

Päivä 2: Web-ohjelmointia

MIMMIT KOODAA -BOOTCAMP 2020
JANI JÄRVINEN, INTERTECHNO TRAINING OY



Web-teknologiat

HTML5-teknologiaperhe

- Termillä ”HTML5” viitataan yleisesti joukkoon web-pohjaisia tekniikoita, joita käytetään modernien sovellusten kehittämisessä
- Virallisesti HTML5 on HTML-kuvauskielen uusin versio
 - Katso <https://www.w3.org/TR/html5/>
 - Standardointi-organisaatio on nimeltään World Wide Web Consortium (W3C)
- Yleisesti HTML5-tekniikoihin lasketaan kuuluvaksi mm. seuraavia tekniikoita:
 - CSS3
 - JavaScript-kielen uudet versiot (versio 6 ja uudemmat)
 - Web Sockets (reaaliaikainen web-liikenne, vrt. chat)
 - Canvas2D
 - Web Storage
 - Semanttisuus



Front-end-tekniikat

- Kolmen tekniikan yhdistelmä: HTML5, CSS3 ja JavaScript
- JavaScriptin kirjoittaminen ja osaaminen keskeistä
- Nykyisin ei välttämättä kannata kirjoittaa suoraan JavaScriptiä, vaan turvallisempaa/virheitä vähentäviä kieliä
 - CoffeeScript, Dart, Elm, TypeScript, ...
- Keskeisiä JavaScript-kirjastoja tällä hetkellä
 - Angular, React, Vue
 - jQuery on jäämässä pois
- Käytännössä kaikki front-end-kirjastot tekevät samaa asiaa: muokkaavat HTML-sivun sisältöä eli objektimallia lennossa

Semanttisia elementtejä 1

Elementti	Käyttötarkoitus
<article>	Määrittelee artikkelin (esim. Blogissa)
<aside>	Sivuhuomautus
<bdi>	Bi-Directional Isolation; kirjoitussuunnan muutos vasen–oikea ← → oikea–vasen
<details>	Piilotettavia lisätietoja
<dialog>	Keskusteluikkuna (dialog box)
<figcaption>	Kuvateksti figure-elementillä määritellylle kuvalle
<figure>	Määrittelee kuva-elementin
<footer>	Alatunniste
<header>	Ylätunniste

Semanttisia elementtejä 2

Elementti	Käyttötarkoitus
<main>	Sivun pääsisältö
<mark>	Korostettu teksti
<meter>	Jotakin arvoa mittaava mittari
<nav>	Navigaatio-linkki
<progress>	Progress bar, edistymistä kuvaava osio sivulla
<section>	Määrittelee dokumentin osan
<summary>	Yhteenveto dokumentin sisällöstä
<time>	Määrittelee jonkin ajan tai päivämäärän
<wbr>	Word Break; mahdollinen rivinvaihto

HTML:n dokumenttoliomalli eli DOM

- DOM eli Document Object Model
- Muistinvarainen esitysmuoto HTML-sivun sisällölle
- Kaikki HTML-dokumentin elementit ("tägit") muodostavat yhdessä puurakenteen, jonka sisältöä voidaan muokata JavaScript-koodissa
- Jokainen puussa oleva elementti on olio, jolla on sekä ominaisuuksia (properties) että tapahtumia (events), mutta myös attribuutteja (attributes)
- Puun päivittäminen aiheuttaa automaattisesti ruudun päivittymisen
- Vertaa perinteisten ohjelmointikielten käyttöliittymätekniikat, kuten Javan AWT sekä .NET-ympäristön WinForms ja WPF

CSS-kieli

- Lyhenne sanoista Cascading Style Sheets
- Määrittelee ns. tyylisivut, joiden avulla voidaan muokata HTML-dokumenttien ulkoasua, toimintoja (esim. animaatiot) sekä asettelua (esim. elementtien sijainti)
- Alun perin CSS:n kautta määriteltävät asiat kirjoitettiin suoraan HTML-koodiin, mutta tämä sopii huonosti nykyaikaisiin web-sovelluksiin – sisältö ja ulkoasu on tärkeä erotella
- CSS-kielestä on useita eri versiota, uusin on CSS 3.1
- Hyvin ilmaisuvoimainen kieli, mutta selaintuki vaihtelee
- Hyvä tapa tarkistaa tukea eri ympäristöissä
 - <https://caniuse.com/>

