

Vefsmíðar

Kóðinn, HTML og CSS

Þessi bók er hluti af þriggja bóka flokki, hinar eru
Viðmót, hönnun og verklag og
Dreamweaver og önnur tól

Allar bækurnar eru aðgengilegar án endurgjalds á
<http://where.is/handbok>

Tengla til stuðnings og meiri þekkingaröflunar er að finna á
<http://where.is/webdesign>

Útgáfa 12 í ágúst 2007

Gunnar Grímsson - gunnar@where.is

Efnisyfirlit

Inngangur 4

Kóðinn5

Uppbygging skipana 6

Helstu reglur XHTML 7

Munurinn á XHTML og HTML 4 7

Sannreyndu kóðann þinn 7

Hvernig skrífum við XHTML? 8

Skráanöfn, hvað má og hvað ekki 9

Umbrot kóða 9

Villuleit 9

Notaðu vafrana 10

CSS - Stílsnið 10

Vafri finnur ekki skrá 10

Þinn kóði og minn kóði 11

Flýtylklar og vanaverk 12

Einfalt XHTML skjal13

Grunnskipanir 14

CSS 14

Fyrirsagnir - <h1-h6> 15

Einfaldar leturbreytingar - , <i> 15

Umbrot texta 16

<p>Málgrein</p> 16

 16

Faldar athugasemdir 17

Röð íláta 17

Tenglar - <a> 18

Hafðu slóðirnar þínar réttar 18

Prófaðu tenglana þína 19

Litir á tenglum 19

Myndir - 21

Möppuvísanir 23

Listar 24

Bakgrunnsmyndir 25

CSS - Stílsnið27

Grunnatriði 28

Uppbygging CSS-skipana 29

CSS kóði annara skoðaður 29

XHTML skipanir endurskilgreindar 30

Faldar athugasemdir 31

Samhæfni við eldri vafra 31

Klasar (e. classes) 32

Tenglar 33

Ekki breyta leturstærð tengla eftir stöðu 33

Sýna staðsetningu gests í leiðakerfinu 33

Mismunandi útlit á tenglum 34

Breyta bakgrunni eftir stöðu tengla 34

CSS og texti 35

Stærðareiningar 35

Leturgerðir á vefsíðum 36

Útprintun með CSS 36

Aðrar aðferðir við að virkja CSS 37

Röð skipana skiptir máli 37

Box Model CSS 39

Staðsetning með CSS 40

Sérstafir 41

Algildar breytur 42

Litakóðar 42

Töflur - <table>43

Skipanir og eðli taflna 43

Útlitshönnun með töflum 45

Taflan sjálf - <table> 46

Raðir og sellur - <tr> og <td> 48

Töflur inni töflum 49

Töfludæmi með algengustu breytum 50

Tafla með föstum hliðum og flæðandi miðju .. 51

Töflur og myndir vinna saman 51

Rammar - <frame>52

Skipanir ramma 53

Ekkert <body> á yfirsíðum ramma 54

Undirskjöl ramma 55

Vísanir í ramma 55

Innskotsrammar 57

Innsláttarform - <form>58

Kóðinn sjálfur 59

Leitarorð og önnur lýsigögn 62

<base> 63

Hnitamyndir 64

Server side includes 65

Úreltar og heimskulegar skipanir 66

Skilyrði fyrir notkun þessarar bókar 67

Inngangur

Þessi handbók er hluti af þriggja bóka flokki um vefsmíðar. Bækurnar heita:

Vefsmíðar - Viðmót hönnun og verklag
Vefsmíðar - Kóðinn, HTML og CSS
Vefsmíðar - Dreamweaver og önnur tól

Upphaflega voru þær allar hluti af sömu bók sem hét "Vefsmíðar - hugsunin, smíðin og tæknin" en sökum umfangs og til þæginda hefur þeim verið skipt í þrjár bækur.

Bókin (og nú bækurnar) er búin að vera að þróast í gegnum kennsluferil minn í vefsmíðum, m.a. hjá Endurmenntunarstofnun Háskóla Íslands, Listaháskóla Íslands, Prenttæknistofnun og Margmiðlunarskólanum (ítarlegri lista yfir þá staði sem ég hef kennt og notað þessa bók á er að finna á <http://where.is/gunnar>).

Bókunum er ekki ætlað að vera tæmandi upplýsingabrunnur um allt sem snýr að vefsmíðum heldur frekar góður grunnur um þau atriði sem mestu skipta máli ef smíða á góðan vef.

Ég uppfæri bækurnar ekki sjaldnar en tvisvar á ári, í upphafi haust- og voranna og **til að tryggja að ekki séu úreltar útgáfur í notkun þá er óheimilt að nota í kennslu eldri útgáfur en eins árs**, enda sé nýrri útgáfa aðgengileg á where.is/handbok. Það er öllum í hag að virða það. **Skoðaðu "Skilyrði fyrir notkun þessarar bókar" aftast í bókinni ef þú ætlar að nota hana á einhvern annan hátt en fyrir þig sjálfan í þínu sjálfsnámi í vefsmíðum.**

Þessi bók sem þú heldur á fjallar um **fjallar um HTML og CSS kóðann** sem er á bakvið vefsíðurnar og er markmiðið þar að sýna allan þann kóða sem þarf til að smíða fallegar vefsíður sem eru lesnæfar í öllum vefskoðurum. **Þó að í dag sé vel hægt að smíða ágæta vefi án þess að kunna staf í kóða, þá er margt sem ekki er hægt að gera án þess að kunna hann** auk þess sem sá sem skilur og kann kóðann getur lagfært villur sem myndræn vefsmíðatól geta ekki. Skilningur á möguleikum og takmörkunum HTML er þar að auki öllum nauðsynlegur sem koma að hönnun og smíði vefsvæða.

Kóðinn

Á bakvið allar vefsíður er kóði sem stýrir því hvernig vefsíðan lítur út, þessi kóði kallast HTML og CSS. Í gegnum tíðina hafa verið til margar útgáfur af HTML og í raun hafa vafraframleiðendur einnig búið til sínar eigin útgáfur, með því að bæta við skipunum sem síðan eingöngu virka í einni tegund vafra. Sú tíð virðist blessunarlega vera að miklu leyti liðin og flestir framleiðendur styðja að miklu leyti nýjustu útgáfu HTML sem heitir XHTML 1.1. Ástæðan fyrir X-inu er að þetta er HTML sem fer eftir XML reglum.

Í bókinni er farið í gegnum XHTML skipanir í þeirri röð sem mér finnst vitrænt og eðlilegt fyrir byrjendur. Nákvæmlega hvaða útgáfu af XHTML þú notar er ekki aðalmálið, þessi bók byggir á XHTML Transitional sem leyfir ýmsar HTML 4 skipanir en stór meirihluti hennar er XHTML Strict.

Transitional þýðir ekki að kóðinn sé lélegri en Strict, fæstar vefsíður og vafrar styðja í dag XHTML að öllu leyti en sá staðall er engu að síður sá sem rétt er að fylgja. **Ef við gerum það þá tryggjum við að þær muni verða leshæfar í vöfrum framtíðarinnar, öfugt við þær síður sem engum stöðlum fylgja.**

Hvers vegna kóða?

Fyrir það fyrsta er varla hægt að smíða vefsíðu án þess að búa til kóða, spurningin er frekar um leiðirnar til að smíða hann og hversu mikla þekkingu vefsmíður þarf að hafa á honum til að geta brugðist við vandamálum.

En ástæðurnar fyrir því að læra og skilja kóðann eru margar. Það auðveldar þér mjög að nota þau tól og vinna þá vinnu sem tengist vefsmíðum ef þú hefur skilning á virkni kóðans auk þess sem villuleit og lagfæringar geta hreinlega kallað á þessa þekkingu.

Forsendur kaflans

Útgangspunktur kóðahluta þessarar handbókar er að hann innihaldi einungis þær skipanir XHTML staðalsins sem vitrænt og gott er að nota. Þó þetta sé vissulega afstætt þá eiga þær skipanir sem hér er fjallað um það sameiginlegt að vera notaðar af metnaðarfullum vefsmíðum um heim allan. Því fer fjarri að hér séu allar skipanir XHTML, heldur frekar úrval þeirra sem virka vel í nýlegum vöfrum, langflestar þær sem þarf til að smíða góða vefi.

Uppbygging skipana

Flestar XHTML skipanir byggjast upp á einföldu kerfi skipana, breyta og gilda.

```
<div align="center">
```

Skipun breyta gildi

Sumar skipanir þurfa engar breytur til að virka en aðrar gera ekkert ef ekki eru settar á þær viðeigandi breytur. Það sem oft veitist erfiðast byrjendum er að vita hvaða breytur eiga við hvaða skipun, en það kemur fljótt með æfingunni.

1. Ílát

Þau eru algengasta form XHTML skipana og þekkjast þau á því að þau eru opnuð þar sem þau byrja að virka og þeim er lokað þegar virkni á að ljúka.

```
<body> <- Ílát opnað
```

Hér inn í kemur það efni sem sést í vafra.

```
</body> <- Íláti lokað
```

2. Einstæðar skipanir

Nokkrar XHTML skipanir eru til sem þarf ekki að loka með sérskipun, þær virka þar sem þær eru, dæmi eru `
` sem býr til línubil og `` sem kallar á mynd.

Taktu eftir skástrikinu fyrir framan hægri odddklofann, það þýðir í raun að skipuninni er lokað um leið og það hefur verið framkvæmt sem hún kallar á. Þetta er nauðsynlegt til að fylgja XHTML staðlinum, sem krefst þess að öllum skipunum sé lokað. Skipanirnar sem notaðar eru í þessari bók sem þetta á við um eru: `
`, ``, `<frame />`, `<meta />` og einhverjar fleiri á stangli. Reglan er einföld: **Ef skipun hefur ekki lokun síðar í skjali þá þarf að loka henni á þennan hátt.**

3. Köll í stafi

Síðan eru skipanir sem kalla á ákveðna stafi, eru í raun kóðuð leið til að kalla á sjaldgæfa stafi. Þær byrja allar á `&` og enda á `;`; Dæmi: `Þ` kallar á stórt Þ, `©` kallar á © osfrv. Þetta skipanaform er einnig hægt að nota með númerum, þá skrifast © með `©`;

4. Skilgreining skjals - Document Type Definition (DTD)

Að síðustu er skipun sem á að fara allra efst í skjalið okkar, hún segir til um hvaða HTML staðli við teljum okkur vera að fylgja í því skjali. Hún er sett fyrir ofan `<html>` í skjalið. Sú sem ég nota þessa dagana er:

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
```

Segja má að allar XHTML skipanir séu ílát, því bæði liðir 1 og 2 að ofan eru ílát, liður 3 eru ekki skipanir og liður 4 er í raun ekki XHTML skipun heldur segir vafra hvaða útgáfu af HTML skjalið byggir á.

Helstu reglur XHTML

Allur kóði verður að vera í lágstöfum.

`<body>` en ekki `<BODY>` eða `<Body>`.

Allar skipanir verða að lokast með skástrikslokun aftar í kóða.

`<p class="midjad">`Hér sé málsgrein`</p>` Þó er leyfilegt, þegar við á, að loka í enda sjálfrar skipunarinnar og er þá skástrikið sett affast í hana, t.d: `
` og ``. Athugaðu að þá verður að vera bil á undan / því annars ruglast eldri vafrar, þekkja ekki skipunina.

(Inline) skipanir sem vinna með texta (a, b, span ofl.) mega ekki innihalda (block) skipanir sem vinna með rými (p, h1-h6, table, div o.fl.)

Þú mátt skrifa `<div>Efni</div>` en ekki `<div>Efni</div>`

Setja verður rétt DTD á síður.

Þetta að neðan er það sem ég nota oftast.

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
```

Segir hverskonar skjal þetta er, hvaða tungumál og staðal það fylgir og hvar hann er að finna.

Ílát mega ekki krossast en geta verið hvort inni öðru.

``Feitletraður og `<i>`skáletraður texti`</i>` er rétt notkun en ``Feitletraður og `<i>`skáletraður texti` </i>` er röng notkun og getur valdið því að skipanir virki ekki rétt, þó allt virðist í lagi í þínum vafra.

Gæsalappir verða að umlykja gildi breyta.

`<p class=midjad>` er bannað, á að vera `<p class="midjad">`. Gæsalöppum verður að loka! Annars er skjalið í tætlum framað næstu gæsalöpp, það sem er þar á milli hverfur í sumum vöfrum.

Munurinn á XHTML og HTML 4

Munurinn er í sjálfu sér ekki mikill og fyrir þá sem hafa skrifað HTML á ákveðinn hátt er lítið mál að skipta yfir í XHTML. XHTML er í raun HTML sem fylgir XML reglum, er strangari en HTML 4 staðallinn og leyfir síður villur. Margt af því sem að ofan er talið er dæmi um hluti sem mátti í eldri útgáfum HTML en má ekki lengur. Margir hafa lengi skrifað HTML eftir flestum þessum reglum og er þá lítið mál að skipta í XHTML.

Sannreyndu kóðann þinn

Besta leiðin til að sannreyna hvort kóðinn þinn stenst staðalinn er að athuga það með gæðastöðlunartóli W3C (World Wide Web Consortium, <http://www.w3.org/>) sem les í gegnum kóðann og segir þér hvort í honum eru villur og þá hverjar. Beina tengla í sannreyna (validator) fyrir CSS og XHTML er að finna á <http://where.is/webdesign>

Hvernig skrifum við (X)HTML?

(X)HTML skjöl eru venjuleg textaskjöl sem hægt er að vinna í öllum ritvinnsluforritum sem geta vistað hreinan texta. Hægt er að smíða vefsíður í allskonar forritum og eru þau misvel fallinn til þess en öll eiga þau eitt sameiginlegt: Þau búa til (X)HTML kóða sem vefvafrar túlka og birta sem vefsíður. Möguleikarnir eru:

1. Sérhæfðir textaritlar (Text editors)

(X)HTML skjöl eru oft ofin í sérhæfðum textaritlum og eru til margir slíkir. Þeim fjölgar ört sem styðja XHTML og þeir sem ég hef notað eru Homesite frá Macromedia og NoteTab (sem er til í ókeypis útgáfu). Mörg þessara forrita innihald einnig ýmiskonar fjölva sem hjálpa okkur að skrifa kóðann, ef við höfum ákveðna grunnþekkingu.

2. Myndrænir vefstólar (WYSIWYG editors)

Einnig er hægt að vinna vefsíður í myndrænum vefstólum (Dreamweaver, GoLive og Frontpage eru þekktust) og er það sú leið sem flestir nýliðar fara. XHTML-stuðningur þessara forrita er mjög mismunandi, Dreamweaver MX virkar ágætlega.

3. Vefumsjónarkerfi

Leið sem verður sífellt algengari og batnar líka að gæðum, er sú að nota vefumsjónarkerfi. Í þeim er miðað við að notendur með litla tölvuþekkingu geti sett efni á vefsíður án þess að hafa skilning á því sem að baki liggur. Flest svona kerfi nota vefvafra sem biðlara, þ.e. efnið er sett inn á sérstökum vefsíðum. Staðlastuðningur þessara kerfa er jafn misjafn og þau eru mörg, ef þau styðja ekki XHTML þá eru minni líkur á að þau séu metnaðarfull og vel unnin á öðrum sviðum. Góð kerfi leyfa þér að ráða alfarið hvernig kóði er settur inn.

4. Myndvinnsluforrit

Mörg myndvinnsluforrit (ImageReady frá Adobe og Fireworks frá Macromedia eru algengust) geta vistað HTML kóða og eru í sumum tilvikum fljótlegasta leiðin til að flytja grafíska hönnun á HTML form. Þau koma þó aldrei í stað HTML ritla því ekki er hægt að vinna beint með kóðann í þeim, né laga áður unnar síður.

5. Hefðbundin forrit með HTML vistunarmöguleika

Þau eru yfirleitt verulega ofhlaðin möguleikum sem ekki nýtast og eru jafnvel fyrir við vefsíðugerð. Og kóðinn úr þeim stenst sjaldan væntingar, er stundum margfaldur á við þann sem raunverulega þyrfti til. Word, Excel og PowerPoint skila t.d. iðulega miklu magni af mjög slæmum kóða sem enga staðla stenst sem virkar einungis í Internet Explorer.

Skráanöfn, hvað má og hvað ekki

Ekki nota önnur tákni í skráanöfnum en _ (undirstrik), - (bandstrik) og . (punkt). Þó MacOS og Windows leyfi bil í skráanöfnum gerir UNIX það ekki alltaf, notaðu _ (undirstrik) í stað bils. **Notaðu ALDREI íslenska stafi né bil í skráa- eða myndanöfn**, það virkar oft illa á vefnum.

Lang öruggast er að venja sig á að nota alltaf litla stafi í öllum skráa- og myndanöfnum. Sum stýrikerfi líta á stórt A og lítið a sem sama stafinn, önnur ekki. Eins er sjálfgefið í Windows umhverfum að viðskeyti eru falin. Ef þú vistar gif mynd, þá bæta Windows stýrikerfi .gif aftan við nafnið sem þú gafst myndinni, en án þess að sýna þér það. Þú getur stillt Windows til að sýna þér fullt nafn skráa með því að opna Windows Explorer, fara með músina í Tools -> Folder Options, velja þar View og taka af hakið fyrir framan "Hide file extensions for known file types".

