

HTML:n perusteet

© Jari Sarja / Otavan Opisto 2012



Tämä oppimateriaali on lisensoitu [Creative Commons Nimeä-Tarttuva 3.0 -lisensillä](https://creativecommons.org/licenses/by-sa/3.0/)

Sisältö

1. Johdanto	4
1.1. HTML:n lähtökohdat	4
1.2. Millä HTML-sivuja tehdään?	5
1.3. Kuinka tarkastella HTML-sivuja?	7
1.4. HTML-koodin perussäännöt.....	9
Nauhoite 1: HTML:n oikeellisuus	11
Tehtävä 1	11
2. HTML-sivun rakenne.....	12
2.1. Tyylimäärittelytagit div ja span	12
2.1.1. Div.....	12
2.1.2. Span	15
2.2. Kappaleet ja otsikot	17
2.2.1. P-tagit	17
2.2.2. Otsikko-tagit (h1-h6).....	18
Nauhoite 2: Kappaleet ja otsikot	19
Tehtävä 2	20
3. Sisällön muotoilu	21
3.1. Fyysiset tehosteet.....	21
3.2. Loogiset tehosteet	21
3.3. Esimerkki tekstin tehosteista.....	22
3.4. Asemointi.....	23
3.5. Kirjasimet eli fontit	23
3.5.1. Font size eli fontin koko.....	23
3.5.2. Font color eli fontin väri	24
3.5.3. Font Face eli kirjasintyyppi	25
3.5.4. Fonttimäärittelyjen yhdistäminen	25
Nauhoite 3: Asemointi	25
Tehtävä 3	26
4. Linkit	27
4.1. Kuinka linkki tehdään?.....	27
4.2. Viittaus dokumentin sisälle	28
4.3. Target eli minne hyperlinkki avataan.....	29
Nauhoite 4: Linkitys	29
Tehtävä 4	29
5. Listat	30
5.1. UL - järjestämätön lista	30
5.2. OL - järjestetty lista	31
5.3. DL - määritelmäluettelo.....	32
Nauhoite 5: Listat	32
Tehtävä 5	33
6. Kuvat.....	34
6.1. Kompastuskiviä	37
6.2. Interaktiivinen kuva	38
Nauhoite 6: Interaktiiviset kuvat	40
Tehtävä 6	40
7. Viivat ja rivinvaihto	41
Tehtävä 7	42
8. Lomakkeet	43
8.1. FORM - Lomake	43
8.2. INPUT - Syöttökenttä, nappi, yms.....	43
8.3. SELECT - Valintalista	46
8.4. TEXTAREA - Tekstinsyöttökenttä	46
8.5. Esimerkki.....	47

8.6. Lomakkeen viimeistely.....	48
8.6.1. Label for (etiketit)	48
8.6.2. Fieldset ja legend.....	49
8.6.3. Tabindex eli tabulaattori	49
Nauhoite 7: Palautelomake.....	50
Tehtävä 8	51
9. Ulkoasun muotoilu.....	52
9.1. Värit HTML:ssä.....	52
9.2. Kehykset.....	55
9.2.1. Kuinka rakentaa kehykselliset www-sivut?	56
9.3. Iframe	58
9.4. Taulukot.....	61
9.4.1. Taulukon tekeminen	62
9.4.2. Tablen ominaisuudet.....	63
9.4.3. Esimerkkejä	65
Nauhoite 8: iframe	66
Tehtävä 9	66
10. Multimedia.....	67
10.1. Multimedia - ääni	67
10.2. Multimedia: liikkuvaa kuvaa	68
10.2.1. Upotetut videotiedostot.....	68
10.3. Javan liittäminen sivuille	70
10.4. Flashin liittäminen sivuille	71
11. Sivuston taittaminen HTML:n avulla	73
Nauhoite 9: Taulukkotaitto	85
Tehtävä 10	85
Lähteet.....	86

1. Johdanto

HTML:n perusteiden ymmärtäminen on ensiarvoisen tärkeää. HTML:n opiskelu luo pohjan kaikelle, mikä liittyy www-sivujen tekemiseen. Opettelu vie hetken, mutta palkitsee runsain mitoin jo viikon harjoittelun jälkeen. HTML:n hallitsemista pidetään yleisesti tärkeänä, koska kaikki web-sivut on tehty käyttäen HTML-kieltä. Kyseessä ei kuitenkaan ole varsinainen ohjelmointikieli vaan lähinnä HTML on sivunkuvauskieli. Siksi HTML on helppo oppia.

Etkö ole koskaan kuullut puhuttavan HTML:stä? Jos aihe kuvataan lyhyesti, voidaan sanoa, että HTML on liima, jonka avulla verkkosivuja kasataan. Websivut näytetään lähes poikkeuksetta HTML-koodin avulla. Web-sivu voi sisältää esimerkiksi tekstiä, ääntä, videota, animaatiota ja kuvia, mutta ne liitetään sinne aina HTML-koodin avulla. HTML määrittelee sivun rakenteen, se kertoo mihin kuvat sijoitetaan, mille sivuille linkitetään, missä otsikko, missä leipäteksti jne.

HTML on lyhenne sanoista Hypertext Markup Language. Aluksi oli kyse tekstidokumenteista, jotka oli linkitetty toisiinsa nk. hyperlinkkien avulla. Nykyään HTML-dokumentit sisältävät paljon muutakin kuin pelkästään tekstiä ja linkkejä. Nykyaikaiset www-sivut voivat olla toimintoiltaan varsin monipuolisia. Voimme esimerkiksi hoitaa pankkiasioitamme www-sivujen kautta. Voimme myös ostaa tuotteita ja palveluita webin kautta. Tällä kertaa keskitymme kuitenkin vain HTML:n perusteisiin.

HTML-koodia tarvitaan, jotta www-selain voisi tulkita dokumentin sisällön. Selaimella tarkoitetaan ohjelmaa, jonka avulla tarkastelemme www-sivuja. Tunnettuja selaimia ovat Internet Explorer, Firefox ja Chrome. Jotta kaikki selaimet osaisivat näyttää sivut samalla tavalla, on luotu standardi. Tämän hetken standardi tunnetaan nimellä HTML 4.01. Tälläkin opintojaksolla keskitytään lähinnä versioon 4.01.

1.1. HTML:n lähtökohdat

HTML-koodi on varsin yksinkertaista ja loogista, joka tietokoneelta löytyy siihen tarvittavat ohjelmat. Lähtökohtaisesti HTML voidaan tiivistää seuraavasti.

- html-koodin tiedostopäätte on joko .htm tai .html
- html-sivuja luetaan selaimella (esim. Firefox, Chrome)
- html-koodi kirjoitetaan tageilla (esim. tämä lihavoit tekstin)
- html-koodia voidaan kokeilla omalla koneella
- html-sivu näkyy netissä, kun se laitetaan jollekin palvelimelle

HTML-kielestä pääsee helpoiten jyvälle tarkastelemalla valmista koodia.

```
<html>
<head>
<title>Jarín kotisivu</title>
</head>
<body>
<h1>Sivun otsikko</h1>
<p>Tässä yksi kappale tekstiä. Ei mitään
kummoista, mutta jotain kuitenkin. Onpahan
vähän esimerkkiä siitä kuinka homma toimii.</p>
</body>
</html>
```

Sivu näyttää kutakuinkin tällaiselta, kun sitä tutkaillaan webiselaimella.



1.2. Millä HTML-sivuja tehdään?

HTML-koodia ei missään tapauksessa pidä kirjoittaa esim. Word-ohjelmalla, koska se tallentaa tekstin lisäksi omia muotoilujaan. HTML-sivuja tehdään yleensä tekstieditorilla eli tekstimuokkaimella, joka on tarkoitettu puhtaan tekstin kirjoittamiseen ja muokkaamiseen.

Tekstinkäsittelyohjelmista poiketen tekstieditoreissa ei yleensä ole muotoiluominaisuuksia lainkaan eli niillä ei voi esimerkiksi lihavoida tai kursivoida tekstiä. Useimpien käyttöjärjestelmien mukana tulee jonkinlainen tekstieditori. Usein valmiiksi asennetut tekstieditorit, kuten Windowsin Muistio, ovat varsin puutteellisia. Nykyään on onneksi olemassa myös kunnollisia editoreita ja vieläpä ilmaiseksi.

Hyvällä editorilla on oltava seuraavat ominaisuudet:

- syntaksin korostus (esim. eri väri)
- rivinumerointi (helpottaa virheiden etsinnässä)
- mahdollisuus pitää monta ikkunaa auki yhtä aikaa

Nuo kaikki ominaisuudet puuttuvat esimerkiksi Muistiolta (Windows käyttöjärjestelmän mukana), joten se on huono valinta. Parempia vaihtoehtoja on tarjolla myös ilmaiseksi.

Windowsille ilmaisia editoreita

- [Notepad++](#) (suosittelen, paras)
- [Crimson Editor](#)
- [EasyHTML](#) (suomalainen ja hyvä)
- [EditPad Lite](#) (kätevä)
- [NoteTab Light](#) (toinen suosikkini)
- [MysticML](#) (mielenkiintoinen ja suomalainen)
- [Cador](#)
- [Komodo Edit](#)
- [iEdit](#)

Linuxille ilmaisia editoreita

- [Vim](#)
- [Gedit](#)
- [Nano](#)
- [Emacs](#)
- [Komodo Edit](#)
- [Bluefish](#)
- [iEdit](#)

Macintoshille ilmaisia editoreita

- [TextWrangler](#)
- [Carbon Emacs](#)
- [Bluefish](#)
- [Komodo Edit](#)

On olemassa myös helpompia tapoja koodata www-sivuja kuin pelkät tekstieditorit. Niiden käyttäminen ei kuitenkaan ole suositeltavaa. Niillä voi pärjätä ihan hyvin, mutta jos ongelmia ilmenee, olet pulassa. Yleensä WYSIWYG-editorit tekevät sekavaa HTML-koodia. Lisäksi jos tähtää websuunnittelijaksi, unohda WYSIWYG-editorit ainakin hetkeksi.

"WYSIWYG-lyhennettä (engl. What You See Is What You Get eli mitä näet, sitä saat) käytetään viittaamaan sellaisiin ohjelmistoihin, joissa sisältö näyttää muokattaessa hyvin samalta kuin lopputulos. Sen sijaan muotoa ja rakennetta kuvaavaa metadataa ei välttämättä näydetä. Hyvin usein käsitettä käytetään kuvaamaan tekstinkäsittelyohjelmia tai HTML-editoreja." ([Wikipedia.](#))

Hyviä ilmaisia WYSIWYG-editoreita ovat mm. seuraavat:

- [NVU](#) (helppokäyttöinen)
- [Amaya](#) (erikoinen käyttöliittymä, mutta hallitsee myös MathML:n)
- [Trellian](#) (en ole kokeillut, mutta näyttää lupaavalta)

Kaupallisista WYSIWYG-editoreista Dreamweaver on pitkään ollut markkinoiden ykköstilalla. Microsoft yritti aluksi haastaa sitä ohjelmalla FrontPage, mutta ei onnistunut. FrontPage sai väistyä ja tilalle tuli Microsoft Expression Web. Nykyään kaupalliset editorit alkavat jo olla aika hyviä ja ovat löytäneet oman vankkumattoman käyttäjäkuntansa.

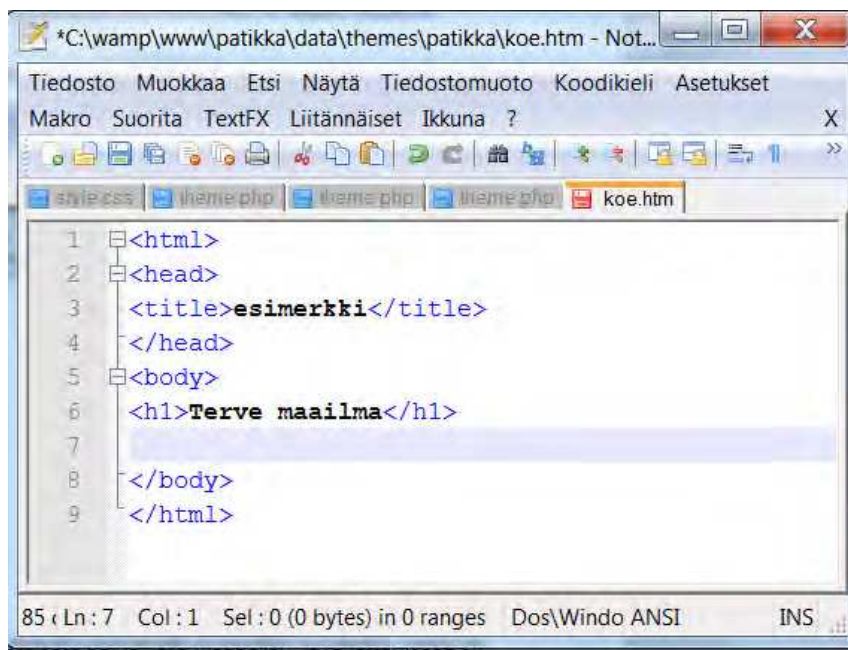
Nykyisin on olemassa myös niin kutsuttuja online-editoreita. Ne ovat yleensä osa jotakin julkaisujärjestelmää, kuten Joomla, Drupal tai Wordpress. Online-editori hoitaa koodaustyön sivujen päivittäjän puolesta. Online-editori mahdollistaa websivujen päivittämisen suoraan webselaimen avulla. Sen etu verrattuna offline-editoriin on siinä, ettei tiedostoja tarvitse siirrellä oman koneen ja webpalvelimen välillä.

Sivujen päivittäminen perinteisesti offline-editorin avulla tarkoittaa sitä, että sivu siirretään ensin muokattavaksi tietokoneelle, johon on asennettu offline-editori. Muokkauksen jälkeen sivu siirretään takaisin palvelimelle. Offline-editoreiden suurin heikkous on siinä, että kaikkien, jotka sivuja muokkaavat, on asennettava koneelleen kyseinen ohjelma.

1.3. Kuinka tarkastella HTML-sivuja?

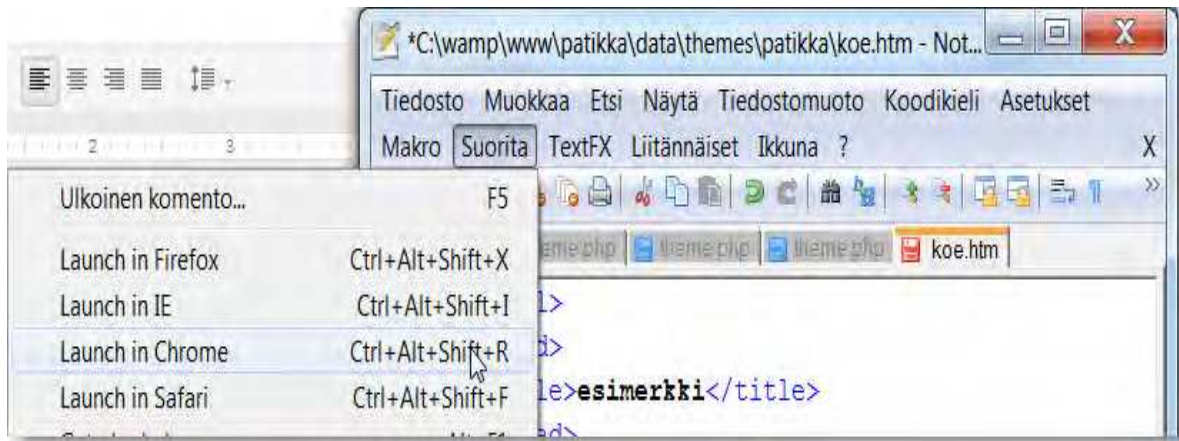
Web-sivujen teko HTML-kielellä on kiitollista puuhaa siinä suhteessa, että näet helposti työsi jäljen, vaikka et laittaisi sivua välittömästi nettiin. Nykyään jokaisessa tietokoneessa on selain, muuta ei sivun tarkastelemiseen tarvita.

Kirjoita pieni esimerkki HTML-kielellä vaikka tähän tapaan.



```
*C:\wamp\www\patikka\data\themes\patikka\koe.htm - Not...
Tiedosto Muokkaa Etsi Näytä Tiedostomuoto Koodikieli Asetukset
Makro Suorita TextFX Liitännäiset Ikkuna ?
C:\wamp\www\patikka\data\themes\patikka\koe.htm
1 <html>
2 <head>
3 <title>esimerkki</title>
4 </head>
5 <body>
6 <h1>Terve maailma</h1>
7
8 </body>
9 </html>
85 Ln: 7 Col: 1 Sel: 0 (0 bytes) in 0 ranges Dos\Windo ANSI INS
```

NotePad++:n kautta pääset tarkastelemaan lopputulosta valitsemalla **Suorita** ja sen jälkeen valitsemalla haluamasi selain.

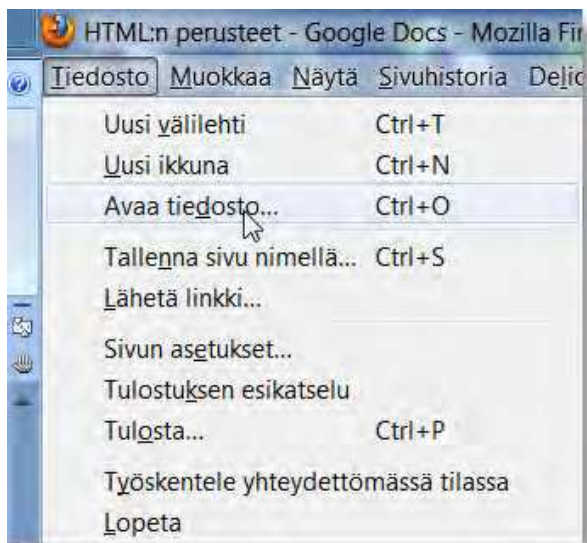


Tämän jälkeen valittu selain näyttää koodatun sivun sisällön.

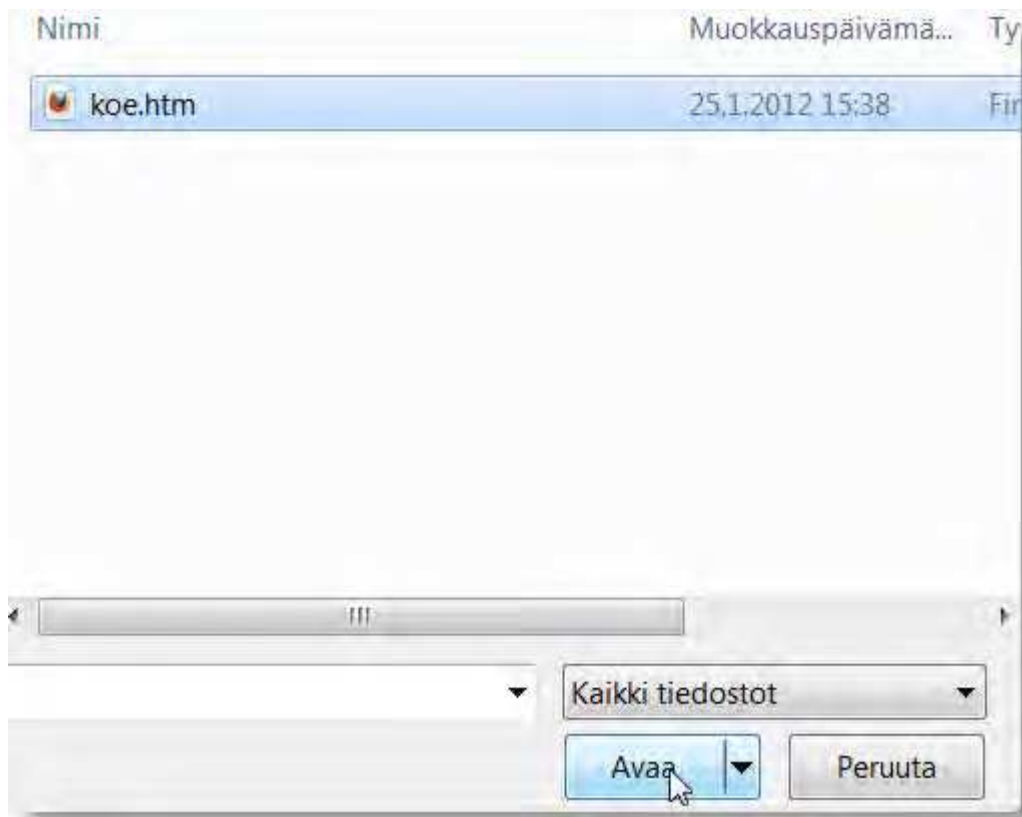


Katsotaan vielä, kuinka homma hoituu selaimen kautta.

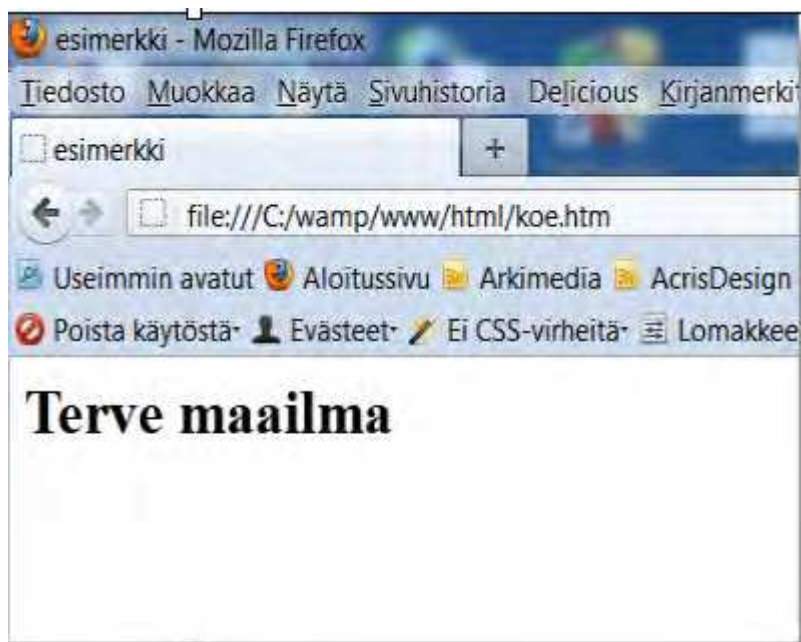
1) Avaa www-selain, esimerkiksi Firefox. Valitse **Tiedosto > Avaa tiedosto...**



2) Etsi tiedosto ja klikkaa **Avaa**. Tässä tapauksessa tiedosto on koe.htm.



3) Valitse ok ja katso lopputulos.



1.4. HTML-koodin perussäännöt

HTML ei ole varsinainen ohjelmointikieli, joten syntaksin eli sääntöjen osalta ei ole paljon muistettavaa. Tässä kuitenkin perussäännöt, joita olisi hyvä noudattaa.

Ei ole väliä, kirjoitatko tagit isolla vai pienellä kirjaimella. XHTML:ää ajatellen olisi hyvä opetella kirjoittamaan tagit pienellä, esimerkiksi näin **lihavoitua tekstiä**.

Tagit tulee aina päättää ja mieluiten oikeassa järjestyksessä. Oikea järjestys on se, että ensin alkanut tagi loppuu viimeisenä. Esimerkiksi lihavointi, joka alkaa kappaleen sisällä ei voi loppua sen ulkopuolella.

väärin

```
<p><b>tässä lihavoitua tekstiä</b></p>
```

oikein

```
<p><b>tässä lihavoitua tekstiä</b></p>
```

väärin

```
<p>olipahan paljon tekstiä
```

oikein

```
<p>olipahan paljon tekstiä</p>
```

Sivun rakenne on aina se, että ensin html, sitten head-osio ja viimeisenä body-osio.

```
1 <html>
2 <head>
3   <title>Jarim kotisivu</title>
4 </head>
5
6 <body>
7   <h1>Tähän otsikko</h1>
8   <p>Tähän kappale yksi, jossa tekstiä.
9   Ei mitään kummallista, mutta jotakin</p>
10  <p>Toisessa kappaleessa jotakin asiaa,
11  ainakin toivoisin niin</p>
12 </body>
13 </html>
```

Sivut toimivat, vaikka rikkoisit edellä olevia sääntöjä. Olisi kuitenkin hyvä oppia alusta saakka kirjoittamaan hyvää HTML-koodia, koska siitä on hyötyä mm. opeteltaessa PHP-ohjelmointia. Lisäksi hyvin koodatut sivut näkyvät aina oikein ja latautuvat nopeasti.

