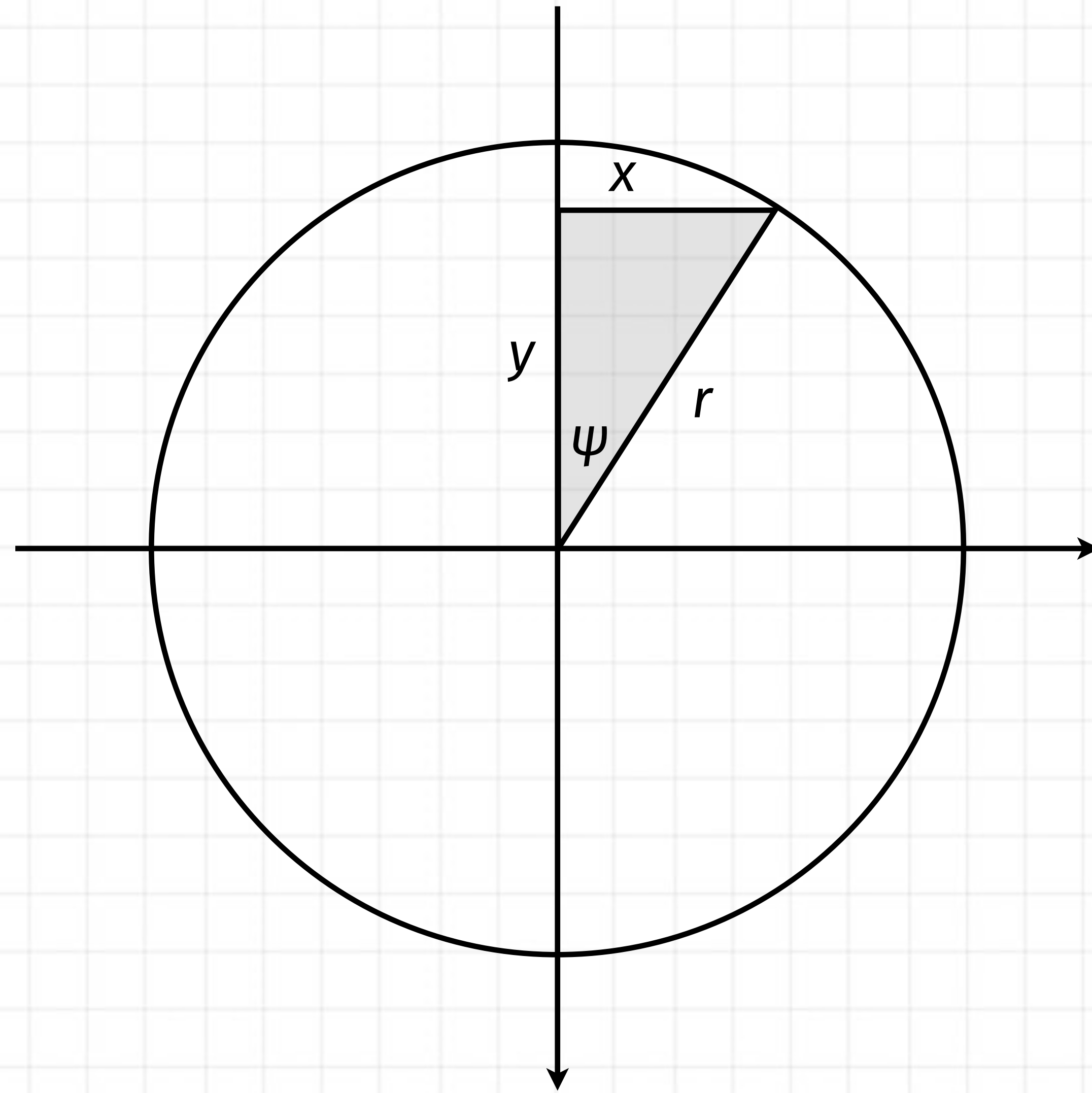
$$\sin \varphi = y / r$$

$$\cos \varphi = x / r$$

$$\tan \varphi = y / x$$

$$y = r \sin \varphi$$

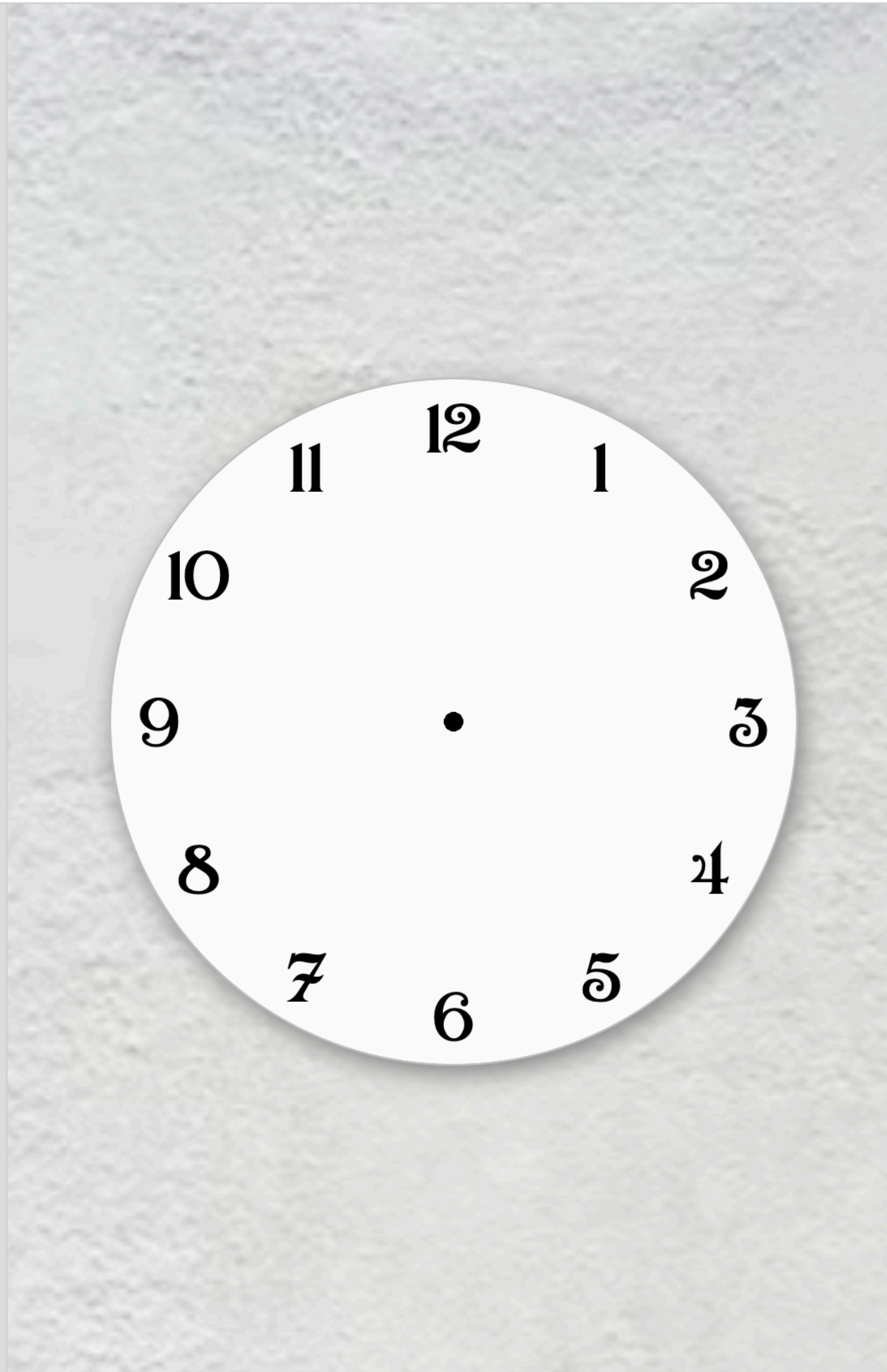
$$x = r \cos \varphi$$



$$x = r \sin \psi$$

$$y = -r \cos \psi$$

```
HTML
CSS (SCSS)
29
30 & li {
31   font: 10cqw/1 'Alchemist', 'Bodoni 72', serif;
32   font-feature-settings: 'ss01';
33   font-variant-numeric: proportional-nums lining-nums;
34   grid-column: 1;
35   grid-row: 1;
36   place-self: center;
37   text-align: center;
38   --radius: 43cqw;
39   --angle: calc(var(--index) / var(--count) * 1turn);
40   translate:
41     calc(var(--radius) * sin(var(--angle)))
42     calc(var(--radius) * -1 * cos(var(--angle)));
43
44 &::before {
45   content: counter(list-item);
46   display: block;
47   padding-block: 3cqw 1.2cqw; /* magic number, depending on used font */
48 }
49 }
50 }
```



```
JS
```