Notaðu alltaf viðskeytin .html eða .htm fyrir html-skjöl og .gif og .jpg fyrir myndanöfn. Það einfaldar þér töluvert að hafa yfirlit yfir innihald vefsins og einnig eiga sumir vafrar það til að sýna skrár sem ekki heita .html eða .htm sem hreinan texta, þ.e. kóðinn birtist á skjánum. Enginn munur er á því hvort notað er .html eða .htm í skjölum. Fleiri viðskeyti eru til fyrir sértækar lausnir, tæpt er á sumum þeirra í síðasta hluta bókarinnar.

Umbrot kóða

Það auðveldar mjög alla vinnu við XHTML-skjöl ef þú hefur bil á milli lína þar sem það á við, eins og gert er í dæmum í bókinni. Ef skjali er þannig skipt niður í rökréttar einingar ertu miklu fljótari að átta þig á hvað gerir hvað, þegar þú kemur að verkinu næst.

XHTML er alveg sama um línubil og inndrætti sem ekki eru skilgreind með XHTML skipunum. Eins getur þú notað inndrög ef þér finnst það þægilegra. DW notar inndrög.

Villuleit

Ef kóði birtist í vafraglugganum er ástæðan oftast einni gæsalöpp of eða van. Ef eitthvað virkar ekki, lestu þá kóðann í kringum vandamálið vel yfir, það er oft ein gæsalöpp sem vantar, stór stafur í tengli í skrá, eða eitthvað slíkt sem að er. Gott er að nota staðreyni (validator) til að finna slíkar villur.

CSS - Stílsnið

Málgreinar merkar CSS vísa til þess hvaða CSS breytur er handhægt að nota á skipanirnar sem fjallað er um á þeirri síðu. Þær eru þó aldrei tæmandi þar sem breytur CSS eru mjög margar og virka ekki allar í öllum algengum vöfrum. Í CSS kaflanum er síðan fjallað um hvernig þú notar stílsniðin.

Ef þú veist ekki hvað CSS er þá geturðu auðvitað kíkt á CSS kaflann aftar í bókinni en ég mæli ekki með að þú skoðir hann fyrr en þú hefur náð góðum tökum á (X)HTML, kóðinn er nægilega ólíkur til að rugla flesta í ríminu sem reyna að læra þá á sama tíma. Í stuttu máli þá er CSS notað til að stilla útlit og umbrot efnisins en XHTML kóðinn setur upp grunneiningarnar og oft líka staðsetningu þeirra á skjánum. Mismunandi er hversu langt menn ganga í að nota CSS en leturstærðir ættirðu skilyrðislaust að stilla með stílsniðum.

Notaðu vafrana

Vefsmíðaforrit eru ekki vafrar og þú verður því að skoða vefina þína í vefskoðurunum sjálfum, í það minnsta í Firefox og Explorer, því það notar enginn WYSIWYG forrit til að skoða vefsíðurnar þínar og ekkert þeirra sýnir þær nákvæmlega eins og vafrarnir sjálfir.

Vafri finnur ekki skrá

Ef þú lendir í vandræðum með að vafri virðist ekki finna skrá eða mynd sem þú telur að sé í ákveðinni möppu þá er til ágætis leið til að komast nær rót vandans. Þú byrjar á að opna einhverja skrá sem er í þessari ákveðnu möppu og eyðir síðan skráarnafninu aftan af slóðinni í Location/Address glugganum og ýtir á Return. Þá kemur upp listi með öllum skjölum í möppunni ásamt stærðum þeirra. Þá getur þú smellt á skránar og ef þær opnast ekki í vafranum þá veistu að skráin sjálf er eitthvað vangæf (t.d. TIF mynd sem heitir .gif eða eitthvað þessháttar). Þessi aðferð virkar ekki í öllum útgáfum Explorer f/Windows.

Flýtiminnis vandamál

Eitt vandamál sem kemur stundum upp, sérstaklega við smíði rammasíðna og innsláttarforma er að breytingar virðast ekki skila sér inní vafrann. Ef þetta gerist hjá þér þá skaltu byrja á að fara í Tools -> Internet Options -> Settings og stilla þar á Every visit to the page. Þetta þýðir að IE á alltaf að hlaða inn nýjustu útgáfu síðunnar.

Næsta sem þú gerir er að venja þig á að nota alltaf svokallað Super Reload (Slaufa-Reload á Mac en SHIFT-Reload á Windows) þegar þú vinnur slíkar síður.

Þetta tvennt dugir yfirleitt til að hægt sé að vefa án vandræða. Athugaðu að það er ekkert víst að þú lendir í þessu en það er hvítleitt þegar það gerist.

Þinn kóði og minn kóði

Skoðaðu kóða annara

Sú leið sem allir notuðu áður fyrr til að læra HTML og fullt af fólki notar í dag er að skoða HTML kóðann á síðum sem aðrir vefa.

Þú getur skoðað kóðann á bak við vefsíður með tveimur aðferðum:

1. Ferð í **View -> Source** í vafra þegar síðan er opin eða
2. Vistar opna síðu í vafra með **File -> Save**

Þú getur notað **View - Source** til að skoða kóða venjulegra vefsíðna og yfirsíðna ramma en þarft að smella með hægri takkanum yfir þeim ramma sem þú vilt sjá kóðann fyrir og velur þar **View - Source** (ekki í öllum vöfrum). Þá á að koma upp gluggi með kóðanum fyrir þessa vefsíðu og þá getur þú skoðað hann, vistað og síðan prófað að fikta og breyta til að sjá hvernig síðan er uppbyggð.

Þegar þú notar **SaveAs** til að ná í síðurnar með myndum og öðrum tengdum skjölum þarftu að passa þig á því að allir vafrar breyta kóðanum eitthvað (t.d. vísunum í myndir) og IE breytir kóðanum meira en þörf er á, setur inn eigin skipanir og breytir sumum þeirra sem eru í honum. Síðan eru líka til sérhæfð tól til að ná í heilu vefina sem er betra að nota ef þú vilt meira en eina síðu.

Skoðaðu líka **<base>** kaflann til að sjá aðra aðferð að sama marki sem ég nota stundum sjálfur.

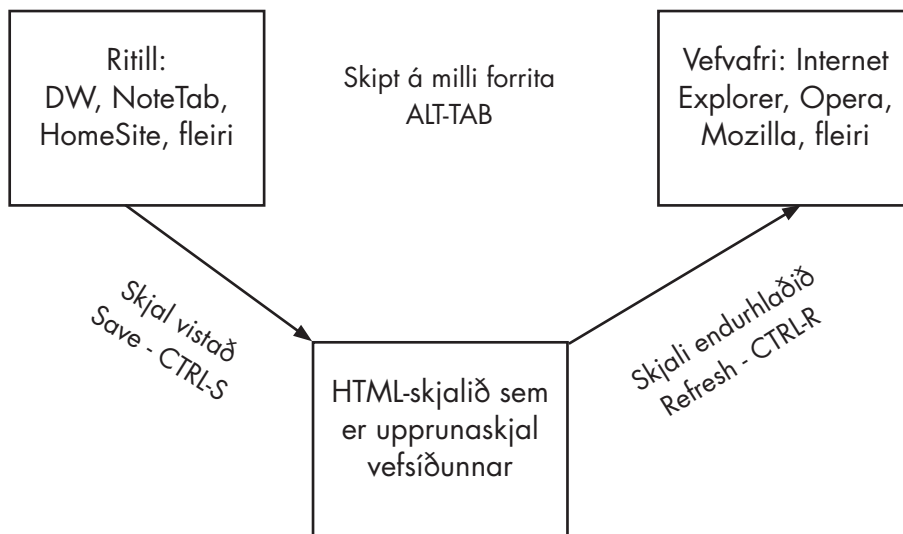
Eftir að þú ert búinn að ná í síðurnar er gott að setja **border** á töflurnar til að sjá uppbyggingu þeirra, þetta geri ég oft í Dreamweaver. Þá ættirðu að sjá mjög vel hvernig viðkomandi vefsmiður hugsaði uppbygginguna og lært sf því.

Höfundarréttur

Margt og mikið hefur verið rætt um höfundarrétt og þjófnað í sambandi við vefsíðugerð og notkun þessarar aðferðar. Það hlýtur alltaf að vera matsatriði að hversu miklu leyti óhætt er að afrita annarra manna hugverk á þessu sviði, en það breytir því ekki að HTML er umbrotsmál (að mestu leyti) og ég sé ekki fyrir mér að einhver taki út einkaleyfi á feitiletruðum texta eða öðru slíku. Púmalputtareglan sem margir hafa, er sú að nýta sér stundum annarra manna síður til að byrja á vef sem síðan þróast áfram og verður á endanum nýr vefur. Að nota grafíska hönnun annara er síðan augljóst brot á alþjóðlegum höfundarréttarlögum og það sama á við um JavaScript og álíka hluti, nema á þeim vefsvæðum þar sem skýrt er tekið fram að slíkt megi.

Vinum saman, dreifum hjólunum

Það er útilokað að ætlast til þess að allir finni upp hjólið aftur og aftur í þessum efnum, frekar en í öðrum. Reynum frekar að bæta við þau hjól sem við höfum nú þegar, og leyfum öðrum að njóta okkar verka til eigin nota. Það er hvort eð er ekki hægt að hindra fólk í að skoða og nota kóða síðna þinna.



Flýtilyklar og vanaverk

Það er augljóst flestum þeim sem unnið hafa mikið við tölvur að þó músin sé að mörgu leyti þægileg, þá er hún frekar seinlegt tæki, sérstaklega þegar verið er að framkvæma sömu aðgerðir aftur og aftur. Og í vefsmíði er einmitt verið að framkvæma sömu aðgerðir aftur og aftur og aftur og aftur ...

Grunnaðgerðir vefsmíða

Ferillinn er á þessa leið: Skrifað í ritli, vistað, hlaðið inn í vafra til að skoða. Þessar aðgerðir er yfirleitt allar hægt að gera með flýtilyklum sem innbyggðir eru í vöfrum og þeim ritli sem þú notar.

Vista síðu í ritli = CTRL-S (slaufa-S á Mac).

Skipta á milli forrita = halda niðri ALT og ýta á TAB

Endurhlaða síðu í vafra = CTRL-R (slaufa-R á Mac)

Í mörgum vefsmíðaforritum geturðu ýtt á einn takka (**F12 í Dreamweaver**) og þá birtist vefsíðan í vafranum. Ef þú notar þessa aðferð, notaðu hana þá eingöngu og ekki reyna að nota Reload til að fá uppfærslur fram, það virkar t.d. ekki í Dreamweaver sem notar bráðabirgðaskrár fyrir vafrann, í sumum forritum er í lagi að blanda þessu saman.

Copy/paste - afrita/líma

Notaðu "afrita og líma" (copy and paste) eins mikið og þú getur! Það flýtir mjög fyrir þér og snarminnkar líkur á innsláttarvillum sem eru mjög algengt vandamál í vefsíðugerð. Þetta á sérstaklega við um vefslóðir (URL), þær er lang öruggast að afrita úr Location glugga vafra þegar vefurinn sem þú ætlar að vísa á er opinn þar.

Afrita = Ctrl-C (slaufa-C á Mac)

Klippa = Ctrl-X (slaufa-X á Mac)

Líma = Ctrl-V (slaufa-V á Mac)

Einfalt XHTML skjal

XHTML er mjög einfalt í grunninn, en möguleikar miklir þegar litið er á öll tög (skipanir) sem til eru í málinu. Grunnskipanirnar eru fáar og sjást þær á næstu síðu.

Sett upp á réttan hátt lítur einfalt XHTMLskjal svona út:

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

<html>

<head>
<title>Jón Jónsson</title>
<meta http-equiv="content-type" content="text/html; charset=iso-8859-1" />
</head>

<body>
  Ég heiti Jón Jónsson og ég á þessa vefsíðu.
</body>
</html>
```

Það sem sést í vefvafra er:

Ég heiti Jón Jónsson og ég á þessa síðu.

Grunnskipanir

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
```

Þetta kallast Document Type Declaration (DTD) og segir til um hvaða staðli skjalið fylgir og hvar hann er að finna. Lista yfir DTD er að finna á www.w3c.org og er einnig hægt að kalla fram í DW og fleiri forritum.

<html>

segir vafra (browser) að skjalið sé HTML skjal. Lokast með </html> í enda skjals.

<head>

Heldur utanum skipanir sem koma ekki fram á síðu en gefa vafra ýmsar upplýsingar. </head> lokar head.

<title>

Texti sem birtist í Favorites/Bookmarks, sem titill síðu efst í vafraglugga og á verkslá neðst á skjá. Lokast með </title> Athugaðu að <title> ílátíð verður að vera inni <head> ílátinu.

Athugaðu að það skiptir miklu máli að hafa lýsandi texta í <title> til að fólk finni aftur síðuna þína sem það hafði fyrir að setja í Favorites/Bookmarks.

```
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1">
```

Þessi lína segir til um hvaða stafasett vefsíðan notar, þetta er nauðsynlegt að hafa til að allir vafrar viti hvaða stafasett á að nota til að birta síðuna. Athugaðu að <meta> ílátíð verður að vera inni <head> ílátinu.

<body>

Segir vafra að hér að neðan sé efni skjals. Í <body> skipun er einnig hægt að setja breytur sem stýra litavali í skjali og spássíu ef þú vilt þjónusta vafra foraldar. Mun betra er að venju að nota CSS. Lokast með </body>.

Aðeins má hafa eina <body> skipun í hverju skjali.

CSS

```
body {
background-color: white;
color : black; }
```

Fyrirsagnir - <h1> til <h6>

<h1> - <h6> Í þessum skipunum höfum við tilbúna 6 mismunandi stærðir fyrirsagna sem við getum síðan endurskilgreint að vild með CSS. Ef við þurfum fleiri tegundir þá er einfalt að bæta á þær CSS-klösum.

Grunnotkun er svona: <h1>Hér er stór fyrirsögn</h1>

En það er ýmislegt sem er ekki eins og við ættum kannski von á.

1. <hx> skipanirnar innihalda <p> og því kemur tvöfalt línubil (ný málsgrein) fyrir ofan og neðan innihald skipunnarinnar. Þessu er hægt að breyta með stílsniðum (sjá CSS-kassa hér fyrir neðan).

2. <h6> er minnst og <h1> er stærsta fyrirsögnin, þetta venst þó frekar fljótt.

3. Þar sem skipanirnar innihalda tölu þá þarf alltaf að breyta lokunum í hvert skipti sem fyrirsögn er minnkuð eða stækkuð. Þetta kemur þó ekki að sök ef CSS er notað til að stýra stærðum og heldur ekki í myndrænum vefsmíðatólum.

Athugaðu að úrelt leið til að setja upp fyrirsagnir er að nota skipunina, þú getur séð þá leið á mörgum eldri vefsíðum, vefsmíðatólum og bókum. Notaðu mun frekar CSS en skipunina.

CSS

```
h1 {  
font-family: sans-serif;  
font-size : 30px;  
padding-left: 30px;  
padding-top: 16px;  
margin-top: 0px;  
margin-bottom: 1px;  
}
```

Einfaldar leturbreytingar

, <i>, <sup> & <sub>

Feitletraður texti (XHTML vill að þú notir)

<i>Skáletur - notist með varúð v/leshæfni</i> - það er óþægilegt að lesa skáletraðan veftexta í mörgum vöfrum. (XHTML vill að þú notir)

^{Yfirskrift} _{Undirskrift} - Þú getur auðveldlega breytt útilitinu með CSS.

Umbrot texta

<p>Málgrein</p>

<p>Skilgreinir málgrein (paragraph)</p>. Þýr til tvöfalt línubil fyrir ofan og neðan skipun. Í raun er sama virkni í því að nota

 fyrir ofan og neðan efni eins og að nota <p> en meiri sveigjanleiki og skynsamlegri uppbygging skjals næst með notkun <p>. Mest notaða breytan er líklega **align**, <p align="center | left | right | justify">Efni</p> en eins og áður er mun betra að nota CSS.

Brýtur línu (line break), sama og ýta einu sinni á Enter í textavinnslu. Hægt er að setja fleiri
 saman og búa þannig til meira línubil en það er sjaldnast æskilegt, tvö eru yfirleitt hámark þess sem notað er.

Textasvæði

 afmarkar texta án þess að hafa áhrif á útlit hans. er ekki rýmisskipun, heldur nær hún yfir ákveðið magn af texta og afmarkar hans útlit, en ekki hegðun. Þú getur síðan skilgreint að vild útlit á það svæði með CSS. Ég nota þessa skipun lítið núorðið, reyni frekar að nota klasa á viðeigandi skipanir, t.d. , , <i> eða .

<div>Svæði</div>

Skipun sem getur tekið sömu breytur og <p> og virkar svipað við fyrstu sýn er <div>. Hún býr til tvöfalt línubil fyrir ofan og neðan efnið sem er í ílátinu en hefur sjálf engin áhrif önnur. Síðan er hægt að setja á hana CSS klasa ef útlit innan <div> ílátsins </div> á að breytast. <div> er í raun hugsuð sem rýmisskipun, þ.e. ætluð til að skilgreina svæði á láréttuna og lóðréttuna og þess vegna virka t.d. breytur eins og **text-align: justify** á <div> á meðan þær virka ekki á . <div> er grunnskipunin þegar útlit er smíðað með CSS í stað taflna.

