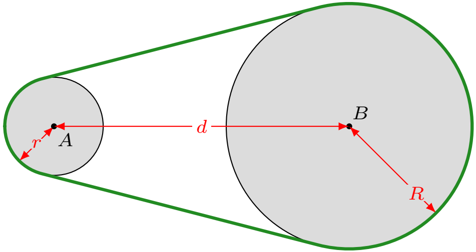
Nerdstuff

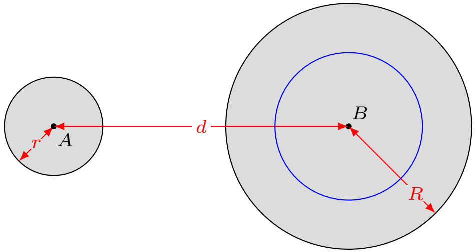
Source code at: https://github.com/dmorgorg/LaTeX/blob/master/misc/misc.pdf

Last updated on February 27, 2020



Two pulleys, centred at A and B, have radii r and R. The distance from A to B is d.

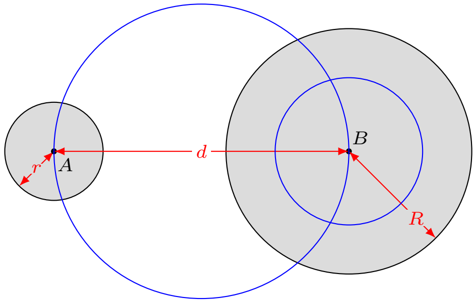
Determine the length of the belt required to go round both pulleys.



Two pulleys, centred at A and B, have radii r and R. The distance from A to B is d.

Determine the length of the belt required to go round both pulleys.

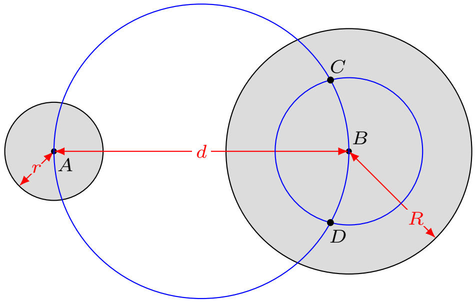
1. Construct a circle, diameter $R-r$, centred at B.



Two pulleys, centred at A and B, have radii r and R. The distance from A to B is d.

Determine the length of the belt required to go round both pulleys.

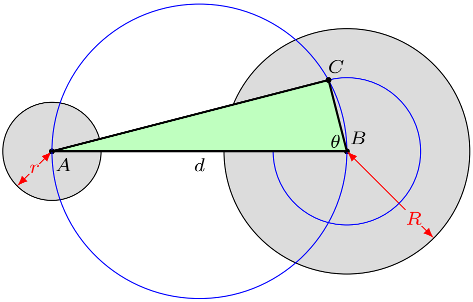
1. Construct a circle, diameter $R-r$, centred at B.
2. Construct a circle with diameter AB.



Two pulleys, centred at A and B, have radii r and R. The distance from A to B is d.

Determine the length of the belt required to go round both pulleys.

1. Construct a circle, diameter $R-r$, centred at B.
2. Construct a circle with diameter AB.
3. These two circles intersect at C and D. Due to the horizontal axis of symmetry through A and B, we only need perform calculations on one half of the system.



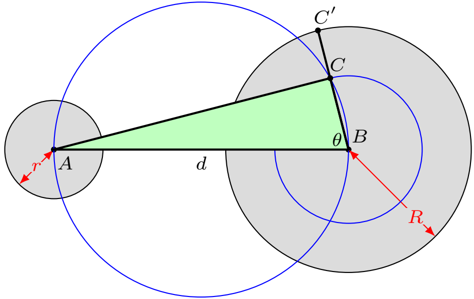
Two pulleys, centred at A and B, have radii r and R. The distance from A to B is d.

Determine the length of the belt required to go round both pulleys.

