



# HTML ja CSS – Verkkosivun toteutus

CS-C1180 Verkkojulkaisemisen perusteet  
21.1.2020 ja 28.1.2020  
Pekka Pulli

# HTML

Rakenne

# CSS

Tyyli

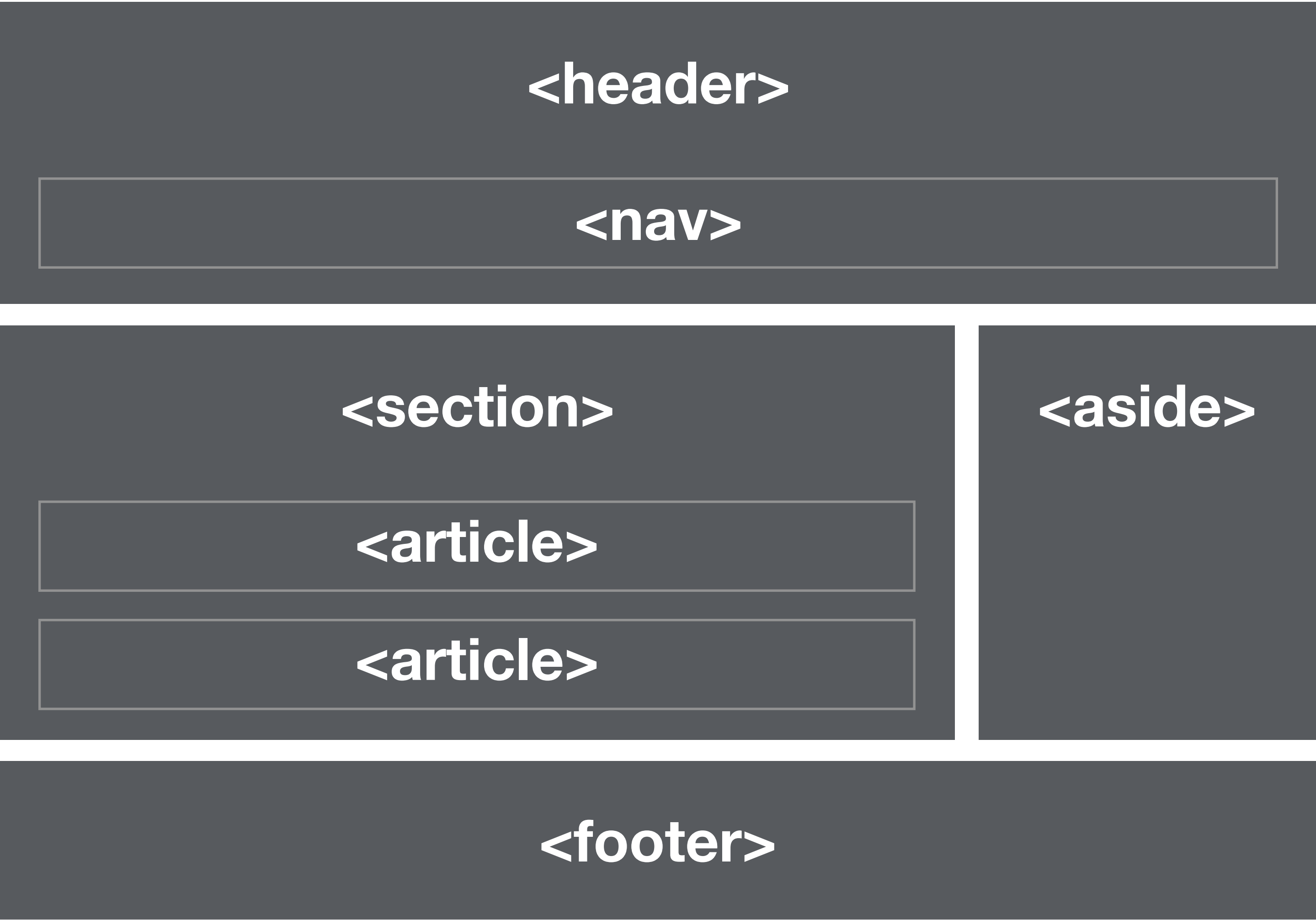
# Javascript

Muuttuva  
sisältö