Láréttar línur - <hr />

<hr /> á varla heima hér en ekkert betur annars staðar. Þessi skipun var mikið notuð áður fyrr, en takmarkanir hennar urðu til þess að flestir fóru að nota myndir í hennar stað. Hún býr til einfalda lárétta línu á skjáinn en með CSS er hægt að hafa nokkur áhrif á útlit línunnar, t.d. með bakgrunnsmynd eða breiddar- og hæðarstillingum. Ég nota þetta sáralítið sjálfur.

Faldar athugasemdir

Til að setja inn texta sem ekki birtist á vefsíðu er hægt að nota

`<!-- Ósýnilegur texti fer hér -->`. Þetta er handhægt að nota til að punkta ábendingar um hvað gerir hvað, sem auðveldar þér að skilja hvað þú varst eiginlega að gera þegar þú ætlar að bæta við og breyta síðar meir. Hvort sem er fyrir þig sjálfan eða efnisritara sem vinna í kóða, þú merkir t.d. `<!-- Hér hefst meginmál-->`. Þetta er einnig hægt að nota til að geyma kóða sem þú ætlar að nota síðar. Hafðu í huga að hægt er að lesa þennan texta þegar kóði er skoðaður úr vafra.

Vefsíður eru aldrei eins hjá öllum

Sá sem skoðar síðurnar þínar gæti verið með aðra uppsetningu á leturgerðum og stærðum en þú. Vefsíðan þín lítur þá allt öðruvísi út hjá honum. Munur á útliti síðna getur verið mjög mikill, sérstaklega ef notandi hefur breytt stillingu leturs. Ef þú vilt vera viss um að vefir þínir líti vel út allsstaðar þá er eina leiðin sú að skoða þá í mismunandi veflesurum og stýrikerfum.

Röð íláta

Athugaðu að ekki mega allar XHTML-skipanir vera ílát fyrir hvaða skipun sem er. Skipanir sem skilgreina rými eins og (`div`, `form`, `h1-h6`, `ol`, `p`, `pre`, `table`, `ul`, `dd`, `li`, `td`, `tr` o.f.l., kallast block á ensku) mega t.d. ekki vera inni í skipunum sem einungis hafa áhrif á texta (`a`, `b`, `br`, `em`, `i`, `img`, `input`, `span`, `strong`, `sub`, `sup`, `iframe` o.f.l., kallast inline á ensku).

Þetta er því miður aðeins flóknara en að ofan er talið, t.d. mega `<p>` og `<h1-6>` ekki fara inn í `<p>` eða `<h1-6>` en `<td>` verður að fara inn í `<tr>` sem verður að fara inn í `<table>`. Ef þú ert í vafa þá skaltu sannreyna (validate) skrána þína.

Tenglar - <a> - Stiklur, krækjur, hlekkir

Tenglar eru grunnur Veraldarvefsins og það sem hann byggir tilvist sína á. Vefsíður væru lítils virði ef þær tengdust ekki hvor annari og án þeirra væri enginn veraldarvefur.

Þeir eru mjög einfaldir í notkun, helsta hættan felst í stafsetningarvillum í slóðum, slíkar leiða alltaf til þess að tengillinn virkar ekki og þá er honum oftast betur sleppt.

Textatengill í vef Háskóla Íslands:

```
<a href="http://www.hi.is/">Vefur Háskóla Íslands</a>
```

Textatengill í skjal í sömu vefmöppu og skjalið sem tengillinn er í:

```
<a href="amma.html">Amma mín</a>
```

Myndtengill í vef Háskóla Íslands

```
<a href="http://www.hi.is/"></a>
```

Pósttengill

```
<a href="mailto:gunnar@where.is">Sendu mér póst</a> Ef smellt er á mailto tengil þá kemur upp "bréfsefni" til að senda viðkomandi tölvupóst.
```

Textatengill í skjal í undirmöppu sem heitir english:

```
<a href="english/amma.html">Amma mín</a>
```

Hafðu slóðirnar þínar réttar

Til að minnka líkur á brotnum útverum tenglum er best að byrja á því að opna í vafra, vefinn sem þú ætlar að tengja í. Þegar hann hefur opnast í vafranum hjá þér þá ferðu í Address/Location reitinn og afritar slóðina þaðan og límir á réttan stað í vefsmíðaforriti þínu. Þannig tryggirðu að slóðin sjálf sé rétt.

Prófaðu tenglana þína

Prófaðu alla tengla vel áður en þú lætur vefinn þinn frá þér. Brotinn tengill þýðir að vefgestur kemst ekki á síðuna sem hann ætlaði á, innihaldið þitt þarf að vera mjög áhugavert til að hann nenni að koma aftur! Sum vefsmíðaforrit eru með innbyggðum fidusum sem geta prófað tengla, þau geta þó aðeins prófað hvort tengillinn virkar en ekki hvort hann vísar á rétta síðu, það er bara hægt að gera með því að smella á hann og sjá hvert hann fer.

Litir á tenglum

Rétta leiðin til að stýra litum og öðru útliti á tenglum er að skilgreina CSS breyturnar sem þú sérð hér neðst á síðunni. Ekki breyta litnum á tenglum með því að endurskilgreina `<hx>` eða samsvarandi. Þú getur líka notað gömlu leiðina; að setja breytur á `<body>`, `<body link="red" vlink="blue" alink="blue">`. en mun skynsamlegra er að nota tenglaskipanir CSS.

CSS

Hér eru algengar grunnbreytur á tengla í CSS. Röðin skiptir máli hér, röðin að neðan er rétt. Meira um þetta í CSS-kaflanum.

Textabreyturnar aftarlega í CSS kaflanum virka langflestar á tengla, forðastu samt að stækka letrið eða fita það (bold) nema þá fyrir allar útgáfur tenglanna, ef þú t.d. stækkar `hover` letur þá hoppar öll síðan til og frá þegar farið er með músina yfir tenglana.

```
A{text-decoration: none;}  
  
A:link {color: #990000;}  
A:visited {color: #009900;}  
A:hover {color: #999900;}  
A:active {color: #009900;}
```

Þessi gildi tækju af undirstrikun á tenglum og settu rauðan sem lit á tengla sem ekki er búið að skoða, grænan á tengla í vefi sem búið væri að skoða og gulan þegar farið er með músina yfir.

Afhverju eru alink og vlink eins?

Taktu eftir að sami litur er á alink og vlink, það er vegna óstaðlaðrar hegðunar Internet Explorer, sem notar alink breytuna vitlaust. Í tilvikum þar sem IE ætti að nota vlink notar hann stundum alink sem gerir það að verkum að liturinn þarf að vera sá sami ef vel á að fara.

Tenglar inni skjöl

`` Segir vafra að hér sé merki 1 staðsett. Hér er einnig hægt að nota `` og er það líklegra til að virka í eldri vöfrum á meðan `id` er leiðin sem staðallinn beinir okkur. IE skilur líka stundum `id` illa. Ef þú vilt vera alveg viss um að merkin þín virki í öllum vöfrum þá skaltu nota `name`, `id` virkar t.d. ekki í NS4, þér er líka frjálst að nota hvorutveggja. Þú getur sett `id` breytuna beint á allar skipanir, t.d. `<body id="top">` ef þú vilt hafa merki efst á síðunni. Þetta síðasta virkar ekki með `name` breytunni.

``Að merki í sama skjali`` Segir vafra að fara að merki 1 í sama skjali.

``Að merki í öðru skjali`` Segir vafra að ná í skjalið og fara að merki 1 í því.

Tenglar í nýja og nefnda glugga

``Opna í öðrum glugga`` Þegar smelt er á þennan tengil þá opnast `skjal.html` í glugga sem er nefndur `main`.

Reglan er sú að ef til staðar er gluggi með gildi `target` breytunnar þá opnast skjalið í honum, annars opnast nýr gluggi. Það pirrar marga ef oft er verið að opna nýja glugga og ég mæli með að þú haldir þessu í lágmarki.

Þú getur ekki stillt stærð gluggans né staðsetningu með þessari breytu heldur þarftu Javascript til þess.

Nokkur gildi eru frátekin í XHTML og best að forðast að nota `_` (undirstrik) í upphafi nafni gildis. Dæmi um frátekin gildi eru `_top` og `_blank`. Meira um þetta í rammakaflanum því target er mikið notað í `<frames>`.

Tenglaminni hreinsað

Eftir að þú ert búinn að vera að vinna vefinn þinn þá ertu yfirleitt búinn að opna allar síður hans í vafra og sérð því alltaf farna (`vlink`) litinn á þeim, öfugt við nýja gesti sem munu sjá alla tengla á forsíðu með ófarna (`link`) litnum. Til að sjá vefinn þinn eins og gestirnir sjá hann þá getur þú látið Internet Explorer gleyma því hvaða síður þú ert búinn að heimsækja. Þú ferð í **Tools -> Internet Options -> Clear History**. Í Firefox ferðu í **Tools -> Options -> Privacy -> History Clear** eða setur upp Web Developer viðbótina sem gefur þér þessa virkni og meira til í einföldu fellivali.

Áttaðu þig samt á því að þetta þýðir að IE man ekki lengur neinar slóðir sem þú hefur heimsótt og klárar þær því ekki fyrir þig þegar þú ert að slá þær inn. Hugleiddu að nota einfaldlega annan vafra til að flakka á Netinu og nota IE eingöngu til vefsmíða, þá leysirðu þetta vandamál. Aðrir vafrar gætu t.d. verið Opera, Mozilla eða Firefox (strípuð útgáfa af Mozilla sem ég nota mikið þessa dagana).

Myndir -

Grunnskipun mynda er ``

Helstu breytur eru þessar, þær ljósu eru minna notaðar:

```

```

`alt` texti er notaður af effirtöldum ástæðum, í mikilvægisröð:

1. Talgervlar blindra og sjónskertra nota þann texta til að segja vefgesti hvert myndatenglar vísa.
2. XHTML staðallinn krefst breytunnar á allar myndir.
3. Kemur í stað myndar ef hún skilar sér ekki til vefgests.
4. Hjá þeim sem nota vafra sem ekki hafa myndræna möguleika.

Burtséð frá því að `alt` **breytan er orðin hluti af XHTML staðlinum**, þá er sama hvort gengið er út frá manngildissjónarmiðum eða peningalegum: Blindir og sjónskertir eru það stór hópur að það að gera þeim verulega erfitt fyrir að finna upplýsingar á vefnum þínum er hrein heimska, sérstaklega þegar lítið mál er að setja `alt` breytu á allar myndir sem eru tenglar. Athugað að `alt` breytunni er ekki ætlað að koma sem bólutexti yfir mynd (þó að IE geri það) heldur áttu að nota `title` breytuna til þess.

Breyturnar sem koma hér fyrir neðan er allar líka hægt að stilla með CSS, sjá CSS kóða neðar og annars staðar í þessum kafla.

`width` og `height` skilgreina stærð myndar, hægt er að teygja og toga myndir með þessum skipunum þó ekkert vit sé í því nema til að prófa. Hægt er að skoða hæð og breidd mynda með því að velja View Image þegar hægri músartakka er haldið yfir mynd. Í Explorer er smellið með hægri takka yfir mynd og valið Properties en það sýnir þér ekki endilega raunverulega stærð myndarinnar, heldur stærðina sem skilgreind er með `width` og `height` í kóðanum.

Mesta notagildi `width` og `height` breytanna er að ef þær eru skilgreindar, þá hleðst texti síðunnar strax inn og hægt er að lesa hann á meðan vafrinn er að hlaða inn myndum. Ekkert vit er í að nota þessar skipanir til að minnka myndir, það lengir einungis hleðslutímann og ef þú notar þær til að stækka myndirnar þá færðu ljótar myndir. Réttu leiðin er að stilla rétta stærð myndanna í myndvinnsluforriti.

Afturámóti er hægt að nýta sér **width** og **height** til að prófa sig áfram með hvað myndir eiga að vera stórar í endanlegri útgáfu. Ef aðeins annaðhvort er skilgreint helst myndin í réttum hlutföllum.

align möguleikar sem vitrænt er að nota eru **right** og **left**, en ekki til að staðsetja myndinar (sem þær voru upphaflega ætlaðar í) heldur til að láta texta flæða með þeim. Staðsetningu mynda (og annars efnis) er best að skilgreina með töflum.

border stillir hvort rammi myndast í kringum myndir sem eru tenglar. **border="0"** gefur engan ramma og er sjálfgefinn í DW þegar tengill er settur á mynd. Einfalt að gera í CSS fyrir allar myndir á vefnum.

vspace og **hspace** skilgreina autt pláss í kringum myndir, þ.e. ýta texta og öðrum myndum frá, **vspace** á lóðréttuna en **hspace** á láréttuna. Gildi eru í skjápunktum (pixels). Í raun er mun betra að nota CSS en þetta er heldur fljótlegra þegar hver mynd er einstök.

CSS

Ég nota CSS frekar lítið með myndum, nema þá bakgrunnsmýndum. En þú getur t.d. slökkt á ramma utan um mynd. Þessi rammi kemur sjálfkrafa ef myndin er tengill.

```
img {border: 0;}
```

Ef þú vilt stilla spássíu fyrir aðeins eina hlið gætirðu stillt það með **margin-left** eða **margin-right**, ef þú værir með margar myndir í sömu stærðum þá gætirðu líka stillt **width** og **height**. Þú þarft auðvitað að nota klasa í það ef myndirnar þínar eru ekki allar jafnstórar.

Ef þú vilt fá grannan svartan ramma utan um efni, í þessu tilfalli mynd þá notarðu eftirfarandi kóða:

```
img
{
border-width: 1px;
border-color: black;
border-style: solid;
}
```

img {float: left|right;} - Sama og **align** HTML-breytan

p { clear: left;} - Lætur allt efni í <p> fara niður fyrir mynd sem er vinstra megin.

Möppuvísanir

Ráðlegt er að geyma myndir í sér möppum, sérstaklega í stærri vefjum. Til að vísa í mynd sem geymd er í möppu er möppunafninu bætt framan við, í dæminu að neðan heitir mappan **myndir**.

```

```

Ef ætlunin er að skipuleggja vef í mörgum möppum er nauðsynlegt að kunna að vísa á milli þeirra. Ef þú ert með tvær möppur sem liggja hlið við hlið í vefmöppunni þinni og ert með skjal í annari sem á að innihalda mynd sem er í hinni möppunni (sem heitir ljósmyndir) þá lítur kóðinn svona út:

```

```

Ef þú þarft að fara upp um tvær möppur þá:

```

```

Listar

Listarnir eru mjög þægilegir til að byggja upp leiðakerfi, enda eru valmyndir ekkert annað en listar. Tvær tegundir þeirra eru mest notaðar; raðaðir listar `` og óraðaðir listar ``. Einnig eru til skilgreiningarlistar `<dd>` sem eru minna notaðir. Stærstu gallar listanna voru lengi vel þeir að fyrir ofan og neðan alla lista kom tvöfalt línubil, ný málsgrein. Þetta hentar alls ekki alltaf og leiddi til þess að sumir fóru að nota reddingu sem skilaði ólöglegum kóða. En með tilkomu CSS er hægt að ráða við þetta vandamál með því að stilla **margin-top** og **margin-bottom**. Einnig er hægt að nota mynd sem punkt (bullet) og sést neðarlega í CSS kódadæmunum hvernig það er gert.

En listarnir eru notaðir á þennan hátt:

`` stendur fyrir óraðaður listi (unordered list) og niðurstaðan úr þessu er punktalisti (bullet list), hægt er að stilla hann með CSS.

```
<ul>
<li>Kaffi</li>           • Kaffi
<li>Te</li>              • Te
</ul>
```

`` býr til raðaðan lista (ordered list) og hægt er að skilgreina ýmiskonar talningaráðferðir og fleira með CSS. `` segir til um liði listans (list item).

```
<ol>
<li>Kaffi</li>           1. Kaffi
<li>Te</li>              2. Te
</ol>
```

CSS

Möguleikar í samstillingu CSS og lista eru mjög miklir, margfalt meiri en hér sést. Á where.is/webdesign er að finna tengla í vefi sem vísa þér veginn til að smíða falleg leiðakerfi á þennan hátt.

Settu þetta á allar listaskilgreiningar, til að fá Firefox og IE til að haga sér eins

```
{
padding: 0;
margin: 0;
}
ol {
list-style-type: none | decimal | lower-roman | upper-roman | lower-alpha | upper-alpha;
margin-top: 1px;
margin-bottom: 1px;
}

ul {
list-style-type: disc | circle | square | none;
list-style-image: url("mynd.gif");
}
```


Bakgrunnsmyndir

Þær eru settar inn í `<body>` eða `<td>` í hefðbundnu HTML og lítur kóðinn þá svona út: `<body background="mynd.gif">` eða `<td background="mynd.jpg">`. Þessi leið er mjög takmörkuð og kallar oft á stórar myndir til að ná settu markmiði. **Einnig er búið að fjarlægja möguleikann á að setja bakgrunnsmynd með þessum hætti á `<td>` úr XHTML-staðlinum, það gæti þýtt að vafrastuðningur hlýði staðlinum og einfaldlega birti ekki þessar myndir í náinni framtíð.**

Mun þægilegra er að nota CSS í bakgrunnsmyndir, gefur meiri stjórn auk annara kosta CSS. Þá er hægt að setja bakgrunnsmyndir á flestar skipanir.