1. Construct a circle, diameter $R-r$, centred at B.
2. Construct a circle with diameter AB.
3. These two circles intersect at C and D. Due to the horizontal axis of symmetry through A and B, we only need perform calculations on one half of the system.
4. Consider $\triangle ABC$: $\angle ACB = 90^\circ$ since it is an angle inscribed in a semicircle. Then:

$$AC = \sqrt{d^2 - (R-r)^2}$$

$$\theta = \sin^{-1}\left(\frac{\sqrt{d^2 - (R-r)^2}}{d}\right)$$



Two pulleys, centred at A and B, have radii r and R. The distance from A to B is d.

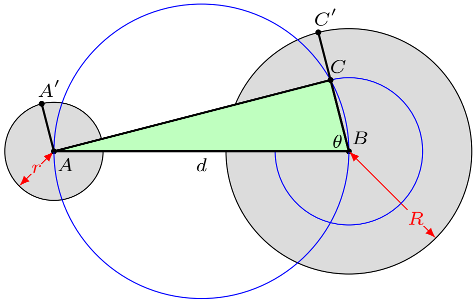
Determine the length of the belt required to go round both pulleys.

5. Extend line BC to C' on the circumference of pulley B. CC' has length r.

4. Consider $\triangle ABC$: $\angle ACB = 90^\circ$ since it is an angle inscribed in a semicircle. Then:

$$AC = \sqrt{d^2 - (R-r)^2}$$

$$\theta = \sin^{-1}\left(\frac{\sqrt{d^2 - (R-r)^2}}{d}\right)$$



Two pulleys, centred at A and B, have radii r and R. The distance from A to B is d.

Determine the length of the belt required to go round both pulleys.

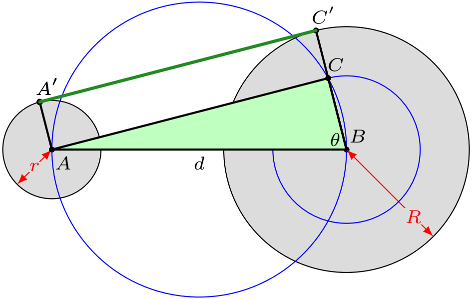
4. Consider $\triangle ABC$: $\angle ACB = 90^\circ$ since it is an angle inscribed in a semicircle. Then:

$$AC = \sqrt{d^2 - (R-r)^2}$$

$$\theta = \sin^{-1}\left(\frac{\sqrt{d^2-(R-r)^2}}{d}\right)$$

5. Extend line BC to C' on the circumference of pulley B. CC' has length r.

6. Draw AA', of length r and parallel to CC', as shown.



Two pulleys, centred at A and B, have radii r and R. The distance from A to B is d.

Determine the length of the belt required to go round both pulleys.

5 Extend line BC to C' on the circumference of pulley B. CC' has length r.

6 Draw AA', of length r and parallel to CC', as shown.

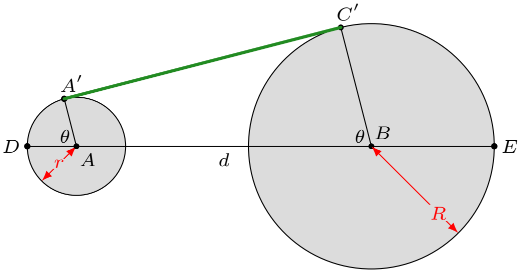
7 Draw $A'C'$: $A'C'CA$ is a rectangle so

$$A'C' = AC = \sqrt{d^2 - (R - r)^2}$$

4 Consider $\triangle ABC$: $\angle ACB = 90^\circ$ since it is an angle inscribed in a semicircle. Then:

$$AC = \sqrt{d^2 - (R - r)^2}$$

$$\theta = \sin^{-1}\left(\frac{\sqrt{d^2 - (R - r)^2}}{d}\right)$$



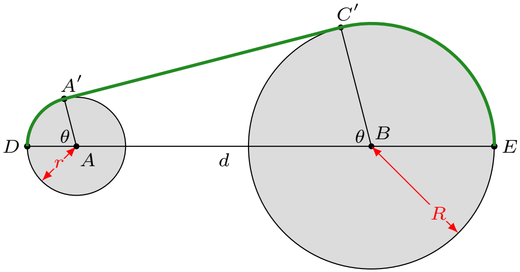
Two pulleys, centred at A and B, have radii r and R. The distance from A to B is d.