HTML-koodia voidaan kirjoittaa miten sattuu ja silti selain näyttää kaiken oikein? Se johtuu siitä, että silloin selain toimii "quirksmodessa" eli korjaa tekemäsi virheet. Parempi tapa on tehdä sivut oikeiksi ilman, että selain hoitaa työn puolestasi. Lisäksi eri selaimet suorittavat korjaukset eri tavoin, voit joutua kierteeseen jossa sivut näkyvät eri selaimilla aina erilaisina.



Nauhoite 1: HTML:n oikeellisuus

kesto: 11:35

URL: <http://youtu.be/aysDhfg0FkE>

Tehtävä 1

Mitä virheitä löydät alla olevasta HTML-koodista? Luettele ne kaikki.

```
<HTML>
<HEAD>
</HEAD>

<h1>Sivun otsikko</h1>
<BODY>
<p>Tässä on yksi kappale tekstiä. Ei mitään
kumoista, mutta jotain kuitenkin. Onpahan vähän
esimerkkiä siitä kuinka homma toimii.

</BODY>
</HTML>
```

2. HTML-sivun rakenne

HTML-koodissa on seitsemän tagia, jotka määrittelevät pelkästään sivun rakennetta. Ne ovat html, head, body, div ja span. Kaksi edellisestä listasta puuttuvaa tagia ovat h ja p, joista myöhemmin lisää.

HTML

Jokainen HTML-sivu alkaa tagilla <html> ja päättyy tagiin </html>. Sillä ei ole väliä kirjoitetaanko <html> vai <HTML>. HTML on juurielementti eli kaikkien muiden elementtien on oltava täysin sen sisällä. Minimissään HTML-sivu sisältää tagit html, head ja body.

HEAD

Head-tagin sisään kirjoitetaan metatietoja, tyylitietoja ja skriptejä. Tärkein elementti sen sisällä on <title>, joka määrittää sivulle otsikon. Head-tagin päätetään juuri ennen body-tagia. HEAD-elementti on pakollinen, ja sitä käytetään vain yhden kerran per sivu. Sivulla ei voi olla kahta head-tagia.

BODY

Body-tagin sisään tulee sivun kaikki sisältö. Body-tagin alku on välittömästi kun head-tagin päättyy. Jokaisen sivun toiseksi viimeisen tagin tulisi olla </body>.

2.1. Tyylimäärittelytagit div ja span

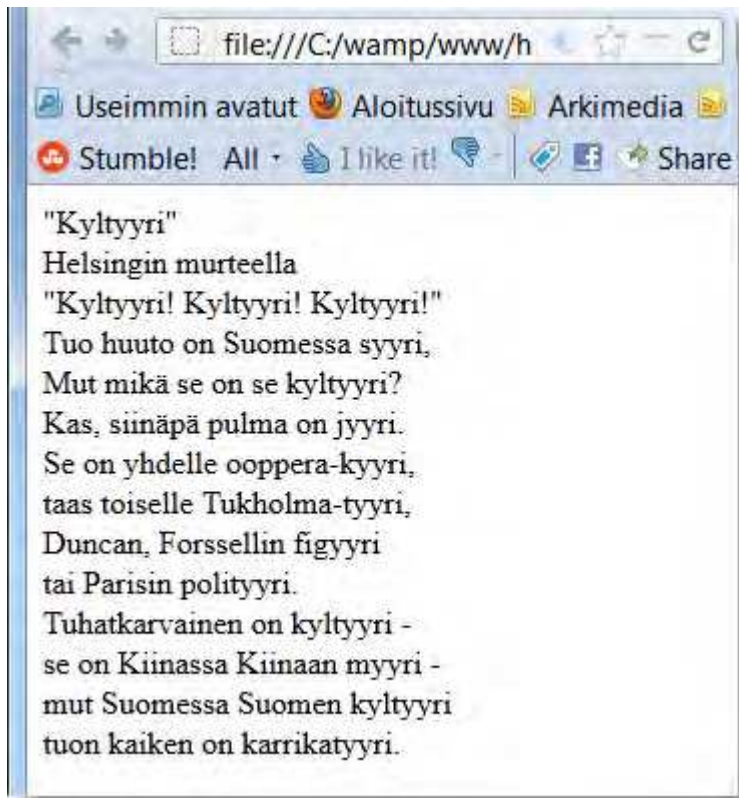
Div ja span ovat HTML-elementtejä, joiden käyttöön liittyvät aina tyylimääritykset. Erityisesti div-elementillä on tärkeä rooli www-sivun taitossa. Ilman tyylimäärittelyjä kyseiset tagit eivät näytä tekevän sivuston ulkoasun kannalta paljoakaan. Siksi div- ja span-tagit voivat tuntua tarpeettomilta. Niistä saadaankin hyöty irti vasta, kun niihin lisätään CSS-tyylimäärittelyt.

2.1.1. Div

Div-tagilla voidaan ryhmitellä lohkoelementtejä. Sitä käytetään esimerkiksi dokumentin tietyn osan fontin, taustaväri tai sijoittelun apuna. Yleensä div-tagit liittyvät CSS-muotoiluun eli ne eivät tee itsessään mitään ulkoasumuunnoksia, mutta ne muodostavat oman lohkon. Esimerkiksi Eino Leinon runo voidaan kirjoittaa div-tageilla (ks. seuraava sivu).

```
C:\wamp\www\html\koe.htm - Notepad++
Tiedosto Muokkaa Etsi Näytä Tiedostomuoto Koodikieli Asetukset Makro Suorita TextFX
Liitännäiset Ikkuna ?
koe.htm
1 <html>
2 <head>
3 <title>Eino Leinin runo Kyltyyri</title>
4 </head>
5
6 <body>
7 <div>"Kyltyyri" </div>
8 <div></div>
9 <div>Helsingin murteella</div>
10
11 <div>"Kyltyyri! Kyltyyri! Kyltyyri!"</div>
12 <div>Tuo huuto on Suomessa syyri,</div>
13 <div>Mut mikä se on se kyltyyri?</div>
14 <div>Kas, siinäpä pulma on jyyri.</div>
15 <div></div>
16 <div>Se on yhdelle ooppera-kyyri,</div>
17 <div>taas toiselle Tukholma-tyyri,</div>
18 <div>Duncan, Forssellin figyyri</div>
19 <div>tai Parisin polityyri.</div>
20 <div></div>
21 <div>Tuhatkarvainen on kyltyyri -</div>
22 <div>se on Kiinassa Kiinaan myyri -</div>
23 <div>mut Suomessa Suomen kyltyyri</div>
24 <div>tuon kaiken on karrikatyyri.</div>
25 </body>
26 </html>
629 chars 679 bytes 26 lines Ln:8 Col:12 Sel:0 (0 bytes) in 0 ranges Dos\Windo ANSI INS
```

Selaimen kautta katseltuna runo näyttää kuten alla.



Jokainen div-tagii muodostaa oman lohkonsa eli tässä tapauksessa suorakaiteen, joka alkaa tagilla <div> ja päättyy tagiin </div>. Jos alueen kokoa ei ole määritetty, ulottuu div-tagin rajaama alue koko rivin mittaiseksi. Tällä tavoin tehdyt lohkot sijoittuvat oletusarvoisesti allekkain.



Yleensä diviä tarvitaan CSS:ää varten, kun esimerkiksi palstoitetaan sivua.

2.1.2. Span

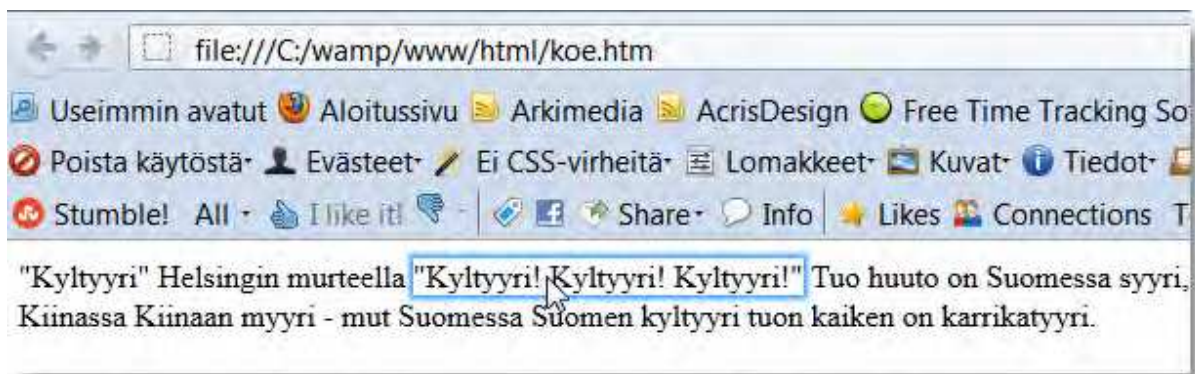
Span ja div toimivat pitkälti samalla tavalla, mutta span ei tee rivitystä. Span jatkaa samalla rivillä niin kauan kuin mahdollista. Eli sama runo nyt span-tagin avulla kirjoitettuna.

```
koe.htm
1 <html>
2 <head>
3 <title>Eino Leinon runo Kyltyyri</title>
4 </head>
5
6 <body>
7 <span>"Kyltyyri" </span>
8 <span></span>
9 <span>Helsingin murteella</span>
10
11 <span>"Kyltyyri! Kyltyyri! Kyltyyri!"</span>
12 <span>Tuo huuto on Suomessa syyri,</span>
13 <span>Mut mikä se on se kyltyyri?</span>
14 <span>Kas, siinäpä pulma on jyyri.</span>
15 <span></span>
16 <span>Se on yhdelle ooppera-kyyri,</span>
17 <span>taas toiselle Tukholma-tyyri,</span>
18 <span>Duncan, Forssellin figyyri</span>
19 <span>tai Parisin polityyri.</span>
20 <span></span>
21 <span>Tuhatkarvainen on kyltyyri -</span>
22 <span>se on Kiinassa Kiinaan myyri -</span>
23 <span>mut Suomessa Suomen kyltyyri</span>
24 <span>tuon kaiken on karrikatyyri.</span>
25 </body>
26 </html>
```

Ja tältä se näyttää selaimen kautta katseltuna.



HUOM! Span-taggi ei muodosta kappaletta tekstistä, teksti tulee pötköön eli inline.



Rakenteellisesti oikein tehty HTML-sivu on kuten alla olevassa kuvassa.

```
koe.htm
1 <html>
2 <head>
3   <title>Jarín kotisivu</title>
4 </head>
5
6 <body>
7   <h1>Tähán otsikko</h1>
8   <p>Tähán kappale yksi, jossa tekstiá.
9   Ei mitáán kummallista, mutta jotakin</p>
10  <p>Toisessa kappaleessa jotakin asiaa,
11  ainakin toivoisin niin</p>
12 </body>
13 </html>
```


Selaimen kautta katseltuna sivu näyttää kuten alla.



2.2. Kappaleet ja otsikot

Nyt keskitytään siihen, kuinka sivuille saadaan tekstiä. HTML:ssä, kuten tekstinkäsittelyssä yleensä, on tärkeää erottaa kappaleet toisistaan. Lisäksi tarvitaan otsikoita, joita HTML:ssä on kaikkiaan kuusi erilaista. Yleensä niistä käytetään kuitenkin vain kolmea ensimmäistä.

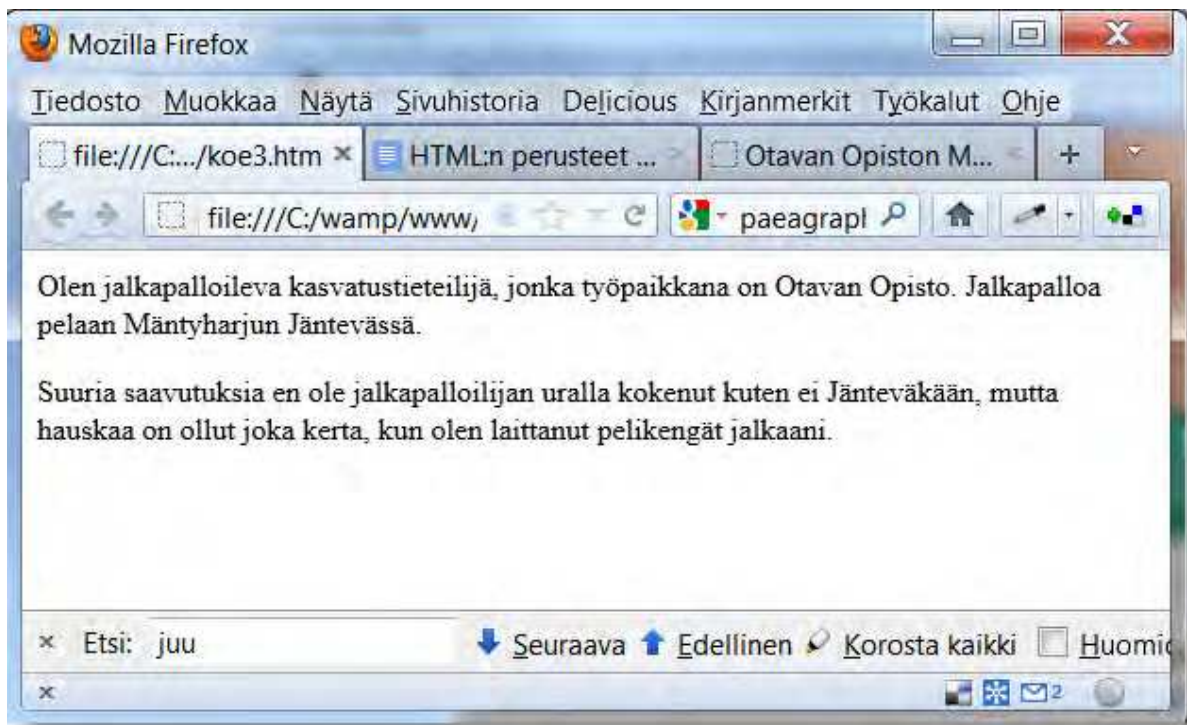
2.2.1. P-taggi

P-tagia <p> (engl. paragraph) tarvitaan lähinnä leipätekstiä varten. Sen avulla kappaleet erotellaan toisistaan. Tässä pieni esimerkki.

A screenshot of a code editor window with two tabs: koe.htm and koe3.htm. The code editor shows the following HTML code:

```
1 <h1>Jari Sarja</h1><p>Olen jalkapalloileva kasvatustieteilijä,  
2 jonka työpaikkana on Otavan Opisto. Jalkapalloa  
3 pelaan Mäntyharjun Jäntevässä. </p><p>Suuria saavutuksia  
4 en ole jalkapalloilijan uralla kokenut kuten ei  
5 Jänteväkään, mutta hauskaa on ollut joka kerta,  
6 kun olen laittanut pelikengät jalkaani.  
7 </p>
```

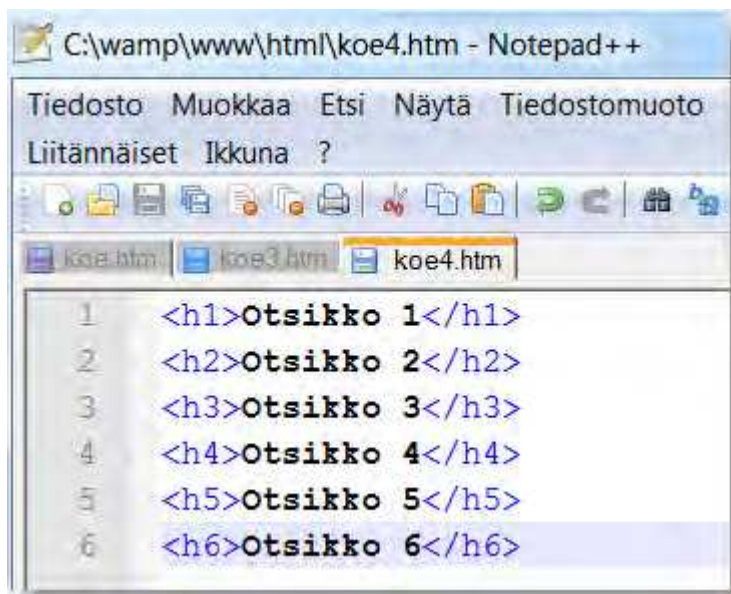
Selaimessa koodi näyttää kuten alla olevassa kuvassa.



HUOM! Vaikka editorissa painetaan enter, se ei tarkoita rivinvaihtoa html-koodissa. Kappalet erotetaan toisistaan p-tagin avulla.

2.2.2. Otsikko-tagit (h1-h6)

Otsikko-tagien avulla sivulle luodaan rakennetta. Sivun tärkein otsikko merkataan yleensä tagilla h1. Seuraavan tason otsikko tagilla h2, ja kun siitä mennään taso alemmas, käytetään tagia h3.



Selaimen kautta tarkasteltuna yllä oleva koodi näyttää kuten alla olevassa kuvassa.



Nauhoite 2: Kappaleet ja otsikot

kesto: 07:20

URL: <http://youtu.be/Op4rtLzueA>

Tehtävä 2

Kirjoita alla oleva teksti html-koodilla, jotta se näyttäisi samalta kuin tässä alla. Käytettävissä ovat vain HTML-tagit <html>, <head>, <body>, <p> ja <h3>.

Eksosomaattinen oppiminen

Fyysisten ja kielellisten välineitten avulla ihminen voi ylittää itsensä. Vuorovaikutuksemme kyseisten välineiden kanssa on keskeistä oppimisen kannalta. Yhdessä nämä erilaiset välineet muodostavat osan kulttuurisista voimavaroistamme. Jo pikainen silmäys omaan asuntoomme osoittaa kuinka tärkeäksi erilaiset fyysiset välineet ovat tulleet erilaisten toimintojen ylläpidossa ja erilaisten ongelmien ratkaisuisissa. Elämme autojen, puhelinten, CD-soittimien, mikroaaltouunien, televisioiden, kahvinkeitinien ja tietokoneiden maailmassa. (Säljö 2001, 72.) Nämä kaikki em. välineet sijaitsevat oman kehomme ulkopuolella, ne ovat siis eksosomaattisia.

Oppimisen suhteen onkin kysyttävä, että onko tekniikka ihmisen sisä- vai ulkopuolella? Toinen yhtä keskeinen kysymys on, että onko oppiminen aina samanlaista? Tekniikan kehittyessä meidän kommunikointimme on muuttunut, samalla tavalla myös oppiminen on muuttunut. Oppiminen eilen ja tänään eivät siis ole täysin samanlaisia tapahtumia. Onkin hyvä luoda katsaus taaksepäin, ja tarkastella sitä kuinka ihmiset ennen kommunikoivat ja oppivat. Yksi olennainen ero eilispäivän ja nykypäivän välillä on siinä, että aina ei ole ollut instituutiota nimeltä koulu. Kuitenkin myös ennen ihmiset kommunikoivat ja oppivat.

Tässä sama raakatekstinä, jonka voit kopioida harjoitusta varten:

Eksosomaattinen oppiminen Fyysisten ja kielellisten välineitten avulla ihminen voi ylittää itsensä. Vuorovaikutuksemme kyseisten välineiden kanssa on keskeistä oppimisen kannalta. Yhdessä nämä erilaiset välineet muodostavat osan kulttuurisista voimavaroistamme. Jo pikainen silmäys omaan asuntoomme osoittaa kuinka tärkeäksi erilaiset fyysiset välineet ovat tulleet erilaisten toimintojen ylläpidossa ja erilaisten ongelmien ratkaisuisissa. Elämme autojen, puhelinten, CD-soittimien, mikroaaltouunien, televisioiden, kahvinkeitinien ja tietokoneiden maailmassa. (Säljö 2001, 72.) Nämä kaikki em. välineet sijaitsevat oman kehomme ulkopuolella, ne ovat siis eksosomaattisia. Oppimisen suhteen onkin kysyttävä, että onko tekniikka ihmisen sisä- vai ulkopuolella? Toinen yhtä keskeinen kysymys on, että onko oppiminen aina samanlaista? Tekniikan kehittyessä meidän kommunikointimme on muuttunut, samalla tavalla myös oppiminen on muuttunut. Oppiminen eilen ja tänään eivät siis ole täysin samanlaisia tapahtumia. Onkin hyvä luoda katsaus taaksepäin, ja tarkastella sitä kuinka ihmiset ennen kommunikoivat ja oppivat. Yksi olennainen ero eilispäivän ja nykypäivän välillä on siinä, että aina ei ole ollut instituutiota nimeltä koulu. Kuitenkin myös ennen ihmiset kommunikoivat ja oppivat.

3. Sisällön muotoilu

Sisällön muotoilun avulla voidaan parantaa sivujen luettavuutta. Se mahdollistaa myös asioiden korostamisen. HTML on kuitenkin sisällön muotoilun osalta kaukana siitä, mihin nykyiset tekstinkäsittelyohjelmat kykenevät. Perustason muotoilu sen sijaan onnistuu. Tehokeinojen avulla voidaan korostaa tai merkitä tiettyä osaa tekstistä. Kyse on esimerkiksi tekstin lihavoinnista ja kursivoinnista, mutta toisaalta myös tavasta kirjoittaa lainaukset ja määritelmät.

3.1. Fyysiset tehosteet

Tavallisia fyysisiä tehosteita ovat lihavointi `` ja kursivointi `<i>`. Mutta muitakin löytyy, kuten käy ilmi alla olevasta taulukosta. Varsinkin matematiikassa tarvitaan yläindeksejä.

miltä näyttää?	englanniksi	tagi
lihavointi	bold	<code>lihavointi</code>
kursivointi	italic	<code><i>kursivointi</i></code>
poistettu	strike	<code><strike>poistettu</strike></code>
$E=MC^2$	superscript	<code><sup>yläindeksi</sup></code>
alaindeksi ₂	subscript	<code><sub>alaindeksi</sub></code>

HUOM! Kursivointia ei kannata käyttää weisivuilla, koska tekstistä tulee vaikealukuista.

3.2. Loogiset tehosteet

Loogiset tekstitehosteet määrittelevät tehosteen antaman vaikutelman, mutta eivät sitä, miltä lopullinen teksti näyttää. Eniten näistä tehosteista käytetään tehostetta `<blockquote>`, jonka avulla tekstiä voidaan sisentää. Toinen varsin käyttökelpoinen looginen tehoste on `<code>`, jonka avulla weisivulle saadaan koodi kätevästi Courier-kirjasimella kirjoitettuna.

Tagi	Mihin sopii?	Miten vaikuttaa?
<code><ADDRESS></ADDRESS></code>	Sivun tekijän tiedot	Kursiivi
<code><BLOCKQUOTE></BLOCKQUOTE></code>	Pitkä lainaus	Kappaleen sisennys, kappaleen ylä- ja alapuolelle jätetään vähän tyhjää tilaa
<code><CITE></CITE></code>	Lainaus	Kursiivi
<code><CODE></CODE></code>	Ohjelmakoodi	Courier-fontti
<code><DFN></DFN></code>	Määritelmä	Kursiivi
<code></code>	Painotus	Kursiivi
<code><KBD></KBD></code>	Näppäinteksti	Lihavoitu courier-fontti

<SAMP></SAMP>	Esimerkki	Courier-fontti
	Voimakas painotus	Lihavoitu
<VAR></VAR>	Korvattava muutuja	Kursiivi

3.3. Esimerkki tekstin tehosteista

```
<html>
<head>
<title>Esimerkki tehosteista</title>
</head>
<body>
<h2>Haasteena muutos ja oppiminen</h2>
```

```
<p><b>Tietotekniikka</b> on maataloilla arkipäiväinen juttu. <i>Lehmällä on panta</i>, joka sisältää digitaalisessa muodossa tietoja lehmästä. Pannan tietojen perusteella tietokone osaa kertoa mitä lehmä on syönyt ja paljonko se luovuttaa maitoa lypsykertaa kohti. </p>
```

```
<p>Perinteisiäkin <b>navettoja</b> on vielä olemassa, joissa lehmät ovat parressa ja lypsäminen sujuu kannukoneella. Ne ovat nykyaikaiselle maatalouslomittajalle eksoottisia harvinaisuuksia. Muutos on vaatinut <em>maataloilta</em>, ja varsinkin maatalouslomittajilta oppimista ja sopeutumista tekniikan suomiin mahdollisuuksiin ja oikkuihin. Jopa yleensä muutoksia karsastavat <b>lehmät</b> ovat oppineet uusille tavoille.</p>
```

```
<address>Jari Sarja, Mäntyharju</address>
```

```
</body>
</html>
```

Selaimen kautta esimerkki näyttää kuten alla olevassa kuvassa.



