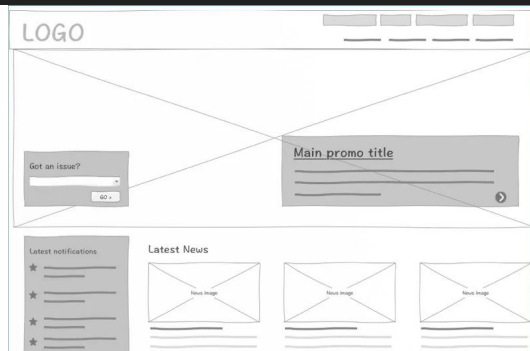
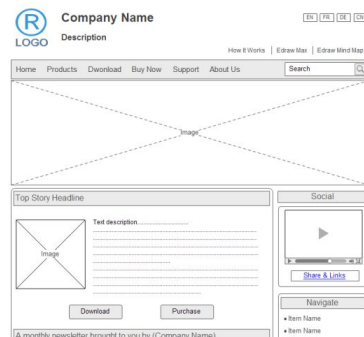
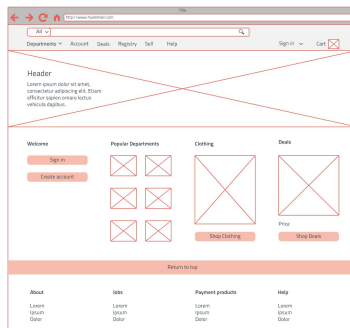
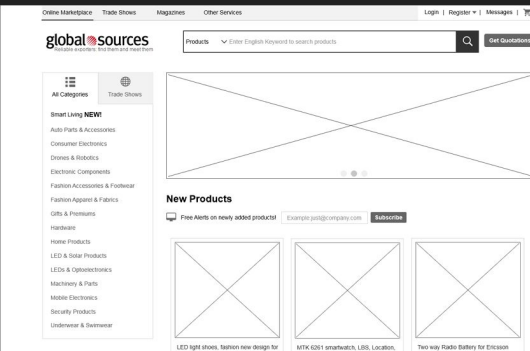


Allt är block!

inline-element, block-element och box-model

och lite om runda hörn...



inline-element och block-element

CSS-egenskapen **display** avgör hur ett element visas upp av webbläsaren.

Vanliga värden:

- block
- inline
- inline-block
- none

https://www.w3schools.com/html/html_blocks.asp

display: block;

- Ett **block** börjar på en ny rad och tar upp hela förälderns bredd.
- **<p>** och **<div>** är exempel på element som har **block** som default
- element med **display: block;** kan man sätta storlek på med CSS.

display: block;

- Ett **block** börjar på en ny rad och tar upp hela sidans bredd.
- **<p>** och **<div>** är exempel på element som har **block** som default
- element med **display: block;** kan man sätta storlek på med CSS.

```
<div class="div1">Text i div 1</div>
```

```
<div class="div2">Text i div 2</div>
```

Text i div 1

Text i div 2

display: inline;

- `inline` tar ingen extra plats. Den bryter inte rad.
- `` är exempel på element som har `inline` som default
- element med `display: inline;` kan man **INTE** sätta storlek på med CSS.

display: inline;

- `inline` tar ingen extra plats.
- `` är exempel på element som har `inline` som default
- element med `display: inline;` kan man INTE sätta storlek på med CSS.

Kolla

```
<span class="blue">Här är text</span>
```

```
<span class="green"> med olika färg.</span>
```

Kolla Här är text med olika färg.

display: inline-block;

- inline-block visas inline.
- ``, `<button>` och `<input>` är exempel på element som har inline-block som default.
- element med `display: inline-block;` kan man sätta storlek på med CSS.



`display: none;`

`display: none;` döljer ett element så att det inte tar någon plats i sidan.

att jämföra med

`visibility: hidden;` döljer ett element men det behåller sin plats i sidan.