Determine the length of the belt required to go round both pulleys.

5. Extend line BC to C' on the circumference of pulley B. CC' has length r.
6. Draw AA', of length r and parallel to CC', as shown.
7. Draw $A'C'$: $A'C'CA$ is a rectangle so

$$A'C' = AC = \sqrt{d^2 - (R - r)^2}$$

8. $A'C'$ is the (top) part of the belt that is tangential to the pulleys at A' and C'. We now need to find the arc-lengths from D to A' and from C' to E.



Two pulleys, centred at A and B, have radii r and R. The distance from A to B is d.

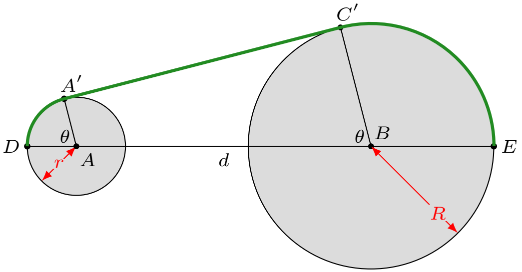
Determine the length of the belt required to go round both pulleys.

5. Extend line BC to C' on the circumference of pulley B. CC' has length r.
6. Draw AA', of length r and parallel to CC', as shown.
7. Draw $A'C'$: $A'C'CA$ is a rectangle so

$$A'C' = AC = \sqrt{d^2 - (R-r)^2}$$

8. $A'C'$ is the (top) part of the belt that is tangential to the pulleys at A' and C'. We now need to find the arc-lengths from D to A' and from C' to E.
9. The angles (θ and $\pi - \theta$) that these arcs subtend at the pulley centres, with the radius of each pulley, are used to determine the arc-lengths (θ in radians):

$$DA' = r\theta \text{ and } C'D = R(\pi - \theta)$$



Two pulleys, centred at A and B, have radii r and R. The distance from A to B is d.

Determine the length of the belt required to go round both pulleys.

10 Belt-length:

$$= 2\left(DA' + A'C' + C'E\right)$$
$$= 2\left(r\theta + \sqrt{d^2 - (R-r)^2} + R(\pi - \theta)\right)$$

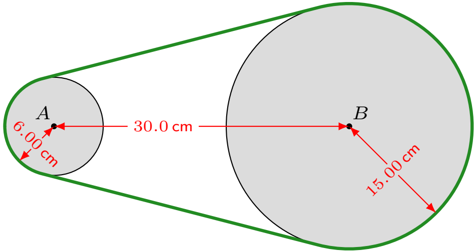
where

$$\theta = \sin^{-1}\left(\frac{\sqrt{d^2-(R-r)^2}}{d}\right)$$

8 $A'C'$ is the (top) part of the belt that is tangential to the pulleys at A' and C'. We now need to find the arc-lengths from D to A' and from C' to E.

9 The angles (θ and $\pi - \theta$) that these arcs subtend at the pulley centres, with the radius of each pulley, are used to determine the arc-lengths (θ in radians):

$$DA' = r\theta \text{ and } C'D = R(\pi - \theta)$$



Two pulleys, centred at A and B, have radii r and R. The distance from A to B is d.

Determine the length of the belt required to go round both pulleys.

Example:

$$\theta = \sin^{-1}\left(\frac{\sqrt{d^2-(R-r)^2}}{d}\right) = \sin^{-1}\left(\frac{\sqrt{6.00^2-1.50^2}}{6.00}\right) = 1.3181 \text{ (radians)}$$

$$\text{B-L} = 2\left(6.00 \times 1.3181 + \sqrt{6.00^2-1.50^2} + 15.00 \times (\pi - 1.3181)\right) = 82.141$$

The belt length is 82.1 cm.