HTML

\* CSS (SCSS)

```
1 div {
2   background: white;
3   box-shadow: 2px 2px 20px darkgray;
4   height: min(80vh, 120vw);
5   aspect-ratio: 1/sqrt(2); /* `sqrt()` not supported in Chromium yet */
6 }
7
8 body {
9   box-sizing: border-box;
10  margin: 0;
11  padding: 1em;
12  height: 100vh;
13  background: url(https://images.unsplash.com/photo-1611072337226-1140ab367200?auto=format&fit=crop&q=80&w=1000&ixlib=rb-4.0.3&ixid=M3wxMjA3fDB8MHxwaG90by1wYWdlfHx8fGVufDB8fHx8fA%3D%3D) center/cover;
   /* background image by Simon Berger on Unsplash */
14  display: grid;
15  place-items: center;
16 }
17
18 html {
19  filter: grayscale(70%);
20 }
21
```

JS



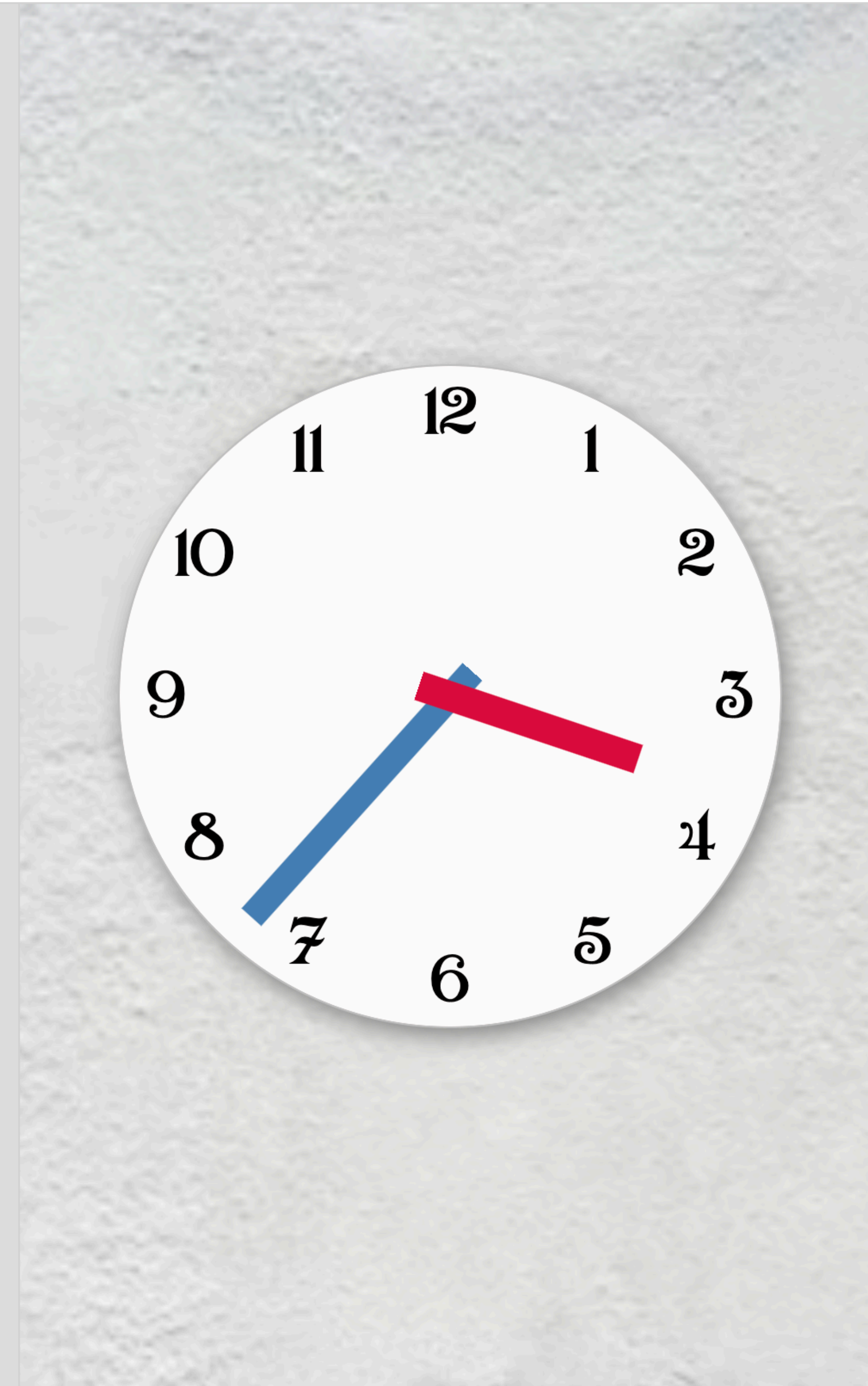
HTML

\* CSS (SCSS)

```
33 &::before {
34   background: linear-gradient(
35     transparent 0 5%,
36     steelblue 0 55%,
37     transparent 0
38   );
39   width: 4cqw;
40   rotate: calc(var(--partOfHour) * 1turn);
41   transition: rotate linear calc(var(--refresh, 0) * 1ms/2);
42 }
```

JS

```
1 const clockElement = document.querySelector('.clock');
2 const refresh = 500; // ms
3 if (refresh > 750) {
4   clockElement.style.setProperty('--refresh', refresh);
5 }
6
7 clockElement.setTime = function() {
8   window.requestAnimationFrame(() => {
9     const now = new Date();
10    clockElement.style.setProperty('--hours', now.getHours());
11    clockElement.style.setProperty('--minutes', now.getMinutes());
12    clockElement.style.setProperty('--seconds', now.getSeconds() +
now.getMilliseconds()/1000);
13  });
14 }
```



```
HTML
```

```
* CSS (SCSS)
```

```
33 &::after {  
34   background: linear-gradient(  
35     transparent 0 5%,  
36     steelblue 0 55%,  
37     transparent 0  
38   );  
39   width: 4cqw;  
40   rotate: calc(var(--partOfHour) * 1turn);  
41   transition: rotate linear calc(var(--refresh, 0) * 1ms/2);  
42 }
```

```