Kuten kuvasta näkyy, kannattaa tekstin tehokeinoja käyttää säästeliäästi. Varsinkin kursivointi toimii tietokoneen näytöllä huonosti.

3.4. Asemointi

Asemoinnissa on käytössä kolme eri mahdollisuutta samaan tapaan kuin vaikkapa Word-tekstinkäsittelyohjelmassa. Voit asemoida tekstin oikealle, vasemmalle tai keskelle. Voit myös tasata sekä oikean että vasemman reunuksen, mikä ei tosin webissä ole suotavaa puutteellisen tavutuksen vuoksi. Tässä alla esitellään kaikki asemointitavat esimerkin kera.

<p align=left> Vasen tasaus

Olen jalkapalloileva kasvatustieteilijä, jonka työpaikkana on Otavan Opisto. Jalkapalloa pelaan Mäntyharjun Jäntevässä. Suuria saavutuksia en ole jalkapalloilijan uralla kokenut kuten ei Jänteväkään, mutta hauskaa on ollut joka kerta, kun olen laittanut pelikengät jalkaani.

<p align=center> Keskitetty

Olen jalkapalloileva kasvatustieteilijä, jonka työpaikkana on Otavan Opisto. Jalkapalloa pelaan Mäntyharjun Jäntevässä. Suuria saavutuksia en ole jalkapalloilijan uralla kokenut kuten ei Jänteväkään, mutta hauskaa on ollut joka kerta, kun olen laittanut pelikengät jalkaani.

<p align=right> Oikea tasaus

Olen jalkapalloileva kasvatustieteilijä, jonka työpaikkana on Otavan Opisto. Jalkapalloa pelaan Mäntyharjun Jäntevässä. Suuria saavutuksia en ole jalkapalloilijan uralla kokenut kuten ei Jänteväkään, mutta hauskaa on ollut joka kerta, kun olen laittanut pelikengät jalkaani.

<p align=justify> Molemmat tasan

Olen jalkapalloileva kasvatustieteilijä, jonka työpaikkana on Otavan Opisto. Jalkapalloa pelaan Mäntyharjun Jäntevässä. Suuria saavutuksia en ole jalkapalloilijan uralla kokenut kuten ei Jänteväkään, mutta hauskaa on ollut joka kerta, kun olen laittanut pelikengät jalkaani.

3.5. Kirjasimet eli fontit

HTML mahdollistaa fonttien eli kirjasimien määrittämisen. HTML:n avulla voidaan säätää kirjasimien kokoa, väriä ja kirjasinperhettä. Kirjasimia ei ole pakko määritellä. Jos ne jätetään määrittelemättä, näyttää selain kirjasimet oletusarvonsa mukaisesti.

3.5.1. Font size eli fontin koko

Fontin koko määritellään tagilla . Fonttien koko ilmoitetaan asteikolla 1...7 tai suhteellisina arvoina, esim. size="+1". Peruskoko on 3. Tässä esimerkkejä:

```
<p><font size="1">kirjasin</font></p>
<p><font size="2">kirjasin</font></p>
<p><font size="3">kirjasin</font></p>
<p><font size="4">kirjasin</font></p>
<p><font size="5">kirjasin</font></p>
<p><font size="6">kirjasin</font></p>
<p><font size="7">kirjasin</font></p>
```



3.5.2. Font color eli fontin väri

Väri ilmoitetaan joko RGB-arvona tai nimenä. Tässä pari esimerkkiä:

```
<p><font color="red">punainen</font></p>  
<p><font color="green">vihreä</font></p>  
<p><font color="blue">sininen</font></p>  
<p><font color="#ff0000">punainen</font></p>  
<p><font color="#00ff00">vihreä</font></p>  
<p><font color="0000ff">sininen</font></p>
```



3.5.3. Font Face eli kirjasintyyppi

Kirjasintyyppi määritellään face-attribuutin avulla. Tässä esimerkkejä:

```
<p><font face="Arial">Jari Sarja: Arial</font></p>
<p><font face="Times">Jari Sarja: Times</font></p>
<p><font face="Verdana">Jari Sarja: Verdana</font></p>
<p><font face="Bookman">Jari Sarja: Bookman</font></p>
```



HUOM! Yleensä webissä käytetään päätteettömiä kirjasintyyppiä kuten asim. Arial ja Verdana.

3.5.4. Fonttimäärittelyjen yhdistäminen

Kirjasimien määrittelyjä voidaan myös yhdistää kuten esimerkissä tässä alla.

```
<font face="Arial,Helvetica,sans-serif" size="4" color="blue">Fonttien yhdistelmää</font>
```



Nauhoite 3: Asemointi

kesto: 06:00

URL: <http://youtu.be/UYGbt0Kf6IQ>

Tehtävä 3

Mitä seuraava tagi tekee? `<p align=justify>`
Miksi sitä ei kannata käyttää webissä?

4. Linkit

Linkkien avulla teksti alkaa elää omaa elämäänsä. Dokumenttia, jossa on linkki toiseen dokumenttiin, kutsutaan hypertekstiksi. Vaikka www-sivut linkkisanoinen ovat monen ensi kosketus hypertekstiin, myös monet perinteisistä teksteistä houkuttelevat käyttäjänsä hypymään kohdasta ja kappaleesta toiseen - tai vielä pitemmälle: lukijan omaan alitajuntaan. Hyperteksti antaa avoimesti lukijalleen luvan jälleenrakentaa tekstin mieleisekseen tulkintoja tehden, assosiaatioita seuraten ja rakennetta muokaten.

4.1. Kuinka linkki tehdään?

Linkki koostuu kahdesta osasta. Ensimmäisen osan muodostaa sivun osoite ``.

Toisen osan muodostaa linkkiteksti, jonka jälkeen linkki-tagin suljetaan ``.
`Otavan Opisto`

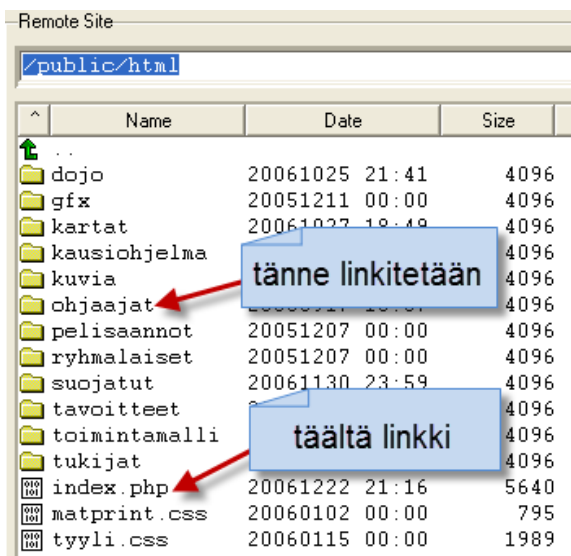
Yllä oleva esimerkki näkyy sivulla kuten tässä alla.

[Otavan Opisto](#)

Tagin muistaminen voi olla hankalaa, mutta muistamista voi helpottaa, jos tietää tagin taustan. Sen ensimmäinen osa eli `a` on lyhenne sanasta anchor, joka on suomeksi ankkuri. Toinen osa eli `href` määrittelee hyperlinkin. Se on lyhenne sanoista "hyperlink reference". Suomennettuna hyperlink reference on kutakuinkin sama kuin "hyperlinkin viittaus".

Linkin osoitteen voi kirjoittaa kahdella tavalla. On olemassa suhteellinen ja absoluuttinen tapa linkittää. **Absoluuttisessa tavassa** linkin osoite kirjoitetaan kokonaisena osoitteena. **Suhteellisessa tavassa** otetaan lähtökohdaksi kansio, jossa dokumentti linkkeineen sijaitsee ja viitataan ylös- tai alaspäin hakemistorakenteessa. Tämä tapa on yleisesti käytössä oman sivuston hakemiston sisälle linkitettäessä.

Esimerkki selventäne asiaa. Kyseessä on Monilajisen liikuntakerhon etusivu, `index.php`. Sieltä sivulta pitäisi saada linkki sivulle `index.htm`, joka sijaitsee kansiossa `ohjaajat`. FTP-ohjelman kautta tarkasteltuna kansiorakenne näyttää tällaiselta.



Mikäli linkki kirjoitetaan absoluuttisesti, on kirjoitettava koko osoite:

```
<a href="http://www.monilaji.net/ohjaajat/index.htm">Ohjaajat</a>
```

Mikäli linkki kirjoitetaan suhteellisesti, on päästävä kansion sisään viittaamalla suoraan kansion nimeen (ohjaajat).

```
<a href="ohjaajat/index.htm">Ohjaajat</a>
```

Mikäli halutaan tehdä linkki toiseen suuntaan, on hakemistorakennetta kivuttava ylöspäin. Seuraavaksi tehdään linkki ohjaajat-kansiosta pääsivulle. Siihen tarvitaan merkkejä "../", jolla päästään yksi taso ylöspäin hakemistorakenteessa. Kaksi tasoa ylöspäin päästään kirjoittamalla "../../".

```
<a href="../../index.php">Etusivu</a>
```

4.2. Viittaus dokumentin sisälle

Varsinkin pitkissä dokumenteissa voi esiintyä tarvetta linkittää tiettyyn kohtaan dokumenttia. Silloin käytetään apuna ankkureita. Ankkuri määritellään seuraavasti:

```
<a name="ankkurin_nimi">Viitesana</a>
```

Siihen voidaan viitata seuraavasti:

```
<a href="tiedosto.html#ankkurin_nimi">Viittaus</a>
```

Pitkissä dokumenteissa on yleensä kohteliasta laittaa alkuun sisällysluettelo, jolloin turha sivujen rullaaminen ylös ja alas voidaan välttää. Myös aakkosellinen hakemisto on mahdollinen ankkureilla toteutettuna. Tässä vielä yksi esimerkki, jossa sivun yläreunassa on linkki, joka viittaa sivun alareunaan. Linkissä oleva # merkki tarkoittaa samaa dokumenttia eli sivulta on linkki samalle sivulle.

```
<BODY>
```

```
<a href="#eka">Eka linkki</a>
```

```
<H2>Eksosomaattinen oppiminen</H2>
```

```
<P>Fyysisten ja kielellisten välineitten avulla ihminen voi ylittää itsensä. Vuorovaikutuksemme kyseisten välineiden kanssa on keskeistä oppimisen kannalta. Yhdessä nämä erilaiset välineet muodostavat osan kulttuurisista voimavaroistamme. Jo pikainen silmäys omaan asuntoomme osoittaa kuinka tärkeäksi erilaiset fyysiset välineet ovat tulleet erilaisten toimintojen ylläpidossa ja erilaisten ongelmien ratkaisuisissa. Elämme autojen, puhelinten, CD-soittimien, mikroaaltouunien, televisioiden, kahvinkeitinien ja tietokoneiden maailmassa. (Säljö 2001, 72.) Nämä kaikki em. välineet sijaitsevat oman kehomme ulkopuolella, ne ovat siis eksosomaattisia.</P>
```

```
<P>Oppimisen suhteen onkin kysyttävä, että onko tekniikka ihmisen sisä- vai ulkopuolella? Toinen yhtä keskeinen kysymys on, että onko oppiminen aina samanlaista? Tekniikan kehityksessä meidän kommunikointimme on muuttunut, samalla tavalla myös oppiminen on muuttunut. Oppiminen eilen ja tänään eivät siis ole täysin samanlaisia tapahtumia. Onkin hyvä luoda katsaus taaksepäin, ja tarkastella sitä kuinka ihmiset ennen kommunikoivat ja oppivat. Yksi olennainen ero eilispäivän ja nykypäivän välillä on siinä, että aina ei ole ollut instituutioita nimeltä koulu. Kuitenkin myös ennen ihmiset kommunikoivat ja oppivat.</P>
```

<P>Ihminen on monessa suhteessa heikko ja rajoittunut olento. Hän ei ole erityisen voimakas, nopea eikä kestävä verrattuna muihin lajeihin. Lisäksi ihmisen muisti on rajallinen. Yksi psykologian historian merkittävimmistä tutkimuksista on Millerin tutkimus vuodelta 1956, jonka mukaan me kykenemme käsittelemään vain rajallisen määrän tietoa. Työmuistimme kapasiteetti on vain 7 +/-2, mikä on pieni määrä olennolle, joka sentään rakentaa tietoyhteiskuntaa. Muistimme rajallisuus ilmenee myös silloin, kun yritämme muistella mitä meidän tulisi tehdä seuraavalla viikolla. Olemme myös heikolla, kun joudumme päässä laskemaan kertolaskuja kolminumeroisilla luvuilla. (Säljö 1999, 3)</P>

<H3>Artefaktit ja mediointi</H3>

4.3. Target eli minne hyperlinkki avataan

Linkki voidaan avata selainäkymän päälle tai uuteen ikkunaan. Yleensä www-sivut rakennetaan siten, että sivuston linkit avautuvat aina kulloisenkin selainäkymän päälle. Jos sivulta on hyperlinkkejä sivuston ulkopuolelle, avautuvat ne uusiin selainikkunoihin. Valintoja on periaatteessa vain kaksi.

target=_blank (aukeaa uuteen ikkunaan)

Otavan Opisto

target=_top (aukeaa samaan selainäkymään)

Otavan Opisto

Jos jättää määrittelemättä targetin, uusi sivu avautuu entisen päälle.



Nauhoite 4: Linkitys

Kesto: 15:32

URL: http://youtu.be/tMWCv_AoKXI

Tehtävä 4

Tee www-sivu, jossa on linkit seuraaville sivuille: Sonera, DNA, Elisa. Tee kaikki linkit sellaisiksi, että ne avautuvat uuteen ikkunaan.

5. Listat

Listat ovat tehokkaita, kun halutaan esittää tietoa tiiviissä muodossa. Kyseessä on ikään kuin webin ranskalaiset viivat. Listat sopivat erittäin hyvin luettelonomaisen tiedon esittämiseen. Webissä listat lisäävät yleensä sivujen luettavuutta.

5.1. UL - järjestämätön lista

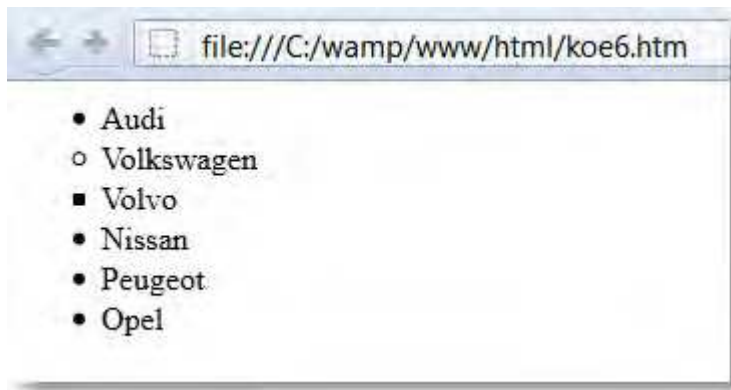
UL-tagilla saa aikaan järjestämättömän listan, jossa LI-tagin määritetyt kohdat erotetaan toisistaan. UL on lyhenne englannin kielen sanoista *unordered list*. LI on puolestaan lyhenne sanoista *list item* eli listaelementit.

```
<ul>
<li>Audi</li>
<li>Volkswagen</li>
<li>Volvo</li>
<li>Nissan</li>
<li>Peugeot</li>
<li>Opel</li>
</ul>
```



Listan tyyliin voi vaikuttaa attribuutilla type. Katso, kuinka pallukasta saadaan esimerkiksi neliö.

```
<ul>
<li type="disc">Audi</li>
<li type="circle">Volkswagen</li>
<li type="square">Volvo</li>
<li type="disc">Nissan</li>
<li type="disc">Peugeot</li>
<li type="disc">Opel</li>
</ul>
```



5.2. OL - järjestetty lista

Järjestetty lista saadaan aikaan tagilla . OL on lyhenne sanoista *ordered list*. Tässä tapauksessa pallukan tilalle tulee järjestysnumero.

```
<ol>
<li>Audi</li>
<li>Volkswagen</li>
<li>Volvo</li>
<li>Nissan</li>
<li>Peugeot</li>
<li>Opel</li>
</ol>
```



Tyyliä voi vaihtaa attribuutilla **type**. **Start** puolestaan kertoo, mistä numerosta tai kirjaimesta aloitetaan. Alla olevassa esimerkissä laitetaan numeroiden tilalle aakkoset ja lista aloitetaan kirjaimesta c.

```
<ol type="a" start="3">
<li>Audi</li>
<li>Volkswagen</li>
<li>Volvo</li>
<li>Nissan</li>
<li>Peugeot</li>
<li>Opel</li>
</ol>
```

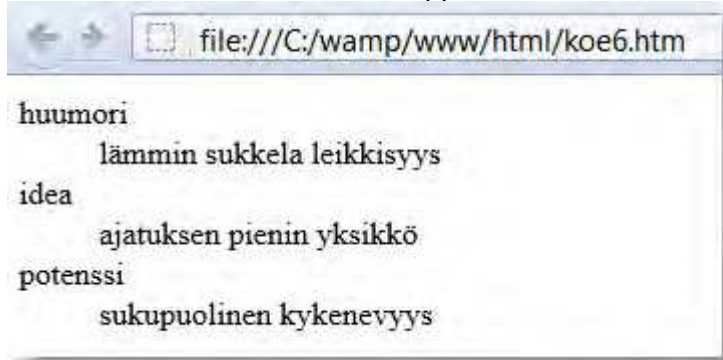


5.3. DL - määritelmäluettelo

DL-tagin (engl. *definition list*) määrittää määritelmäluettelon, jossa DT-tagilla ilmaistaan määriteltävä termi ja DD-tagilla määritelmä.

```
<dl>
<dt>huumori
<dd>lämmin sukkela leikkisyys
<dt>idea
<dd>ajatuksen pienin yksikkö
<dt>potenssi
<dd>sukupuolinen kykenevyys
</dl>
```

Selaimen kautta tarkasteltuna lopputulos on kuten alla olevassa kuvassa.



Nauhoite 5: Listat

kesto: 05:15

URL: <http://youtu.be/IWHsKVwFjZs>

Tehtävä 5

Tee HTML-sivu, jossa seuraavat webselaimet on listattu listaelementin avulla: Chrome, Firefox, IE, Safari ja Opera.

6. Kuvat

Kuvan lisääminen tekstin joukkoon käy IMG-tagilla, joka on lyhenne englannin kielen sanasta *image* (suomeksi kuva). SRC-attribuutti, joka on lyhenne sanasta *source*, kertoo kuvan sijainnin. Sijainti voi olla joko absoluuttinen tai relatiivinen. Olisi hyvä, jos IMG-tagisi sisältäisi SRC-attribuutin lisäksi vähintäänkin ALT-attribuutin. ALT:lla määritellään teksti, joka näytetään, jos kuvaa ei pystytä näyttämään. Tekstin pitäisi kuvata mahdollisimman hyvin kuvan sisältö. Suositeltavaa on myös kuvan koon määritteleminen.

```

```

Height ja width -attribuutit kertovat kuvan korkeuden ja leveyden. Tämä tieto on tärkeää graafisille selaimille, jotta ne osaavat varata kuville oikean kokoisen tilan ennen kuin kuva on haettu palvelimelta. ALT-attribuuttia tarvitaan, jotta sivuista tulisi esteettömät eli kyse on siitä, että kaikki voivat selailla sivuja.

Tässä kuva laitetaan tekstin oikealle puolelle komennolla align="right".

```

```



Kyborgit tulevat

Ihon alle sijoitettava mikrosiru voi pelastaa kuulon, näön tai liikkumiskyvyn. Ihmeiden aika ei siis ole ohi, sillä mikrosirujen avulla kuurot voivat jälleen erottaa ääniä, sokeat hahmottaa ääriä ja liikuntakyvyttömät ottaa askelia. Lisäksi mikrosirut mahdollistavat oven avaamisen kulkijalle tai vaikka valon sytyttämisen. Tulevaisuudessa sen avulla voidaan ohjata tietokonetta ajatuksen voimalla. (Kompuutteri 2000, 48.)

Professori Kevin Warwickia on leikkimielisesti nimetty maailman ensimmäiseksi kyborgiksi eli tietokoneen ja ihmisen yhteensulauksi. Hän kykeni käsivarteen sijoitetulla mikrosirun avulla mm. avaamaan ovia pelkästään ojentamalla kättään. Ohut siru oli vain 23 millimetrin pituinen ja 3 millimetrin levyinen. Warwickin käsivarressa ollut siru oli lasia. Sirun rikkoutumisen pelossa hän ei käyttänyt sitä kauan, mutta ehti silti todistaa tekniikan toimivaksi. (Kompuutteri 2000, 49-50)



Tässä kuva laitetaan tekstin vasemmalle puolelle komennolla align="left".

```

```

Kyborgit tulevat

Ihon alle sijoitettava mikrosiru voi pelastaa kuulon, näön tai liikkumiskyvyn. Ihmeiden aika ei siis ole ohi, sillä mikrosirujen avulla kuurot voivat jälleen erottaa ääniä, sokeat hahmottaa äärivirvoja ja liikuntakyvyttömät ottaa askelia. Lisäksi mikrosirut mahdollistavat oven avaamisen kulkijalle tai vaikka valon sytyttämisen. Tulevaisuudessa sen avulla voidaan ohjata tietokonetta ajatuksen voimalla. (Kompuutteri 2000, 48.)



toimivaksi. (Kompuutteri 2000, 49-50)

Professori Kevin Warwickia on leikkimielisesti nimetty maailman ensimmäiseksi kyborgiksi eli tietokoneen ja ihmisen yhteensulaumaksi. Hän kykeni käsivarteen sijoitetulla mikrosirun avulla mm. avaamaan ovia pelkästään ojentamalla kättään. Ohut siru oli vain 23 millimetrin pituinen ja 3 millimetrin levyinen. Warwickin käsivarressa ollut siru oli lasia. Sirun rikkoutumisen pelossa hän ei käyttänyt sitä kauan, mutta ehti silti todistaa tekniikan

Kuvan asemoinnin suhteen ovat käytössä myös vaihtoehdot: bottom, top ja absmiddle. Jos valitaan bottom, alkaa teksti kuvan alareunan tasalta. Jos valitaan top, alkaa teksti kuvan yläreunasta. Valinta absmiddle keskittää tekstin suhteessa kuvaan.



Kokeilussa: align="bottom".



Kokeilussa: align="top".



Kokeilussa: align="absmiddle".

Align-attribuutilla vaikutetaan kuvan sijoittumiseen tekstin suhteen. Border-attribuutti määrittää kuvan ympärillä olevan kehyksen koon. Linkkinä toimivan kuvan ympärille lisätään oletuksena kehys. Sillä tavoin käyttäjälle tehdään selväksi, että kuva toimii linkkinä.