CSS

Þú getur skilgreint bakgrunnsmyndir bæði beint á skipanir eða sem klasa og vísað í þá úr skipununum með `class="bakgrunnur"`. Oftast á betur við að nota klasana.

background-image: url(bakgrunnur.gif);
Vísar á myndina sem nota á sem bakgrunnsmynd.

background-repeat : repeat-y | repeat-x | no-repeat | repeat;
Endurtekning myndar, **repeat** er eins og í hefðbundnu HTML, fyllir upp í skjáinn frá vinstri til hægri og að ofan og niður. **repeat-y** og **repeat-x** staflar myndinni niður (y) og til hægri (x) en **no-repeat** er engin endurtekning.

background-attachment : scroll eða fixed;
Ef **scroll** þá skrunar myndin með síðunni en annars situr hún föst á þeim stað sem hún birtist upphaflega á.

background-position: center center;
background-position: 100px 100px;
Stillir hvar bakgrunnsmynd birtist, fyrri möguleikinn miðjar myndina í glugganum en sá seinni setur hana 100 skjápunkta frá vinstri og efri kanti.

Skilgreiningarlistar (definition lists, lítið notaðir)

```
<d1>
<dt>Kaffi</dt>
<dd>Heitur svartur drykkur</dd>
<dt>Mjólk</dt>
<dd>Kaldur hvítur drykkur</dd>
</d1>
```

Kaffi
Heitur svartur drykkur
Mjólk
Kaldur hvítur drykkur

Listar inn í listum

Ef þú ætlar að setja lista inn í lista þá þarftu að gera það eins og hér sést. Innri listinn verður ílát inn í , ljósgrár kassi er utan um hann.

```
<ul>
  <li>Kaffi</li>
  <li>Te
    <ul>
      <li>Svart te</li>
      <li>Grænt te</li>
    </ul>
  </li>
  <li>Mjólk</li>
</ul>
```

- Kaffi
 - Te
 - Svart te
 - Grænt te
 - Mjólk
-

CSS - Stílsnið

CSS eða Cascading Style Sheets eru í raun ekki hluti af XHTML staðlinum, heldur annar staðall sem situr við hlið XHTML og gerir suma þá hluti sem XHTML er ekki ætlað að gera. Þessi staðreynd endurspeglast í því að á sumum síðum þessarar bókar er að finna CSS kóða sem sýna nokkrar gagnlegar CSS skipanir sem virka vel með þeim XHTML skipunum sem rætt er um þar.

Grunnskiptingin er sú að XHTML sér um að skipta skjalinu upp í einingar en CSS sér um að stýra útliti þeirra eininga. Innihaldið kemur síðan sem efni þessara eininga. Mjög mismunandi er hversu langt menn ganga í að nota CSS til að stýra útliti, hægt er að smíða síður án taflna sem nota CSS í stað þeirra. Í það minnsta ættirðu að nota stílsniðin til að stýra útliti leturs, það sparar þér mikla vinnu.

CSS einfaldar okkur mjög viðhald og smíði vefja og bætir við fjölmörgum möguleikum frá eldra HTML. Helsti gallinn er þó sá að vafrastuðningur er misgóður en hægt er að komast framhjá því vandamáli með því að nota bara það sem þokkalega er stutt og prófa vel í sem flestum vöfrum. Stóri flöskuhálsinn í dag er Internet Explorer, nýjasta útgáfa hans (6) er orðin gömul og úrelt en því miður virðist vera langt í nýja útgáfu af IE.

Sumar þeirra skipana sem vel eru studdar er fjallað um hér á eftir, en alls ekki allar. Að sjálfsgöðu eru til vefir sem sérhæfa sig í að miðla þessum upplýsingum og eru tenglar í slíka á <http://where.is/webdesign>. Það er rétt að taka fram að CSS stuðningur eykst mjög hratt og eldri vafrar hverfa smám saman og það stýttist í að við getum svellköld notað stóran hluta staðalsins.

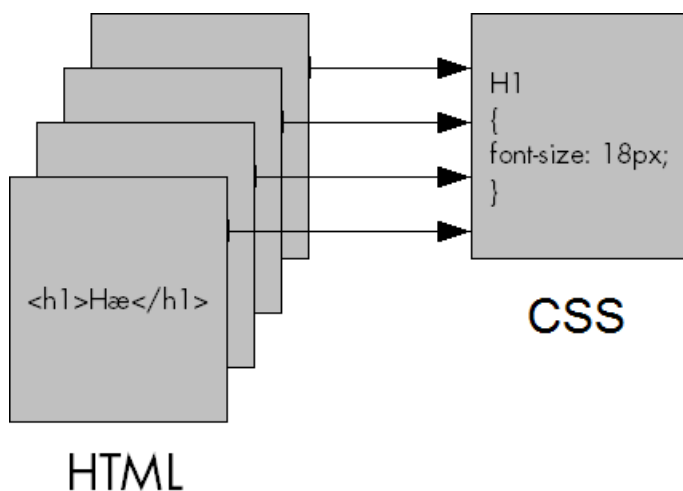
Athugaðu að það eru til fleiri leiðir en hér er lýst til að virkja stílsniðin. Það sem á eftir kemur er það sem mér hefur reynt virka best í vöfrum og vera þægilegast og eðlilegast í notkun. Ef gera á flóknari hluti þarf oftast að grípa til flóknari leiða. Og rétt er að benda á að hér er aðeins skoðaður toppurinn af toppnum á ísjakanum, CSS hefur mun fleiri möguleika en koma fram í þessari bók.

Kostir:

- Hægt að skilgreina allar skipanir í einni skrá sem stýrir útlit fjölda vefsíðna
- Hægt að skilgreina fleiri atriði og nákvæmar en í venjulegu HTML
- Mögulegt að skilgreina mismunandi útlit fyrir útprentun og skjá
- Margt, margt fleira

Gallar:

- Stuðningur nýrri vafra er misgóður
- Virkar ekki í mjög gömlum vöfrum
- Erfitt að láta efni líta **alveg** eins út í mismunandi vefvöfrum
- Óþægilega flókið að stýra útliti vegna mismunandi vafrastuðnings, sérstaklega Internet Explorer



Grunnatriði

Sú leið sem mér finnst langbest til að nota CSS er að vera með sérskrá sem inniheldur allar stílsniðsskipanirnar. Stærsti kosturinn við þetta er að þú getur t.d. með því að breyta einum staf, breytt stærð alls meginmálstexta á vefnum þínum, óháð fjölda síðna hans.

Til að kalla á skrána sem inniheldur stílsniðið notar þú eftirfarandi skipun:

```
<head>  
<link rel="stylesheet" type="text/css" href="utlit.css" />  
</head>
```

Þetta er sú aðferð sem langflestir vefsmíðir og vefsmíðatól nota, en rétt er að benda á að sumar breytur CSS eru vitlaust skilgreindar í eldri vöfrum sem getur skemmt leshæfni. Ef þú vilt að eldri vafrar fái eingöngu hráa síðu án útlitsskilgreininga þá er betra að nota aðra aðferð. Eldri vafrar þekkja ekki `import` skipunina og birta því vefinn hráan, án stílsniða ef gert er eins og að neðan.

```
<style><!-- @import url(utlit.css); --></style>
```

Fyrir nýrri vafra kemur þetta út á eitt, báðar virka fínt.

Það eru þrjár grunnleiðir til að nota CSS skipanir og þær eru:

1. Endurskilgreina HTML skipanir
2. Nota klasa (class) eða einkenni (id) sem tengd eru við skipanir
3. Skilgreina útlit tengla (pseudo classes)

Athugaðu að fleiri leiðir eru notaðar til að skilgreina CSS en þetta eru þær helstu og ættu að fleyta þér ansi langt.

Hvað þarf til að CSS virki?

1. Einhverja skipun í HTML-skjali, t.d. `<h1>`
2. Tengingu í CSS-skjalið úr HTML-skjalinu, t.d. `<link href="stilar.css" />`
3. Skilgreiningu á `<h1>` í CSS-skjalinu

Uppbygging CSS-skipana

CSS-skráin er samansafn af skilgreiningum, ein fyrir hverja skipun eða undirskipun og líta þær eins út, óháð því hvort verið er smíða klasa eða endurskilgreina XHTML skipun. Tenglabreyturnar eru síðan aðeins öðruvísi eins og sést aftar í kaflanum.

En áður en við skoðum muninn á þessum þremur leiðum þá er rétt að skoða grunnkerfi CSS-breyta. Mér finnst þægilegast að setja þær upp eins og sést hér á næstu síðu, sérstaklega þegar margar breytur eru komnar á skipun eða klasa. Annarsstaðar í bókinni þjappa ég þessu aðeins betur, plássins vegna.

```
h1
{
font-size: 26px;
}
```

Þetta er fyrsta CSS-skipunin sem við skoðum og þú sérð að þetta er ekki flókið. Við erum hér að endurskilgreina <h1> skipunina þannig að alltaf þegar henni er beitt í XHTML kóðanum þá verði allur texti innan hennar 26 skjápunktar á stærð. Fyrst kemur skipunin (h1), síðan slaufusvigi, þarnaest breytan (font-size) með tvípunkti á eftir og loks gildi breytunnar (26px) með semikomu á eftir. Síðan lokast slaufusviginn og skilgreiningin þar með. Aðrar skilgreiningar koma síðan þar á eftir.

Athugaðu að semíkomman ; er nauðsynleg á eftir öllum breytum, nema þeirri síðustu í hverri skilgreiningu. En þar sem hún má vera þar, þá finnst mér langþægilegast að setja hana allsstaðar, það auðveldar afritun á breytum milli skilgreininga og CSS-skráa.

CSS kóði annara skoðaður

Alveg eins og með HTML-kóða þá er hægt að skoða CSS-kóða annara. Það er aðeins meira mál en að nota View Source en ætti þó ekki að vefjast fyrir þér.

Í sumum vefskoðurum er hægt að slá inn slóðina á CSS skjalið í Address/Location gluggann og ýta á Enter, þá birtist gluggi sem spyr okkur hvar við viljum vista skjalið sem við síðan opnum í þeim ritli sem við viljum eða jafnvel birtir stílsniðin á skjánum eins og Opera og Mozilla. Til að skoða .css skrár á vefjum annara í IE er ekki alltaf hægt að nota þessa aðferð heldur getur þurft að búa til vefsíðu með tengli í skrána (finnið nafnið með View-Source), smella á hann með hægri músartakka og velja Save Link As eða Save Target As og vista skrána. Opna síðan í ritli.

Þægilegasta leiðin er sú að setja upp WebDeveloper viðbótina við Firefox sem gerir okkur kleift að smella á einn takka og fá upp CSS-kóðann í hliðarglugga þar sem hægt er að fikta í honum að vild og sjá strax áhrif breytinga á vefsíðuna. Önnur mjög gagnleg viðbót í stílsniðavinnu er Firebug, líka fyrir Firefox.

XHTML skipanir endurskilgreindar

```
BODY
{
font-family: sans-serif; /* Meira á bls. 31 og 36 */
}

H1
{
font-size: 18pt;
color: #ff0000; /* textalitur í þessu tilfelli, í raun litur innihalds */
line-height: 120%; /* línubil, best í % */
text-indent: 10px; /* inndrag texta */
text-align: center; /* hliðrun texta, left, right, center eða justify */
}

H2
{
font-weight: normal; /* þykkt leturs */
color: #ff0000;
}
```

Þarna skilgreinum við þrjár XHTML skipanir: `<body>`, `<h1>` og `<h2>`. Allur meginmálstexti mun verða í sans-serif lettri. Þessar stillingar eiga að erfast á allan texta sem er í `<body>`, nema að skilgreiningar annara skipana í .css skránni segi að leturgerð eða stærð skuli breytast. (Í raun er samt vandamál með sumar útgáfur af IE sem veldur því að skilgreiningar á `body` skila sér ekki inn í töflur og þarf því að skilgreina `td` á sama hátt, það er auðveldast að gera með því að bæta `TD` í efstu línuna: `BODY, TD`).

Í dæminu `<h1>Opið öllum</h1>` mun textinn "Opið öllum" birtast í 18pt. serif lettri með 10 skjápunkta inndragi, línubil verður 26pt. og textinn verður rauður og miðjaður.

`<h2>` mun afturámóti nota sjálfgefna stærð og leturgerð en með því að segja að leturþyngd (`font-weight`) sé `normal` þá verður það ekki feitletrað.

Hægt er að nota margar tegundir stærðareininga og breytur eru mun fleiri en hér eru sýndar. Best er að nota stærðir sem stækka og minnka í öllum vöfrum eftir stillingu vefgesta á allan meginmálstexta, % hafa reynst mér best, sjá ábendingar hvað þetta varðar á bls. 35.

Nokkrar breytur útskýrðar

font-family: "Trebuchet MS", sans-serif;

Skilgreinir leturgerð, nóg er að hafa aftast annaðhvort serif (Times), sans-serif (Arial) eða mono (Courier) en einnig er hægt að skilgreina aðrar leturgerðir eins og sést að ofan. Vafrinn athugar þá hvort hann eigi leturgerðirnar sem eru fremst og endar síðan á grunnleturgerðinni sem þú átt alltaf að setja síðast (serif, sans-serif eða mono). Ef ekkert er skilgreint þá nota flestir vafrar serif.

Þegar fleiri en ein leturgerð er skilgreind þá athugar vafrinn fyrst hvort fyrsta leturgerðin er til á vélinni sem skoðar, ef ekki þá næstu leturgerð og síðan koll af kolli þar til hann endar á þeirri síðustu. Því er mjög mikilvægt að enda þessar skilgreiningar á serif, sans-serif eða mono til að fá eitthvað sem er nálægt því sem óskað er eftir.

Ef ekkert er skilgreint þá eru flestir vafrar með serif letur skilgreint sem sjálfgefið.

font-size stýrir leturstærð, sjá Stærðareiningar á bls. 35.

line-height stýrir línubili málsgreinar, þú getur notað sömu einingar og í leturstærð en % er oftast öruggast og best.

text-align stillir hliðrun texta, hægt að nota left, right, center og justify.

color stýrir textalit, lit innihalds íláts. Gildi geta verið litakóði (bls. 42) eða nafn.

font-weight stýrir þykkt leturs, eina sem raunhæft er að nota í dag er bold og normal

background-color stýrir bakgrunnslit íláts

text-indent dregur inn fyrstu línu málsgreinar.

Faldar athugasemdir

Í .css skjölum notum við aðra aðferð en í XHTML þegar við ætlum að setja inn athugasemdir eða fela skipanir, gera þær óvirkar. Þar setjum við /* í byrjun og */ í enda.

Dæmi: /* Ábendingar sem ekki sjást í vafra */

Litakóðar

Það er hægt að nota mörg algeng litanöfn úr ensku en það er öruggara og nákvæmara að nota litakóða sem eru útskýrðir á bls. 43.

Samhæfni við eldri vafra

Hafðu í huga að ef þér er mjög umhugað um að útlit þitt sé eins í eldri vöfrum sem mjög fáir eru enn að nota, þá er betra að nota HTML skipanir heldur en CSS jafngildi þeirra. Ég mæli samt með að þú einbeitir þér að því að hægt sé að lesa síðurnar þínar í eldri vöfrum en hafir minni áhyggjur af útlitinu þar. Þannig geturðu farið eftir XHTML staðlinum og minnkað líkurnar á því að þú þurfir að laga síðurnar þínar síðar. Skoðaðu líka ábendingu um notkun **import** í stað **link** á bls. 28.

Klasar (e. classes)

Eitt sem eykur töluvert á þægindi og sveigjanleika er möguleikinn á því að nota klasa og búa þar með til margar mismunandi útgáfur af hverri skipun.

Það er gert á eftirfarandi hátt:

```
.storraudur
{font-size: 220%;
 color: #ff0000;}

.litillblar
{font-size: 80%;
 color: #0000ff;}
```

Með þessari aðferð í notkun klasa er hægt að nota þá í öllum ílátum sem á annað borð er hægt að skilgreina með CSS, t.d. `p`, `div`, `span`, `a` ofl.

Og svona er kallað í klasa:

```
<p class="storraudur">Stórt rautt letur</p> og <span
class="litillblar">lítið blátt</span>
```

Eða svona:

```
<span class="storraudur">Stórt rautt letur</span> og <h5
class="litillblar">lítið blátt</h5>
```

Einnig er hægt að festa klasa við ákveðnar skipanir, það lítur svona út:

```
h2.litill-blar
{font-size: 9pt;
 color: #0000ff;}
```

Þeir virka þá ekki með öðrum skipunum og því ekki oft ástæða til að nota þetta, einna helst þegar þörf er á fleiri en einni tegundum tengla.

Það er hægt að virkja tvo eða fleiri klasa á hverja skipun og er það gert svona:

```
<span class="klasi1 klasi2">Báðir klasar virkjaðir</span>
```

Tenglar

Að síðustu eru það svokölluð *pseudo-elements*, þau snúa að útliti tengla, oftast byrjum við þó á því að skilgreina **a** skipunina því hún er yfirflokkur þeirra sem að neðan koma. Athugaðu að röðin á `link`, `visited`, `hover` og `active` skiptir máli, hún er rétt hér að neðan.

```
A {text-decoration: none; }
```

```
A:link      { color: #990000; }  
A:visited  { color: #009900; }  
A:hover    { color: #999900; }  
A:active   { color: #009900; }
```

`link` er útlit á tenglum sem vísa á síður sem ekki er búið að skoða, `visited` á þeim sem vísa á síður sem búið er að skoða og `active` er útlit á tenglum akkúrat á meðan smelt er á hann. `hover` breytan er sá stíll sem kemur þegar farið er með músina yfir tengilinn.

Þetta þýðir að allir tenglar verða án undirstrikunar og taka síðan þá liti sem skilgreindir eru.