JS
```

```
1 const clockElement = document.querySelector('.clock');  
2 const refresh = 500; // ms  
3 if (refresh > 750) {  
4   clockElement.style.setProperty('--refresh', refresh);  
5 }  
6  
7 clockElement.setTime = function() {  
8   window.requestAnimationFrame(() => {  
9     const now = new Date();  
10    clockElement.style.setProperty('--hours', now.getHours());  
11    clockElement.style.setProperty('--minutes', now.getMinutes());  
12    clockElement.style.setProperty('--seconds', now.getSeconds() +  
13    now.getMilliseconds()/1000);  
14  });  
15 }
```



```
HTML  
1 <figure>  
2 <ol data-timezone="America/New_York" class="clock" role="image"></ol>  
16 <figcaption>  
17 Manhattan  
18 </figcaption>  
19 </figure>  
20 <figure>  
21 <ol data-timezone="Europe/Berlin" class="clock" role="image"></ol>  
35 <figcaption>  
36 Berlin  
37 </figcaption>
```

```
* CSS (SCSS)
```

```
JS  
9 clockElement.setTime = function() {  
10 window.requestAnimationFrame(() => {  
11 const now = new Date();  
12 [h, m, s] = Intl.DateTimeFormat('de-DE', {  
13 hour: "numeric",  
14 minute: "numeric",  
15 second: "numeric",  
16 timeZone: clockElement.dataset.timezone,  
17 }).format(now).split(':').map(x => parseInt(x));  
18 clockElement.style.setProperty('--hours', h);  
19 clockElement.style.setProperty('--minutes', m);  
20 clockElement.style.setProperty('--seconds', s);  
21 });  
22 }
```



HTML

```
1 <figure>
2   <ol data-
   timezone="America/New
   _York" class="clock"
   role="image"></ol>
16 <figcaption>
17   Manhattan
18 </figcaption>
19 </figure>
20 <figure>
21 <ol data-
```

\* CSS (SCSS)

JS

```
9 clockElement.setTime
  = function() {
10 window.requestAnimati
   onFrame(() => {
11   const now = new
   Date();
12   [h, m, s] =
   Intl.DateTimeFormat('
   de-DE', {
13     hour:
   "numeric",
14     minute:
```