# HTML

# HTML eli HyperText Mark-up Language

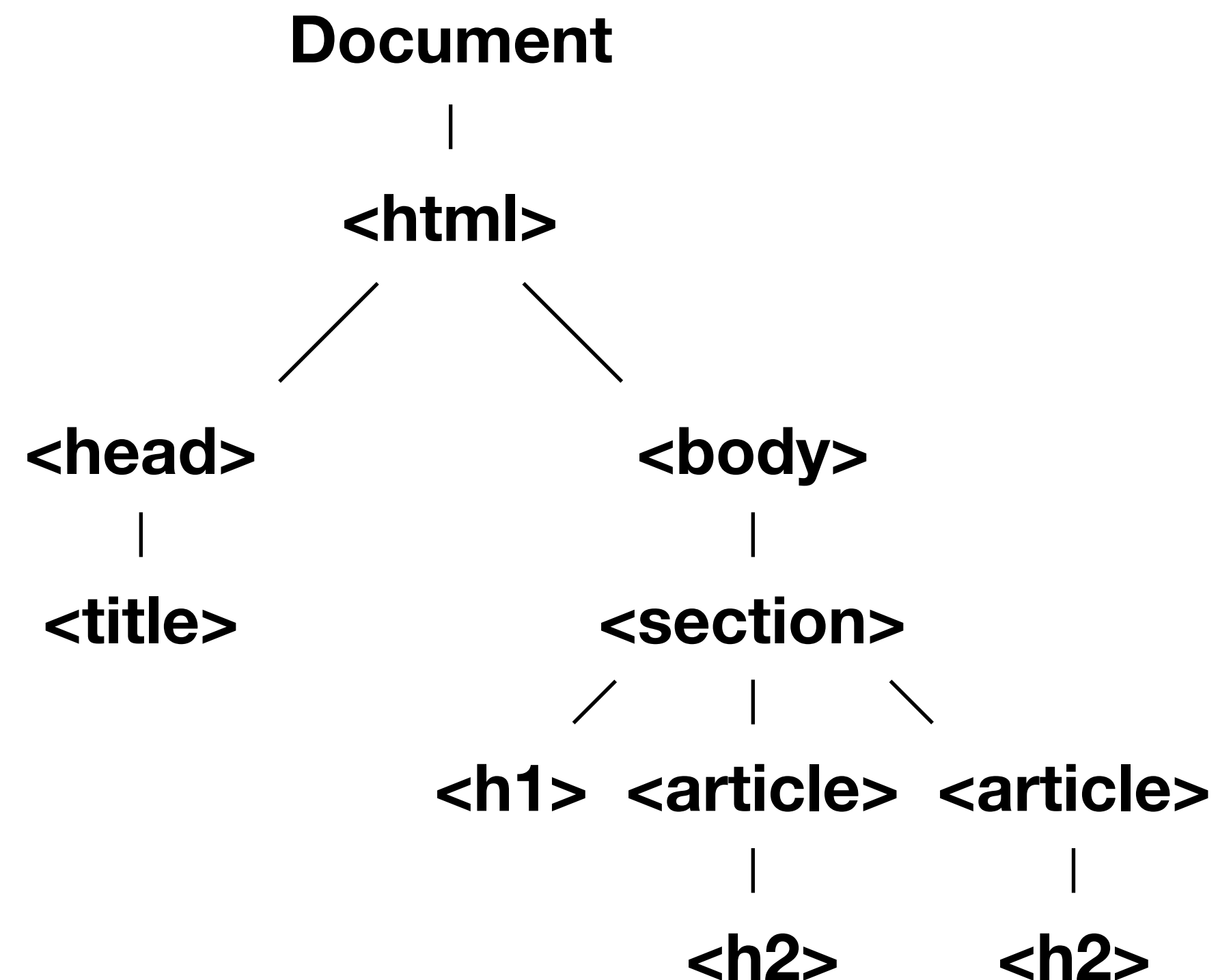
- Verkkosivun rakenteen kuvauskieli
- Tällä hetkellä käytetään versiota HTML5
- Muoto: `<tag attribute="value">content</tag>`
- Sisäkkäiset tagit ja tekstisisältö niiden sisällä muodostavat HTML-sivun rakenteen.