Web :: Dynamic font sizes with CSS

- Responsive websites usually require font sizes that change with device (or browser window) width. A font size of 20px that works well on a large monitor is unlikely to be suitable for a smaller tablet or a phone.

Web :: Dynamic font sizes with CSS

- Responsive websites usually require font sizes that change with device (or browser window) width. A font size of 20px that works well on a large monitor is unlikely to be suitable for a smaller tablet or a phone.
- With media queries, you can set a different font size for each range of device sizes; this is perfectly adequate in many cases. It does have the disadvantage that when resizing a browser, the user will see the font size jumping from, for example, 18px to 16px to 14px. (But it's usually only designers who spend too much time resizing windows.)

Web :: Dynamic font sizes with CSS

- Responsive websites usually require font sizes that change with device (or browser window) width. A font size of 20px that works well on a large monitor is unlikely to be suitable for a smaller tablet or a phone.
- With media queries, you can set a different font size for each range of device sizes; this is perfectly adequate in many cases. It does have the disadvantage that when resizing a browser, the user will see the font size jumping from, for example, 18px to 16px to 14px. (But it's usually only designers who spend too much time resizing windows.)
- For a more fluid result, viewport units are useful (where `1vw = 1/100` of the window width). So, for example, if you set your css to `font-size:2vw;` and your phone is 400px wide, the font will be 8px. If your window width is 1200px, font size will be 24px.

Web :: Dynamic font sizes with CSS

- Responsive websites usually require font sizes that change with device (or browser window) width. A font size of 20px that works well on a large monitor is unlikely to be suitable for a smaller tablet or a phone.
- With media queries, you can set a different font size for each range of device sizes; this is perfectly adequate in many cases. It does have the disadvantage that when resizing a browser, the user will see the font size jumping from, for example, 18px to 16px to 14px. (But it's usually only designers who spend too much time resizing windows.)
- For a more fluid result, viewport units are useful (where `1vw = 1/100` of the window width). So, for example, if you set your css to `font-size:2vw;` and your phone is 400px wide, the font will be 8px. If your window width is 1200px, font size will be 24px.
- Using viewport units alone may tend to make the fonts too small on small screens and too large on large screens; one can introduce some fixed sizes as well: `font-size:calc(9px + 1vw);` sets font size at 13px for phone width of 400px and font size of 21px for window size of 1200px.

  Still not exactly what you want? Maybe `font-size:calc(4px + 1.5vw);`?

Web :: Dynamic font sizes with CSS

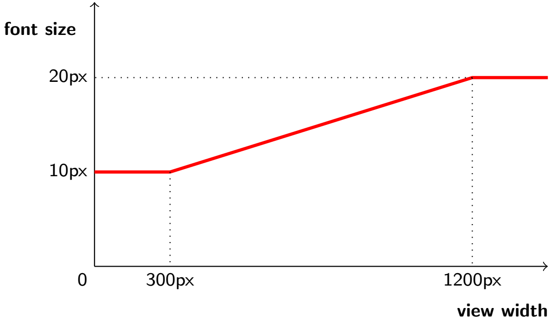
- Responsive websites usually require font sizes that change with device (or browser window) width. A font size of 20px that works well on a large monitor is unlikely to be suitable for a smaller tablet or a phone.
- With media queries, you can set a different font size for each range of device sizes; this is perfectly adequate in many cases. It does have the disadvantage that when resizing a browser, the user will see the font size jumping from, for example, 18px to 16px to 14px. (But it's usually only designers who spend too much time resizing windows.)
- For a more fluid result, viewport units are useful (where `1vw = 1/100` of the window width). So, for example, if you set your css to `font-size:2vw;` and your phone is 400px wide, the font will be 8px. If your window width is 1200px, font size will be 24px.
- Using viewport units alone may tend to make the fonts too small on small screens and too large on large screens; one can introduce some fixed sizes as well: `font-size:calc(9px + 1vw);` sets font size at 13px for phone width of 400px and font size of 21px for window size of 1200px.