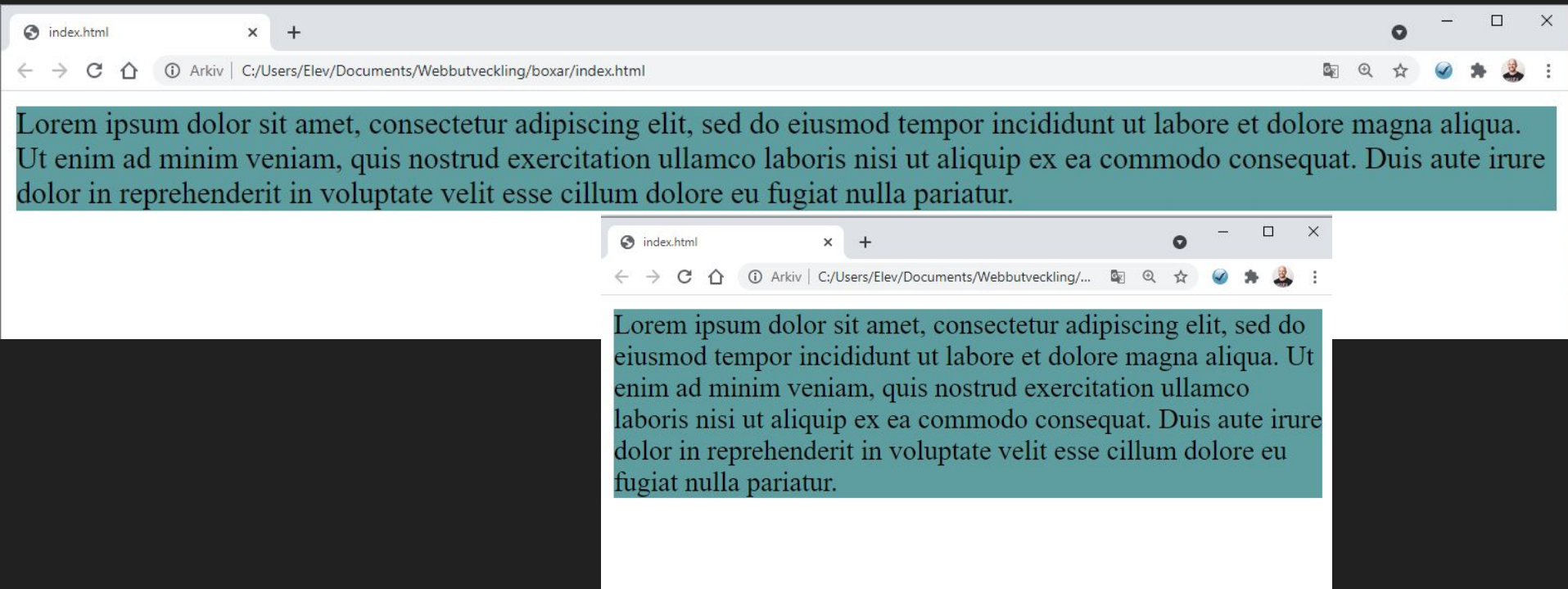
Jag visar...

Hur får ett block sin storlek?

- Webbläsaren räknar ut det för att allt innehåll ska få plats.
- Vill man själv bestämma storlek måste man bestämma hur webbläsaren ska hantera det som inte får plats.