Kyborgit tulevat

Ihon alle sijoitettava mikrosiru voi pelastaa kuulon, näön tai liikkumiskyvyn. Ihmeiden aika ei siis ole ohi, sillä mikrosirujen avulla kuurot voivat jälleen erottaa ääniä, sokeat hahmottaa äärivirvoja ja liikuntakyvyttömät ottaa askelia. Lisäksi mikrosirut mahdollistavat oven avaamisen kulkijalle tai vaikka valon sytyttämisen. Tulevaisuudessa sen avulla voidaan ohjata tietokonetta ajatuksen voimalla. (Kompuutteri 2000, 48.)

Professori Kevin Warwickia on leikkimielisesti nimetty maailman ensimmäiseksi kyborgiksi eli tietokoneen ja ihmisen yhteensulaumaksi. Hän kykeni käsivarteen sijoitetulla mikrosirun avulla mm. avaamaan ovia pelkästään ojentamalla kättään. Ohut siru oli vain 23 millimetrin pituinen ja 3 millimetrin levyinen. Warwickin käsivarressa ollut siru oli lasia. Sirun rikkoutumisen pelossa hän ei käyttänyt sitä kauan, mutta ehti silti todistaa tekniikan toimivaksi. (Kompuutteri 2000, 49-50)

Huom! Osa uusimmista selaimista, kuten Firefox, jättää piirtämättä tuon sinisen kehyksen. Helpoimmin hoidat kuitenkin ongelman määrittelemällä, että kehys on nolla pikseliä.

```
<a href="http://www.jarisarja.fi"></a>
```

Hspace ja **vspace**-attribuuteilla määritetään kuvan ympärille jätettävä tila. VSPACE kertoo ylä- ja alapuolelle, HSPACE oikealle ja vasemmalle puolelle jätettävän tilan.

```
<a href="http://www.jarisarja.fi"></a>
```

Kyborgit tulevat

Ihon alle sijoitettava mikrosiru voi pelastaa kuulon, näön tai liikkumiskyvyn. Ihmeiden aika ei siis ole ohi, sillä mikrosirujen avulla kuurot voivat jälleen erottaa ääniä, sokeat hahmottaa äärivirvoja ja liikuntakyvyttömät ottaa askelia. Lisäksi mikrosirut mahdollistavat oven avaamisen kulkijalle tai vaikka valon sytyttämisen. Tulevaisuudessa voidaan ohjata tietokonetta ajatuksen voimalla. (Kompuutteri 2000, 48.)

Professori Kevin Warwickia on leikkimielisesti nimitelty maailman ensimmäiseksi kyborgiksi eli tietokoneen ja ihmisen yhteensulaumaksi. Hän kykeni käsivarteen sijoitetulla mikrosirun avulla mm. avaamaan ovia pelkästään ojentamalla kättään. Ohut siru oli vain 23 millimetrin pituinen ja 3 millimetrin levyinen. Warwickin käsivarressa ollut siru oli lasia. Sirun rikkoutumisen pelossa hän ei käyttänyt sitä kauan, mutta ehti silti todistaa tekniikan toimivaksi. (Kompuutteri 2000, 49-50)

6.1. Kompastuskiviä

Kun kuvasta tehdään linkki, kuvan ympärille tulee sininen kehys. Ongelman voi ratkaista tavalla, joka kerrottiin jo edellä eli merkitsemällä, että on border=0. Kuvaan lisätty linkki voisi olla esimerkiksi tällainen:

```
<a href="http://sarja.internetix.fi" target="_blank"></a>
```

Toinen kompastuskivi liittyy hakemistorakenteeseen. Voi käydä niin, että kun sivut siirretään nettiin huomataan, että kaverin koneella kuvat eivät näy, vaikka omalla koneella ne näkyvät ihan normaalisti. Tällaisessa tapauksessa on yleensä käytetty editoria, joka on rakentanut polun oman koneen hakemistoon. Virheen huomaa siitä, että viittaus on omalle koneelle eli se alkaa "file:///".

```

```

Kolmas kompastuskivi liittyy myös hakemistorakenteeseen ja tilanteeseen, jossa kuva ei näy sivulla. Jos esimerkiksi kissa.jpg on samassa kohtaa hakemistorakennetta (samassa kansiossa) kuin sivu, linkitys tehdään näin.

```

```

Jos kuva onkin saman kansion kuvat-alakansiossa, linkitys tehdään näin.

```

```

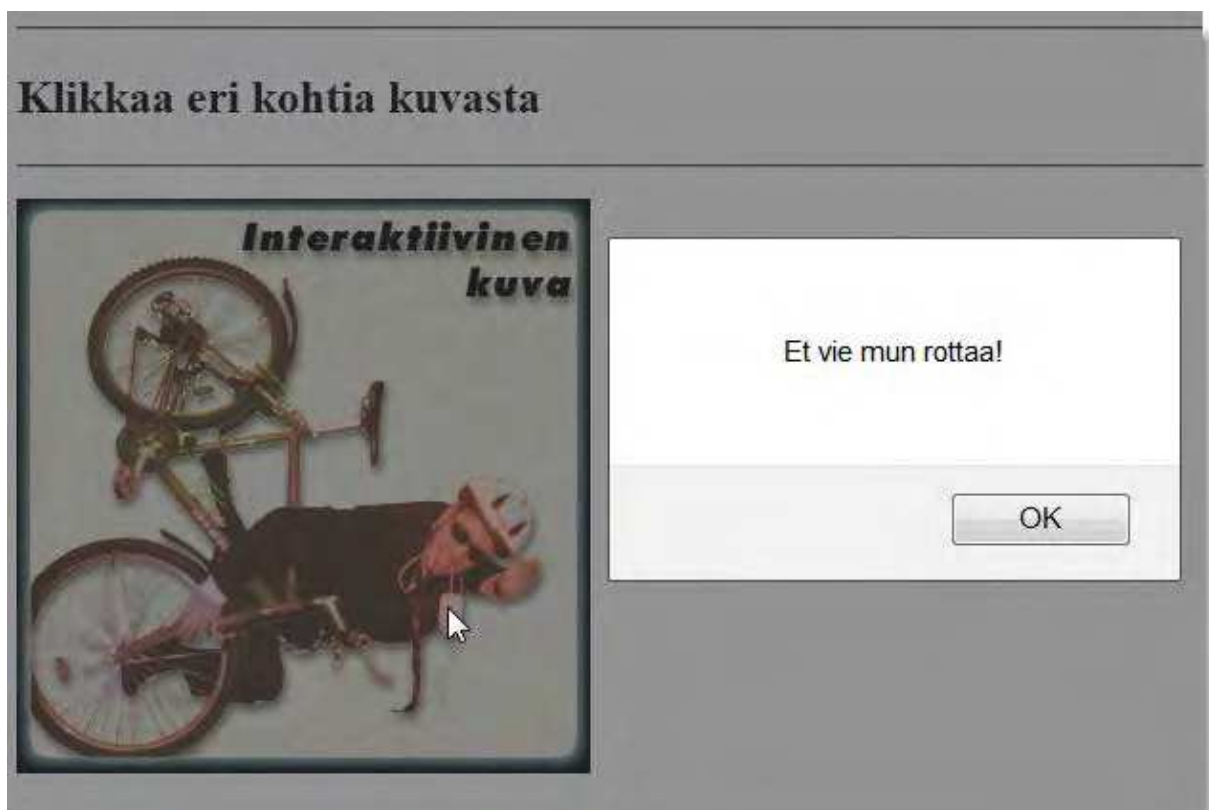
Jos kuva puolestaan on hakemistorakenteessa yhden portaan verran ylempänä kuvat-kansiossa, niin linkitys tehdään näin.

```

```

6.2. Interaktiivinen kuva

Aktiivinen kuva tai kosketusherkkä kuva (*image map*) on websivuun liitetty kuva, johon on upotettu alueita, jotka toimivat linkkeinä. Näitä alueita sanotaan joskus "kuumiksi alueiksi", "*hot spots*". Interaktiivisen kuvan avulla saadaan aikaan näyttäviä ja havainnollisia kokonaisuuksia. Tässä alla yksi esimerkki. Kun kuvassa näkyvää hiirtä napsautetaan, avautuu ikkuna, joka sisältää viestin "Et vie mun rottaa". Tällaista kuvakarttaa voi hyödyntää esimerkiksi joulukalenterin luokkuja tehdessä. Silloin kuvassa olisi 24 luokkua, joita voisi napsauttaa.



Interaktiivista kuvaa tehdessä on tehtävä aina tietyt perusasiat:

1. Luodaan kuva, josta halutaan tehdä interaktiivinen.
2. Luodaan html-tiedosto, johon kuva sijoitetaan.
3. Määritellään kuvaan alueet, joista tehdään linkkejä.
4. Määritellään kohteet, johon alueet viittaavat.
5. Nimetään koodipätkä, joka viittaa tiettyyn kuvaan.

Käytännössä interaktiivisen kuvan voi luoda helpoiten jollakin siihen tarkoitukseen tehdyllä ohjelmalla.

- [Image Mapper](#) (Windows)
- [Xmap](#) (Windows)

- [Free Imagemapper](#) (Windows)
- [Image Mapper 3.0](#) (Mac)
- [Map Spinner](#) (Mac)
- [kimagemapeditor](#) (Ubuntu KDE)

Ohjelmasta saat valmiin koodin sijoitettavaksi sivulle. Koodi sisältää seuraavat osatekijät.

- kuvan liittäminen sivulle (img src -tagi)
- usemap (imagemapin eli interaktiivisen kuvan nimi)
- map name (viittaus interaktiiviseen kuvaan)
- area (muoto ja koordinaatit)

Tältä näyttää esim. erään joulukalenterin interaktiivinen kuva, jossa klikataan luokkuja.

```

<map name="kalenteri">
<area shape="rect" coords="265,540,213,499" href="1.php" target="_BLANK">
<area shape="rect" coords="69,540,22,504" href="2.php" target="_BLANK">
<area shape="rect" coords="81,258,26,220" href="3.php" target="_BLANK">
<area shape="rect" coords="161,231,113,192" href="4.php" target="_BLANK">
<area shape="rect" coords="60,315,11,274" href="5.php" target="_BLANK">
<area shape="rect" coords="712,248,660,211" href="6.php" target="_BLANK">
<area shape="rect" coords="651,299,597,261" href="7.php" target="_BLANK">
<area shape="rect" coords="723,302,670,264" href="8.php" target="_BLANK">
<area shape="rect" coords="206,288,157,253" href="9.php" target="_BLANK">
<area shape="rect" coords="289,316,242,283" href="10.php" target="_BLANK">
<area shape="rect" coords="144,332,94,293" href="11.php" target="_BLANK">
<area shape="rect" coords="164,535,110,493" href="12.php" target="_BLANK">
<area shape="rect" coords="431,255,378,215" href="13.php" target="_BLANK">
<area shape="rect" coords="454,325,396,285" href="14.php" target="_BLANK">
<area shape="rect" coords="670,362,619,323" href="15.php" target="_BLANK">
<area shape="rect" coords="633,421,581,382" href="16.php" target="_BLANK">
<area shape="rect" coords="334,522,278,484" href="17.php" target="_BLANK">
<area shape="rect" coords="421,528,365,491" href="18.php" target="_BLANK">
<area shape="rect" coords="534,433,470,394" href="19.php" target="_BLANK">
<area shape="rect" coords="435,472,383,436" href="20.php" target="_BLANK">
<area shape="rect" coords="728,507,670,465" href="21.php" target="_BLANK">
<area shape="rect" coords="753,447,698,410" href="22.php" target="_BLANK">
<area shape="rect" coords="305,373,261,341" href="23.php" target="_BLANK">
<area shape="rect" coords="537,379,463,328" href="24.php" target="_BLANK">
</map>
```



Nauhoite 6: Interaktiiviset kuvat

kesto: 05:56

URL: <http://youtu.be/pbb6QstXNrk>

Tehtävä 6

Kirjoita koodi, joka tekee seuraavaa:
sijoita etusivulle (etusivu.htm) kuva jari.jpg, joka on kansiossa jari
tee kuvasta linkki ilman kehystä sivulle <http://www.otavanopisto.fi>

Lataa ja pura [zip-paketti](#), jossa tiedostot sijaitsevat. Muokkaa tiedostoa etusivu.htm, jotta kuva näkyisi ja linkki toimisi. Tallenna lopuksi muokkaamasi koodi (etusivu.htm).

Hakemistorakenne näkyy alla.



7. Viivat ja rivinvaihto

HTML-koodilla voidaan piirtää vaakasuoria viivoja. Varsinkin webin alkuaikoina viivat olivat tärkeitä, koska niiden avulla voitiin sommitella sisältöä. Nykyään viivoja näkee eniten vieras-kirjoissa.

Koodina viivan tekeminen on yksinkertainen, koska siihen riittää pelkkä tagi `<hr>`. Tagi on lyhenne englannin kielen sanoista *horizontal rule*.

Viivalle voidaan antaa lisämääreitä, jotka näkyvät seuraavassa taulukossa.

attribuutti (lisämääre)	toiminto
noshade	ei kolmiulotteisuutta
size	viivan korkeus
width	viivan pituus
align	kuinka viiva asemoituu (oikea / vasen)
color	viivan väri

Lopuksi esimerkki erilaisista viivoista

Viiva 1

```
<hr noshade size="10">
```

Viiva 2

```
<hr size="8">
```

Viiva 3

```
<hr color="#00FF00">
```

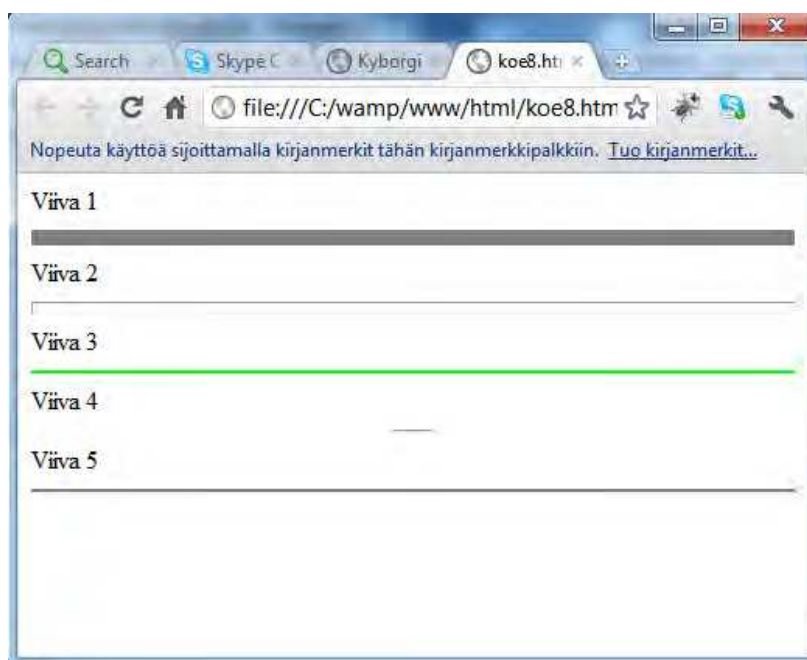
Viiva 4

```
<hr height="10" width="23" align="center">
```

Viiva 5

```
<hr noshade>
```

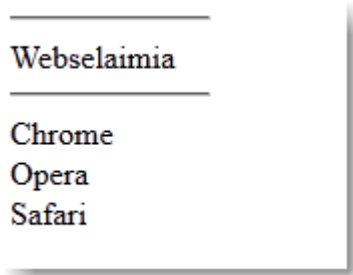
Lopputulokset näin (selaimen ikkuna on kavennettu ja kuvaa pienennetty).



Joskus sivuille tarvitaan pelkästään rivinvaihto ilman viivaa. Se hoituu tagilla **
**, joka on lyhenne englannin kielisestä sanasta *break*.

Tehtävä 7

Tee HTML-sivu, joka on samannäköinen kuin alla olevassa kuvassa.



8. Lomakkeet

Lomakkeet antavat käyttäjälle mahdollisuuden lähettää tietoa www-palvelimelle. Lomake vaatii toimiakseen ohjelman, joka käsittelee lomakkeella välitetyn tiedon. Lomakkeen tietoja voi lähettää myös suoraan sähköpostiin komennolla mailto, mutta se tapa ei ole suositeltava, koska:

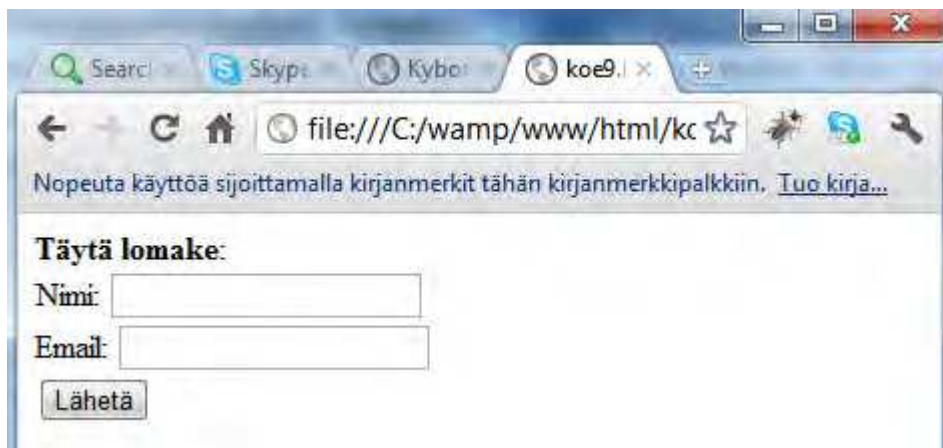
- sähköpostiosoite lukee koodissa eli roskapostia on tiedossa
- lomakkeen voi lähettää vain koneelta, johon on asennettu sähköpostiohjelma ja sähköpostitili
- lomakkeen tiedot tulevat sähköpostiisi huonosti muotoiltuina

Paremmista menetelmistä sen verran, että lomake kannattaa tehdä joko [PHP:llä](#) tai lähettää se cgi:n hoidettavaksi. Usein webhotelli tarjoaa valmiita ratkaisua kyseiseen ongelmaan.

8.1. FORM - Lomake

HTML-lomake luodaan FORM-tagien sisään. ACTION kertoo lomakkeen käsittelevän ohjelman osoitteen. METHOD määrää tavan, jolla tieto lähetetään.

```
<form action="vastaanota.php" method="post">
<p>
<b> Täytä lomake</b>:<br>
Nimi: <input type="text" name="nimi"><br>
Email: <input type="text" name="email"><br>
<input type="submit" value="Lähetä">
</p>
</form>
```



8.2. INPUT - Syöttökenttä, nappi, yms.

INPUT-koodilla tehdään lomakkeelle syöttökenttiä. NAME-attribuutti nimeää kentän. Selain lähettää kenttien nimet tietojen mukana, jotta www-palvelimella toimiva ohjelma osaa tunnistaa tiedot. ALIGN vaikuttaa kenttien sijoittumiseen. Muut parametrit riippuvat TYPE-parametrasta.

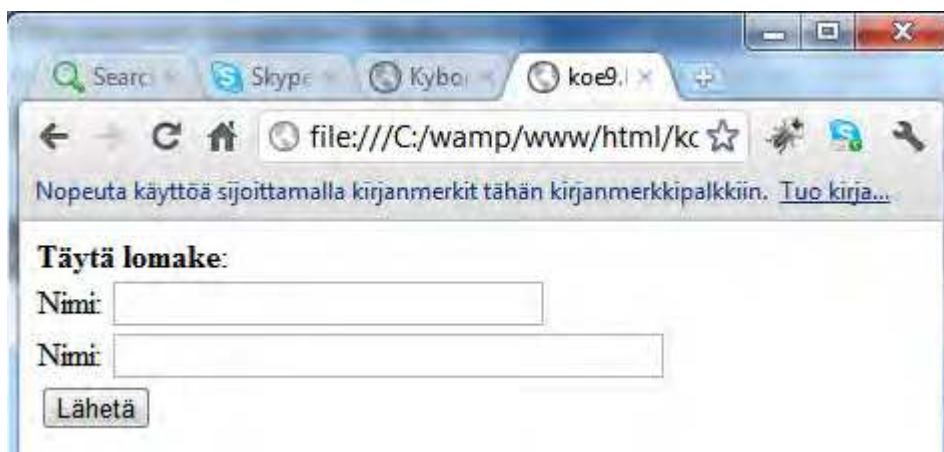
TYPE=text

Tekee tekstinsyöttökentän (ks. alla oleva kuva). Size määrittää kentän pituuden eli sen, kuinka iso laatikko piirtyy selaimen. Maxlength määrittää kentän maksimipituuden.

```

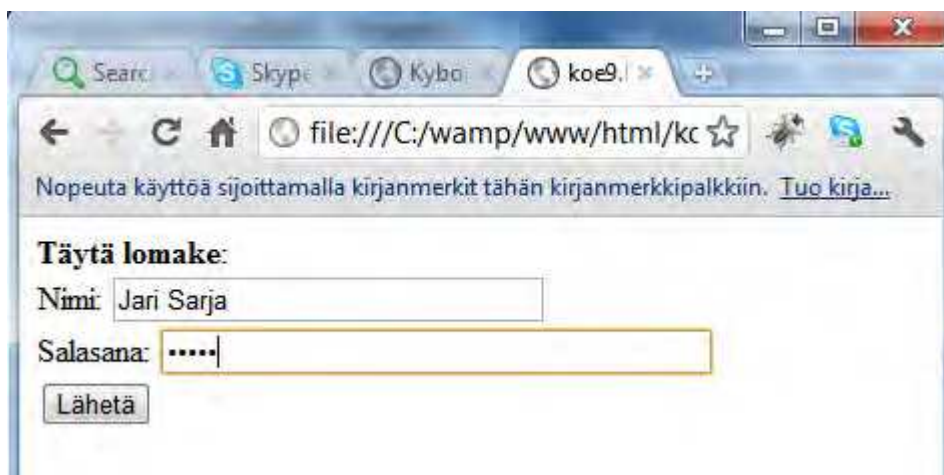
<form action="vastaanota.php" method="post">
<p>
<b> Täytä lomake</b>:<br>
Nimi: <input type="text" name="nimi" size="30"><br>
Nimi: <input type="text" name="email" size="40"><br>
<input type="submit" value="Lähetä">
</p>
</form>

```



TYPE=password

Tekee tekstinsyöttökentän, jossa syötetty teksti näkyy pallukoina tai tähtinä riippuen selaimesta.