  Still not exactly what you want? Maybe `font-size:calc(4px + 1.5vw);`?
- For more precise control, without trial and error that rapidly becomes frustrating, I turned to this excellent CSS-tricks page showing examples such as

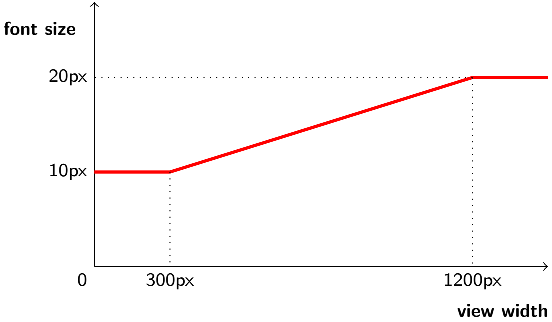
  ```
  font-size:  calc(16px + 6 * ((100vw - 320px) / 680));
  ```

  What are these magic numbers? At the moment, I can tell that, at a screen width of 320px, the font size is 16px. Font size increases smoothly until, at a screen width of 1000px, the font size of 22px. But I probably won't remember how to figure that out next week! The CSS-tricks page doesn't show how these numbers are derived...

  ...but it's just some (relatively) simple high-school math. All you need to recall from high-school is the equation of a line in the form: $\dfrac{y - y_1}{x - x_1} = \dfrac{y_2 - y_1}{x_2 - x_1}$

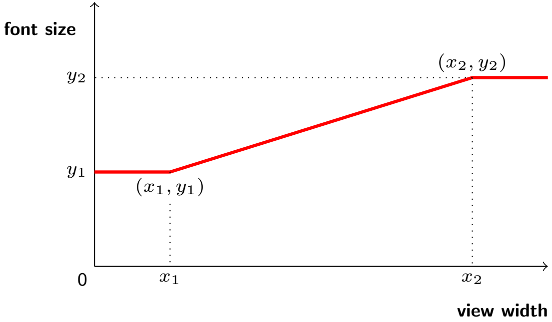


1. We can graph our desired font size against view width as shown. In this case: view widths less than 300px have a font size of 10px; font sizes grow uniformly from 10px at 300px view width to a font-size 20px at 1200px window width; and, for window sizes over 1200px, the font size remains 20px. How do we achieve this with CSS?

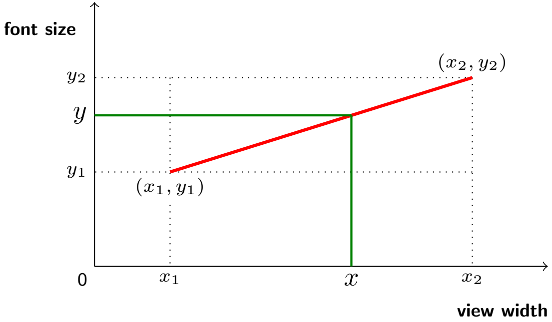


1. We can graph our desired font size against view width as shown. In this case: view widths less than 300px have a font size of 10px; font sizes grow uniformly from 10px at 300px view width to a font-size 20px at 1200px window width; and, for window sizes over 1200px, the font size remains 20px. How do we achieve this with CSS?
2. The constant font sizes below 300px and above 1200px can be easily handled with media queries; we'll come back to them later. What is more interesting is the uniformly increasing font size calculation between view widths of 300px and 1200px.