# Katsotaanpas aalto.fi:tä!



# DOM-puu (Document Object Model)



```
<html>
  <head>
    <title>Portfolio</title>
  </head>
  <body>
    <header>Portfolio</header>
    <section>
      <h1>...</h1>
      <article>
        <h2>...</h2>
      </article>
    </section>
  </body>
</html>
```

# Harjoitus

- Katso oman portfoliosivustosi suunnitelmaa. Kirjoita auki sen rakenne ja nimeä osiot parhaasi mukaan (esim. “otsikko”, “navigaatio” jne.)
- Ei tarvitse käyttää oikeita HTML-tageja, mutta voit yrittää sisentää alielementit niin, että muodostat jonkinlaisen rakenteen



# <html>

- Sivulle pitää määrittää dokumenttityyppi. HTML5-dokumenteissa se on aina sivun alussa oleva `<!doctype html>`.
- HTML4-sivuilla `<!DOCTYPE HTML PUBLIC “-//W3C//DTD HTML 4.10//EN” http://www.w3c.org/TR/html4/strict.dtd>`
- Dokumenttityypin jälkeen kaikki muu sisältö menee `<html>`-tagin sisään

# <head>

- Sivuston metatiedot, eli esimerkiksi

```
<title>Sivuston otsikko</title>
```

```
<meta charset="UTF-8">
```

```
<meta name="description" content="Sivuston kuvaus">
```

```
<meta name="keywords" content="Sivuston avainsanat  
pilkulla eroteltuna">
```

```
<meta name="author" content="Nimi Nminen">
```

```
<meta name="viewport" content="width=device-width,  
initial-scale=1.0">
```

# <body>

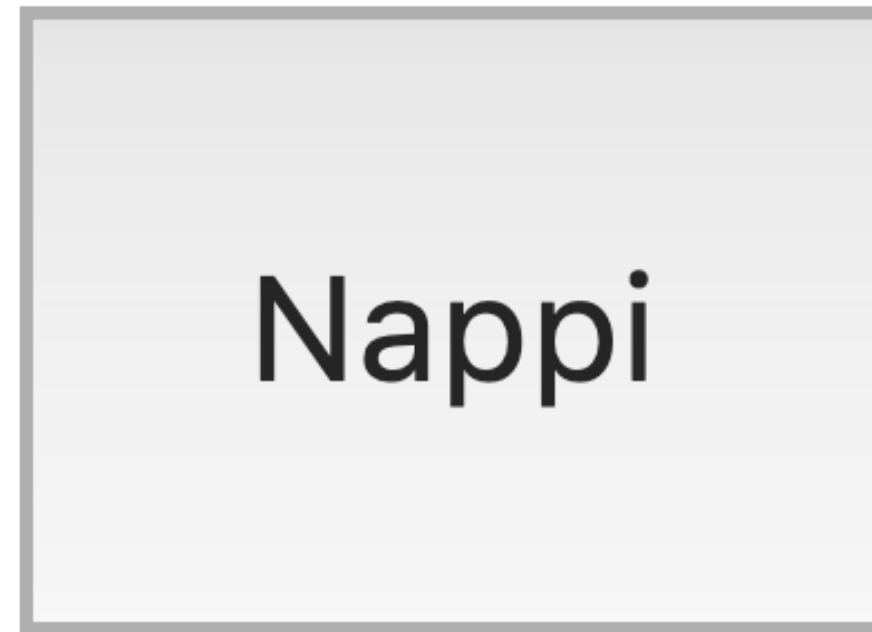
- Kaikki sivulla näkyvä on <body>-tagin sisällä – Täyden listan löydät täältä: <https://www.w3schools.com/tags/>
- HTML4:
  - <div>-tagi (division) määrittää alueita, <span>-tagilla voi ryhmitellä vaikka tekstin sisällä olevia elementtejä
  - Muita tärkeitä tageja muun muassa <h1>-<h6> (heading), <p> (paragraph), <a> (hyperlinkki), <img> (image), <input> (kaavakkeen kenttä) ja <button> (nappi)
- HTML5:
  - Semanttinen nimeäminen käyttötarkoituksen mukaan, muun muassa <header>, <nav>, <section>, <article> ja <footer>