TYPE=checkbox

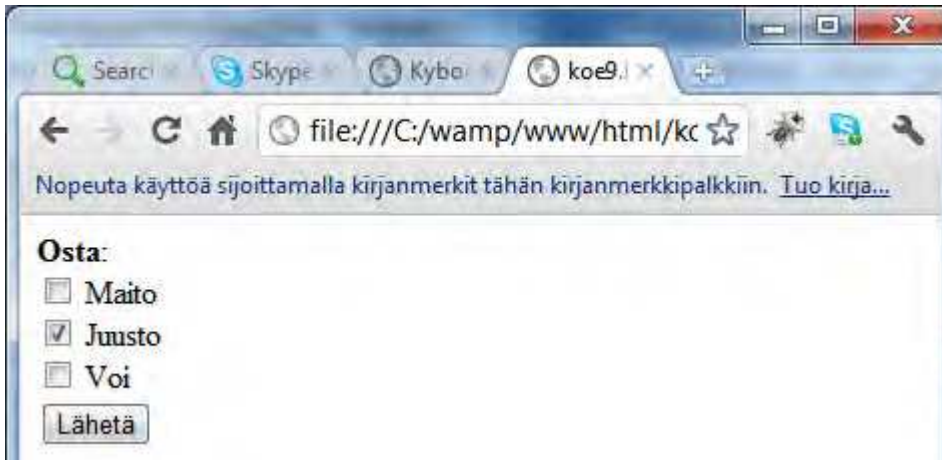
Tekee valintaruudun, joka voi olla joko valittu tai ei. CHECKED tekee ruudusta oletusvalinnan eli ruudussa on valmiina ruksi.

```

<form action="vastaanota.php" method="post">
<p>
<b> Osta</b>:<br>
<input type="checkbox" name="maito" value="Maito"> Maito<br>
<input type="checkbox" name="juusto" value="Juusto" checked> Juusto<br>
<input type="checkbox" name="voi" value="Voi"> Voi<br>

```

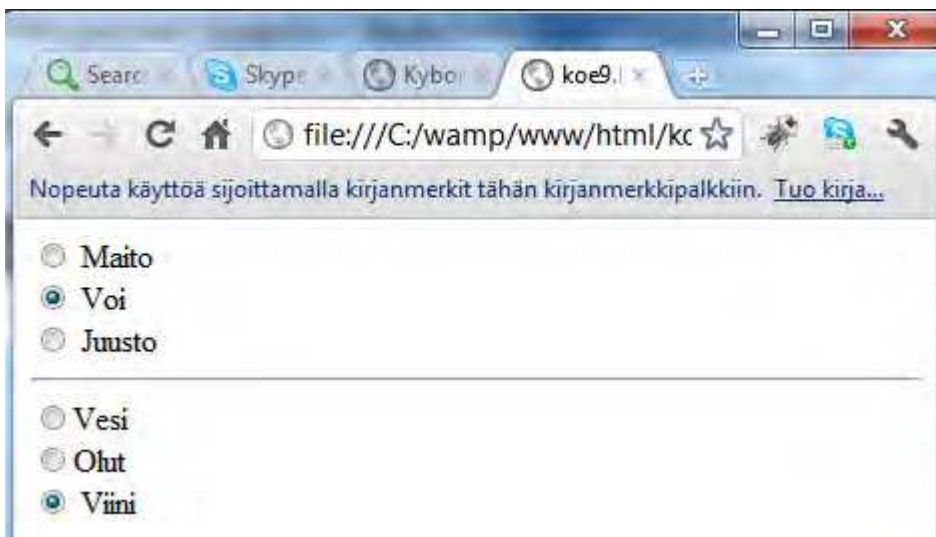
```
<input type="submit" value="Lähetä">
</p>
</form>.
```



TYPE=radio

Tekee valintanapin. Napit ovat aina ryhmissä ja vain yksi ryhmän napeista voi olla valittuna. Samaan ryhmään kuuluvilla napeilla pitää olla sama NAME-määre ja eri VALUE-määre. Checked tarkoittaa oletusvalintaa. Tässä esimerkissä on valmiiksi valittuina Juusto ja Viini.

```
<input type="radio" name="ryhma1" value="Maito"> Maito<br>
<input type="radio" name="ryhma1" value="Voi" checked> Voi<br>
<input type="radio" name="ryhma1" value="Juusto"> Juusto
<hr>
<input type="radio" name="ryhma2" value="Vesi">Vesi<br>
<input type="radio" name="ryhma2" value="Olut">Olut<br>
<input type="radio" name="ryhma2" value="Viini" checked> Viini<br>
```



TYPE=submit

Tekee painikkeen, jota painamalla voi lähettää lomakkeen tiedot. Lomakkeella voi olla useita lähetä-painikkeita, mutta niillä tulee olla eri NAME-määre.

```
<input type="submit" value="Lähetä">
```

TYPE=reset

Tekee painikkeen, jota painamalla voi tyhjentää lomakkeen.

```
<input type="reset" value="Tyhjennä lomake">
```

TYPE=hidden

Tekee piilotetun kentän. Sitä tarvitaan, kun halutaan viedä tietoa lomakkeenkäsittelyohjelmalle ilman, että tietoa näkyy selaimessa (näkyä tosin lähdekoodissa).

8.3. SELECT - Valintalista

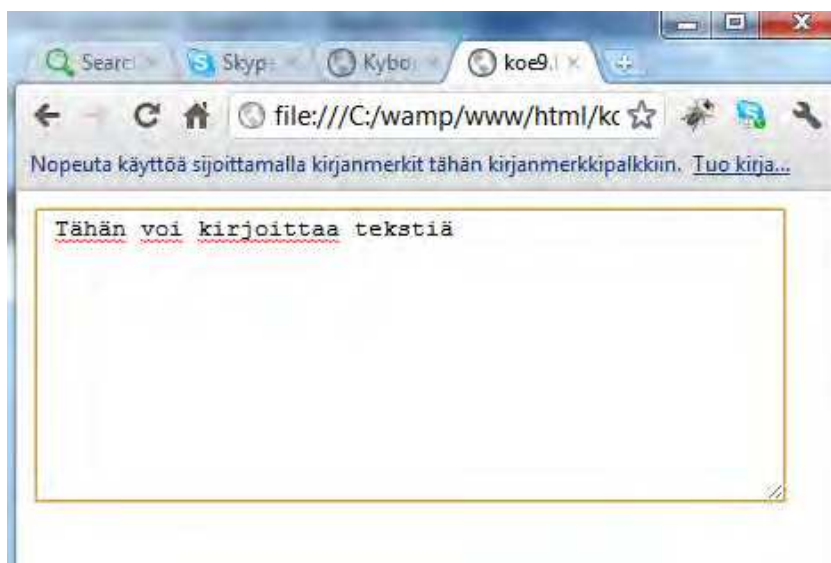
SELECT-koodilla saa aikaiseksi valintalistan, jossa on useita vaihtoehtoja, joista käyttäjä voi valita yhden tai useamman. Valintalistan vaihtoehdot määritellään OPTION-koodein. NAME määrittää kentälle nimen, SIZE kerralla näkyvien rivien lukumäärän ja MULTIPLE mahdollistaa usean vaihtoehdon valitsemisen. VALUE-attribuutilla määrätään lähetettävän vaihtoehdon arvo.

```
<FORM>  
<SELECT NAME=vaihtoehdot>  
<OPTION VALUE=1>Vaihtoehto 1  
<OPTION VALUE=2>Vaihtoehto 2  
<OPTION VALUE=3>Vaihtoehto 3  
</SELECT>  
</FORM>
```

8.4. TEXTAREA - Tekstinsyöttökenttä

Tekee tekstinsyöttökentän, jolle voi syöttää useita rivejä tekstiä. NAME kertoo kentän nimen, ROWS rivien lukumäärän ja COLS kentän leveyden. Koodin sisälle kirjoitettu teksti näkyy tekstikentässä.

```
<TEXTAREA NAME="tekstii", ROWS=10, COLS=50> </TEXTAREA>
```



8.5. Esimerkki

Alla esimerkki valmiista palautelomakkeesta.

Search Skype Kybo koe9,1 x

file:///C:/wamp/www/html/kc

Nopeuta käyttöä sijoittamalla kirjanmerkit tähän kirjanmerkkipalkkiin. [Tuo kirja...](#)

Palautteen aihe:
kysyttävää

Nimeni:
Jari Sarja

Sähköpostiosoitteeni:
jari.sarja@otavanopisto.fi

Palaute:
Saavatko oppilaat omat kannettavat tänä vuonna?

lähetä

Ja koodina sama esimerkki.

```
<form action="palaute.php" method="post">
```

```
<p><font face="Arial"><strong>Palautteen aihe:</strong><br>
```

```
<select name="palautetta" size="1">
```

```
<option>opetussuunnitelmat </option>
```

```
<option>opetusharjoittelu</option>
```

```
<option>kodin ja koulun yhteistyö</option>
```

```
<option>idea tai aloite</option>
```

```
<option>kysyttävää</option>
```

```
<option>muu aihe</option>
```

```
</select></font></p>
```

```
<p><font face="Arial"><strong>Nimeni:</strong><br>
```

```
<input type="text" size="25" name="nimi"><br>
```

```
<strong>Sähköpostiosoitteeni:</strong><br>
```

```
<input type="text" size="25" name="email"><br>
```

```
<strong>Palaute:</strong><br>
```

```
<textarea name="palautteeni" rows="4" cols="40"></textarea>
</font></p>
<p><font face="Arial"><input type="submit" value="lähetä">
<input type="reset" value="tyhjennä"> </font></p>
</form>
```

8.6. Lomakkeen viimeistely

HTML-lomaketta voidaan vielä kehittää ulkonäön, käytettävyyden ja esteettömyyden osalta.

8.6.1. Label for (etiketit)

Lisätään lomakkeen esteettömyyttä ja käytettävyyttä etikettien avulla.

```
<label for="nimi">Nimi:</label>
<form id="form" name="form" method="post" action="palaute.php">

<label for="nimi">Nimi:</label><br>
<input type="text" name="nimi" id="nimi" /> <br>

<label for="sposti">Sähköposti:</label><br>
<input type="text" name="sposti" id="sposti" /> <br>

<label for="viesti">Viesti:</label><br>
<textarea name="viesti" id="viesti" cols="45" rows="5"></textarea> <br>

<input type="submit" name="laheta" id="laheta" value="Lähetä palaute" />
</form>
```

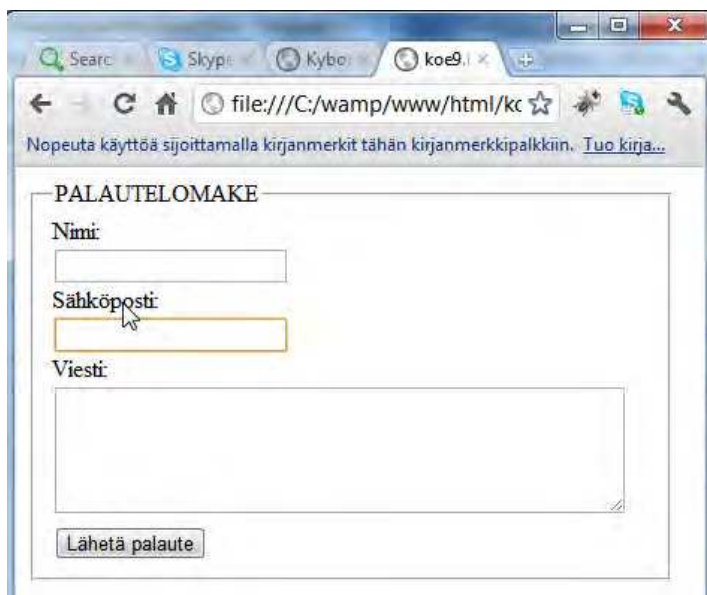
Nyt tarvitsee klikata vain etikettiä eli esimerkiksi tekstiä Sähköposti, niin kursori on valmiina oikeassa paikassa ja kyseinen kenttä aktivoituu.



8.6.2. Fieldset ja legend

Laittamalla mukaan fieldsetin ja legendin parannat esteettömyyttä ja samalla vaikutat ulkoasuun. Fieldset tarkoittaa lomake-elementtien ryhmittelyä eli koko ryhmä niputetaan yhteen kehyksellä. Legend tarkoittaa elementtiryhmän otsikkoa.

```
<form id="form" name="form" method="post" action="palaute.php">  
<fieldset style="width: 400px;">  
<legend>PALAUTELOMAKE</legend>  
  
<label for="nimi">Nimi:</label><br />  
<input type="text" name="nimi" id="nimi" /> <br />  
<label for="sposti">Sähköposti:</label><br />  
<input type="text" name="sposti" id="sposti" /> <br />  
<label for="viesti">Viesti:</label><br />  
  
<textarea name="viesti" id="viesti" cols="45" rows="5"></textarea> <br />  
  
<input type="submit" name="laheta" id="laheta" value="Lähetä palaute" />  
</fieldset>  
</form>
```



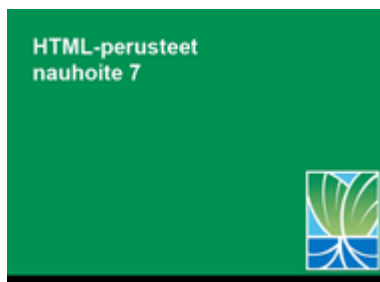
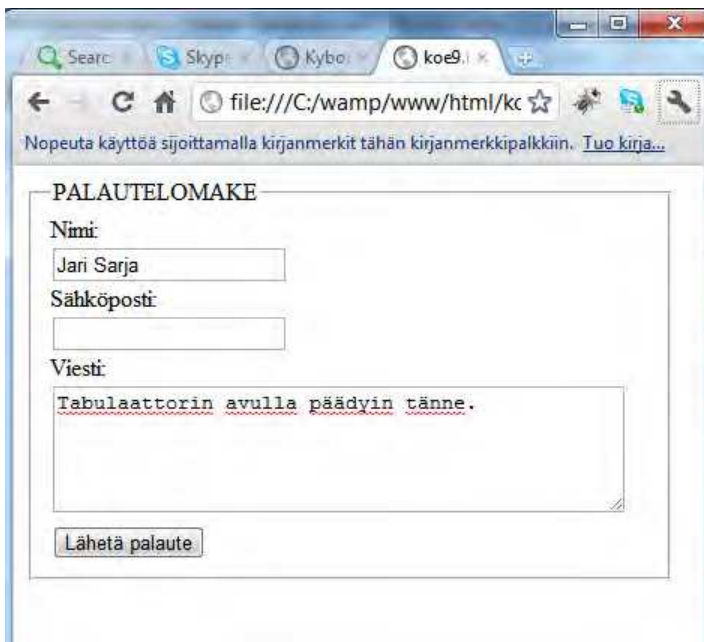
8.6.3. Tabindex eli tabulaattori

Tabulaattori on näppärä, ja sen saa toimimaan myös webissä. Käsky on **tabindex**. Ideana on se, että pääset seuraavaan kenttään napsauttamalla tab-näppäintä (tabulaattori). Se säästää aikaa, koska ei tarvitse hiiren avulla siirtää osoitinta seuraavaan kenttään. Seuraa-

vassa esimerkissä lomake laitetaan toimimaan siten, että ensin on nimi, sitten viesti ja vasta viimeisenä sähköposti.

```
<form id="form" name="form" method="post" action="palaute.php">
<fieldset style="width: 400px;">
<legend>PALAUTELOMAKE</legend>

<label for="nimi">Nimi:</label><br />
<input type="text" name="nimi" id="nimi" tabindex="1"/> <br />
<label for="sposti">Sähköposti:</label><br />
<input type="text" name="sposti" id="sposti" tabindex="3" /> <br />
<label for="viesti">Viesti:</label><br />
<textarea name="viesti" id="viesti" tabindex="2" cols="45" rows="5" wrap="virtual">
</textarea> <br />
<input type="submit" name="laheta" id="laheta" value="Lähetä palaute" />
</fieldset>
</form>
```



Nauhoite 7: Palautelomake

kesto: 15:50

URL: <http://youtu.be/bFOcisWkbi8>

Tehtävä 8

Kirjoita HTML:llä seuraavanlainen lomake:

PALAUTELOMAKE

Nimi:

Sähköposti:

Palautteen laji:

Risuja Ruusuja

Palaute:

Lähetä palaute

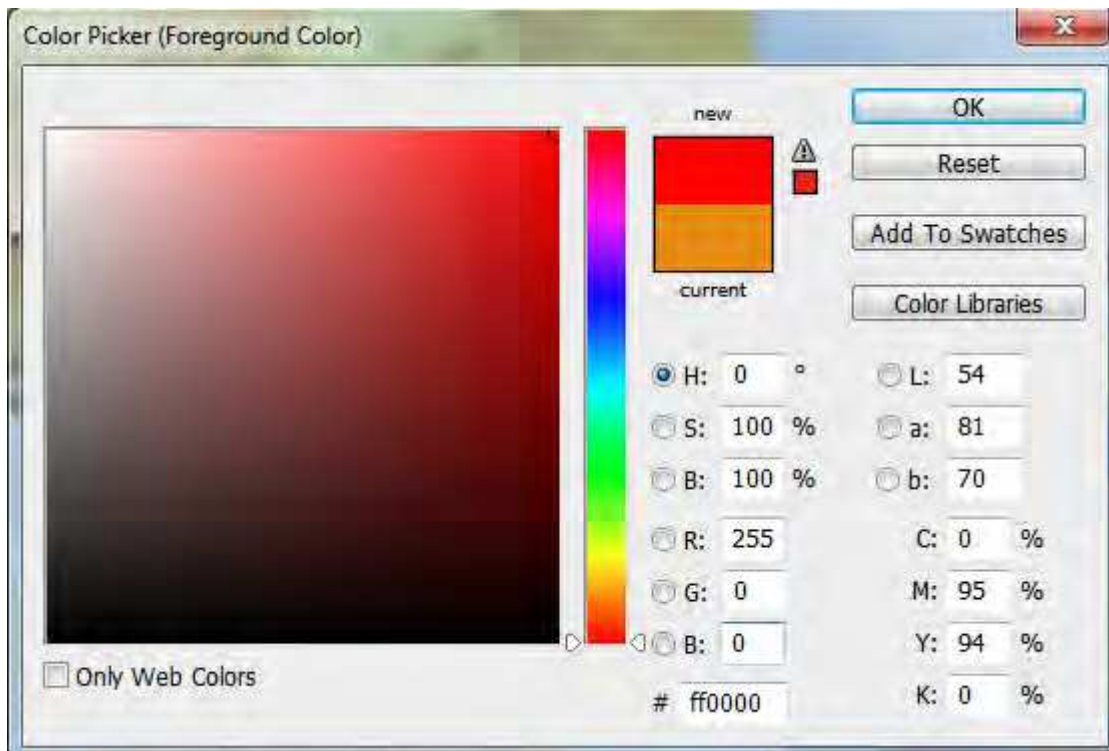
9. Ulkoasun muotoilu

HTML:n avulla voidaan tehdä myös graafista taittoa eli vaikuttaa sivuston ulkoasuun. HTML:n avulla sivuille saadaan perusilme, kuten esimerkiksi sivuston koko ja värimaailma. HTML:n avulla sivulle voidaan sijoittaa myös muun muassa yläpalkki ja navigaatio.

9.1. Värit HTML:ssä

Värikoodi ilmaistaan HTML-sivulla heksadesimaalilukuna. Lisäksi värin eteen tulee risuaita #. Kuvankäsittelyohjelmissa värit määritellään RGB-järjestelmän avulla (Red-Green-Blue). Kaksi ensimmäistä lukua kuvaavat punaista väriä, kaksi seuraavaa vihreää ja kaksi viimeistä kuvaavat sinistä. Värit muodostetaan sotkemalla keskenään kolmea edellä mainittua väriä. Kuvankäsittelyohjelmassa suurin arvo kullekin värille on 255 ja pienin on 0. Heksadesimaalissa pienin luku on 0, mutta suurin on F.

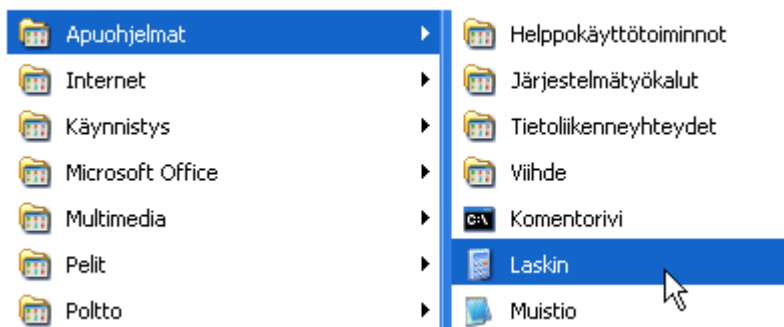
Tässä esimerkki siitä, kuinka Adoben PhotoShop -ohjelma ilmoittaa värit sekä RGB-arvoina että heksadesimaalilukuina. Vasemmassa sarakkeessa näkyvät punaisen värin RGB-arvot eli R=255, G=0 ja B=0. Sama näkyy heksadesimaalina risuaidan kohdalla eli # = ff0000.



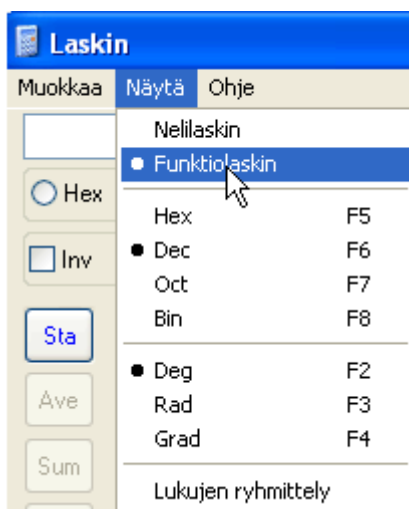
Tarkastellaan vielä värejä taulukon avulla esimerkkien kera.

Väri	RGB-väri	Heksadesimaali	Esimerkki
valkoinen	R=255 G=255 B=255	#FFFFFF	<body bgcolor="#FFFFFF"> Sivun taustaväriksi valkoinen
punainen	R=255 G=0 B=0	#FF0000	 Fontin väriksi punainen
sininen	R=000 G=000 B=255	#0000FF	 Fontin väriksi sininen
musta	R=000 G=000 B=000	#000000	 Fontin väriksi musta

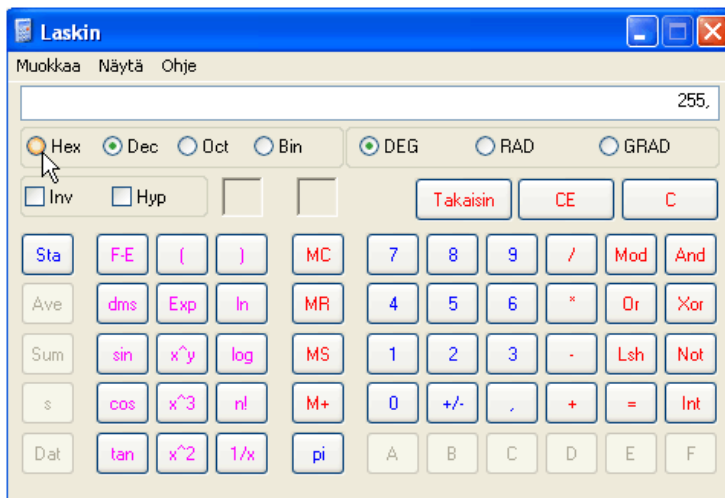
Kymmenlukujärjestelmästä voi olla hankala muuttaa lukuja heksadesimaaliluvuiksi päässä laskien. Onkin syytä muistaa, että tietokoneen laskimesta on apua: Avaa aluksi laskin.



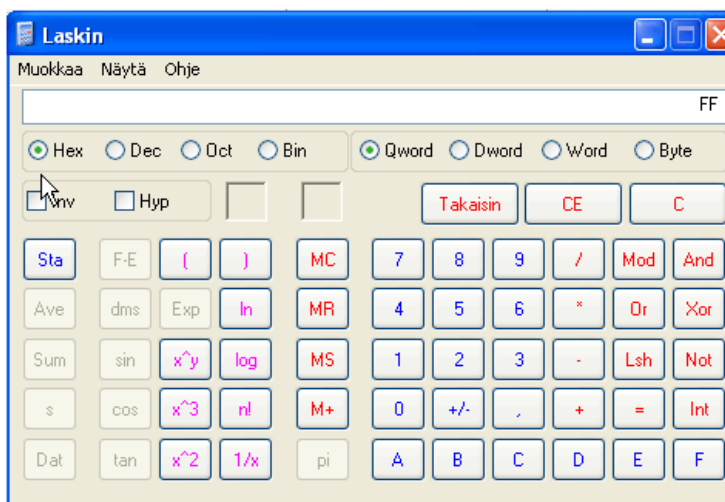
Valitse "Funktio-laskin" (Näytä > Funktio-laskin).



Kirjoita RGB-arvo.



Paina painiketta "Hex"



RGB-arvo 255 on heksadesimaalilukuna siis FF.

HUOM! Windows 7:ssä sama muunnos onnistuu, kun valitset ohjelmointilaskimen.

On myös ohjelmia, jotka ilmoittavat värikoodin mistä tahansa (klikkaat vaan kyseistä kohtaa). Yksi suositeltava vaihtoehto Windowsin käyttäjille on ilmaisohjelma [Hex Color Finder 3.0](#). Macintoshille löytyy ainakin ohjelma nimeltä [Color Snapper](#), mutta se on maksullinen. Tosin myös Firefoxiin saa asennettua ohjelmia, joiden avulla värikoodin saa helposti selvitettyä. Kannattaa kokeilla hakusanaa color picker.