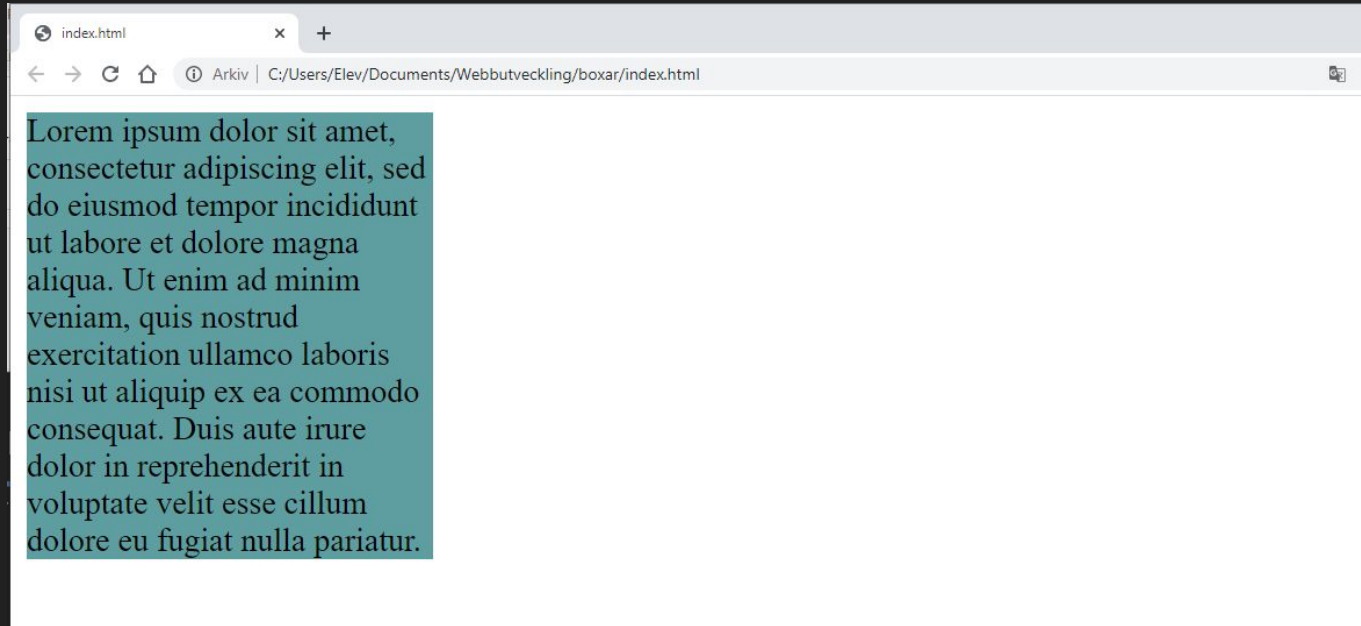
Hur får ett block sin storlek?

En standard-`<div>` tar hela förälders bredd. Blir så hög att all text får plats.



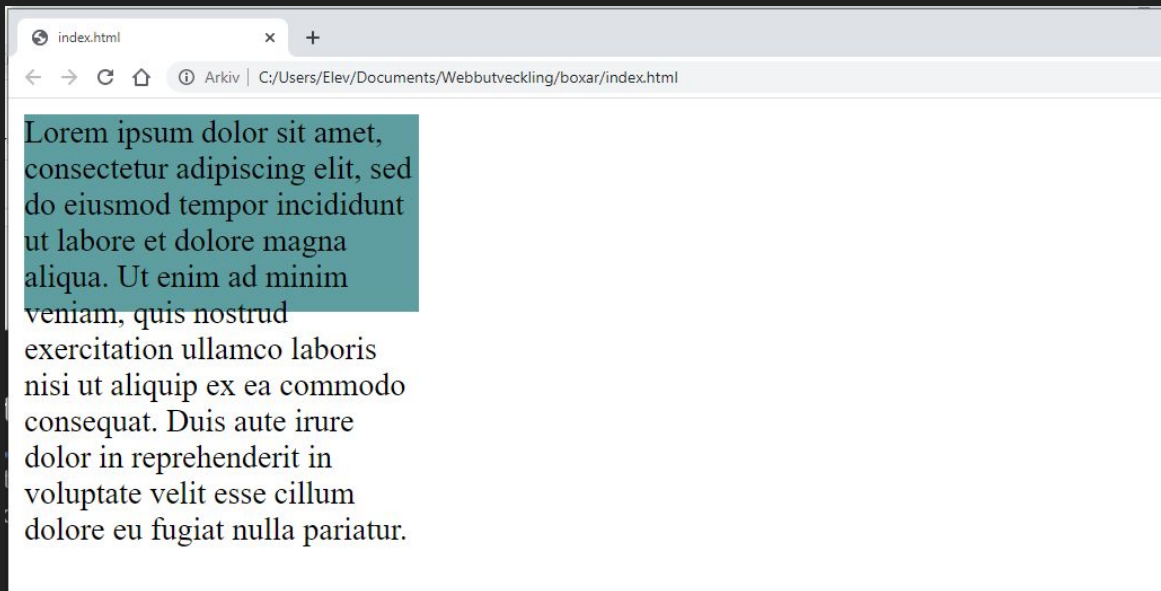
Hur får ett block sin storlek?