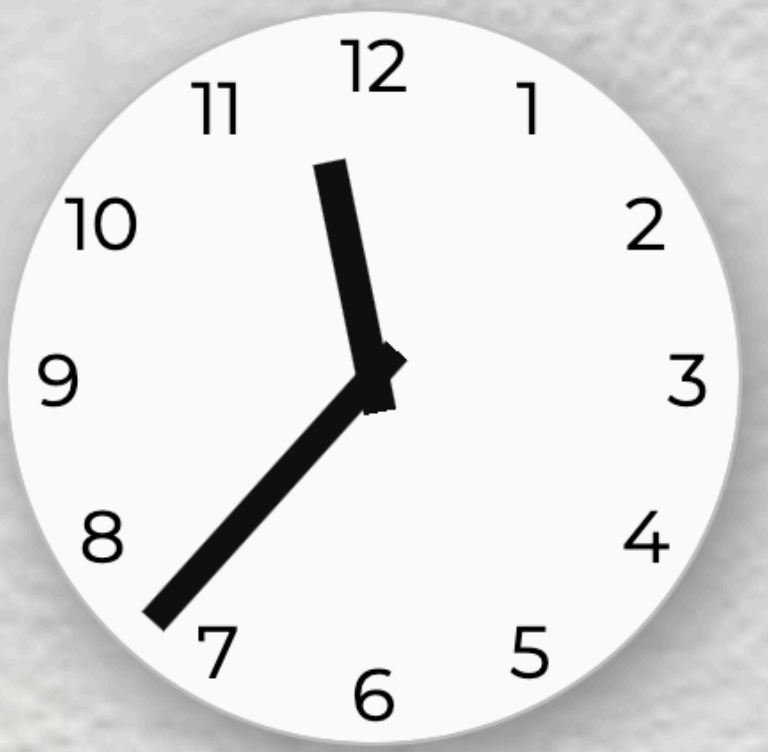
MANHATTAN



BERLIN



BOLLYWOOD



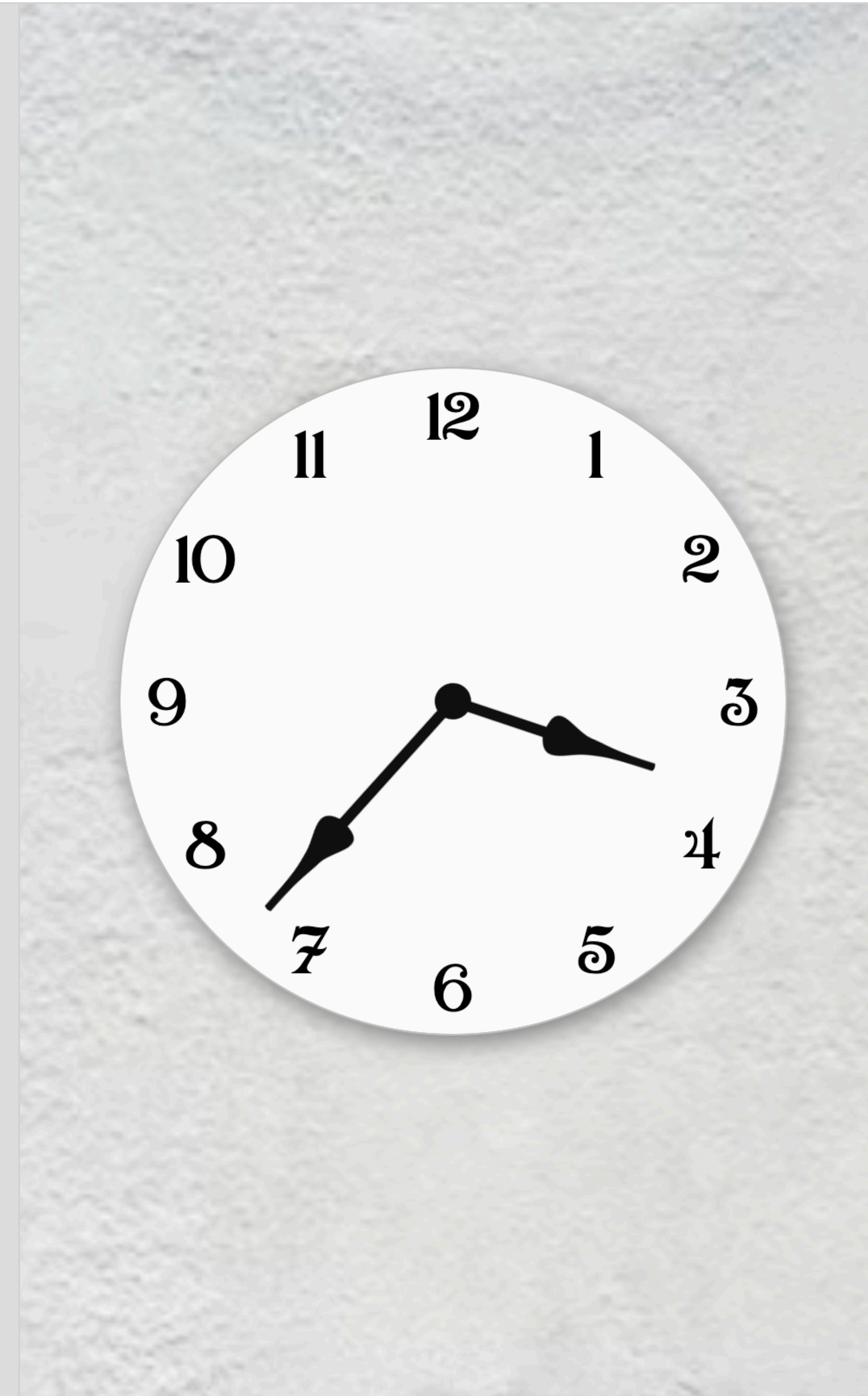
SEOUL

HTML

\* CSS (SCSS)

```
26
27 &::before, &::after {
28   content: '';
29   grid-column: 1;
30   grid-row: 1;
31   width: 7cqw;
32 }
33
34 &::after {
35   background: url("data:image/svg+xml,%3Csvg xmlns='http://www.w3.org
43 /2000/svg' viewBox='-50 -800 100 1600' fill='%231111'%3E%3Cpath d='m13-41a43
43 0 11-26 0v-352c-74-10-12-65-8-112s11-57 9-164q11-4 19 1c1 103 3 114 13
161s59 97-7 115z'/%3E%3C/svg%3E") no-repeat center;
36   rotate: calc(var(--partOfHour) * 1turn);
37   transition: rotate linear calc(var(--refresh, 0) * 1ms/2);
38 }
39
40 &::before {
41   background: url("data:image/svg+xml,%3Csvg xmlns='http://www.w3.org
42 /2000/svg' viewBox='-50 -800 100 1600' fill='%231111'%3E%3Cpath d='m13-41a43
43 0 11-26 0v-192c-74-10-12-65-8-112s11-57 9-164q11-4 19 1c1 103 3 114 13
161s59 97-7 115z'/%3E%3C/svg%3E") no-repeat center;