# HTML5-tagit

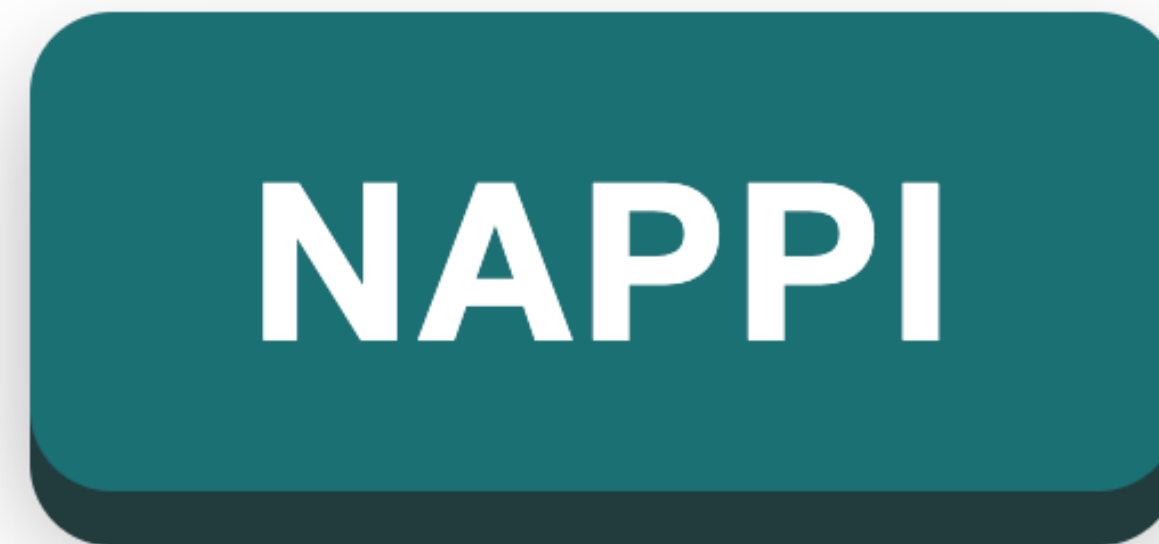
- `<header>`: Sivuston otsikko-osio
- `<nav>`: Navigaatio
- `<section>`: Rakenteellinen osio, esim. sivun pääsisältö
- `<article>`: Toistuvan sisällön osa, yksittäinen artikkeli esim. blogissa
- `<aside>`: Lisäsisältö (esim. oikeassa laidassa)
- `<footer>`: Sivuston alaosa, jossa yleensä majailevat esim. yhteystiedot

# CSS

```
<button class="my-button">Nappi</button>
```



```
.my-button {  
}
```



```
.my-button {  
  padding: 8px 16px;  
  background-color: #1C6F74;  
  border: none;  
  box-shadow: 0 4px #223B3D, 0 4px 8px rgba(0,0,0,0.3);  
  border-radius: 6px;  
  color: white;  
  text-transform: uppercase;  
  font-weight: bold;  
  font-size: 1rem;  
  font-family: 'Helvetica Neue', Helvetica, Arial, sans-serif;  
  letter-spacing: 0.04em;  
}
```

# CSS eli Cascading Style Sheets

- Verkkosivun tyylit määritellään CSS:llä. Nykyinen jo melko laajasti toimiva versio on CSS3.
- CSS:ää käytetään sekä koko sivuston asetteluun että yksittäisten elementtien tyylittelyyn.
- Kolme tapaa määritellä CSS-tyylejä:
  - Ulkoinen .css-tiedosto: `<link rel="stylesheet" type="text/css" href="style.css"/>`
  - Sisäinen tyylimäärittely: `<style type="text/css">...</style>`
  - Elementin tyylimäärittely: `<div style="...">...</div>`