Om man sätter bredd (**width: 200px;**) på sin `<div>` beräknar webbläsaren höjden.



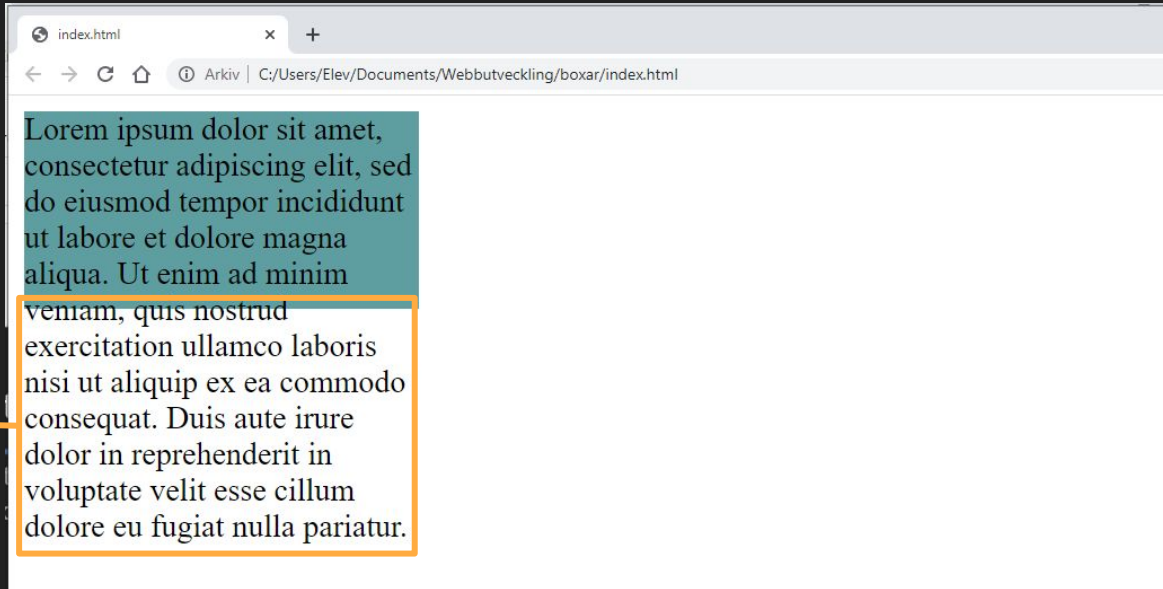
Hur får ett block sin storlek?

Om man sätter bredd (**width: 200px;**) och höjd (**height: 100px;**) på sin <div>. Vad händer då?