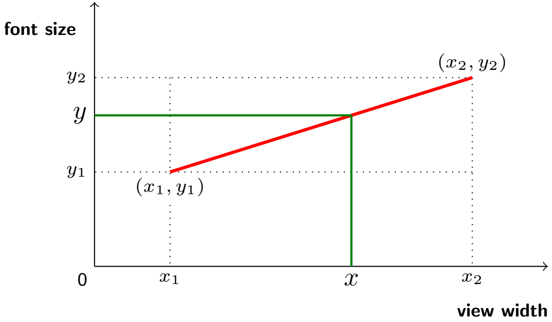
Web :: Dynamic font sizes with CSS



1. We can graph our desired font size against view width as shown. In this case: view widths less than 300px have a font size of 10px; font sizes grow uniformly from 10px at 300px view width to a font-size 20px at 1200px window width; and, for window sizes over 1200px, the font size remains 20px. How do we achieve this with CSS?
2. The constant font sizes below 300px and above 1200px can be easily handled with media queries; we'll come back to them later. What is more interesting is the uniformly increasing font size calculation between view widths of 300px and 1200px.
3. Instead of using the fixed numbers, we'll generalise and use variables so we can easily adjust our formula for different required values.



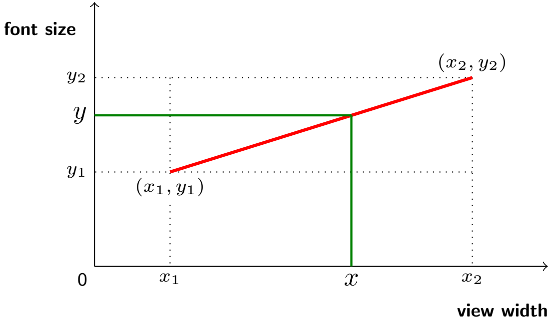
4 For now, just focus on the sloped line as a function from view width x to font size y.



4 For now, just focus on the sloped line as a function from view width x to font size y.

5 The equation of that line is given by:

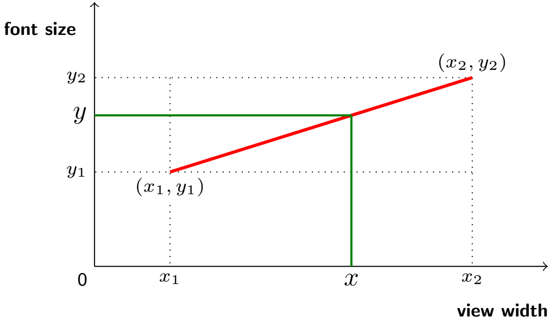
$$\frac{y - y_1}{x - x_1} = \frac{y_2 - y_1}{x_2 - x_1}$$



4 For now, just focus on the sloped line as a function from view width x to font size y.

5 The equation of that line is given by:

$$\frac{y - y_1}{x - x_1} = \frac{y_2 - y_1}{x_2 - x_1}$$
$$\Rightarrow y - y_1 = (x - x_1) \cdot \frac{y_2 - y_1}{x_2 - x_1}$$



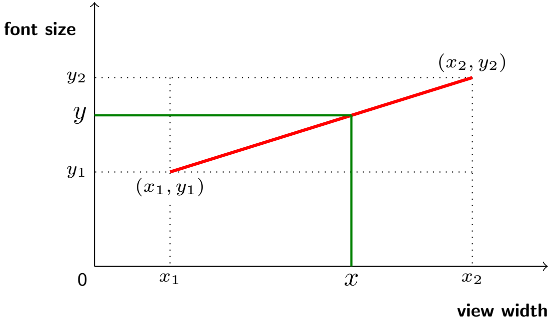
4 For now, just focus on the sloped line as a function from view width x to font size y.

5 The equation of that line is given by:

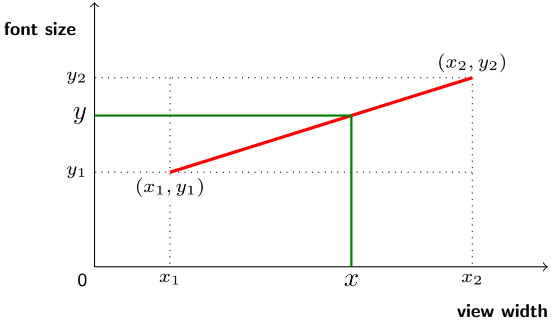
$$\frac{y - y_1}{x - x_1} = \frac{y_2 - y_1}{x_2 - x_1}$$