# CSS-tuki vaihtelee selaimittain

- Vaikka useimmat selaimet tukevat jo hyvin CSS3:a, pieniä eroja tulkinnoissa on.
- <https://necolas.github.io/normalize.css/> on työkalu selainten erojen tasoittamiseen. <http://www.cssreset.com/> sisältää muita hyviä välineitä.
- <https://caniuse.com> pitää kirjaa eri selainversioiden tuesta CSS-ominaisuuksille
- Käytännössä vain Internet Explorer 11 (vuodelta 2013!) aiheuttaa enää isoja ongelmia, ja sitä käyttävät lähinnä julkiset laitokset

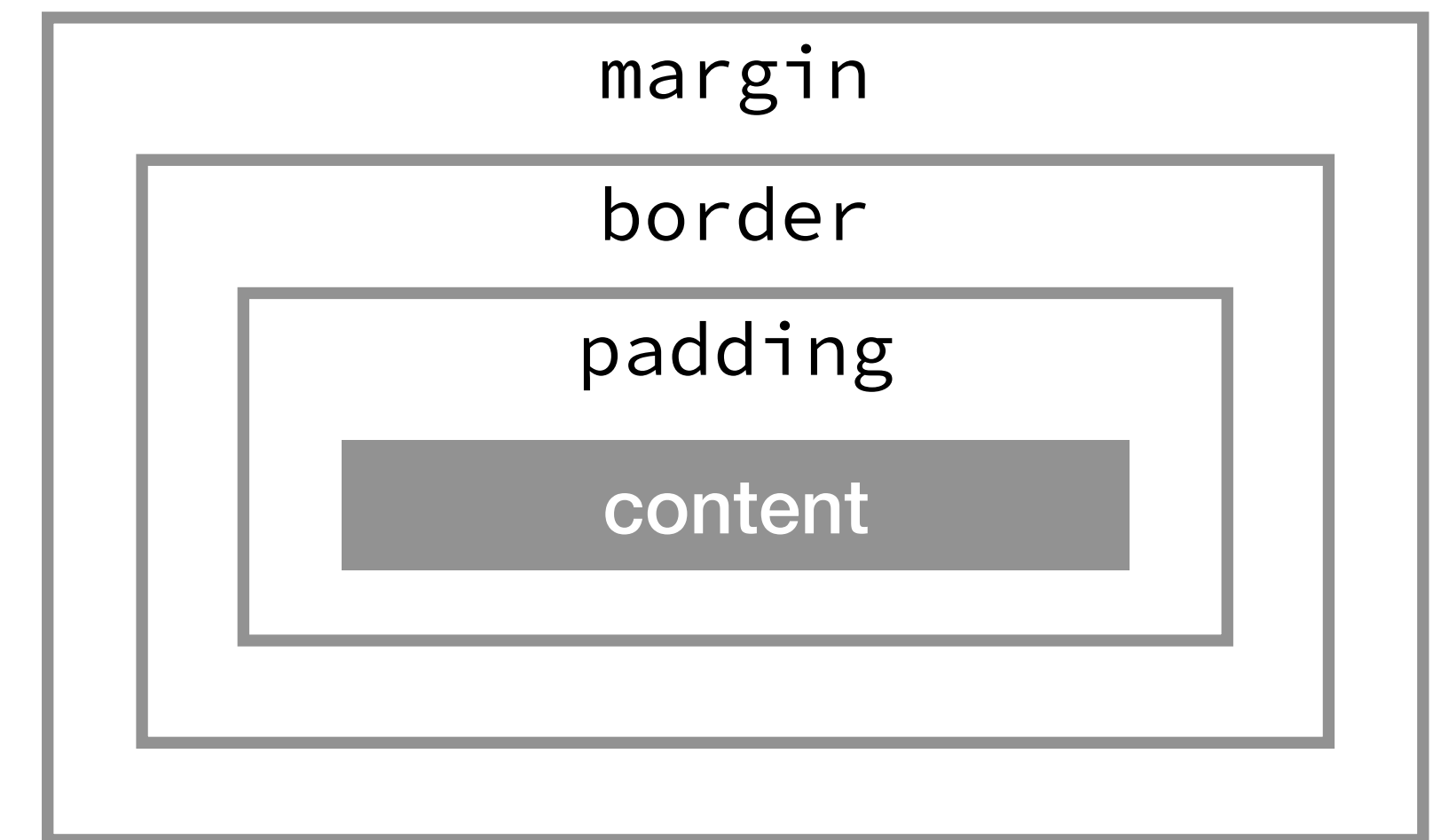


# Elementtien valinta CSS:llä

- CSS:ssä valitaan ensin muotoiltava elementti.
  - Tagi (esim. `<h1>`)
  - Luokka (esim. `.my-button`, `<button class="my-button">`)
  - Uniikki id (esim. `#main-button`, `<button id="main-button">`)
- Tagi ja luokka valitsevat kaikki sopivat elementit, id:tä käytetään harvoin valinnassa koska se pätee vain yhteen elementtiin ja usein halutaan tehdä uudelleenkäytettäviä komponentteja
- Valintapeli! <http://flukeout.github.io/>