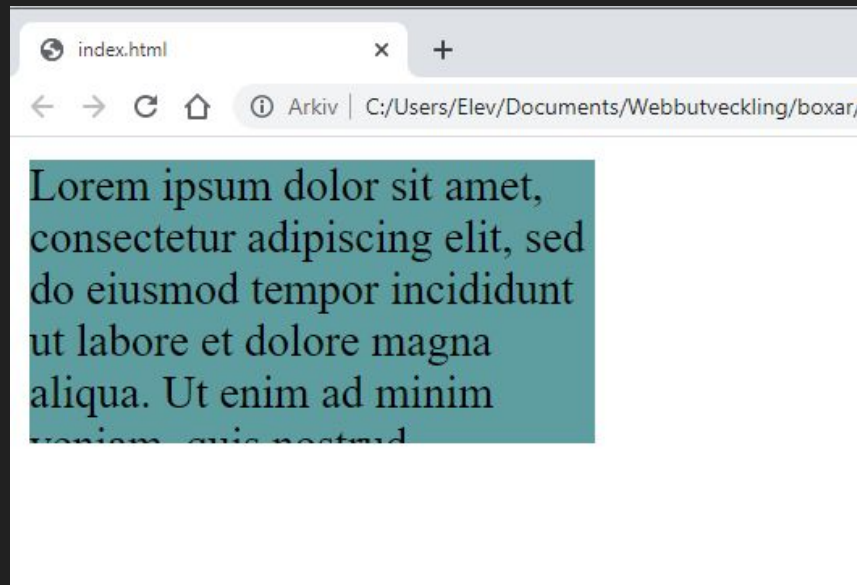
Det som inte får plats är overflow

overflow



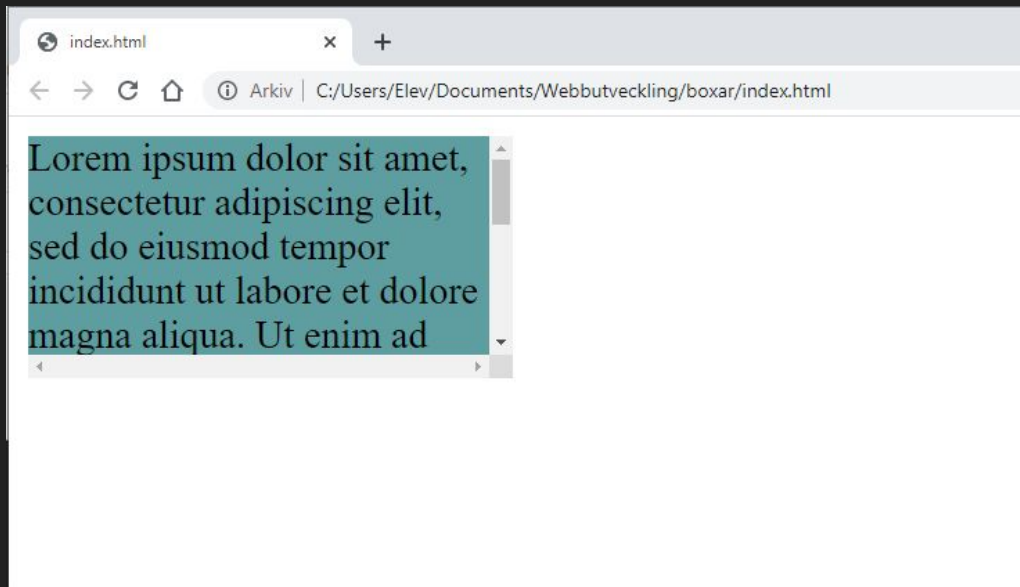
overflow: hidden

```
.div-cadet {  
  background-color: cadetblue;  
  width: 200px;  
  height: 100px;  
  overflow: hidden;  
}
```



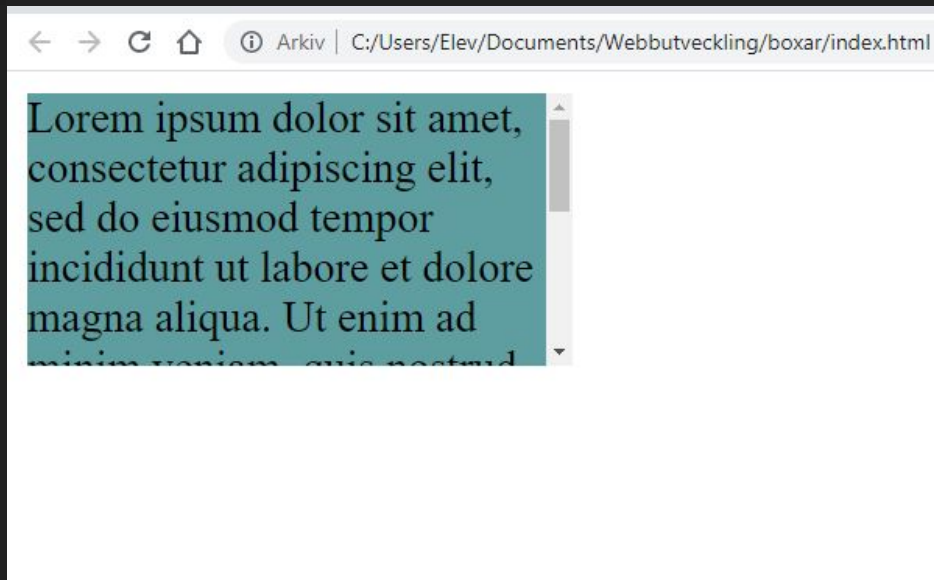
overflow: scroll

```
.div-cadet {  
    background-color: cadetblue;  
    width: 200px;  
    height: 100px;  
    overflow: scroll;  
}
```