Prófaðu að gera tilraunir með bakgrunnsleti, bakgrunnsmyndir og fleira á tenglana þína, í mörgum tilfellum getur þú komist hjá því að nota Javascript til að breyta útliti tengla þegar farið er með músina yfir.

Stýra svæði inn í svæði - Contextual Selectors

Þú getur látið ákveðna skipun einungis virka ef hún er inn í annari skipun. Sama með klasa. Þú getur t.d. sett klasa á `` og látið alla tengla sem eru innan `` skipana haga sér öðruvísi en aðrir tenglar. Tvö einföld dæmi:

```
p b { font-size: 60px; }
```

`` sem er inn í `<p>` verður 60px á stærð en önnur `` verða eðlileg.

```
.stort b { font-size: 60px; }
```

Sem virkjar stórt letur á þau `` sem eru innan klasans `stort`.

Ekki breyta leturstærð tengla eftir stöðu þeirra

Passaðu þig á að breyta ekki leturstærð á neinn hátt með undirklösum **A**. Það lítur hræðilega út ef farið er með músina yfir og letrið stækkar, stundum hreyfist öll síðan með! Feitletrun stækkar letur, ekki gleyma því.

Sýndu staðsetningu gests í leiðakerfinu

Það er engin sjálfvirk leið til að gera þetta því CSS-kóðinn veit ekki á hvaða síðu þú ert staddur. En þú getur búið til `class` eða `id` og virkjað það á þeim tengli sem vísar á síðuna sem gesturinn er á.

Mismunandi útlit á tenglum

Ef þú vilt vera með fleiri en eina tegund tengla á vefnum þínum, þá geturðu búið til sérklasa fyrir þá, eins og hér sést:

A.nytt

```
{background-color: #333333;  
text-decoration: none;}
```

```
A.nytt:link {color: #ff0000;}  
A.nytt:visited {color: #00ff00;}  
A.nytt:hover {color: #ffff00;}  
A.nytt:active {color: #00ff00;}
```

Þú virkjar þetta síðan á þennan hátt:

```
<a href="blabla" class="nytt">texti</a>
```

Þú getur líka virkjað þetta með því að stýra svæði inn í svæði með Contextual Selectors, sjá bls. 33, það er oft betri aðferð.

Passaðu þig samt á því að rugla ekki vefgestina þína með mörgum mismunandi tegundum tengla. Það er sjálfsagt að hafa leiðakerfið öðruvísi en tengla í meginmáli en fleiri tegundir tengla þurfa að hafa góða ástæðu fyrir tilvist sinni.

Þú getur reyndar líka sett venjulegan klasa á **a** skipunina en missir þá möguleikann á því að fá mismunandi tenglahegðun eftir hvort búið er að skoða síðuna sem tengillinn bendir á. HTML kóðinn lítur þá eins út og að ofan en CSS kóðinn er bara venjulegur klasi.

Breyta bakgrunni eftir stöðu tengla

Þetta er frekar lítið mál en til að IE sé ekki með leiðindi þarf að stilla allar fjórar breytur eins og sést hér að neðan.

```
a {padding-left: 20px;} /*Texta ýtt frá bakgrunnsmynd */
```

```
a:link {background-image: url(grar.gif);  
background-repeat: no-repeat;}
```

```
a:visited {background-image: url(blar.gif);  
background-repeat: no-repeat;}
```

```
a:hover {background-image: url(raudur.gif);  
background-repeat: no-repeat;}
```

```
a:active {background-image: url(gulur.gif);  
background-repeat: no-repeat;}
```

Tenglastaða hreinsuð

Til að sjá vefinn þinn eins og sá sem kemur þangað í fyrsta sinn þá þarftu að segja vafranum þínum að hreinsa allar minningar um hvaða síður hafa verið opnaðar í honum. Þetta er einfalt:

Internet Explorer: **Tools -> Internet Options -> Clear History**

Firefox/Mozilla: **Tools -> Options -> Privacy -> History ->Clear**

CSS og texti

Eftirfarandi CSS breytur virka á flestar skipanir sem hafa áhrif á umbrot texta og er gagnlegt að nota. Þetta er þó engan veginn tæmandi listi, heldur einungis það sem mér hefur þótt vera gagnlegt. Ef þú vilt skoða tæmandi lista þá er best að fara á where.is/webdesign og fylgja CSS tenglunum sem þar eru.

```
BODY{
    font-family : Verdana, Geneva, sans-serif;
    color : black;
    line-height : 160%;
}

H2 {
    margin-top: 11px;
    margin-bottom :0px;
    margin-left : 10px;
    color : #003322;
    font-family : sans-serif;
    font-size : 100%;
    font-weight : bold;
    text-align : left | right | center;
}

.inndrag
{
    text-indent : 10px;
}
```

Ef þú vilt einungis draga fyrstu línu inn, notaðu þá text-indent og slepptu margin-left sem er fyrir ofan.

Stærðareiningar

Þær stærðareiningar sem ég nota í eilífrí leit að stillingum sem virka vel allsstaðar eru: **px**, **em** og **%**, fleiri eru til og auðvelt að finna upplýsingar um slíkar á vefnum. Ég er farinn að hallast að því að nota **%** í öllum meginmálstexta, þ.e. í öllum texta sem einhver gæti hugsanlega viljað stækka en **px** á þann texta sem ég vil ekki að hægt sé að stækka, fyrirsagnir og leiðakerfi. Í seinna tilvikinu passa ég mig líka mjög vel á því að allir geti lesið þennan texta.

Athugaðu samt að **%** og aðrar hlutfallseiningar í CSS eru túlkaðar mismunandi af vöfrum og ef þú passar þig ekki á erfðunum í CSS (stillir fyrst t.d. **body** og síðan **p**) með hlutfallsstillingum þá er alls óvíst hver innsta stærðin verður. Ég fer einfaldlega þá leið að passa upp á að leturstærð sé bara stillt á einum stað og losna við þetta á þann hátt.

Hvað aðrar stærðir en leturstillingar varðar er oft gott að nota **px** en einnig **%** og **em**.

Leturgerðir á vefsíðum

Þú getur í raun vísað í hvaða leturgerð sem er með CSS en rétt er að enda allar vísanir á serif, sans-serif eða monospace.

```
h1 { font-family: Trebuchet MS, sans-serif; }  
body { font-family: Georgia, serif; }
```

Þú getur líka sleppt því að gefa upp leturgerðarnafn og aðeins gefið upp einhvern af þessum þremur kostum, þá er notað sjálfgefið slíkt letur í tölvu vefgests. Vafrinn reynir fyrst að nota þær leturgerðir sem gefnar eru upp fyrst.

Útprintun með CSS

Eins og talað var um framfarir í bókinni þá er 750 skjápunkta upplausn of breið til að komast fyrir á A4 blaði, sem leiðir oftast til þess að skerst hægra megin af texta. Auðvitað er hægt að stilla prentarann á að prenta í langsníði (landscape) en fæstir notendur átta sig á því.

Með CSS er hægt að gera ýmsar kúnstir í prentun og ein þeirra er að fela ákveðna hluta vefsíðu þegar prentað er út. Þetta kallar á sitthvora skrá fyrir útprintun og skjá, vafrar eiga að lesa þær báðar og aðeins þar sem einhverju munar verður breyting. Stílsníðs-skrá útprintunar er t.d. alltaf mun styttri hjá mér, hún inniheldur bara þær stillingar sem ég vil hafa öðruvísi þar.

Fyrst þarftu að kalla á sitthvora .css skrána fyrir skjá og prent. Taktu eftir hver er munurinn á línunum:

```
<link rel="stylesheet" href="utlit.css" type="text/css"  
media="screen">  
<link rel="stylesheet" href="prent.css" type="text/css"  
media="print">
```

Síðan seturðu þetta inn í CSS-skjalið fyrir útprintun, utlit-prent.css:

```
.leftmenu {display: none}  
.topmenu {display: none}
```

og í XHTML-kóðann fer t.d. eftirfarandi:

```
<td rowspan="2" class="leftmenu">
```

Það sem er í þessari sellu mun ekki birtast á prenti.

Og auðvitað er hægt að fela annað en sellur, veldu bara úr XHTML-skipunasettinu. Og þetta er líka hægt að nota á hinn veginn, birta hluti í útprintun en ekki á skjá. Þetta er líka hægt að nota til að vera með sitthvort lettrið á sitthvorum miðlinum.

Rétt er að taka fram að einhverjar eldri útgáfur Internet Explorer lesa ekki inn fyrri skrána og verður því útprintun þar frekar hrá, innihaldið ætti þó að skila sér. Og þetta virkar ekki í Netscape 4.

Aðrar aðferðir við að virkja CSS

Þú getur sett CSS beint inn í HTML kóðann en ég nota það sárásjaldan, eingöngu þegar ein síða er allt öðruvísi en hinar. Þetta er gert svona:

```

```

Og það er líka hægt að setja CSS kóða fremst í hvert skjal en sama gildir um það og ofanskráð, ég nota þetta eingöngu við mjög sértækar síður. Þetta lítur svona út:

```
<style type="text/css">
<!--
body {
  background-color: white;
  margin: 0px;
  font-family: sans-serif; }
-->
</style>
```

<-- og --> er notað til að fela kóðann fyrir vöfrum sem ekki skilja CSS. Þarna er auðvitað hægt að setja hvaða CSS kóða sem vill.

Fleiri en einn klasi á skipun

Þú getur vísað í fleiri en einn klasa með `<p class="stort rautt">`

Styttri kóði

Það er hægt að skrifa CSS-kóða í knappari stíl (shorthand) sem er að vísu heldur erfiðari lestrar. `border: 1px solid #000000;` gefur t.d. þunnan svartan ramma á allar hliðar.

Ósamhæfni vafra í padding og margin

Til að fá Firefox og IE til að haga sér eins þá getur verið gott að setja þessar skilgreiningar á skipanir.

```
{
padding: 0;
margin: 0;
}
```

Röð skipana skiptir máli

Athugað að röð skilgreininga í CSS-skjali skiptir máli. Sumar skipanir eru n.k. yfirskipanir, þ.e. þær stýra mörgum undirskipunum sem síðan er hægt að fínstilla ef vill. T.d. er `border-style` yfirskipun yfir `border-left-style`, `border-right-style`, `border-top-style` og `border-bottom-style` og ef `border-style` kemur neðar í skilgreiningunni en hinar þá yfirskrifar hún það sem í þeim er.

Athugaðu að þetta á við á fleiri stöðum, t.d. hvað tenglabreyturnar varðar. Þar verður röðin á breytunum að vera þessi ef allt á að virka rétt:

```
link, visited, hover, active
```

Dæmi um einfalt CSS-skjal

Auðvitað þarf síðan að kóða XHTML-skjalið til samræmis.

```
body { color: #000;
        padding: 0px;
        margin: 0px;
        font-family: verdana, sans-serif;
        background-color: #fff; }

table {margin: 10px; }

td { vertical-align: top; }

p {   line-height: 140%;
      font-size: 96%; }

h1 {  font-size: 180%;
      padding-left: 0px;
      padding-top: 22px; }

h2 {  font-size: 140%;
      padding: 10px; }

h3 {  font-size: 120%;
      padding-left: 30px;
      margin-top: 2px;
      margin-bottom: 1px; }

.skuggi {
  color: #fff;
  width: 200px;
  background-color: #999;
  text-align: center;

  border-width: 3px;
  border-style: solid;

  border-bottom-color: #555;
  border-right-color: #555;
  border-top-color: #aaa;
  border-left-color: #aaa;
}

/* möguleikar í border-style eru:
solid, dashed, dotted, groove, inset,
outset, ridge, double. Sumir þeirra
virka ekki nema þykkt border sé 3px
eða meira.*/

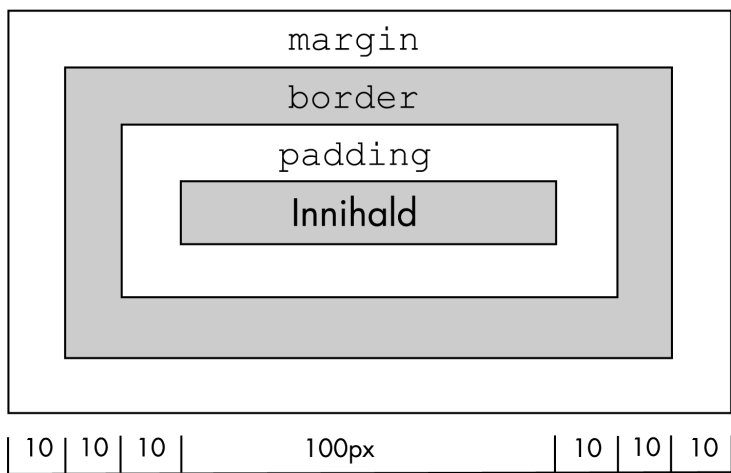
.mailto { color: #000;
          background-color : #fff; }

A { text-decoration: none; }

A:link { color: #000; }
A:visited { color: #030; }
A:hover { color: #000;
          text-decoration: underline; }
A:active { color: #090; }
```

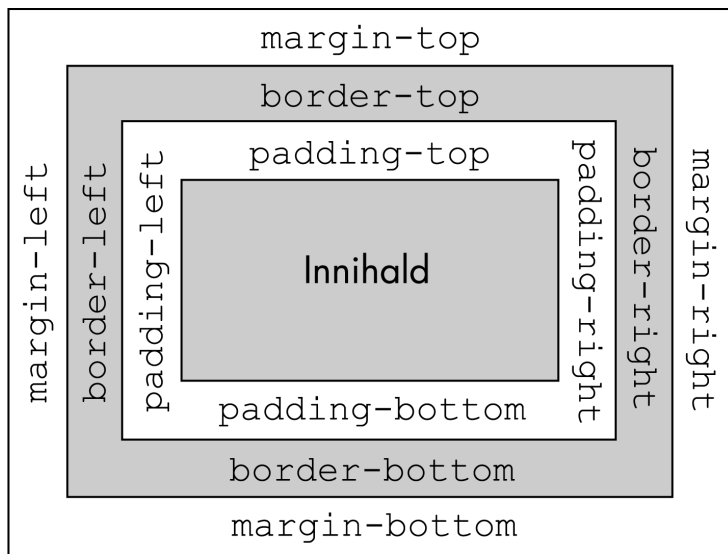
Box Model CSS

Svokallað box model í CSS er leið til að útskýra hvernig ýmsar rýmisskipanir hegða sér. Þessar skipanir eru **padding**, **margin** og **border** og undirskipanir þeirra. Þetta er best útskýrt myndrænt eins og sést hér að neðan.



Samtals 160 skjápunktar á breidd

Undirskipanir eru síðan notaðar til að stilla rými sér fyrir hverja hlið, eins og sést að neðan. Athugaðu að ef þú setur grunnskipanir fyrir neðan undirskipanir þá eru þær sterkari og eyða hinum í raun út.



Staðsetning með CSS

Í dag er hægt að nota CSS til að stýra algjörlega útliti vefsvæða en það getur verið þrautin þyngri að fá það til að hegða sér eins í öllum vöfrum sem markhópurinn notar. Þetta er svið sem ég hef ekki kannað af neinu viti og því er hér lítið að finna á þessu stigi. Eftirfarandi dæmi er fengið úr bókinni **HTML Utopia - Designing without tables using CSS**, fin bók sem ég mæli með fyrir þá sem vilja kynna sér þetta mál. Þessi síða lítur nokkurn veginn eins út í öllum vöfrum á Windows en eilítið öðruvísi á Internet Explorer fyrir Macintosh. Athugaðu að þetta er aðeins ein leið af mörgum til að stýra útliti með CSS og flestar þeirra nota einhverskonar trix til að komast fyrir mismunandi stuðning vafra.

HTML kóðinn lítur svona út:

```
<div id="top"><h1>Hér er efni hausins</h1></div>
<div id="left"><p>Efni vinstri dálks</p></div>
<div id="center"><p>Efni miðjunnar</p></div>
<div id="right"><p>Efni hægri dálks</p></div>
```

Og CSS kóðinn er svona:

```
body {
  margin: 0;
  padding: 0; }

#top {
  margin: 20px;
  padding: 10px;
  background: #ccc;
  height: 100px; }

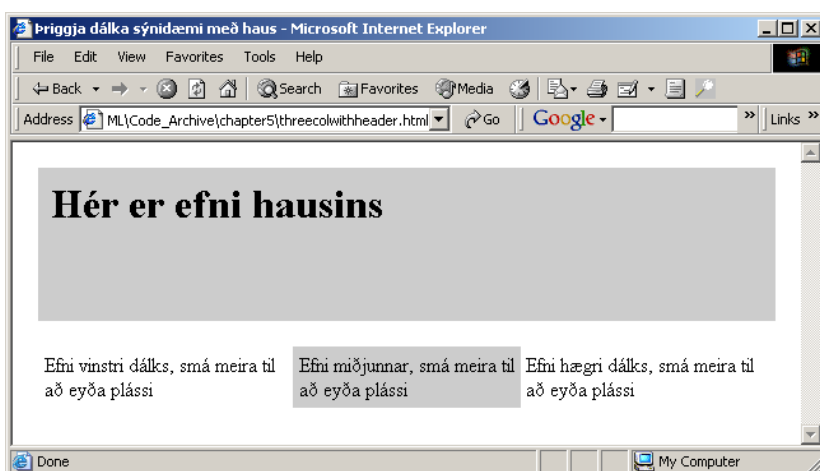
#left {
  position: absolute;
  left: 20px;
  top: 160px;
  width: 200px; }

#center {
  background: #ccc;
  margin-top: 0;
  margin-left: 220px;
  margin-right: 220px; }

#right {
  position: absolute;
  right: 20px;
  top: 160px;
  width: 200px; }

p {padding: 5px; }

/* Fjarlægir sjálfgefið top margin fyrir Mozilla */
div > p {margin-top: 0; }
```



id eða class

Þegar unnið er með umbrot í CSS eru oftast notuð **id** í stað **class** eins og sést í þessu dæmi. **id** er með # (myllu) fyrir framan en **class** með punkti. Eftir því sem ég kemst næst þá er eini munurinn á **id** og **class** sá að **id** má bara nota einu sinni í hverju skjali en **class** eins oft og vill, enda virkar dæmið alveg eins þó að **id** sé breytt í **class**.