Selainten F12-työkalut

- Kaikissa nykyisissä työpöytä-selaimissa on joukko hyödyllisiä kehittäjätyökaluja
- Käynnistyksen pikanäppäimeksi vakiintunut F12-näppäin
- Keskeisiä toimintoja
 - HTML DOM-mallin seuraaminen, yleensä reaaliajassa
 - Alkuperäisen lähdekoodin katsominen
 - Tietoliikenteen seuraaminen, HTTP-pyynnöt ja vastaukset
 - JavaScript-koodin virheenjäljitys
 - Konsoli, jolla voi ajaa JavaScript-koodia ("komentorivi") ja johon tulevat mm. virheilmoitukset sekä mahdolliset omat lokimerkinnät
- Lisäksi profilointiin (muistinkulutus, suorituskyky) liittyviä työkaluja

Web-sovellukset

- Yleensä termillä tarkoitetaan interaktiivisia sovelluksia, joita käytetään web-selaimen avulla
- Merkittävänä etuna päätelaiteriippumattomuus
 - Kehittäjän tarvitsee toteuttaa sovellus vain kertaalleen ja silti sovellus on käyttökelpoinen tuhansilla erilaisilla laitteilla mobiilista tabletteihin ja pc-koneisiin
- Web-sovellusten kehittäminen vaatii tuntemusta vähintään neljästä asiasta
 - Taustajärjestelmän ohjelmoinnista
 - HTML-kielestä
 - CSS-kielestä
 - JavaScript-kielestä päätelaitteen toimintojen kehittämistä varten (nk. *front-end* -kehitys)
- Esimerkkejä: Google, Facebook, Twitter, ...

Erilaisia toteutustekniikoita

- Nykyisin yksinkertainen web-kehittäminen voidaan tehdä alusta loppuun tekstieditorilla ja sopivalla pilvipalvelulla – jopa ilmaiseksi
- Tunnettuja front-end -tekniikoita
 - AngularJS, ReactJS, Vue
 - Backbone.JS, Meteor
- Tunnettuja back-end -tekniikoita
 - PHP, Python, Perl, Ruby, Scala
 - ASP.NET, Django, Zend
- Työpöytäsovelluksia web-tekniikoilla?
 - Esim. Electron: <http://electron.atom.io/>

TypeScript-kieli

- Microsoftilta lähtöisin oleva JavaScript-pohjainen ohjelmointikieli, joka on nykyisin avointa lähdekoodia ja toimii kaikilla alustoilla
- Lyhyesti "TS", kääntäjä on nimeltään "TSC" (vertaa "CSC", eli C#:n kääntäjä)
- JavaScript-kielen ylijoukko (superset)
 - Kaikki mikä on kelpollista JavaScriptiä, on myös TypeScriptiä
- TypeScript-kääntäjä tuottaa tuloksena JavaScript-koodia
 - Teknisesti "transpiler" eli lähdekoodista lähdekoodiin kääntäjä, ei "compiler" (kääntäjä)
- Selaimet eivät suoraan ymmärrä TypeScript-kieltä, vaan se on ensin käännettävä JavaScript-kieleksi
- <http://www.typescriptlang.org/>

jQuery-tekniikka

- Yksi ensimmäisiä ja tunnetumpia JavaScript-kirjastoja
- Tehtävänä mahdollistaa HTML DOM-mallin käsittely ja muokkaus tehokkaasti käytetystä selaimesta riippumatta
- Koodissa tunnistaa \$-merkistä
- Sivun alustus
 - ```
$(function () {
 // koodia tähän
});
```
- Osoite: [www.jquery.com](http://www.jquery.com)

# Harjoitukset

---

1. Toteuta perus-HTML-sivu siten, että siihen on liitetty kaksi ulkopuolista tiedostoa: CSS- ja JavaScript-tiedostot. Varmista, että voit CSS- ja JS-tiedostoja muokkaamalla vaikuttaa HTML-sivun ulkoasuun ja toimintoihin.
2. Kytke edellisen tehtävän HTML-sivuun jQuery-kirjasto. Kirjoita jQueryn avulla sivun alustustoiminto (page load), jossa näytät käyttäjälle tervetuloa-viestin.
3. Tutustu Bootstrap-kehikkoon asentamalla se jollekin HTML-sivulle. Kokeile toteuttaa sen avulla nk. Modal-ikkuna. Tee sivulle nappi (button), jota klikkaamalla avaat ruudulle modal-ikkunan. Käytä Bootstrapin mukana tulevia CSS-tyylejä.
4. Alusta uusi ASP.NET MVC -sovellus. Muokkaa sen CSHTML-tiedostoja katsoaksesi, miten voit muokata käyttöliittymiä. Voitko hyödyntää Bootstrap-kehikkoa ASP.NET-sovelluksissa?
5. Lisää HTML-sivullesi TypeScript-kieltä. Mitä sinun pitää tehdä, jotta selain ymmärtää kirjoittamaasi TypeScript-koodia?