42   rotate: calc((var(--hours) + var(--partOfHour))/12 * 1turn);
43 }
44
```

JS





HTML

\* CSS (SCSS)

```
34
35 &::before, &::after {
36   content: '';
37   grid-column: 1;
38   grid-row: 1;
39   width: 7cqw;
40 }
41
42 &::after {
43   background: url("data:image/svg+xml,%3Csvg xmlns='http://www.w3.org
43 /2000/svg' viewBox='-50 -800 100 1600' fill='%231111'%3E%3Cpath d='m13-41a43
43 0 11-26 0v-352c-74-10-12-65-8-112s11-57 9-164q11-4 19 1c1 103 3 114 13
161s59 97-7 115z'/%3E%3C/svg%3E") no-repeat center;
44   rotate: calc(var(--partOfHour) * 1turn);
45   transition: rotate linear calc(var(--refresh, 0) * 1ms/2);
46 }
47
48 &::before {
49   background: url("data:image/svg+xml,%3Csvg xmlns='http://www.w3.org
49 /2000/svg' viewBox='-50 -800 100 1600' fill='%231111'%3E%3Cpath d='m13-41a43
43 0 11-26 0v-192c-74-10-12-65-8-112s11-57 9-164q11-4 19 1c1 103 3 114 13
161s59 97-7 115z'/%3E%3C/svg%3E") no-repeat center;
50   rotate: calc((var(--hours) + var(--partOfHour))/12 * 1turn);
51 }
52
```

JS