Eina ástæðan sem ég sé fyrir þessari venju er að við sannreyningu (validation) sést ef sama **id** er notað oft en einu sinni (þú færð villu) og það er oftast ekki ætlunin í þessari vinnu.

Meiri upplýsingar

Það eru til margir ágætis vefir um þetta mál og eru tenglar í nokkra þeirra á where.is/webdesign Bestur er líklega Position is Everything og eins eru góð dæmi á Little Boxes. Ég nota sjálfur mest Layout Generator frá CSS Creator (<http://www.csscreator.com/version2/pagelayout.php>) til að byrja slíka vefi og laga síðan til eftir þörfum. Þú þarft að vera innskráður notandi til að geta notað græjuna, það er ókeypis að skrá sig hjá þeim.

Sérstafir

** **; Þessi skipun býr til 'órjúfanlegt bil' (no break space) og þarf þessvegna nokkuð margar svona skipanir í röð til að ýta texta inn svo einhverju nemi, þetra er að nota CSS spássíur. Er afturámóti handhægt ef tvö orð mega ekki skiptast á milli lína og er þá sett inn í stað orðabils.

** **; tilheyrir sérstakri deild XHTML skipana sem er ætluð til að geta skrifað sérstök ták og stafi. Nokkur önnur ták eru

©; sem gefur þér © merkið,

&; = & (og merkið)

<; = <

>; = >

­; = Soft hyphen, bandstrik sem birtist einungis ef orð skiptist á milli lína. Í sumum vöfrum birtist annað en ætlað er.

Fjöldi annara slíkra skipana er til og þú getur fundið upplýsingar um þær á vefnum, m.a. í gegnum where.is/webdesign

Mjög mörg vefsmíðaforrit nota síðan þessa kóða til að túlka íslensku stafina, sem er í raun rétta leiðin til að skrifa þá. En það hefur samt engin vandræði í för með sér að skrifa þá á hefðbundinn hátt. Hér að neðan sést hvaða ták túlka hvaða íslensku stafi:

þæðö = **þ** **æ** **ð** **ö**

ÞÆÐÖ = **Þ** **&AELIG;** **Ð** **Ö**

óíúýéá = **ó** **í** **ú** **ý** **é** **á**;

ÓÍÚÝÉÁ = **Ó** **Í** **Ú** **Ý** **É** **Á**;

Algildar breytur

Nokkrar breytur virka á nær allar XHTML skipanir, þó vissulega séu þær misgagnlegar. Þær sem ég nota eru þessar:

```
title="texti sem birtist þegar mús er færð yfir"
```

Þetta er hægt að nota á `body` og allar skipanir sem mega vera inni `body`

```
class="cssklasi"
```

Sjá nánar um þetta í CSS kaflanum

```
id="nafn/merki"
```

Þetta er hægt að nota til margvíslegra hluta, sjá nánar í CSS kaflanum um not í því samhengi. M.a. notað tenglum sem vísa á merkta staði á síðum. Kemur í stað name í mörgum tilvikum.

Litakóðar

Þegar skilgreindur er litur á texta, bakgrunn eða tenglum á vefsíðum eru til tvær leiðir. Önnur er sú að nota litanöfn, `white`, `red`, `blue` osfrv. en hin er að nota þá kóða sem hér verða útskýrðir. Með því að nota kóðann er hægt að skilgreina mun fleiri liti en með nöfnunum. Þessir kóðar virka bæði í HTML og CSS, ég nota þá eingöngu með CSS í dag.

Nokkrir grunnlitir eru:

```
#ffffff = hvítt
```

```
#000000 = svart
```

```
#ff0000 = rautt
```

```
#00ff00 = grænt
```

```
#0000ff = blátt
```

```
#ffff00 = gult
```

Myllan (#) þýðir að kóði kemur þar á eftir. Þetta er sextándakerfis kóði þar sem 0 er minnsti styrkur á lit og F er mesti styrkur. Fyrstu tveir stafirnir eru styrkurinn á rauða litnum, tveir næstu styrkurinn á græna litnum og tveir þeir síðustu stýra styrknum á bláa litnum. Ef enginn styrkur er á neinum lit verður liturinn svartur. Ef fullur styrkur er á öllum litum verður liturinn hvítur.

Athugaðu að ef þú ert að nota samstæð pör þá er líka hægt að skrifa styttri útgáfu: `#fff = #ffffff`, `#369 = #336699`.

Leyleg gildi eru: 0,1,2,3,4,5,6,7,8,9, a,b,c,d,e,f. Til eru ýmis forrit og vefsíður sem hjálpa okkur að finna réttu litina og er hægt að finna þau á Netinu, einnig er gott að nota Photoshop eða Illustrator auk fleiri forrita.

Töflur - <table>

Töflur, <table>, eru sígilda leiðin til að stýra útliti vefsíðna þar sem með þeim er hægt að skilgreina staðsetningu mynda og texta og láta líta eins út í sem flestum vöfrum og stýrikerfum. Þróunin er stöðug í þá átt að nota CSS til að stýra útliti vefsíðna og er sú aðferð líklega búin að ná yfirhöndinni. Hún er þó enn flóknari í framkvæmd en töfluleiðin sökum misgóðs vafrastuðnings.

Í töflunum reynir fyrst á rökrétta hugsun og skipulag. Þær eru viðkvæmar fyrir "smávægilegum" mistökum og sýna þér stundum hina verstu delli ef þú gerir smá vitleysu. Á meðan þær eru einfaldar er þetta lítið mál en verður mun erfiðara viðureignar þegar maður fer að setja töflur inni töflur eða lætur þær ná yfir fleiri en eina röð eða hólf. Þær er þó lítið mál að smíða í forritum eins og Dreamweaver.

Óráðlegt er að treysta fullkomlega á stærðarskilgreiningar í töflum. Ef t.d. mynd er stærri en skilgreining töflunnar á selluni sem myndin er í, þá teygist sellan. Ef stór mynd er í einni sellu, þá minnka hinar. Óruggasta leiðin til að stýra sellustærð er að setja inn í þær myndir af þeirri breidd og/eða hæð sem þú vilt að sellan sé en það er hálf-klunnaleg leið og stuðningur við aðrar aðferðir (XHTML-breytur og CSS-skipanir) orðinn það góður að þessi leið er sjaldnast farin í dag.

Töflur stýrast af innihaldi. Til að sjá hvernig taflan þín lítur raunverulega út þarftu því að hafa efni í sellunum, efni sem endurspeglar það sem þar mun raunverulega verða. . Einnig er hægt að setja töflur inni töflur sem getur verið nauðsynlegt til að leysa ýmis uppsetningarvandamál.

Skiptu stórum síðum í nokkrar töflur

Þar sem flestir vafrar birta ekki innihald taflna fyrr en allt efni þeirra hefur hlaðist inn, þá er ráðlegt að skipta stórum og efnismiklum síðum niður í a.m.k. tvær töflur. Til dæmis gæti hausinn verið í sértöflu og restin í annari töflu. Þá sér gestur hausinn strax og veit að meira er á leiðinni.

Skipanir og eðli taflna

Grunnskipanir taflna eru ekki margar og auðvelt að læra á helstu breytur.

```
<table>
<tr>
<td>
Hér kemur efni
</td>
</tr>
</table>
```

Þetta er fullgild en mjög einföld XHTML tafla. Athugaðu að þú sérð engin merki um að þetta sé tafla nema þú setjir `border="1"` sem breytu á `<table>` skipunina. Í langflestum tilvikum slökkvum við á rammanum áður en vefurinn er tekinn í notkun með `border="0"`:

```
<table border="1">

<tr>
<td>Sella 1</td>
<td>Sella 2</td>
</tr>

<tr>
<td>Sella 3</td>
<td>Sella 4</td>
</tr>

</table>
```

Þessi tafla inniheldur 4 sellur og lítur svona út:

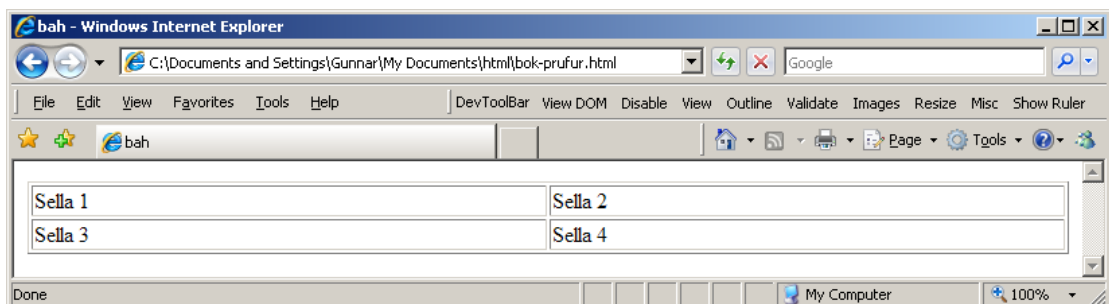
Sella1	Sella2
Sella3	Sella4

Útlitshönnun með töflum

Litlu meira þarf síðan til að útlitshanna einfalda vefsíðu. Ef taflan er stillt til að vera 100% á breidd og hæð þá skiptist vafraglugginn í fjóra 'jafna' hluta og í hverjum fyrir sig er síðan hægt að stilla hvar innihaldið situr með því að setja `align` og `valign` á viðeigandi `<td>`

```
<table border="1" width="100%">
<tr><td>Sella 1</td><td>Sella 2</td></tr>
<tr><td>Sella 3</td><td>Sella 4</td></tr>
</table>
```

Gefur þér töfluna að neðan:



Athugaðu að `height` er ekki lengur lögleg breyta á `<table>` samkvæmt XHTML-staðlinum og þó hægt sé að stilla breidd töflu með CSS, bæði % og px þá á það sama ekki við um hæð. Þar virka bara px ef width breytan er notuð en ekki %.

Taflan sjálf - <table>

<table>Hér inni fara allar töfluskipanir</table>. Það sem er utan við þetta ílát hverfur eða kemur á mjög óæskilegum stað, oftast fyrir ofan töflu.

Allar breytur sem fjallað er um hér eiga sér hliðstæðu í CSS og er auðvitað mun betra að venja sig á að nota þær.

Breytur <table> ílátsins eru m.a:

```
<table width="X" height="x" border="X" cellspacing="x"
cellpadding="X" bgcolor="gray" align="left eða center eða
right">
```

width stýrir breidd þess íláts sem hún er sett inni. Hægt er að skilgreina í skjápunktum **width="xxx"** eða í prósentum af skjástærð, **width="xx%"** Ef ætlunin er að smíða alvöru vef sem allir geta skoðað án óþæginda er rétt gildi **<table width="750">** eða **<table width="100%">**.Ef % eru notaðar stýrist stærð töflunnar af gluggastærð vafra en ef skjápunktar eru notaðir þá er stærðin föst.

Ef ætlunin er að gera fasta útlitshönnun sem lítur eins út í sem flestum vefskoðurum og upplausnum er nauðsynlegt að festa stærðina með skjápunktum. Ég nota yfirleitt sjálfur þá aðferð þar sem % aðferðin felur í sér hættu á of löngum textalínum sem vont er að lesa, læt mig þá frekar hafa það að nýta ekki skjáinn eins vel. En þetta er matsatriði fyrir hvern vef.

height virkar eins og **width**, nema á lóðréttum öxli. Ekki er ráðlegt að setja height á töflur nema öruggt sé að þær ýti þar með ekki öðru efni niðurfyrir glugga vafra. **Athugaðu að height á table er ekki stutt í XHTML.**

border setur ramma á töflur. Til að fá engan ramma er notað **border="0"**. Notaðu síðan CSS til að stýra útliti rammans (kantsins) ef þú vilt.

Stundum er nauðsynlegt að hafa **border="1"** á meðan verið er að vinna við töflur. Ramminn gefur manni vísbendingu um hvað að er þegar útkoman er önnur en sú sem maður ætlaði.

cellspacing myndar autt svæði umhverfis innihald einstakra hólfa. Gildi í skjápunktum. Notaðu frekar CSS, **padding** eða **margin**.

cellpadding myndar fylltan ramma umhverfis innihald einstakra hólfa. Gildi í skjápunktum. Notaðu frekar CSS, **padding** eða **margin**.

Ef ætlunin er að líma saman myndir með töflu, setja þær alveg saman hvor við aðra, þá þarf að gæta að því að **cellpadding="0"** og **cellspacing="0"**. Einnig þarf að gæta að því að vafrar eiga það til að líta á línubil í kóða sem x skjápunkta bil, þannig að í þessum tilvikum getur reynst nauðsynlegt að búa til ljótar, langar línur svo ekki myndist bil milli myndanna. Sama gildir ef stilla á millibil á milli bakgrunnslitaðra sella. Þú getur líka notað CSS:

border-collapse: separate; og
border-spacing: 0 0; sem sett er á **table**.

bgcolor setur bakgrunnslit ílátsins sem það er skilgreint í, hvort sem það er heil tafla, röð eða sella. Notaðu frekar CSS, **background-color**

align stýrir því síðan hvar tafla situr á síðunni á láréttuna.

CSS

Ég nota CSS töluvert til að stýra töflum, stærðum, litum, útliti kanta ofl. Eftir-farandi skipanir nota ég oft en ekki endilega allar saman. Þær virka líka á **<tr>** og **<table>**. Skoðaðu líka kóðabútana á CSS-síðunni hér að framan, þar eru t.d. skipanir sem snúa að útlínum taflna o.f.l.. Fimm efstu hér að neðan stýra augljóslega útliti texta.

```
td{      font-family : Verdana, Arial, sans-serif ;
        font-size : 12pt;
        font-weight: bold;
        line-height: 16pt;
        color: #000000;

        text-align: center;
        vertical-align : top;
        background-color : #ffff00;

        border-width: 1px;
        border-color: #000;
        border-style: solid;

        height: 200px;
        width: 30%;

        padding : 20px;
        padding-left: 10px;
        padding-top: 5 px; }
```

Raðir og sellur - <tr> og <td>

<tr></tr> table row, raðir. Það sem inni þetta ílát fer er allt í láréttri röð. Breytur sem virka í **<tr>** eru **bgcolor**, **valign** og **align**, sjá útskýringu á breytum neðar. Breyturnar virka þá á allar sellur í viðkomandi röð.

<td></td> table data, hólf, sella. Hér inni fara upplýsingarnar, jafnt texti og myndir. Möguleikar inni **<td>** eru:

```
<td align="left eða center eða right" valign="top eða  
middle eða bottom" rowspan="X" colspan="X" bgcolor="white"  
background="mynd.gif">
```

align stýrir því hvar efni hólfis situr inni henni á láréttunni. **align="left eða center eða right"**. CSS, **text-align**

valign stýrir því hvar efni hólfis situr á lóðréttunni. **valign="top eða middle eða bottom"**. CSS, **vertical-align**

colspan gerir svipað og **rowspan** en segir til um hvað hólfíð teygir sig mörg **<td>** (lóðrétt hólf) til hliðar. Ekki hægt með CSS.

rowspan er notað til að fylla uppí göt sem raðir neðar í sömu töflu mynda með því að hafa fleiri **<td></td>** í sér. Það segir semsagt til um hversu mörg **<tr>** (láréttar raðir) röðin spannar niður. Ekki hægt með CSS.

Sjá kóðadæmi á næstu síðu með **colspan** og **rowspan**

Einnig er hægt að hugsa um **rowspan** og **colspan** sem einingar sem töflu er skipt uppí, ef ein **<td>** er með **colspan="8"** og er eina sellan í þeirri **<tr>**, þá er hægt að skipta þeirri töflu í 8 hluta á þvervegin.

Mögulegt er að setja inn **width** og **height** breytur á einstakar sellur en það er ekki örugg leið, þó oftast virki hún vel í nýrri vöfrum. Erfitt er að búa til fast útlit sem kemur eins út í flestum aðstæðu, þú getur stundum þurft að láta myndir stýra stærð sellanna.

Hægt er að nota **background="mynd.gif"** sem breytu á töflu, röð eða sellu. Sömu varúðarreglur gilda þar eins og með **background** breytuna í **<body>** leshæfni vefsins vegna. Þú hefur mun meiri stjórn á bakgrunnsmyndum með CSS.