overflow: auto

```
.div-cadet {  
  background-color: cadetblue;  
  width: 200px;  
  height: 100px;  
  overflow: auto;  
}
```



Övningar - del 1

Skapa en ny mapp med ny index.html och style.css.

Skapa tre div:ar. Fyll dem med [Lorem ipsum](#)-text. Lämna en med default styling, gör en 200px bred, gör den tredje 200px bred och 200px hög. Se till att det endast går att scrolla i texten vertikalt i den tredje div:en.

Skapa en `` runt ett ord i din text. Se till att det 40px margin på båda sidor om ordet (`margin: 40px;`). Tips: du behöver ändra display-egenskapen för att kunna göra det.

Lägg in en bild överst i din sida. Styla den med `display:none` och ändra sedan till `visibility:hidden`. Vad blir skillnaden mellan de olika stilreglerna?

Hur brett blir ett element (totalt) med följande stilregler? Fundera och räkna.

```
width: 100px;  
padding-left: 10px;  
margin-right: 5px;  
border-left: solid black 5px;
```

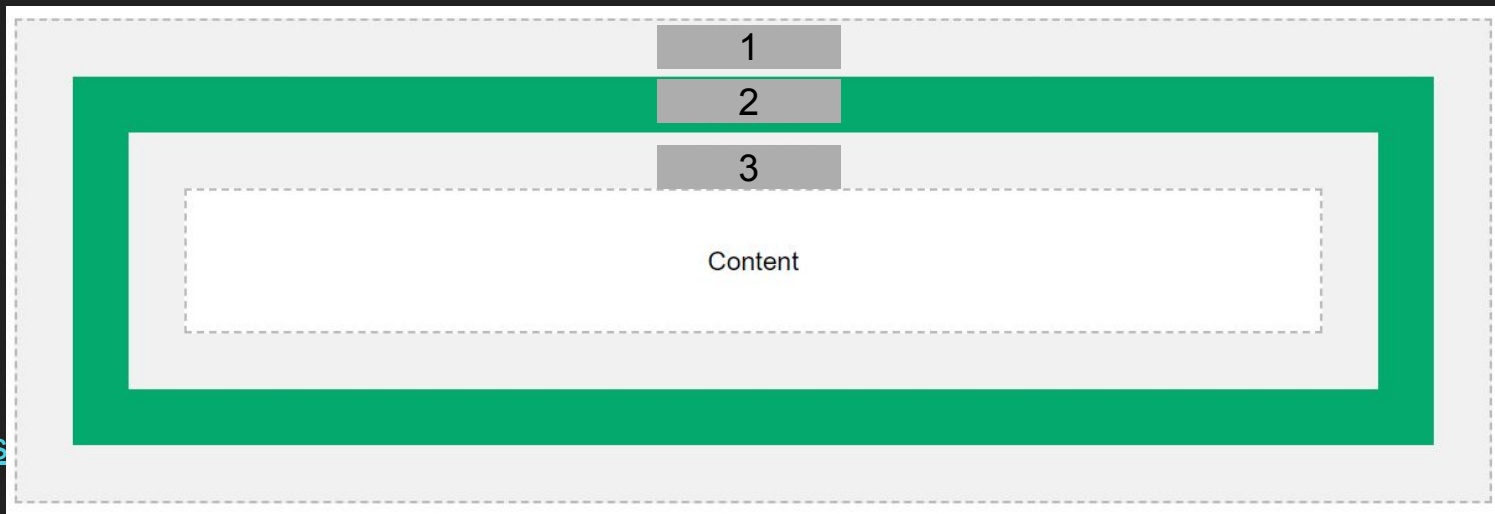
Box model

Box model

Alla element har egenskaperna `margin`, `padding` och `border`.