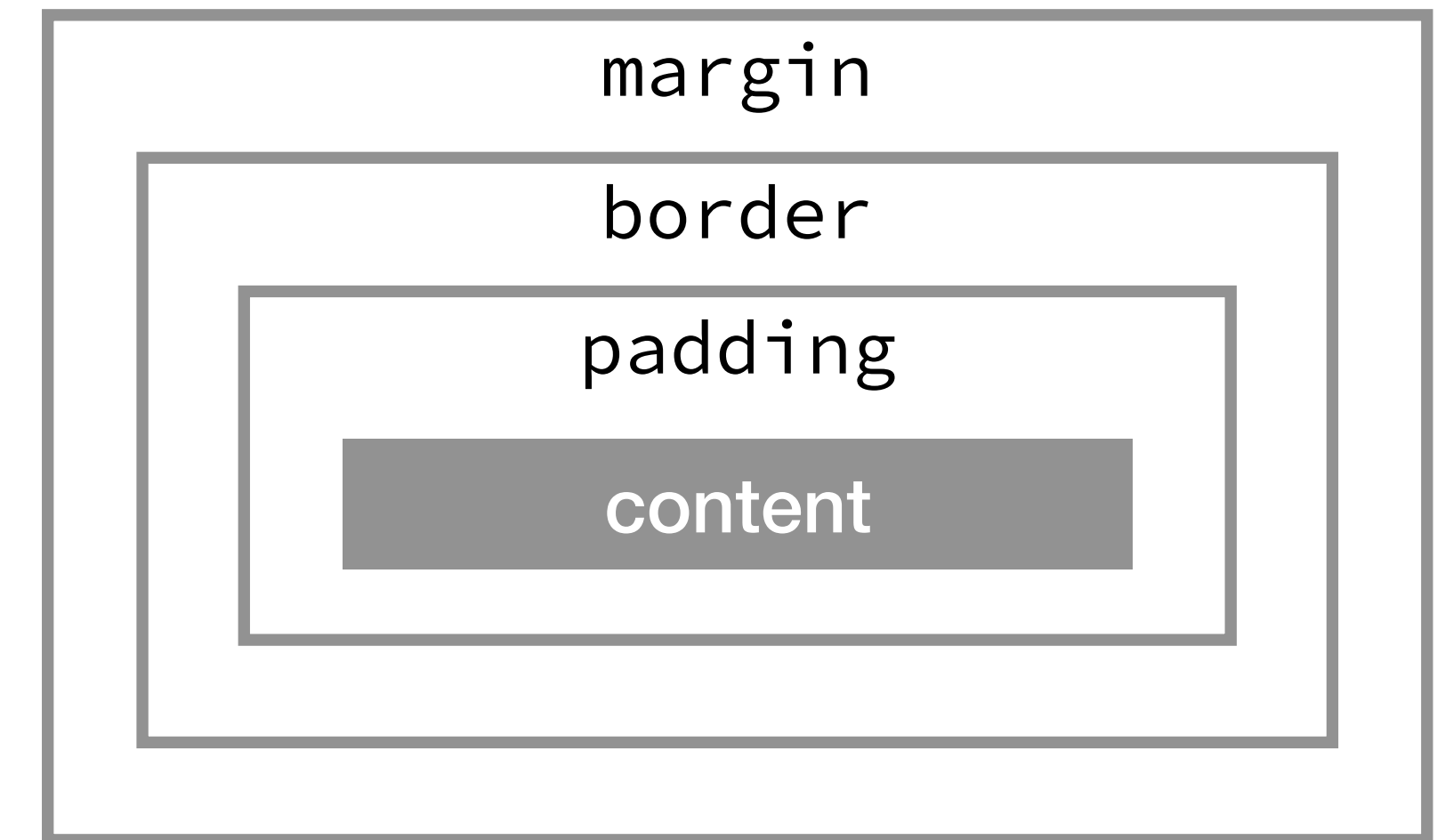
# Elementtien muotoilu

- CSS käyttää elementtien muotoiluun ns. box modelia (oikealla)
- Elementtien koot voidaan määritellä absoluuttisesti (px) tai suhteellisesti (% , rem tai em)
- Prosentit suhteutetaan ympäröivän elementin kokoon (parent), rem-koot suhteutetaan <body>-elementille annetun fontin kokoon ja em-koko ympäröivän elementin fonttikokoon.



# Elementtien muotoilu

- Marginaalit ja paddingit voi asettaa joko erikseen joka suuntaan (esim. `margin-left: 15px;`)  
*tai*  
yhdellä määrittelyksellä (esim. `margin: 10px 15px;`, joka asettaa ylä- ja alamarginaaliksi 10px ja sivuille 15px)
- Border voidaan asettaa muodossa `border: paksuus tyyppi väri;`



# Elementtien muotoilu

- Muita olennaisia elementtien määrittäjiä ovat `background`, `box-shadow` ja `border-radius`
- Värin määrittämiseen usein RGB-heksakoodilla (esim. `#3460a0`), jossa numeroparit viittaavat punaisen, vihreän ja sinisen määrään arvoilla 00-ff. Myös läpinäkyviä värejä voi määrittää `rgba`-koodilla (esim. `rgba(0, 0, 0, 0.5)`), jossa ensimmäiset arvot viittaavat rgb-arvoihin välillä 0-255 ja viimeinen on `opacity` eli peittävyys arvolla 0-1.
- Referenssi: <https://htmlcheatsheet.com/css/>