Colspan og rowspan dæmi

```
<table border="1">
  <tr>
    <td colspan="2">colspan="2"</td>
  </tr>
  <tr>
    <td>Eðlileg</td>
    <td>Eðlileg</td>
  </tr>
</table>
```

colspan="2"	
Eðlileg	Eðlileg

```
<table border="1">
  <tr>
    <td rowspan="2">rowspan="2"</td>
    <td>Eðlileg</td>
  </tr>
  <tr>
    <td>Eðlileg</td>
  </tr>
</table>
```

rowspan="2"	Eðlileg
	Eðlileg

Töflur inni töflum

Oft þarf að setja töflur inn í töflur til að ná ákveðnu umbroti. Stundum gerir maður það líka frekar en búa til mjög flóknar töflur með **colspan** og **rowspan** í aðalhlutverkum. Þetta er í raun mjög einfalt að framkvæma, hér að neðan sést hvernig þú notar þessa aðferð til að setja umbrotið efni (innri taflan) í ytri töflu til að miðja bæði á lóðréttuna og láréttuna, óháð stærð vafraglugga.

```
<table border="1" width="100%" height="100%">
<tr><td valign="middle" align="center">
```

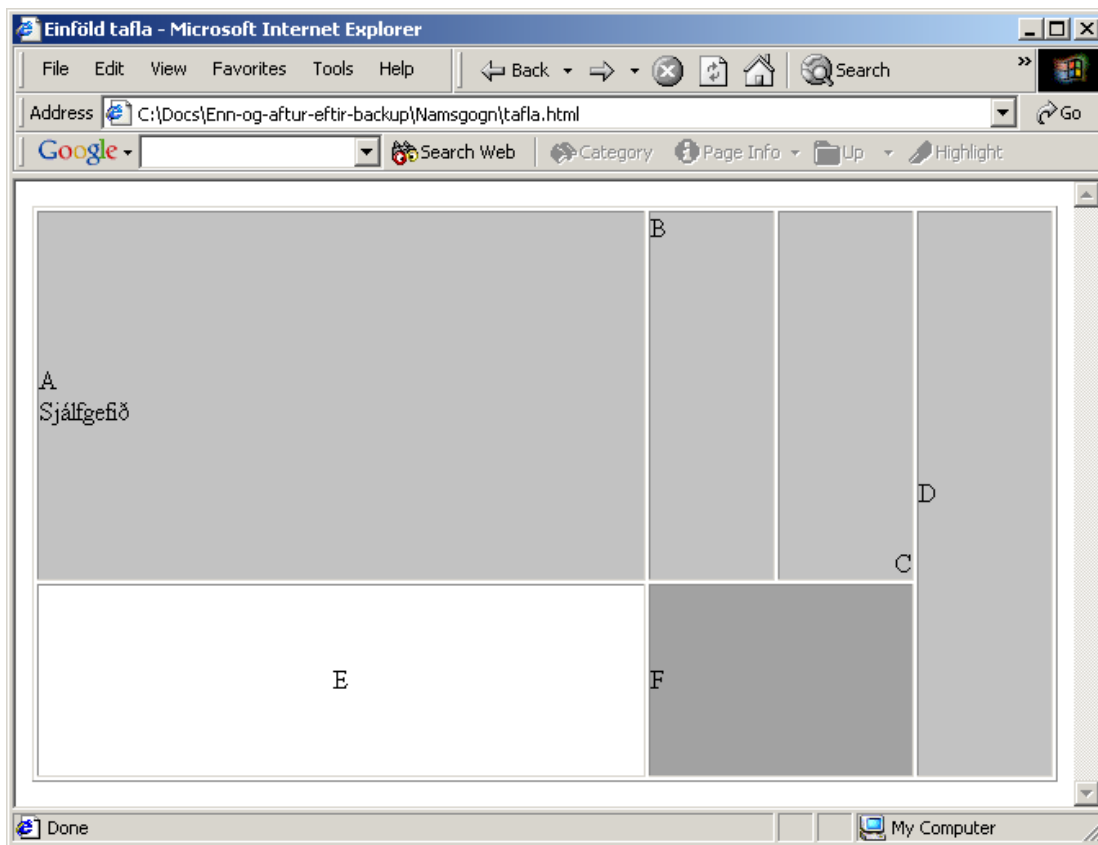
```
<table width="250" height="200">
<tr>
<td>Hér er vinstri selli í miðju</td>
<td>Hér er hægri selli í miðju</td>
</tr>
</table>
```

```
</td></tr>
</table>
```

Töfludæmi með algengustu breytum

```
<table border="1" width="100%" height="100%" cellpadding="0">
<tr bgcolor="#bbbbbb">
<td>A<br />Sjálfgefið</td>
<td valign="top">B</td>
<td rowspan="2" align="right">C</td>
<td colspan="2">D</td>
</tr>
<tr>
<td colspan="2" align="center">E</td>
<td colspan="2" bgcolor="#999999">F</td>
</tr>
</table>
```

Þessi kóði gefur okkar töfluna að neðan:



Tafla með föstum hliðum og flæðandi miðju

Til að smíða töflu sem er með hliðardálkum af fastri breidd og miðdálki sem stækkar og minnkar eftir stærð gluggans þá er það gert svona:

```
<table width="100%">
<tr>
<td width="200">Fastur dálkur</td>
<td>Þetta svæði flæðir eftir gluggastærð</td>
<td width="200">Fastur dálkur</td>
</tr>
</table>
```

Töflur og myndir vinna saman

Hér að neðan sést hvernig myndir eru notaðar til að stýra hæð og breidd töflusella. Þessi leið er að vissu leyti úrelt en virkar í raun enn mjög vel þó CSS hefur að mestu komið í stað þessarar aðferðar.

Taflan er með `width="595"` sem tryggir að útlitið haldist eins þó gluggi vafra sé misstór og samanlögð breidd mynda er líka 595 skjápunktar. Ef ætlunin væri að mjókka vefinn um t.d. 5 skjápunkta væri nóg að minnka myndina af stelpunni og breyta width í 590. Athugaðu að þessi síða var smíðuð á þeim tíma þegar lægsti sameiginlegi nefnari vefgesta var 640 x 480 skjápunkta upplausn, ef hún væri smíðuð í dag væri breiddin u.þ.b. 760 skjápunktar.

<ul style="list-style-type: none"> • Dagskrá • Atriði • Um Listahátíð 		<p style="text-align: right;"><i>Listahátíð</i> Í REYKJAVÍK 1998 16 MAÍ-7 JÚNÍ</p>
	<p style="text-align: center;">Velkomin</p> <p>Í sumarbyrjun 1998 verður haldin Listahátíð í Reykjavík í 15. sinn. Hátíðin er haldin annað hvert ár og er einn stærsti lista- og menningarviðburður á Íslandi. Á hátíðinni verður fjöldi tónleika og myndlistarsýninga, leiksýningar, danssýningar, fjölleikahús og bókmenntaviðburðir.</p> <p>Á dagskránni eru viðamikil erlend atriði sem að öllu jöfnu myndu ekki sækja Ísland heim og innlend atriði sem spegla vel það listalíf sem hér blómstrar. Listahátíð er fyrir alla, unga sem aldna. Það verður hátíð í borginni næsta vor og margt framandi og óvenjulegt á seyði.</p> <p style="text-align: center;">þar sem straumar mætast</p>	<ul style="list-style-type: none"> • MIDASALA OG MIÐAVERÐ • SKRIFSTOFA OG STARFSFÓLK • FULLTRÚARÁÐ • STYRKTRADILAR • AÐRAR HÁTÍÐIR 

Rammar - <frames>

Rammar grundvallast á því að eitt yfirsíð skjal kallar í síður og segir í hvaða ramma þær eiga að fara. Hægt er að kalla á hvaða .html skjal inni ramma en oftast þarf að laga þau að rammahugsun. **Þó að þú vinnir með ramma, þá er oftast nauðsynlegt að nota annað hvort töflur eða stílsnið til að stilla efnið af á undirsíðunum.**

Það á alls ekki alltaf við að nota ramma og rammasiður er erfitt að gera vel, mun erfiðara en hefðbundnar vefsíður. Stórir gallar í túlkun eldri útgáfa Netscape (og Explorer) á römmum og hræðileg misnotkun slæmra vefara gera það að verkum að þeir hafa lengi verið eitt hataðasta fyrirbæri vefsins. Engu að síður eru rammar mjög öflugt tæki í réttum höndum, þegar við á.

Helstu **kostir** þeirra eru að auðvelt er að gera vef skipulegan og gegnsæjan, auk þess sem fastir hlutar síðunnar fara sjaldan/aldrei af skjánum (ef rétt er ofið) sem gerir upplifunina samfelldari. Að auki þarf tölva vefgests ekki að teikna fasta hluti aftur og aftur sem gerir birtingartímann styttri.

Helstu **gallar** þeirra eru:

1. Leitarvélar ráða illa við rammavefi og þær síður sem finnast búa oftast til næsta vandamál sem er:

2. Ef tengill er í einstaka rammasiður frá öðrum vef, þá kemur einungis sú einstaka síða og því vantar í flestum tilfellum þau föstu atriði sem eru í öðrum römmum, tengla, lógó og fleira sem máli getur skipt. Þess vegna er nauðsynlegt að setja í það minnsta fót með grunnupplýsingum um eiganda vefs og tengli í forsiðu á allar síður sem notaðar eru í rammavef.

3. Bókamerki virka ekki vel í Netscape/Firefox né Operu. Ef þú setur bókamerki á rammasiðu þá færðu alltaf fyrstu síðuna sem var í því rammasetti, óháð því hvað þú varst búinn að smella þig djúpt ofaní vefinn. Framhjá þessu er hægt að komast með því að vera með margar yfirsíður, eina fyrir hvern stað sem þú vilt að vefgestir geti merkt hjá sér. Þessi leið skemmir afturámóti töluvert fyrir kostum ramma og kallar á aukna vinnu.

Ef þig langar í lítinn kassa á miðri síðu þar sem birtist skrunrönd ef að efni er of mikið til að passa í svæðið þá skaltu hugleiða að nota **<iframe>** sem þú getur fræðst um á bls. 99.

Skipanir ramma

Einfaldasti rammakóði lítur svona út:

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Frameset//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-frameset.dtd">

<html>
<head>
<title>Forsíðan ræður titlinum</title>
<meta http-equiv="Content-Type" content="text/html;
charset=UTF-8" />
</head>

<frameset cols="150,*">
  <frame name="tenglar" src="tenglar.html" />
  <frame name="meginmal" src="byrja.html" />
</frameset>

</html>
```

Og svona lítur þetta út í vafra:



Athugaðu að innihaldið kemur frá `tenglar.html` og `byrja.html` sem kallað er á í kóðanum að ofan. Það þarf sem sagt þrjú skjöl til að búa til einfalda rammásiðu: Aðalskjalið sem inniheldur kóðann að ofan og tvö til viðbótar sem kallað er í og innihalda efnið sem birtist á síðunni.

Útlit rammanna

Inní `<frameset>` ílætið fara allar skipanir sem lúta að skipulagi og uppsetningu ramma. `</frameset>`

```
<frameset cols="breidd,breidd,breidd"> eða
<frameset rows="hæð,hæð,hæð">
```

Þarna er skilgreint hversu marga ramma vefsíðan skiptist í og hvernig hún skiptist á milli þeirra. Athugaðu að fjöldi hæðar og/eða breiddarskilgreininga stýrir því hversu marga ramma við fáum. Í dæmunum að ofan mun skjárinn skiptast í þrennt.

Hægt er að skilgreina rows og cols á þrjá vegu: Í skjápunktum (pixels), prósentum og hlutfallaskipting.

```
rows/cols=xxx gefur xxx gildi í skjápunktum
rows/cols=xx% gefur gildi í prósentum af heildarstærð ramma
rows/cols=* nýtir það sem eftir er af heildarstærð ramma.
```

Í flestum tilvikum virkar best að blanda saman skjápunktum og *. Varast ber að nota skjápunkta eingöngu, þar sem það getur leitt af sér of breiða ramma fyrir skjábreidd.

```
<frameset rows="100,2*,*"> Þessi skipun gefur fyrstu röð 100 skjápunkta, annari röð 2/3 af því sem eftir er og þeirri þriðju 1/3.
```

`frameborder`, `border` og `bordercolor` eru skipanir sem stýra útliti línanna sem aðskilja rammanna.

`frameborder` tekur gildin `yes` og `no` sem stýra útliti línanna í Firefox en í Explorer á það að afnema línuna ef gildi er `"0"`.

`border` tekur gildi frá 0-x og stýrir þykkt línunnar sem aðskilur rammanna, 0 = enginn lína.

```
bordercolor stýrir lit línunnar og tekur kunnugleg gildi,
bordercolor="#RRGGBB"
```

Rammaleysi

`<noframes>` Hér seturðu efni fyrir veflesara sem ekki skilja ramma, hvort heldur sem er heila síðu, einhverja tengla eða það sem algengast er; tilkynningu um að gestur þurfi að nota vaфра sem styður ramma ef hann ætlar að geta skoðað vefinn. Það sem hér er kemur ekki fram í veflesurum sem skilja ramma `</noframes>`. Þetta er lítið notað í dag, næstum allir vafrar sem er í notkun styðja ramma.

Ekkert `<body>` á yfirsíðum ramma

Á þeim síðum sem rammur eru skilgreindir á ekki að vera neitt `<body>` tag! `<body>` skipanir tilheyra síðunum sem vísað er í. Ef `<body>` er sett inni rammásiðu þá virkar síðan illa eða ekki þar sem rammásiðan inniheldur ekki neitt efni, þ.e. texta eða myndir og skilgreiningar á litum og öðru sem sett er inni `<body>`.

Undirskjöl ramma

`<frame src="nafnsidu.html">` Skilgreinir hvaða skjal á að opnast inni viðkomandi ramma.

Möguleikar eru:

```
<frame src="skra.html" name="nafn" marginwidth="X"
marginheight="X" scrolling="no" noresize="noresize" />
```

`src` vísar í skjal sem hleðst inni ramma í upphafi

`name` er nafn sem þú gefur rammanum til að geta vísað öðrum skjölum inni hann síðar, ekki sleppa þessari breyту.

`marginwidth` notar þú (eða ekki) til að skilgreina spássíur til sinnhvorrar hliðar efnis í skjápunktum

`marginheight` gerir svipað en nú á lóðréttu kantinn

`scrolling=yes/no/auto` segir til um hvort skrunrönd myndast við ramma. `auto` er sjálfgefið en varist gildið `no`, það getur gert vefinn ófæran þeim sem hafa lága skjáupplausn.

`noresize` þýðir að sá sem er að skoða getur ekki breytt stærð rammans hjá sér. Það hefur augljóslega áhrif á næstu ramma og getur gert vefinn ófæran þeim sem hafa lága skjáupplausn.

Vísanir í ramma

Til að vísa ákveðnu skjali í ákveðinn ramma er notað

```
<a href="nafnskjals.html" target="nafn_ramma">
```

Möguleikar í notkun `target` eru nokkrir en sá þeirra sem oftast er notaður er `target="_top"` sem lætur viðkomandi síðu taka yfir vafra gluggann. Ef nafn það sem gefið er upp í `target` er ekki til sem rammi, þá opnast nýr gluggi með viðkomandi síðu og eftir það er hægt að nota það nafn til að vísa öðrum síðum í. Einnig er hægt að nota `target="_blank"` til að þvinga nýjan glugga.

Hægt er að setja á einn stað `target` breyту sem virkar fyrir alla tengla síðunnar. Það er sett inn í `<base>` (sjá `<base>` kafla) og lítur svona út::

```
<base target="nafn_ramma" />
```

Allir tenglar síðunnar munu þá vísa í rammann sem er með `name="nafn_ramma"`

Flóknari rammasíða

Þessi kóði býr til það sem sést á skjámyndinni fyrir neðan, þrískiptir vafraglugganum. Þú getur að sjálfsögðu búið til mun flóknari rammasíður en þessa, en það er sjaldnast góð hugmynd. Bæði fara þá rammarnir að taka mikið pláss á skjánum og eins er sjaldnast nokkur þörf á slíkum rammafjölda.

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Frameset//EN
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-frameset.dtd>

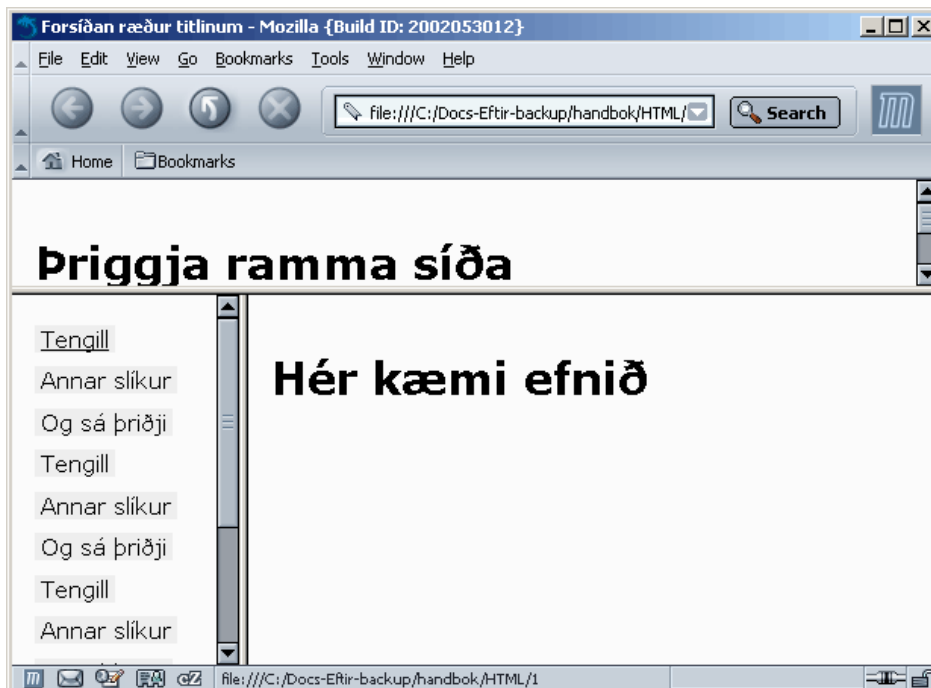
<html>
<head><title>Forsíðan ræður titlinum</title></head>

<frameset rows="70,*">
  <frame name="toppur" src="haus.html" />

<frameset cols="150,*">
  <frame name="tenglar" src="tenglar.html" />
  <frame name="meginmal" src="byrja.html" />
</frameset>

</frameset>

</html>
```



Skrunrendurnar birtast ef þörf er á þeim, þú getur slökkt á því en þá tekurðu áhættuna á að sumir notendur geti ekki nálgast allt efnið þitt.