`margin` och `padding` är transparenta “marginaler”.

`border` är synlig men default är att det inte finns någon



Box model - mellanrum

Mellanrum mellan element skapas ofta genom att ge elementen `padding` eller `margin`.

`padding` och `margin` kan styras och vara olika på olika sidor av elementet.

Box model - mellanrum

Mellanrum mellan element skapas ofta genom att ge elementen `padding` eller `margin`.

`padding` och `margin` kan styras och vara olika på olika sidor av elementet.

```
margin: 10px; /*Ger 10 pixlars margin runt hela elementet*/
```

```
margin-top: 10px;
```

```
margin-bottom: 5px;
```

```
margin-left: 20px;
```

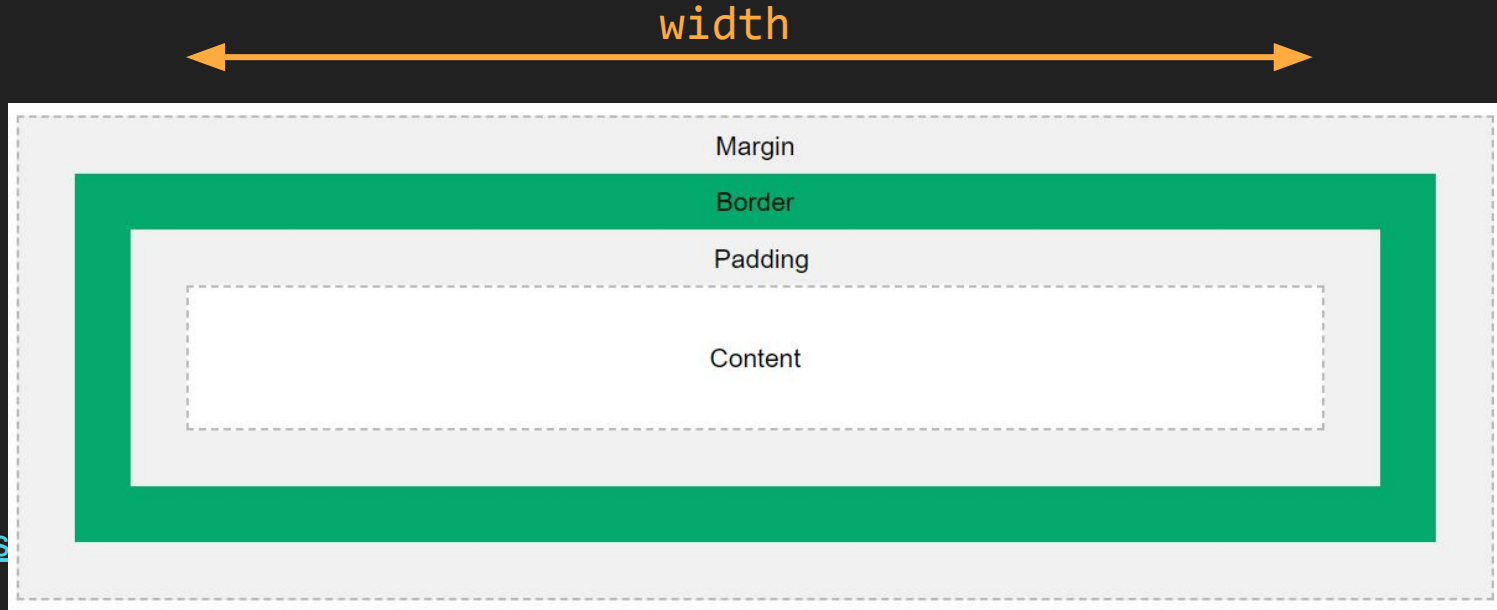
```
margin-right: 0px;
```

https://www.w3schools.com/css/css_boxmodel.asp

Box model - storlek

`width: 200px;` sätter storleken på Content i modellen.

margin, border och padding tillkommer till elementets totala bredd.



Demo

Men cirklar då?