9.2. Kehykset

"Kehykset ("freimit", engl. frames) ovat selainikkunan osia, jotka toimivat joissakin suhteissa itsenäisinä ali-ikkunoina. Kehyksen sisällä näkyy yleensä HTML-dokumentti, mutta siellä voi olla myös kuva, pelkkä tekstitiedosto tms. Kehyksen sisällitys voi olla vieritettävissä (skrollattavissa) ylös ja alas, kenties myös vaakasuunnassa, kuten normaali dokumentti normaalissa selainikkunassa on. Kehykset ovat suorakulmaisia: ikkuna voidaan jakaa kahteen tai useampaan suorakulmaiseen osaan vaakasuunnassa tai pystysuunnassa. Näin saatuja osia voi vielä jakaa edelleen." ([Korpela.](#))

Korpelan mukaan kehysidea lienee peräisin siitä, että useissa ohjelmissa on mahdollista katsella isohkoa dokumenttia siten, että vasemmalla on kapeassa ali-ikkunassa sisällysluettelo ja oikealla isommassa ali-ikkunassa kulloinkin katsottavan osan sisältö. Seuraavassa on tästä esimerkkikuva, jossa Acrobat Readerillä luetaan kirjaa [M niinkuin matematiikka](#).

The screenshot shows the Acrobat Reader interface. On the left, there is a sidebar titled "Kirjanmerkit" (Bookmarks) containing a table of contents with expandable items: Taustat, Reaaliluvut, Luku pii, Neperin luku e, Summa ja tulo, Keskiarvo, Murtoluvut, Alkutekijät, Lukujärjestelmät, Kompleksiluvut, Potenssi, Juuret, Polynomit, Polynomien tekijöihin jako, Yhtälöt, Polynomiyhtälöt, Juuriyhtälöt, Itseisarvoyhtälöt, Transkendenttiyhtälöt, Yhtälöryhmät, and Epäyhtälöt. The main window displays the first page of the book "Taustat" (1/1). The page content includes an introduction to the book, a list of authors, and a preface. The text is in Finnish and discusses the history and purpose of the book, which is a collection of articles from the "MaTa" project. The page number "1/1" is visible in the top right corner of the document area.

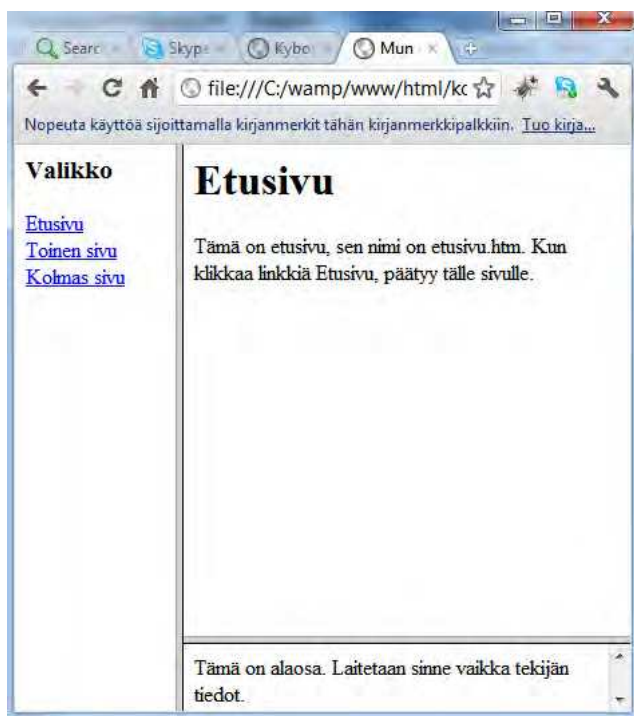
Tällaisella esitystavalla on useita etuja. Käyttäjä voi siirtyä vasemmassa kehyksessä eli "navigointikehyksessä" haluamaansa kohtaan ja valita klikkaamalla. Jos hän huomaa, ettei se sisältänyt sitä mitä hän etsi, hän voi saman tien valita toisen kohdan. Erityisesti hakuteostyyppisissä dokumenteissa tämä voi olla näppärää. ([Korpela.](#))

9.2.1. Kuinka rakentaa kehykselliset www-sivut?

Kehykset olivat muotia ajanjaksolla 1998 - 2002. Nykyään kehyssivuja näkee erittäin harvoin. Niiden käyttöä ei suosittelen kukaan, en minäkään. Jos kuitenkin haluat tehdä kehykselliset www-sivut, on sinun toimittava seuraavan kaavan mukaan.

1. luo sivu, joka sisältää kehyksen määrittelyn (frameset)
2. luo sivut, jotka tulevat kehyksiin (kehysten sisään)

Kehysten avulla tehty sivu voi olla rakenteeltaan kuten tässä alla. Siinä on kapea vasen reuna ja leveämpi oikeanpuoleinen osa on vielä pilkottu kahtia ylä- ja alaosaan.



Koodina yllä oleva kehyssivu menee kuten alla.

```
<html>
<head>
<title>Mun sivu</title>
</head>
<frameset cols="120,*">
<frame src="vasenvalikko.htm" name="menu">
<frameset rows="*,50">
<frame src="etusivu.htm" name="teksti">
<frame src="alaosa.htm" name="alaosa">
</frameset>
</frameset>
</html>
```


Vasen reuna on koodin osalta kuten alla.

```
<h3>Valikko</h3>
```

```
<a href="etusivu.htm">Etusivu</a><br>  
<a href="toka.htm">Toinen sivu</a><br>  
<a href="kolmas.htm">Kolmas sivu</a><br>
```

Kehyksiä käytettäessä on linkkien kanssa oltava tarkkana. Ongelmaksi muodostuu esimerkiksi, että sivut avautuvat kehyksen sisään. Käytettävät perusvaihtoehdot ovat:

- _blank** avaa uuden selainikkunan
- _self** lataa sivun samaan ikkunaan
- _top** avautuu entisen päälle.

Sivuston ulkopuolella oleva linkki on aina `_blank`. Muussa tapauksessa linkitetty sivu näkyy kehyksessä.

```
<h3>Valikko</h3>
```

```
<a href="etusivu.htm">Etusivu</a><br>  
<a href="toka.htm">Toinen sivu</a><br>  
<a href="http://www.otavanopisto.fi">Otavan Opisto</a><br>
```

Nyt jos napsautat linkkiä Otavan Opisto, aukeaa linkki kehyksen sisään ja lopputulos on kaalaman näköinen.



Oikea tapa kirjoittaa ulkopuolinen linkki on siis `_blank`.

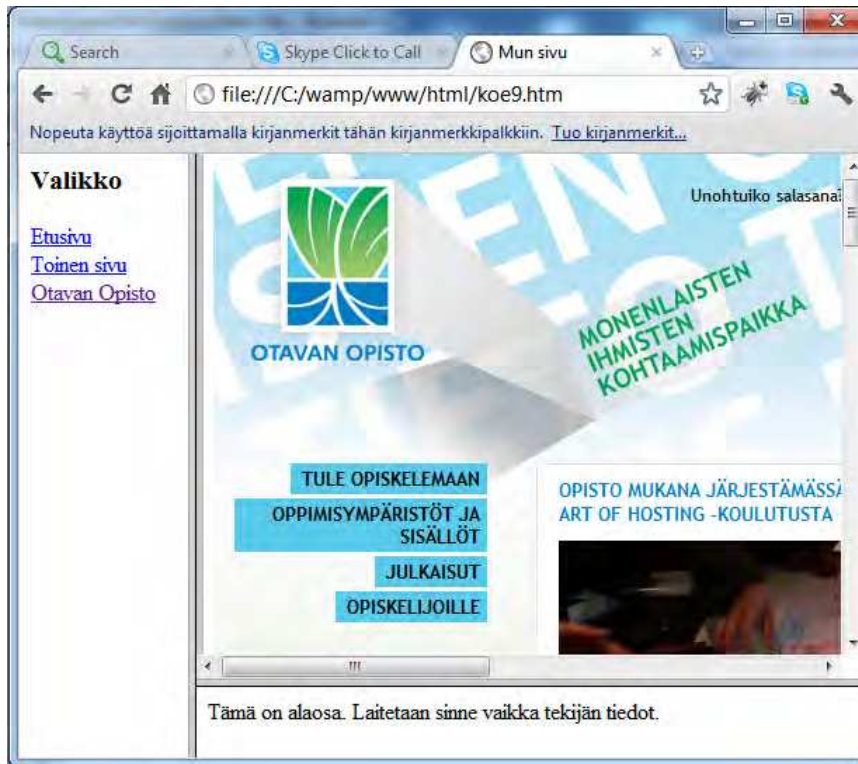
```
<a href="http://www.otavanopisto.fi" target="_blank">Otavan Opisto</a><br>
```

Jos halutaan sivun avautuvan oikeanpuoleiseen kehykseen, niin sitten kuten alla.

```
<a href="http://www.otavanopisto.fi" target="teksti">Otavan Opisto</a><br>
```

HUOM! Määrittelimme, että oikeanpuoleinen kehys on nimeltään teksti.

```
<frame src="etusivu.htm" name="teksti">
```



9.3. Iframe

IFRAME (Inline Floating Frames) on kehys, joka sijoitetaan www-sivulle. Kehyksen hienous piilee siinä, että kehysen sisään voidaan hakea sisältöä muualta, esimerkiksi joltakin toiselta sivustolta. Ero tavallisiin kehyksiin on siinä, ettei koko sivu ole kehys. Iframen avulla voi valita miltä tahansa sivulta alue, johon sisältö tulee muualta. Huono puoli on siinä, etteivät kaikki selaimet välttämättä oletusarvoisesti hyväksy iframen kautta tulevaa sisältöä.

Tässä esimerkki siitä, kuinka ottelutulokset ja videot haetaan HC Crusadersin sivuilta.

```
<BODY>
```

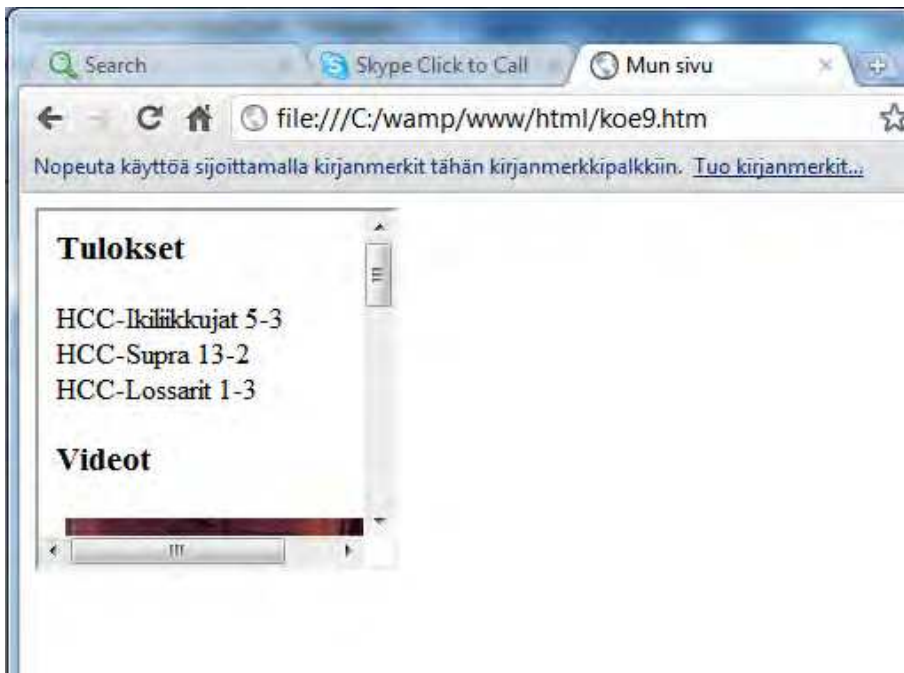
```
<iframe src="http://www.hccrusaders.net/keski.php" name="iframe"
```

```
width="200" marginwidth="10" height="200"
```

```
marginheight="10" scrolling="auto">
```

```
</iframe>
```

```
</BODY>
```



Käytetyt attribuutit ovat:

- **nimi** (identifioi kehyksen ja sitä voidaan käyttää myös targettina kun linkitetään)
- **width** ja **height** (kehyksen koko joko pikseleinä tai prosentteina)
- **margin width** ja **margin height** (sisällön ja kehyksen välinen tila)
- **scrolling** (**auto**=tarvittaessa, **yes**=aina, **no**=ei ikinä, esim. scrolling="auto")
- **frameborder** (näkykö kehyksen reunus vai ei, esim. näin: frameborder="0")

Eriyisen käteviä iframet ovat taulukkopohjaisissa taitoissa, koska silloin kehys saadaan juuri sinne minne halutaan.

```
<BODY>
<table width="450" border="0" cellpadding="10"
cellspacing="0" bgcolor="#DCDCBA">
<tr valign="top">
<td> <p>Ensimmäinen solu </p></td>
<td> <iframe src="http://www.hccrusaders.net/keski.php" name="iframe"
width="300" height="350" scrolling="auto"> </iframe> </td>
<td> <p>Kolmas solu </p> </td>
</tr>
</table>
</BODY>
```



Joskus ongelmana voi olla taustakuvan häviäminen. Silloin ratkaisu on tehdä iframesta läpinäkyvä tähän tapaan (allowtransparency).

```
<iframe src="kehys.htm" width="300" height="300" scrolling="auto"
frameborder="0" allowtransparency="true"></iframe>
```

Linkittäminen iframen sisään onnistuu näin:

```
<a href="sivu.htm" target="iframe">Linkki</a>
```

Linkkiä napsauttamalla kehukseena avautuu linkin takana luuraava sivu eli sivu.htm. Iframea voi tyylitellä css:n avulla esimerkiksi kuten alla, jossa iframelle on piirretty sininen kehys kaksinkertaisena.

```
<BODY>
<table width="450" border="0" cellpadding="10"
cellspacing="0" bgcolor="#DCDCBA">
<tr valign="top">
<td> <p>Ensimmäinen solu </p></td>
<td> <iframe src="http://www.hccrusaders.net/keski.php" name="iframe"
width="300" height="350" scrolling="auto" style="border: 5px double #ACBBEC;">
</iframe> </td>
<td> <p>Kolmas solu</p> </td>
</tr>
</table></BODY>
```



9.4. Taulukot

Taulukoita käytetään kotisivuilla kahdesta eri syystä:

- taulukoiden avulla voidaan esittää informaatio havainnollisesti.
- taulukoita käytetään usein myös kotisivun ulkoasun taittamiseen.

Alun perin oli tarkoitus, että html-tilaukoilla esitettäisiin vain dataa. Hyvin pian websuunnittelijat keksivät hyödyntää taulukkoa www-sivujen taittoon. Taulukon avulla sivujen elementit saatiin halutuille paikoille, esim. yläpalkki ylös, sen alapuolelle navigaatio ja lopuksi sisältö kahteen palstaan. Nyt tosin ollaan siirtymässä pois kyseisestä tavasta taittaa websivu. Korvaavassa tavassa taitto hoituu HTML:n ja CSS:n avulla.

Taulukkopohjainen sivujen taittaminen on jatkuvasti vähenemään päin. Vuonna 2006 löysin vielä helposti taulukkopohjaista taittamista isoiltakin sivuilta. Silloin oli helppoa poimia esimerkkejä, kuten 1.1.2006, jolloin taulukkopohjaisia taittoja löytyi muun muassa seuraavilta isoilta sivustoilta:

- [Sonera](#),
- [Dna](#)
- [MTV3](#).

Lähdekoodi paljasti, että table-tagia oli käytetty ahkerasti, vaikka sivulla ei näkynyt ainuttakaan taulukkoa. Lähemmässä tarkastelussa esimerkiksi MTV3-sivuston koodissa näkyi tageja kuten <table>, <td> ja <tr>. Tässä pätkä ihan sivun alusta:

```
<body bgcolor="#CCCCCC" >
<table width="990" border="0" cellpadding="0" cellspacing="5">
<tr>
<td colspan="2" style="height: 90px; padding: 0px 5px 5px 10px">
<table width="100%" border="0" cellpadding="0" cellspacing="0">
<tr><td>
```

Nyt huhtikuussa 2012 kyseiset isot sivustot eivät ole enää taulukkotaittoiset. Nyt Soneran, DNA:n ja Elisan sivuilla näkyy runsaasti div-tageja eikä bodyn jälkeen ensimmäinen tagi ole table. Table-tagin avulla taittaminen ei kuitenkaan ole täysin poistunut käytöstä. Moni web-suunnittelija taittaa edelleenkin pieniä sivustoja table-tagilla.

9.4.1. Taulukon tekeminen

Taulukon määrittely aloitetaan tagilla **<table>**. Voit laittaa sivuillesi minne tahansa taulukko-tagin, kunhan muistat päättää sen merkillä **</table>**. Yksinkertaisimmillaan taulukko olisi tällainen.

```
<table></table>
```

Tällaisesta taulukosta ei kuitenkaan ole mitään hyötyä, koska siitä puuttuvat rivit ja sarakkeet. Rivi lisätään koodilla **<tr>** ja sarake koodilla **<td>**. Tehdään yksinkertainen esimerkki, jossa on kaksi riviä ja kaksi saraketta.

```
<table>
<tr>
<td>eka rivi ja eka solu</td><td>eka rivi ja toinen solu</td>
</tr>
<tr>
<td>toka rivi ja eka solu</td><td>toka rivi ja toinen solu</td>
</tr></table>
```

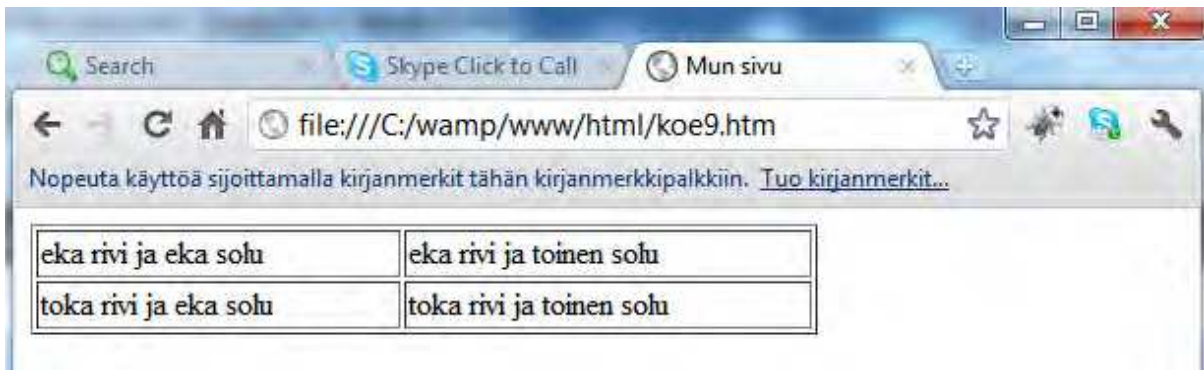
Selaimen kautta tarkasteltuna yllä oleva koodiesimerkki näyttää tältä:



Muokataan koodia ottamalla mukaan myös taulukon reunaviivat sekä määritellään taulukon leveydeksi 400 pikseliä.

```
<table width="400" border="1">
<tr>
<td>eka rivi ja eka solu</td><td>eka rivi ja toinen solu</td>
</tr>
<tr>
<td>toka rivi ja eka solu</td><td>toka rivi ja toinen solu</td>
</tr></table>
```

Nyt taulukon ulkoasu on hieman muuttunut.



9.4.2. Tablen ominaisuudet

Taulukkoa ja sen soluja sekä rivejä voidaan määritellä mm. koon ja värin suhteen. **Table-tagilla** voi olla seuraavanlaisia ominaisuuksia.

Ominaisuus	Kuvaus	Esimerkki
align=left center right	Määrittelee taulukon asemoinnin	<table align="center">
background=tiedostonimi	Taustakuva taulukolle	<table background="koira.jpg">
bgcolor=#rrggbb	Taustaväri taulukolle	<table bgcolor="#00FF00">
border=n	Reunan paksuus	<TABLE border="0">
bordercolor=#rrggbb	Reunan väri	<TABLE border="1" bordercolor="#rrggbb">
bordercolordark=#rrggbb	Reunan varjo	<TABLE border="1" bordercolor="#FFFFFF" bordercolordark="#rrggbb">
cellpadding=n	Solujen väli (kuinka suuri)	<TABLE cellpadding="1">
cellspacing=n	Solun reunan ja sisällön väli	<table cellspacing="2">

valign= top bottom	Sisällön paikka solun sisällä	<table cellspacing="2" valign="top">
width= n,n n,n%	Taulukon vähimmäisleveys pikseleinä ja prosentteina	<table width="23">

TR:n ja TD:n ominaisuudet

Taulukon rivillä <TR> ja sarakkeella <TD> voi olla seuraavanlaisia ominaisuuksia.

Ominaisuus	Kuvaus	Esimerkki
align= left center right	Määrittelee sisällön ase- moinnin solussa.	<td align="center">
background=tiedostonimi	Taustakuva solulle	<td background="koira.jpg">
bgcolor=#rrggbb	Taustaväri solulle	<td bgcolor="#00FF00">
bordercolor=#rrggbb	Reunan väri	<td bordercolor="#rrggbb">
bordercolordark=#rrggbb	Reunan varjo	<td bordercolor="#FFFFFF" bordercolordark="#rrggbb">
valign= top bottom	Sisällön paikka solun sisäl- lä	<td valign="top">
width= n,n n,n%	Solun vähimmäisleveys pikseleinä ja prosentteina	<td width="23"> <td width="50%">
height= n,n n,n%	Solun vähimmäiskorkeus pikseleinä ja prosentteina	<td height="23"> <td height="100%">

TD:n ominaisuudet

Taulukon solulla <TD> voi olla seuraavanlaisia ominaisuuksia.

HUOM! Jos ominaisuus on määritelty TR:lle, niin TD ylittää sen.

Ominaisuus	Kuvaus	Esimerkki
nowrap	Estää rivinvaihdon	<td nowrap>
rowspan=n	Rivien määrä (yhdistää)	<td rowspan="2">
colspan=n	Solujen määrä (yhdistää)	<td colspan="2">

9.4.3. Esimerkkejä

Esimerkki 1.

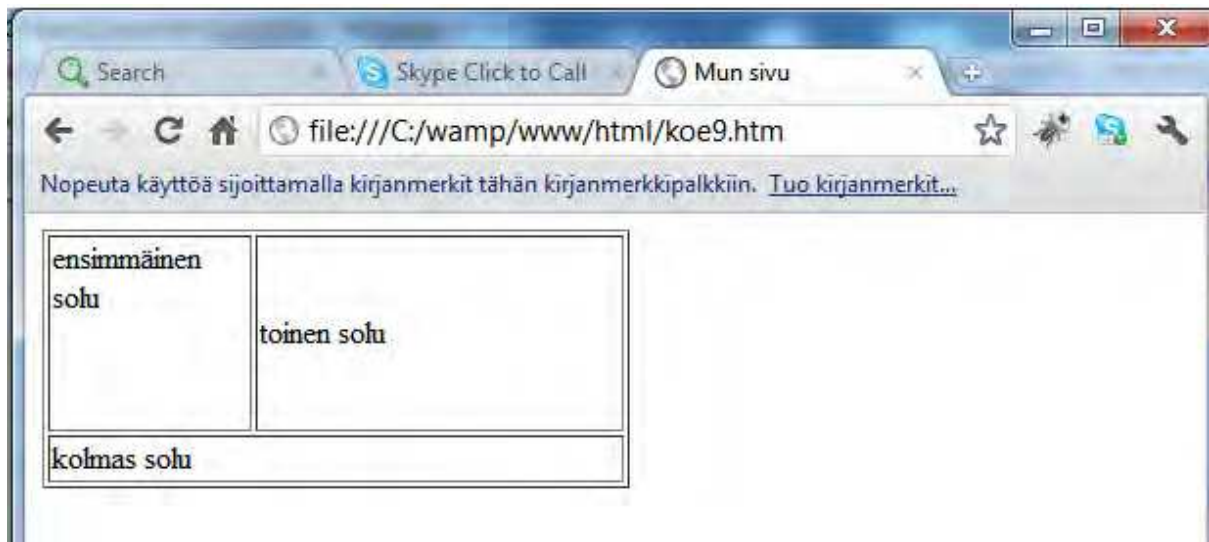
```
<table width="300" cellpadding="2" border="1">
<tr>
<td width="100">eka rivi ja eka solu</td><td width="200">
eka rivi ja toinen solu</td>
</tr>
<tr>
<td bgcolor="#666666">
toka rivi ja eka solu</td><td>toka rivi ja toinen solu,
jossa eniten tekstä och samma på svenska</td>
</tr>
</table>
```



Esimerkki 2

Tässä esimerkissä yhdistetään rivin kaksi solut yhdeksi soluksi käyttäen tagia colspan (<td colspan="2">).

```
<table width="300" border="1">
<tr>
<td width="100" valign="top" height="100"> ensimmäinen solu</td>
<td>toinen solu</td>
</tr>
<tr>
<td colspan="2">kolmas solu</td>
</tr>
</table>
```



Nauhoite 8: iframe

kesto: 07:15

URL: <http://youtu.be/eFXsagy-ufo>

Tehtävä 9

Tee taulukko, jonka koko on 80% ja jossa on kolme riviä ja kolme solua. Solujen koot ovat 30%, 30% ja 40%.