$$\Rightarrow y - y_1 = (x - x_1) \cdot \frac{y_2 - y_1}{x_2 - x_1}$$

$$\Rightarrow y = y_1 + (x - x_1) \cdot \frac{y_2 - y_1}{x_2 - x_1}$$



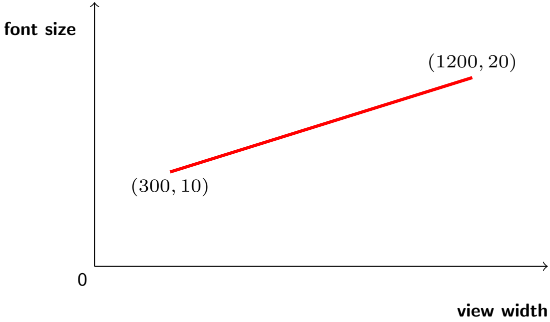
6 We have font size $= y_1 + (x - x_1) \cdot \dfrac{y_2 - y_1}{x_2 - x_1}$ where x_1, y_1, x_2 and y_2 are numbers chosen for our particular design and x is view width. Of course, CSS does not understand x but view width can be represented by 100vw.



6 We have font size $= y_1 + (x - x_1) \cdot \dfrac{y_2 - y_1}{x_2 - x_1}$ where x_1, y_1, x_2 and y_2 are numbers chosen for our particular design and x is view width. Of course, CSS does not understand x but view width can be represented by 100vw.

7 Thus, `font-size:calc(`y_1` + (100vw - `x_1`)*(`$y_2 - y_1$`)/(`$x_2 - x_1$`));`

Web :: Dynamic font sizes with CSS



6 We have font size $= y_1 + (x - x_1) \cdot \dfrac{y_2 - y_1}{x_2 - x_1}$ where x_1, y_1, x_2 and y_2 are numbers chosen for our particular design and x is view width. Of course, CSS does not understand x but view width can be represented by 100vw.

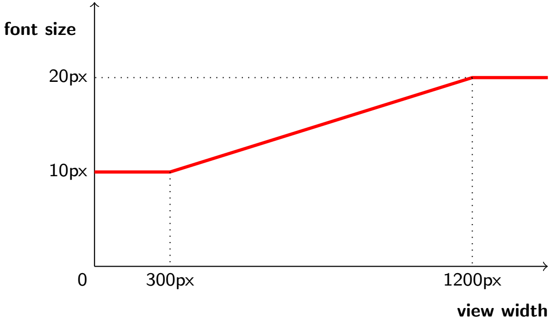
7 Thus, `font-size:calc(`y_1` + (100vw - `x_1`)*(`$y_2 - y_1$`)/(`$x_2 - x_1$`));`

8 From our previous example:

```
font-size:calc(10px + (100vw - 300px)*(20px-10px)/(1200px-300px));
```

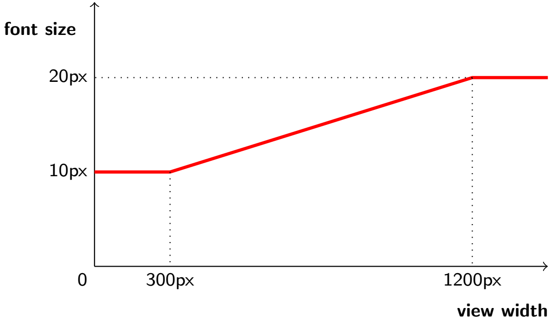
or, more concisely:

```
font-size:  calc(10px + 10 * (100vw - 300px)/900);
```



- CSS for the complete range of view widths:

```
@media screen and (min-width:1200px) {
html { font-size:20px; }
}
@media screen and (max-width:1200px) {
html { font-size:  calc(10px + (100vw - 300px)/90); }
}
@media screen and (max-width:300px) {
html { font-size:10px; }
}
```



10 For ease of editing, using SASS variables for min-font, min-width, max-font, max-width is a better solution. Or write a mixin...