Cirklar är kvadratiska block med rundade hörn

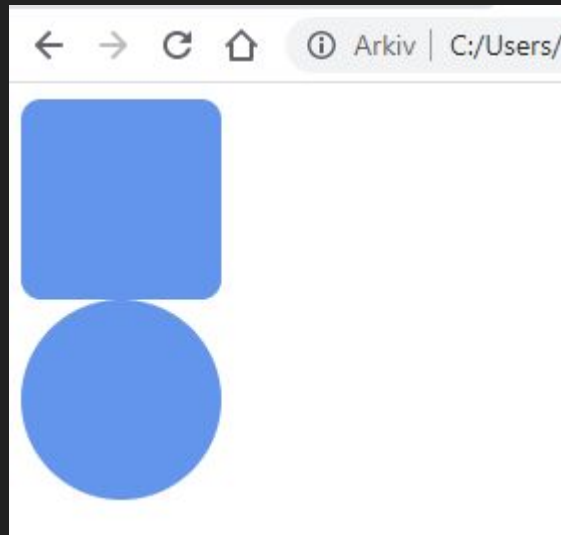
HTML

```
...<body>  
<div></div>  
<div class="round"></div>  
</body>...
```

CSS

```
div {  
    width: 100px;  
    height: 100px;  
    border-radius: 10px;  
    background-color:  
    cornflowerblue;  
}  
  
div.round {  
    border-radius: 50%;  
}
```

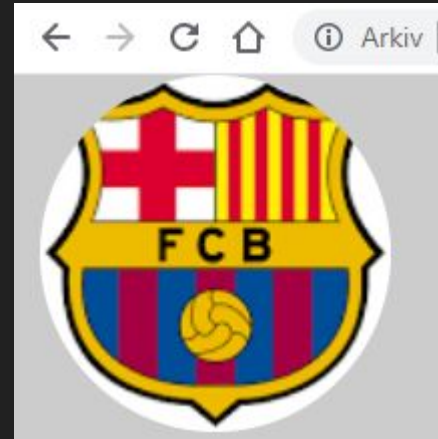
Resultat



https://www.w3schools.com/css/css_border_rounded.asp

Även bilder kan göras runda

```
img {  
  width: 100px;  
  border-radius: 50%;  
}
```

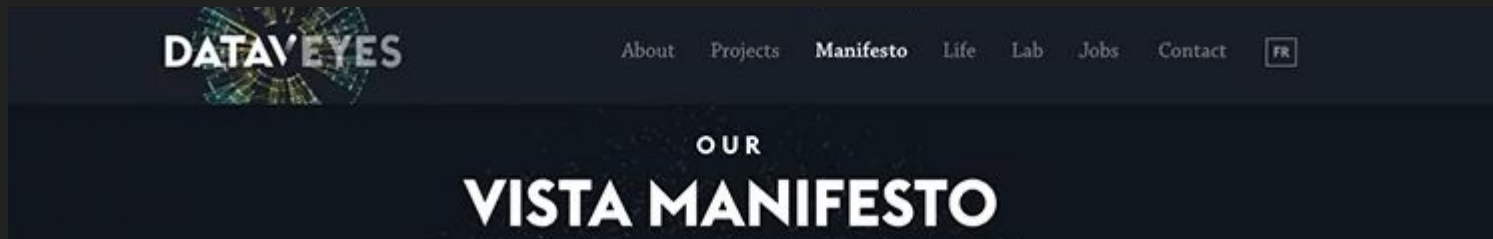


https://www.w3schools.com/css/css_border_rounded.asp

Övningar del 2

Skapa en horisontell lista med länkar. Skapa mellanrum mellan dem med padding eller margin. Tänk att det är en topp-meny på en sida. Använd google för att hitta lösningar.

Styla dina länkar så att de blir snygga. Testa med ljus bakgrund, mörk bakgrund, olika textfärger, olika fonter, olika text-decoration.



Övningar del 3

Till vänster om din meny i del 2 lägger du in en rund bild. Gör den lagom stor, alltså ungefär lika hög som menyn. Det är lättast att få det snyggt om din bild är kvadratisk från början.