10. Multimedia

Web-sivuille voidaan liittää tekstiä, linkkejä ja kuvia - niiden lisäämistä on jo esitelty aiemmin. Sivuille voidaan myös lisätä liikkuvaa kuvaa sekä ääntä. World Wide Webin alkuaikoina selaimet tukivat vain tekstiä. Myöhemmin tulivat selaimet, jotka osasivat näyttää myös kuvia. Nykyään HTML:n avulla sivuille voidaan liittää myös multimediaa eli ääntä ja liikkuvaa kuvaa. HTML:n avulla sivuille voidaan lisätä jopa peli.

10.1. Multimedia - ääni

Ääntä saa sivuille taustamusiikiksi, mutta en suosittelen sitä. Tässä joitakin syitä siihen, miksi en suosittelen:

- moni haluaa ainoastaan lukea sivuja ja katsella kuvia
- jotkut surffailevat öiseen aikaan, jolloin muu väki nukkuu. Yks kaks hiljaisuuden rikookin yllättävä taustamusiikki.
- vaikka musiikki olisi kaunista, lakkaa se soimasta, kun henkilö vaihtaa toiselle sivulle. Taustamusiikki ei kulje samalla tavalla mukana kuin taustaväri.
- kaikki selaimet eivät tue sivuille upotettua ääntä

MIDI (lyhenne englannin kielen sanoista *Musical Instrument Digital Interface*) tarkoittaa musiikkisoittimien digitaalista liitäntää tai rajapintaa. Se on tiedonsiirtojärjestelmä, joka on suunniteltu välittämään viestejä sähköisten musiikkilaitteiden välillä. MIDI-tiedostot ovat hiljattain alkaneet tulla kaikkien kuluttajien ihmeteltäviksi uusimpien matkapuhelinten soittoäänien ominaisuudessa. ([Mikä on MIDI?.](#))

Internet Explorerissa esim. juuri MIDI-tiedoston saa taustääneksi alla olevalla komennolla.

```
<bgsound src="beethoven.mid" />
```

Muita selaimia (esim. Firefox) ajatellen seuraava tapa olisi parempi.

```
<embed src="beethoven.mid" autostart="false" loop="false" height="45" width="170"><noembed>Sorry, selaimesi ei tue embed-tagia</noembed>
```

Autostart määrittää sen, käynnistyykö ääni automaattisesti vai ei. Attribuutit width ja height määrittelevät ohjauspaneelin koon. Lopuksi on vielä koodia siltä varalta, jos käyttäjän selain ei tue tagia embed.

MP3 on lyhenne, joka tulee sanoista MPEG audio layer 3. MPEG on puolestaan tapa pakata aineistoa pieneen tilaan, samalla kuitenkin menetetään osa alkuperäisestä aineistosta. MP3 on "lähes CD-tasoisesta äänestä." Kun alkuperäinen ääni on ollut CD-tasoisesta, 44kHz näytetaajuudella otettua 16-bittistä stereoääntä, on tästä äänestä poistettu asioita, joita ihmiskorva ei helposti erota. Näin äänitiedoston koko on saatu kahdenteentoista osaan alkuperäisestä koosta. MP3 on nykyisin erittäin suosittu formaatti äänelle.

Taustääni ei ole koskaan hyvä vaihtoehto. Parempi on laittaa musa linkin taakse.

```
<a href="01_stone_cold.mp3">Kuuntele bändiä Autonomadic</a>
```

HUOM! Musiikin kanssa pitää olla erityisen tarkkana. Älä laita sivuille musiikkia, joka on tekijänoikeuslakien alaista. Yksi suositeltava vaihtoehto on etsiä ja käyttää musiikkia, joka on julkaistu [Creative Commons](#) -lisenssillä. Esimerkiksi [Jamendosta](#) löytyy myös Creative Commons -lisensoitua musiikkia.

Jos ääni halutaan upottaa sivulle, seuraa ongelmia. Osa selaimista kyselee, saako äänitiedoston soittaa. Toinen ongelma liittyy siihen, että Firefox suosii koodia embed ja IE koodia object. Siksi olisikin tehtävä kuten alla eli huomioi kaiken, myös sen ettei ääntä kuulu lainkaan.

```
<object id="mediaPlayer" width="0" height="0" classid="CLSID:22d6f312-b0f6-11d0-94ab-0080c74c7e95" code-base="http://activex.microsoft.com/activex/controls/mplayer/en/nsmp2inf.cab#Version=5,1,52,701" standby="Loading Microsoft Windows Media Player components..." type="application/x-oleobject" uiMode="invisible">
```

```
  <param name="fileName" value="beethoven.mid">
  <param name="animationatStart" value="true">
  <param name="transparentatStart" value="true">
  <param name="autoStart" value="true">
```

```
  <param name="showControls" value="false">
  <param name="uiMode" value="none">
  <param name="loop" value="true">
```

```
</object>
```

```
<embed type="application/x-mplayer2" width="0" height="0" visible="false"
  pluginspage="http://microsoft.com/windows/mediaplayer/en/download/"
  id="mediaPlayer2" name="mediaPlayer" displaysize="0" autosize="false"
  src="beethoven.mid"
  playCount="9999" autostart="true" designatesp="5311" loop="true"></embed>
```

```
<div style="font-size: 65%">
```

```
Eikö musiikki kuulu? Lataa se <a href="beethoven.mid3">tästä;</a>.
```

```
</div>
```

10.2. Multimedia: liikkuvaa kuvaa

Muistutan heti alkuun, ettei HTML-sivuilla alun perin ollut tarkoitus esittää multimediaa. Ennistä parempien nettiyhteyksien vuoksi ovat videot kuitenkin lisääntyneet netissä. Selaimet eivät kuitenkaan tue videoita suoraan. Videoitten katseluun tarvitaan erilliset katseluohjelmat tai pluginit kuten esim. Windowsin Media Player, QuickTime tai RealPlayer.

Kun videon haluaa laittaa nettiin saataville, on ensin huolehdittava videon koosta. Kiinteät, nopeat nettiyhteydet ovat vielä suhteellisen harvinaisia, joten video olisi hyvä saada mahdollisimman pieneen tilaan, jotta sen kopioiminen ei kestäisi iäisyksiä. Normaali kamerasta saatava ja editoitu DV-muotoinen video vie noin 3,6 megatavua sekunnissa, mikä on aivan liikaa jopa parhaille ADSL- tai kaapelimodeemiyhteyksille.

Mikäli videota ei katsella selainikkunassa, on linkki hyvä vaihtoehto.

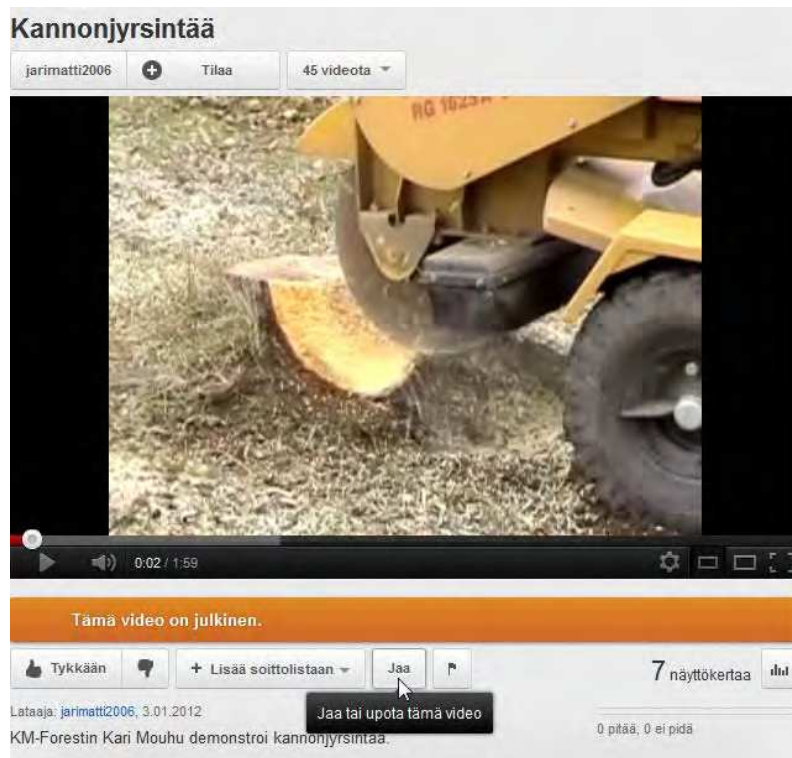
```
<a href="video.avi">Video</a>
```

10.2.1. Upotetut videotiedostot

Tavallisille www-sivuille voidaan upottaa videoita. Silloin ohjelma avautuu ikään kuin www-sivun sisään. Tämä on hyvä tapa siinä mielessä, ettei selaaja joudu pois sivuiltasi kesken

kaiken. Tämä tapa on yleinen esim. suosituissa [YouTubessa](#). Nykyään onkin järkevintä laittaa video YouTubeen ja upottaa se sieltä.

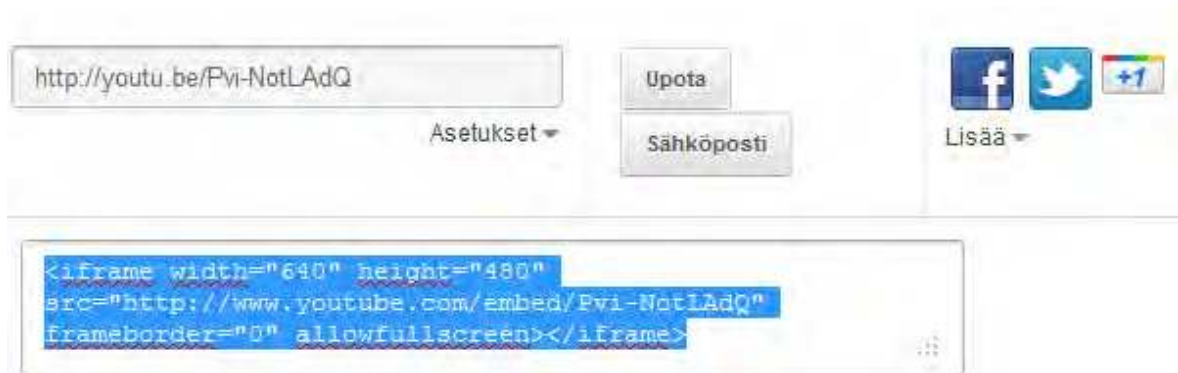
YouTubeesta upottaminen on helppoa. Valitset videon ja sitten painikkeen "Jaa".



Sitten valitset upota.



Lopuksi kopioit koodin sivuillesi ja video on toiminnassa.



10.3. Javan liittäminen sivuille

Java Applet on Java-ohjelmointikielellä tehty pieni ohjelma, jota voidaan käyttää www-sivuilla. HTML-dokumentissa määritellään, mikä applet suoritetaan ja millaisena se suoritetaan (sijainti ja koko www-sivulla, muut ominaisuudet). Java Applettien katsomiseen tarvitaan Java-yhteensopiva selain. Java-yhteensopiva selain tarkoittaa selainta, jossa on joko Sunin kehittämä, erikseen Internetistä ladattava Java Plug-in tai useimmissa selaimissa jo valmiina oleva Java-virtuaalikone. Nykyään appletteja käytetään www-sivuilla harvoin, koska FLASH on syrjäyttänyt ne ainakin osittain.

HTML-kielessä appletti ladataan **<APPLET>...</APPLET>**-tageilla.

```
<applet code="lake.class" width="280" height="440">  
<param name="image" value="kansanopistopeli.jpg">  
<param name="href" value="http://www.ibdprince.com/java.shtml">  
</applet>
```

Nykyään on tosin suositeltavaa käyttää object-tagia applet-tagin sijaan.

```
<OBJECT codetype = "application/java"  
  classid = "java:lake.class"  
  width="250" height="674">  
<param name = "image" value = "petteri.jpg">  
</OBJECT>
```



Varsinkin nuo vedenheijastusta matkivat efektit ovat olleet suosittuja. Kyseiset appletit löydät osoitteesta: <http://www.ibdprince.com/java.shtml>

10.4. Flashin liittäminen sivuille

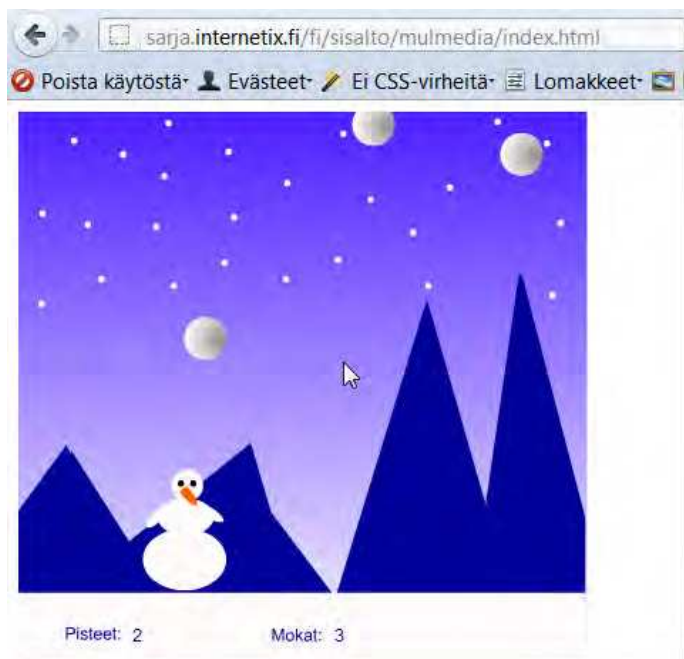
Yhtenä vaihtoehtona Javan käytölle on Macromedian Flash-tekniikka. Flash-tekniikan perus-idea on sama kuin Javassa ainakin siinä mielessä, että HTML-dokumenttiin "upotetaan" toiminnallisesti erillinen osa. Flashin yhteydessä käytettävä dataformaatti on Shockwave Flash (SWF). Flash-ohjelma tekee automaattisesti myös Flash-toteutuksen sisältävän HTML-tiedoston, jota on tarvittaessa helppo muuttaa omiin tarkoituksiin sopivaksi.

Tältä näyttäisi pelin liittäminen sivulle:

```

<HTML>
<HEAD>
<TITLE>index</TITLE>
</HEAD>
<BODY bgcolor="#FFFFFF">
<!-- URL's used in the movie-->
<A HREF="lumiukko.swf"></A> <!-- text used in the movie-->
<!--Polkka lumiukon --><OBJECT classid="clsid:D27CDB6E-AE6D-11cf-96B8-444553540000"
code-
base="http://download.macromedia.com/pub/shockwave/cabs/flash/swflash.cab#version=5,0,0,0"
WIDTH="400" HEIGHT="400">
<PARAM NAME=movie VALUE="index.swf"> <PARAM NAME=quality VALUE="high"> <PARAM
NAME=bgcolor VALUE="#FFFFFF"> <EMBED src="index.swf" quality="high" bgcolor="#FFFFFF"
WIDTH="400" HEIGHT="400" TYPE="application/x-shockwave-flash"
PLUGINS PAGE="http://www.macromedia.com/shockwave/download/index.cgi?P1_Prod_Version=Sh
ockwaveFlash"></EMBED></OBJECT>
</BODY>
</HTML>

```



11. Sivuston taittaminen HTML:n avulla

Sivuston taittaminen tarkoittaa sitä, että kuvasta tehdään HTML-koodia. Toisin sanoen, web-sivu on aluksi vain yksittäinen kuva, joka puretaan vähitellen paloiksi ja liitetään HTML-koodiin.

Laitetaan ensin HTML:n perusrunko kuntoon eli sivun kannalta kaikki pakolliset koodit. Samalla laitetaan sivulle taustaväri.

```
<html>
<head>
<title>HC Crusaders</title>
</head>
<body bgcolor="#d3cbe0">

</body>
<body>
</html>
```



Seuraavassa vaiheessa sivulle luodaan perusrakenne taulukon avulla. Laitetaan taulukko paikoilleen; taulukon leveys on 950 pikseliä. Lisäksi keskitetään taulukko komennolla align. Laitetaan samalla vielä yläpalkki paikoilleen eli se aloittaa uuden rivin (TR) ja tarvitsee yhden uuden solun (TD).

```
<html>
<head>
<title>HC Crusaders</title>
</head>

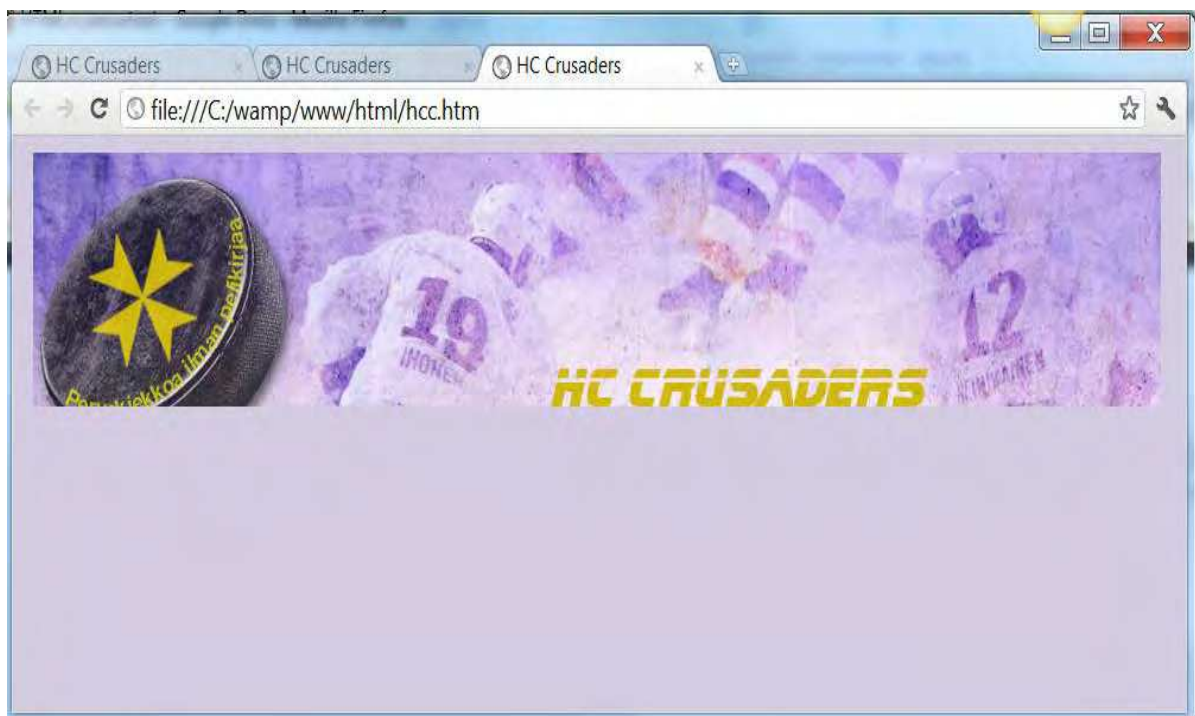
<body bgcolor="#d3cbe0">

<table width="960" align="center">

<tr>
<td width="950" height="169"></td>
</tr>

</table>

<body>
</html>
```



Seuraavaksi laitamme navigaation paikoilleen. Tarvitsemme jälleen uuden rivin ja solun. Navigaation koko on 950 x 31 pikseliä.

```
<html>
<head>
<title>HC Crusaders</title>
</head>
<body bgcolor="#d3cbe0">
<table width="950" align="center">
<tr>
<td width="950" height="169"></td>
</tr>
<tr>
<td width="950" height="31"></td>
</tr>
</table>

</body>
<body>
</html>
```



Kuten kuvasta näkyy, yläpalkin ja navigaation väliin jää rako. Korjataan tilanne määrittämällä tablen koodi uusiksi.

```
<table width="950" align="center" cellpadding="0" cellspacing="0" border="0">
```

Sitten on jäljellä enää alaosa, joka tosin jaetaan kahteen palstaan. Koska alhaalla on kaksi palstaa (solua), täytyy yläpuolen TD eli solu määritellä kahden solun levyiseksi (colspan=2). Lisäksi laitamme kummallekin palstalle taustakuvan.

```
<html>
<head>
<title>HC Crusaders</title>
</head>

<body bgcolor="#d3cbe0">

<table width="950" align="center" cellpadding="0" cellspacing="0"
border="0">

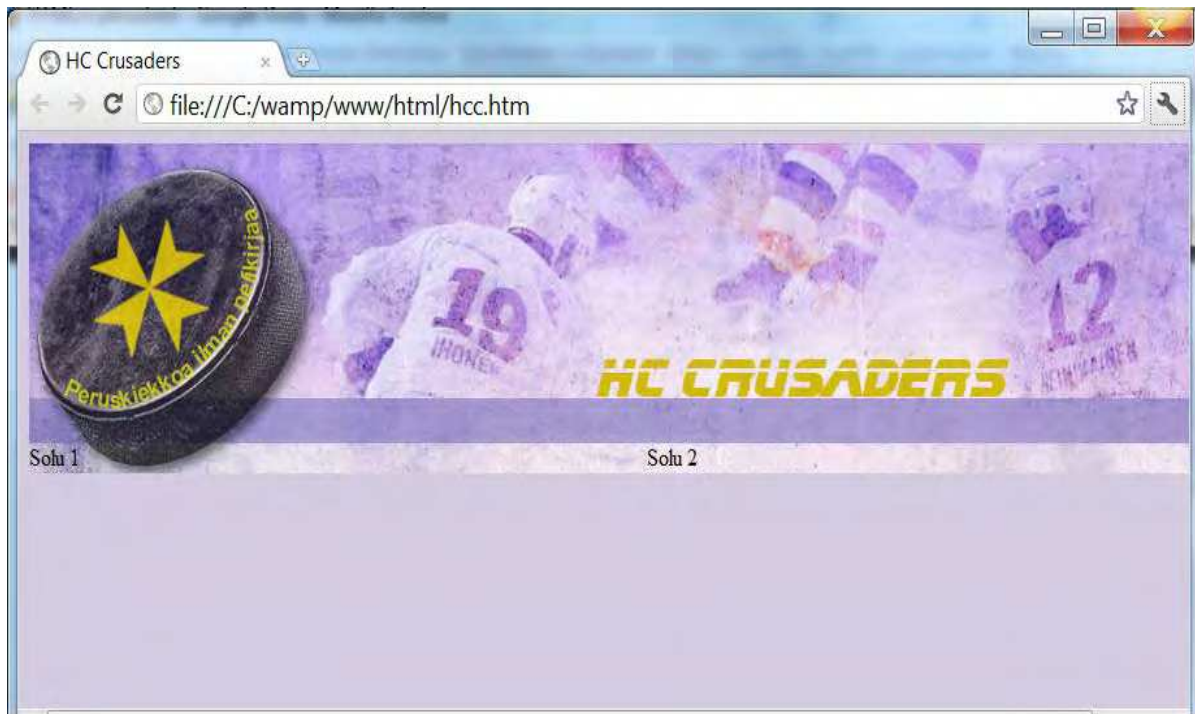
<tr>
<td width="950" height="169" colspan="2"></td>
</tr>

<tr>
<td width="950" height="31" colspan="2"></td>
</tr>

<tr>
<td width="586" background="vasen.jpg" >Solu 1</td>
<td width="364" background="oikea.jpg"> Solu 2</td>
</tr>

</table>

</body>
<body>
</html>
```



Sivu näyttää vielä hieman oudolta, koska tekstiä on niin vähän. Siksi laitamme sinne [lorem ipsum](#) tekstiä, jotta sivu alkaisi näyttää siltä kuten alkuperäisessä kuvassa.