# Tekstin muotoilu

- Fontti määritellään `font-family`-attribuutilla. Tälle annetaan usein fallback-fontteja ja viimeisenä aina jokin perustyyppi kuten `serif` tai `sans-serif`, esim. `font-family: 'Helvetica Neue' Arial sans-serif;`
- Fontin asetuksen lisäksi voi käyttää esim. näitä: `color`, `font-size`, `font-weight`, `text-decoration`, `text-transform`, `line-height`, `letter-spacing`
- Omia fontteja voi tuoda projektiin `@font-face`-attribuutilla, fonttipalveluista fontit tuodaan `<head>`-elementin sisään esimerkiksi näin: `<link href="https://fonts.googleapis.com/css?family=Roboto&display=swap" rel="stylesheet">`

# Komponenttikirjasto CodeSandboxilla

- Pohja portfoliosivun komponenttikirjastolle: <https://codesandbox.io/s/css-komponenttikirjasto-6ebeo>
- Luo oma versio forkkamalla esimerkkipohja itsellesi. Komponentit tulevat lopulta osaksi portfoliosivustoasi, mutta CodeSandbox on nopea tapa muotoilla komponentteja.
- Referenssi, josta löytyy useimmat muotoiltavat aspektit: <https://htmlcheatsheet.com/css/>

# Ensi kerralla (28.1.)

- Ensi kerralla harjoitellaan elementtien asemointia CSS:n avulla ja tehdään harjoitustyötä.
- Helpottaa omaa työskentelyänne, jos saatte siihen mennessä sivunne rakenteen (ja jopa sisällön) valmiiksi.