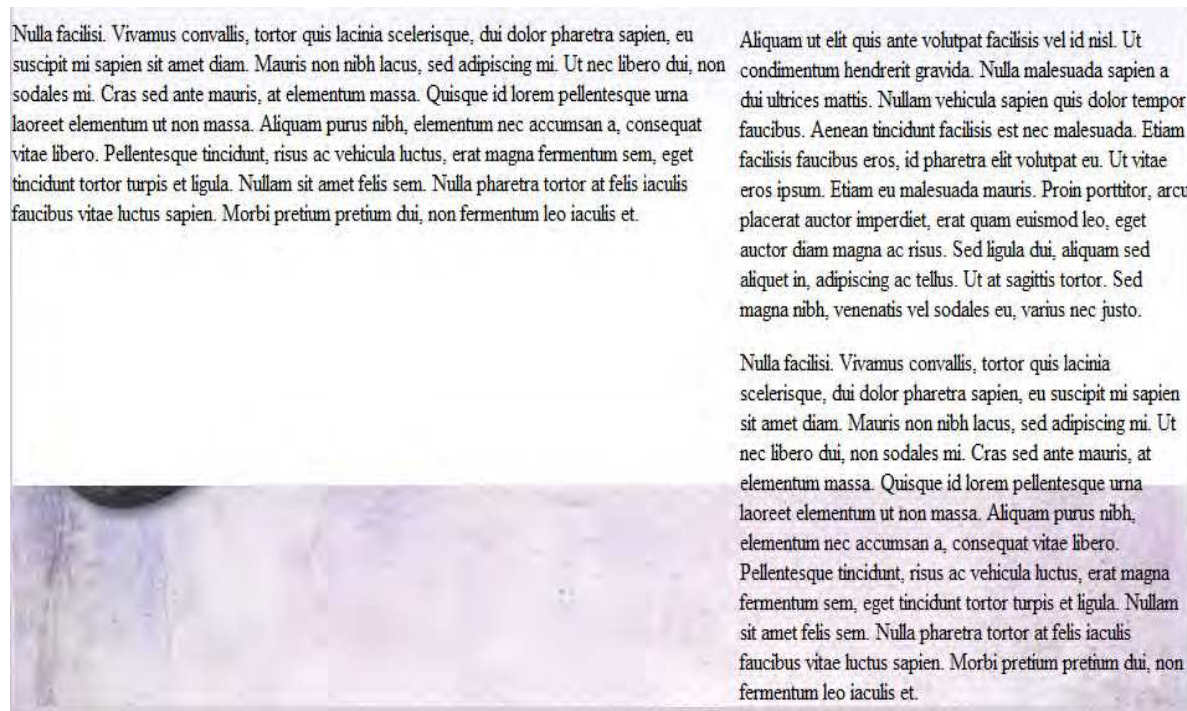


Tässä vaiheessa tulee ongelma, koska teksti ei ala yläreunasta. Siksi laitamme, että TD on muotoa `valign="top"`.

<td width="586" background="vasen.jpg" valign="top">

<td width="586" background="vasen.jpg" valign="top">

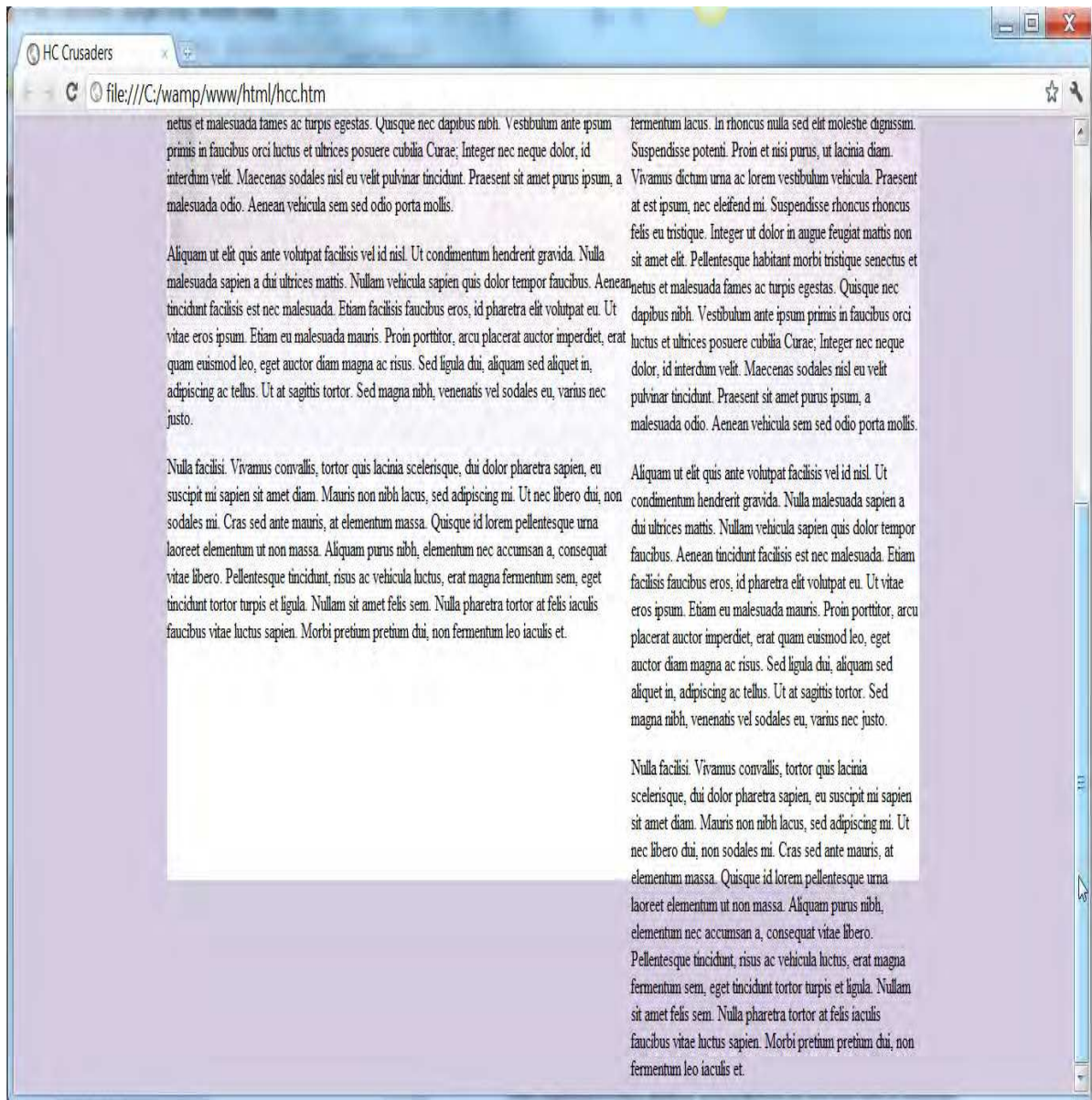
Nyt sivu alkaa olla kuosissa, mutta vielä on sellainen ongelma, että tausta alkaa uudestaan - katso alla olevasta kuvasta vasen alareuna, jossa kiekko uudestaan esillä.



Tuo ongelma johtuu siitä, että HTML:ssä kuva toistuu eikä toistamista voi estää HTML:n avulla. Tähän ongelmaan on kaksi ratkaisua. Ensimmäinen on se, että solun koko on sama kuin taustakuvan koko. Silloin solun korkeus olisi 800 pikseliä. Tämän ratkaisun huono puoli on se, ettei sivu voi olla pidempi kuin mihin se on määritelty. Toinen ratkaisu edellyttää CSS:n ottamista apuun. Silloin voimme estää taustan toistumisen. Koodi kuten alla.

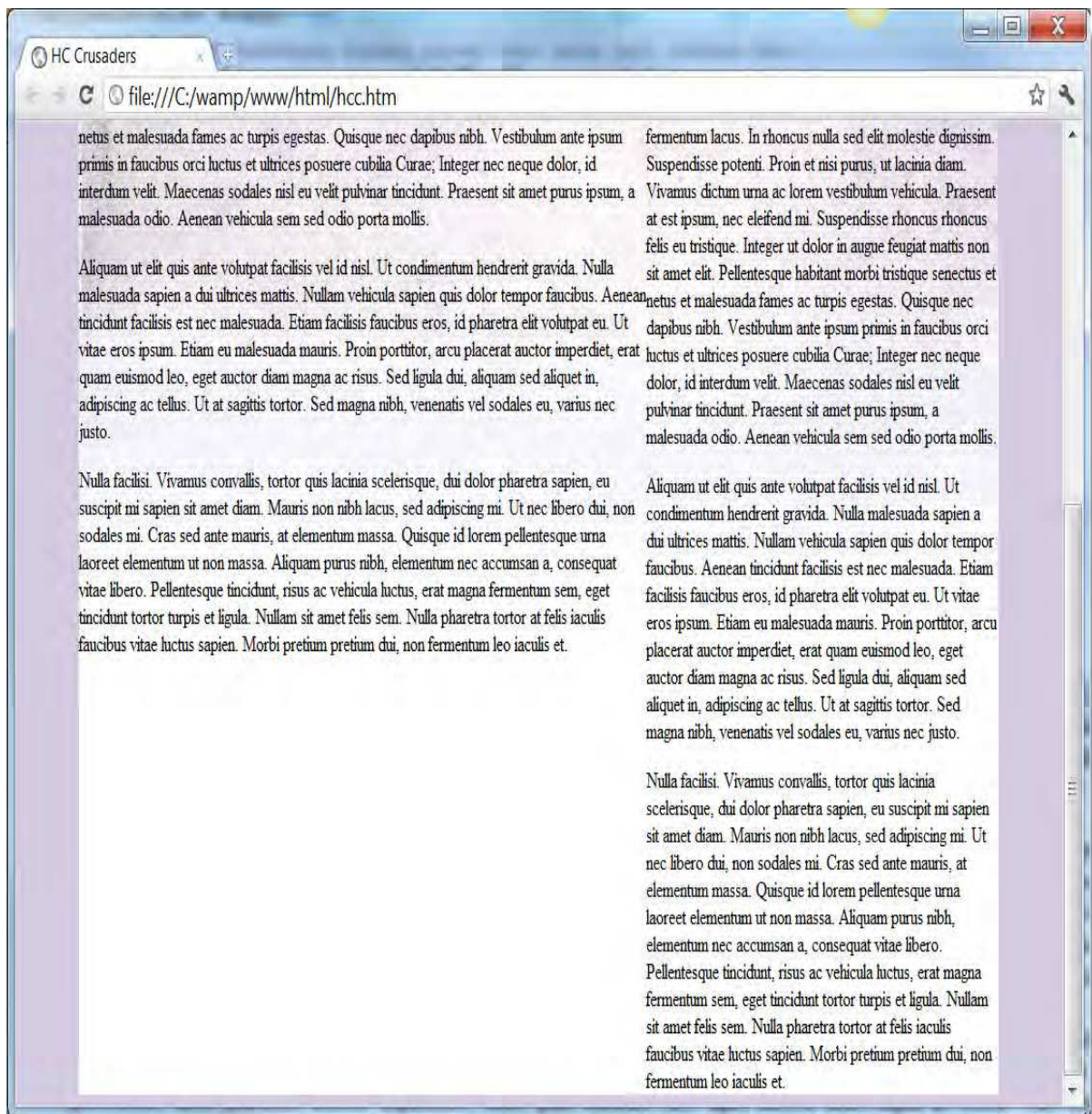
```
<td width="586" background="vasen.jpg" valign="top" style="background-repeat:no-repeat">
```

Tosin sitten on uusi ongelma, koska nyt alareuna ikään kuin loppuu kesken.



Tämän korjaamme sillä tavoin, että määrittelemme tablen taustaväriksi valkoisen.

```
<table bgcolor="#ffffff" width="950" align="center" cellpadding="0"
cellspacing="0" border="0">
```



Lopuksi tehdään vielä navigaatio, niin ensimmäinen sivu on valmis. Aiemmin laitoimme navigaation paikoilleen, mutta sen ongelmana on se, ettei kuvan päälle voi kirjoittaa. Siksi muutamme navigaation taustakuvan taulukon solun taustakuvaksi.

```
<td width="950" height="169" colspan="2" background="navigaatio.jpg"></td>
```

Nyt sen päälle voi kirjoittaa ja laitamme sinne linkit paikoilleen. Navigaatio kuten alla.



Siistitään linkkejä vielä hieman CSS:n avulla.

```

<a href="#" style="text-decoration:none;color:black">Etusivu</a>
<a href="#" style="text-decoration:none;color:black">Historia</a>
<a href="#" style="text-decoration:none;color:black">Info</a>
<a href="#" style="text-decoration:none;color:black">Tukijat</a>
<a href="#" style="text-decoration:none;color:black">Vieraskirja</a>
<a href="#" style="text-decoration:none;color:black">Joukkue</a>
<a href="#" style="text-decoration:none;color:black">Kuvagalleria</a>

```



Lopuksi korjataan se ongelma, että teksti alkaa liian reunasta. Käytämme siihen tagia div.

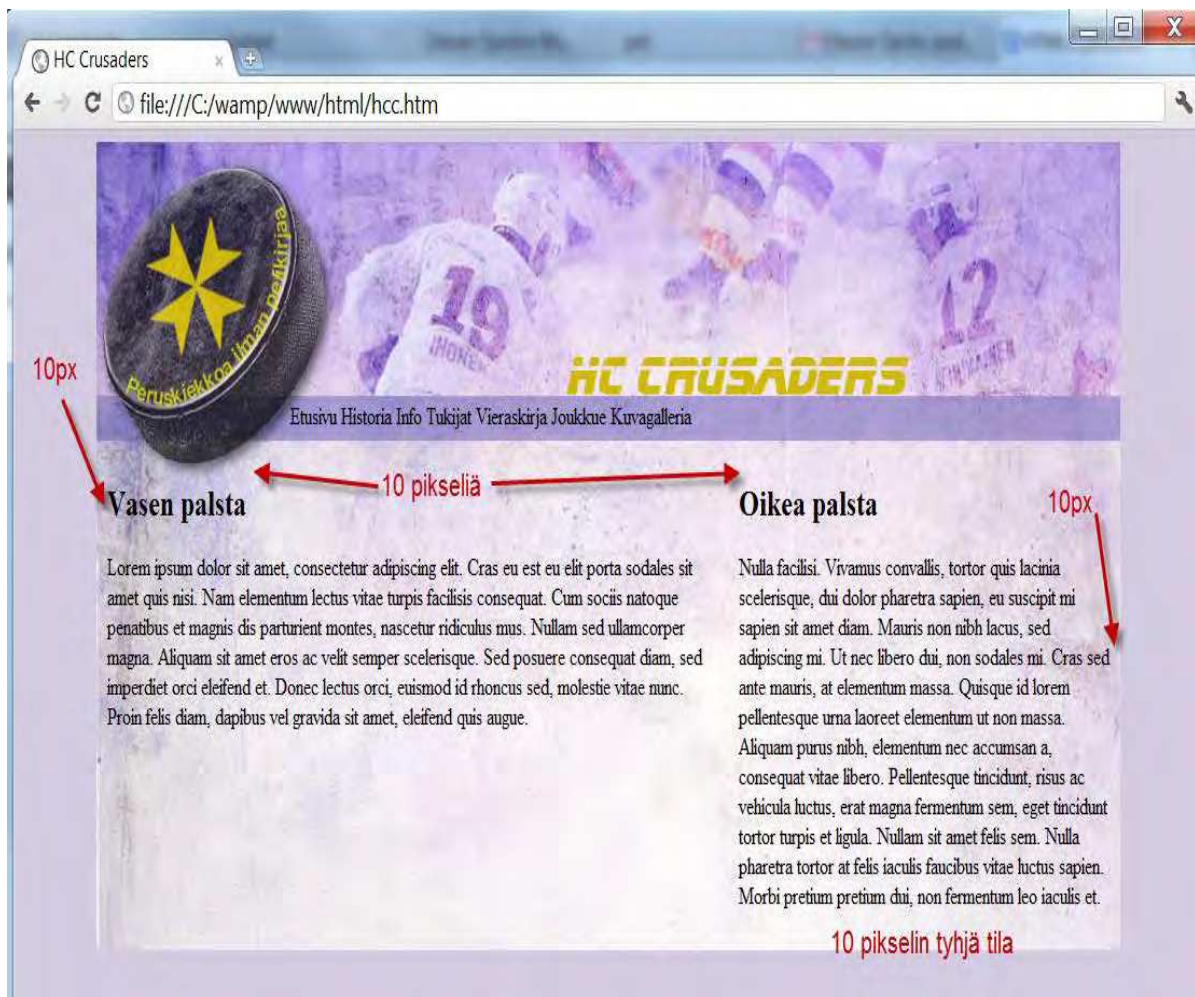
```
<div style="padding: 10px;">
```

```
<h2>Vasen palsta</h2>
```

```
<p>Lorem ipsum dolor sit amet, consectetur adipiscing elit. Cras eu est eu elit porta sodales sit amet quis nisi. Nam elementum lectus vitae turpis facilisis consequat. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Nullam sed ullamcorper magna. Aliquam sit amet eros ac velit semper scelerisque. Sed posuere consequat diam, sed imperdiet orci eleifend et. Donec lectus orci, euismod id rhoncus sed, molestie vitae nunc. Proin felis diam, dapibus vel gravida sit amet, eleifend quis augue.</p>
```

```
</div>
```

Teemme saman myös oikealle palstalle, jolloin ensimmäinen sivu on valmis.



Mikäli haluat tarkastella tehtyä sivua tarkemmin, voit tallentaa esimerkin omalle koneellesi [zip-pakettina](#).

Loppuosa sivuston tekemisestä on sitä, että kopioidaan sivupohja ja nimetään se. Lopuksi laitetaan navigaation linkitys paikoilleen. Ensimmäinen sivu voi olla nimeltään index.htm (kansion oletussivu), toinen sivu voisi olla nimeltään historia.htm ja kolmas info.htm. Valikko olisi silloin seuraavanlainen.

```
<a href="index.htm" style="text-decoration:none;color:black">Etusivu</a>
<a href="historia.htm" style="text-decoration:none;color:black">Historia</a>
<a href="info.htm" style="text-decoration:none;color:black">Info</a>
```



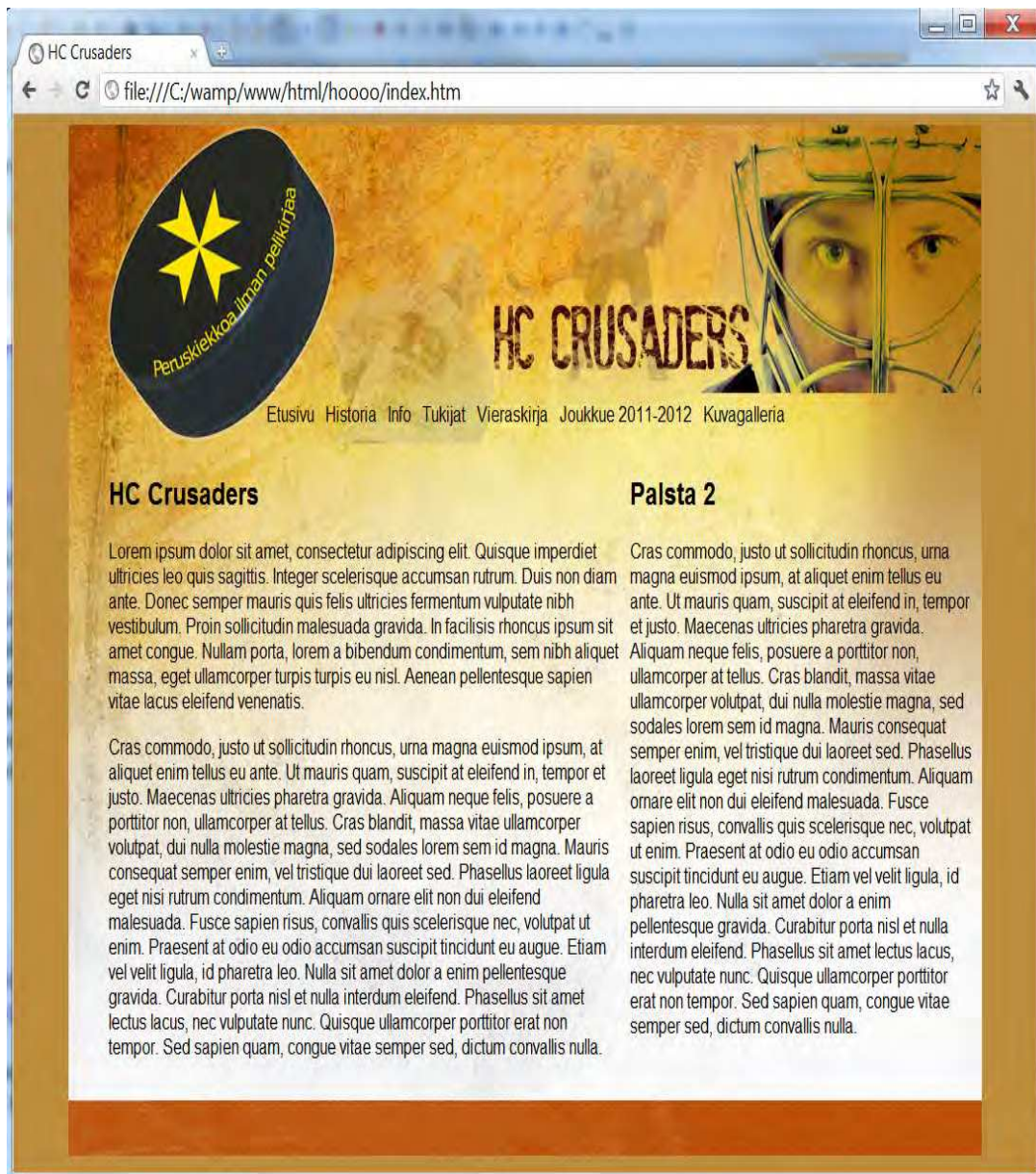
Nauhoite 9: Taulukkotaitto

kesto: 43:24

URL: <http://youtu.be/AUCETKJYrPo>

Tehtävä 10

Tee [kuvista](#) yksi taulukkotaitettu HTML-sivu, jotta se olisi mahdollisimman paljon samankaltainen kuin alla olevassa kuvassa. Käytä CSS:ää ainoastaan linkkien alleviivauksen poistoon ja siihen, että palstoille tulee tyhjää tilaa.



Lähteet

IT Plus (2001). Oikea työkalu: HTML. *IT Plus*(2), 89.

Mikä on MIDI? [Online: <http://www.henrin.net/musiikki/tekniikka/midi>] Luettu 21.1.2007.

WYSIWYG. Wikipedia.[Online: <http://fi.wikipedia.org/wiki/WYSIWYG>] Luettu 14.2.2007.

Materiaalia tehdessä on viitemateriaalina käytetty myös seuraavia web-sivustoja.

HTML Tutorial: <http://www.tizag.com/htmlT/htmlhr.php>

HTML Tutorial: <http://www.w3schools.com/html/>

Java-ohjelmoinnin perusteet:

<http://oppimateriaalit.internetix.fi/fi/avoimet/6tekniikkatalous/java/index>