Innskotsrammar

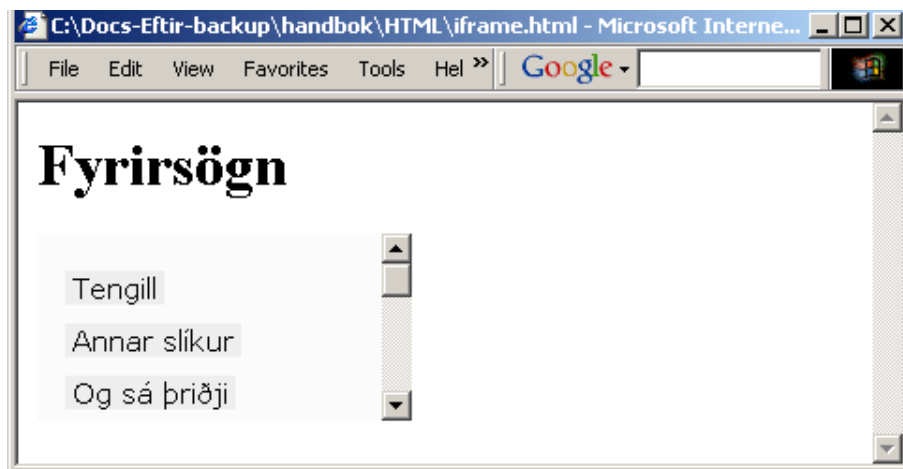
`<iframe>` er tiltölulega nýleg skipun og hugsanlegt er að einhverjir vafrar sem enn eru í notkun skilji hana ekki. Til öryggis er gott að setja inn tilvísanir fyrir þá vafra, það er ekki gert með `<noframes>` heldur eins og sést hér að neðan:

```
<iframe src="tenglar.html" width="200" height="100"
frameborder="0" name="efni">Texti sem birtist í vöfrum
sem ekki skilja þessa skipun, gott getur verið að setja
tengil í skrána hér</iframe>
```

Ef svona er kóðað mun innihald skra.html birtast í 200 x 100 skjápunkta glugga á þeim stað sem skipunin er sett inni skjalið, þú getur sett `iframe` á alla staði sem má setja innihald á.

Athugaðu að innskotsrammar prentast illa út, flestir vafrar prenta út það sem sést á skjánum sem getur leitt til þess að neðri hluta efnisins birtist ekki á pappírnum. Leið til að komast framhjá þessu er að setja tengil (Prenta) í innskotskrána í hana sjálfa og nota target til að láta opnast í nýjum glugga. Ekki góð leið en betri en engin.

Innskotsramma er ekki hægt að vinna með í myndrænu viðmóti DW og NS4 skilur þá ekki. Að neðan sést dæmi um notkun `iframe`:



Vandamál með skrunrendur

Ef IE vill endilega birta skrunrönd utan um rammann þá hef ég ekki fundið neina endanlega lausn á því vandamáli en sambland af eftirfarandi reddar oftast málinu:

1. Passaðu að efnið í skjalinu sem þú ert að vísa í sé ekki of stórt fyrir rammann. Athugaðu að skrunröndin tekur pláss inn í `iframe` rammann.
2. Prófaðu að bæta þessum kóða fremst í skrána sem kallað er í:
`<?xml version="1.0" encoding="iso-8859-1"?>`
3. Prófaðu að breyta skránum þínum í HTML4 eða XHTML, allt eftir hvort þú varst að nota.
4. Settu `scrolling="yes"` breytu á `iframe` skipunina.

Innsláttarform - <form>

Hægt er að setja upp þægileg form fyrir notandann til að biðja um upplýsingar. Þessi form er hægt að nota til margra hluta: Gestabækur, próf, pantanir, vefumsjónarkerfi, atkvæðagreiðslur; allt þetta notar innsláttarform og lengi voru þau eina leiðin okkar til að kalla eftir innleggi frá vefgestum okkar, eina leiðin til að smíða gagnvirkan vef. Notkun þeirra hefur aukist mikið á síðari árum eftir því sem leiðum til að vinna með upplýsingar úr þeim hefur fjölgað.

Unnið með upplýsingar úr formum

1. Til að vinna með upplýsingarnar úr forminu eru til margar leiðir, JSP, PHP, ASP og cgi-bin (Perl eða C) eru þær algengustu í dag. Allar þessar leiðir eiga það sameiginlegt að þurfa ýmiskonar réttindi og sérstaka uppsetningu á vefmiðlara. Mismunandi er eftir hýsingaraðilum hverjar þessara þjónusta eru uppsettar og oft er notandinn takmarkaður við þá aðferð sem hýsingaraðilinn vill nota.

2. Til eru vefþjónustur sem bjóða uppá að vinna með upplýsingarnar fyrir okkur á þann hátt sem við viljum, ef þú ert með einfaldar þarfir þá getur slík þjónusta verið besti kosturinn fyrir þig. Tengill í slíka er á where.is/webdesign. Ef þú ætlar að fara þessa leið þá borgar sig fyrir þig að lesa leiðbeiningarnar þar áður en þú smíðar formið, það gæti sparað þér einhverja leiðréttingarvinnu.

3. Einnig er hægt að nota Javascript til að höndla upplýsingarnar úr forminu en það er ekki öruggt til árangurs sökum mismunandi góðs stuðnings vafra við Javascript og þess að sumir vefgestir eru með slökkt á Javascript sem mun því orsaka að sumir gestir geti ekki notað formið.

4. Síðasta leiðin er svo að nota mailto sem sendingaraðferð. Hún treystir á póstforrit á vél notanda sem ekki er víst að sé rétt uppsett og skilar textanum að auki mjög brengluðum, setur + í stað orðabíla ofl. Þessi leið skilar ekki svarsíðu og skilur notendann eftir í óvissu um hvort sending hafi tekist.

Ég mæli ekki með að neinn noti annað en það sem lýst er í liðum 1 og 2 hér að ofan. Hvað síðan er notað af því sem er í lið 1 veltur á aðstæðum og þekkingu hvers og eins. Ég nota sjálfur nær eingöngu Perl þegar ég þarf að fá niðurstöðurnar í pósti en PHP ef ætlunin er að vista niðurstöðurnar í gagnagrunni.

Kóðinn sjálfur

```
<form method="post" action="http://www.server.is/cgi-bin/post-query">
```

```
<form method="get" action="mailto:gunnar@where.is">
```

`form` segir vafra að hér að neðan sé innsláttarform.

`method` hefur möguleikana `post` og `get` en réttast er að nota `post` því `get` er í úreldingu

`action` segir skjalinu hvar forritið sem höndlar beiðnina er staðsett ef `action="URL"` en sendir póst á netfang sem sett er í `mailto`.

Ef `mailto` er notað þá skilar textinn sér til viðtakanda illæsilegur sökum tákna í stað bila og brenslaðra íslenskra stafa. Til eru forrit sem meðhöndla þennan póst og breyta í skiljanlegt form.

Hér að neðan eru nokkrir helstu möguleikar forma sýndir.

1. Einfaldur textainnsláttargluggi:

Netfang: ` <input name="netfang" />` Eða fyrir aftan

Netfang: Eða fyrir aftan

2. Textainnsláttargluggi með skilgreiningu á lengd:

Hvar geymirðu vefinn þinn?`
<input type="text" name="home" size="40" value="http://" />`

`size` er þá breidd í stöfum og `value` er texti sem birtist inni forminu

Hvar geymirðu vefinn þinn?

3. Valhnappar þar sem þú getur valið fleiri en einn möguleika:

```
<input type="checkbox" name="dottir" value="Y" />Dóttir  
<br />  
<input type="checkbox" name="sonur" value="Y" />Sonur
```

Dóttir
 Sonur

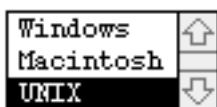
4. Valhnappar þar sem aðeins er hægt að velja einn:

```
<input type="radio" name="foreldrar" value="mamma" />  
Mamma<br />  
<input type="radio" name="foreldrar" value="pabbi"  
checked />Pabbi
```

Mamma
 Pabbi

5. Vallisti þar sem hægt er að velja um fleiri en einn möguleika:

```
<select name="tegund" size="3" multiple="multiple">  
<option>Windows</option>  
<option>Macintosh</option>  
<option selected="selected">UNIX</option>  
</select>
```



Ef `size` breytu er sleppt úr, verður valglugginn að fellivalmynd.

`selected` merkir viðkomandi lið á valmynd

`multiple` þýðir að hægt er að merkja fleiri en einn, til þess þarf gestur að halda SHIFT niðri á meðan valið er

6. Stór textainnsláttargluggi með skilgreiningu á stærð:

```
<textarea name="abendingar" rows="6" cols="40">Texti sem birtist inni glugga</textarea>
```

Texti sem birtist inni glugga

Svona glugga er líka hægt að nota til að birta texta sem skrunar inni skilgreindu svæði, með því að nota `value="texti sem birtast á í skrunglugga"`. Þetta er svipuð virkni og `<iframe>` en er mun takmarkaðri (ekki er hægt að nota umbrotsskipanir á textann) en virkar í öllum vöfrum.

7. Hnappar sem smellt er á til að senda form og hreinsa ef viðkomandi vill byrja uppá nýtt:

```
<input type="submit" value="Ábendingar" />  
<input type="reset" value="Hreinsa" />
```

`</form>` Til að enda innsláttarform

CSS

Til að forðast að t.d. valhnappar verði bæði ljótir og ónothæfir er best að nota klasa á langflestar CSS breytingar á formum. Ekki nota klasa sem undantekningar heldur sem reglu. Flestar almennar CSS breytur virka ágætlega á innsláttarreiti og sendihnappa, heldur færri á fellivalmyndir og fæst gerir góða hluti á valhnappa. Prófaðu vel það sem þú gerir, sérstaklega **padding** og **border-width**, ég myndi ekki nota meira en **1px** gildi á þessar breytur.

```
padding: 1px;  
width: 180px;  
color: #000080;  
background-color: #E0FFFF;  
border-style: solid;  
border-width: 1px;  
border-color: #000000;
```

Leitarorð og önnur lýsigögn

Til að setja inn leitarorð fyrir leitarvélarnotum við `<meta>` skipunina.

`<meta>` skipanir fara inn í `<head>` ílátið.

Efstu tvö er mjög mikilvægt að setja inn fyrir leitarvélarnar, í bókinni "Viðmót, hönnun og verklag" er meira um þetta efni.

1. Lykilorð, 3-5 fyrstu geta nýtst, fyrir leitarvélarnar með kommu á milli leitarorða
`<meta name="keywords" content="Vefsmíðar, heimasíðugerð, viðmót" />`

2. Lýsing á vefnum fyrir leitarvélarnar

`<meta name="description" content="Þessi vefur er ætlaður vefsmíðum og inniheldur bæði tengla og ýmsar upplýsingar sem geta komið slíkum að gagni." />`

3. Þessar þrjár að neðan segja vefvöfrum, proxyþjónum og leitarvélum að skjalið sé í sífelldri endurnýjun, eigi ekki að geymast í flýtiminni þeirra. Sú fyrsta segir að það verði úrelt á dagsetningu þeirri sem er í `"content"`. Örugkast er að nota þær allar en ekki nota þær nema á síður sem oft er breytt því þær tefja birtingu síðunnar.

```
<meta http-equiv="Expires" content="06-Okt-1992
09:30:00 GMT" />
<meta http-equiv="pragma" content="no-cache" />
<meta http-equiv="cache-control" content="no-cache" />
```

4. Meta-refresh beinir vefflökkurum sjálfkrafa á tilgreinda síðu. Ef engin slóð er tilgreind, hleður hún sjálfri sér inn. content tekur gildi í sekúndum.

```
<meta http-equiv="Refresh" content="12; URL=http://
vefslod.midlari.is/skra.html" />
<meta http-equiv="Refresh" content="10" />
```

5. Hægt er að búa til eigin meta name, t.d. til að merkja sér síðurnar eða setja inn atriðisorð fyrir eigin leitarvél.

```
<meta name="eitthvad" content="ymislegt" />
<meta name="author" content="Nafn vefsmíðs" />
```

Favicon, lógó í slóðarreit vafra

Til að setja inn eigið lógó í Address/Location glugga í vafra seturðu inn eftirfarandi kóða og mynd í rót vefsvæðis miðlara.

```
<link href="/favicon.ico" />
```

Meiri upplýsingar um þetta færðu á <http://www.favicon.com/>

<base>

<base> fer inni <head> ílátíð.

```
<base href="http://vefslod.midlari.is/">
```

Base nota ég til að breyta grunnveffangi því sem tenglar á viðkomandi síðu vísa á. Ef base er sett á síðu sem vistuð er á <http://this.is> með ofangreindum gildum, þá breytast allar tilvísanir sem ekki nota <http://>, þ.e. "relative paths" og færast yfir á <http://http://vefslod.midlari.is/> Þetta er þægilegt að nota ef maður vill skoða hvernig vefsíða er smíðuð án þess að þurfa að ná í allar myndirnar með henni. Þá nær maður í eintak af .html skjalinu til sín og bætir

```
<base href="http://http://vefslod.midlari.is/">
```

 inni og getur síðan sett border á töflu, breytt bakgrunni og gert raunar hvaða breytingar sem vill. Ekki klikka á skástrikinu fyrir aftan slóðina.

Hnitamyndir

Hnitamyndir (imagemaps) eru myndir sem innihalda x fjölda tengla sem fara með þig á mismunandi vefföng eftir því hvar smellt er á myndina. Til eru 2 tegundir slíkra mynda, annars vegar sú eldri, "server-side imagemaps", þar sem vísanirnar eru geymdar á vefmiðlaranum og hins vegar: "client-side imagemap" sem tekur vísanirnar úr XHTML skjalinu. Núorðið nota flestir eingöngu "client-side".

Þægilegasta leiðin til að búa til hnitamyndir er að nota þartilgerð forrit þar sem þú merkir inná myndina hvaða svæði þú vilt að vísi hvert. Forritið sér síðan um að breyta merkingum þínum í upplýsingar sem vefmiðlarinn/þjónninn skilur. Þetta er sáraf einfalt í framkvæmd, en erfitt að útskýra. Forrit sem búa til slíkar skrár eru til fyrir flest stýrikerfi og er hægt að nálgast þau á Netinu. Einnig eru þau innbyggð í mörg vefsmíðatól, þ.á.m. Dreamaweaaver.

Til að segja vafra að um hnitamynd sé að ræða er sett inni `img` skipunina ISMAP eða USEMAP. ISMAP er sett inn fyrir server-side, en USEMAP fyrir client-side. Hægt er að setja inn báðar og á þá vafri sem ekki þekkir client-side að ná í hina.

Oft er betra að klippa myndirnar niður og líma saman með töflunum en stundum er hnitamynd eina leiðin, sérstaklega ef vefgrafík er flókin og/eða mjög samfelld, landakort eru t.d. oftast tengd sem hnitamyndir.

Dæmi um tilvísun í usemap kort:

```
<a href="http://www.rhi.hi.is/cgi-bin/kort"></a>
```

Og hnitakóðinn lítur svona út:

```
<map name="kort">  
<area shape="rect" coords="43,262,273,381" href="#">  
<area shape="circle" coords="174,233,27" href="#">  
<area shape="poly" coords="110,138,110,216,214,163,143,1  
71" href="#">  
</map>
```

Í staðinn fyrir # er síðan settur tengill í þá síðu sem þetta svæði á myndinni á að vísa á.

Til að geta sett upp server-side hnitamynd þarf annað hvort aðgang að svokölluðu cgi-bin eða aðstoð kerfisstjóra, oftast er client-side látið duga núorðið.

Server side includes

Server-Side Includes (SSI) er mjög skemmtileg leið til að halda utan um vefi sem ekki þurfa að byggjast upp á gagnagrunnum. SSI byggir á sinn hátt á svipaðri aðferð og rammur. Úr hvaða vefsíðu sem er getur þú vísað í aðra skrá sem smellist inn í vefsíðuna á þeim stað sem þú setur tilvísunina. Í raun má segja að SSI hafi verið undanfari kerfa eins og PHP, ASP og JSP.

Þetta opnar möguleika á að geyma allan texta sem sérstakar skrár sem þú getur síðan vísað í þaðan sem þú vilt. Þessar textaskrár geta síðan vinnufélagar og aðrir þeir sem sjá um textasmíð og viðhald skrifað inni án þess að hafa nokkra minnstu þekkingu á HTML eða vefsíðugerð almennt. Einnig er þá hægt að vera með útlit vefsins í sér skráum og því hægt að skipta um alla grafík og umbrot með því að breyta örfáum skráum.

Flestir vefmiðlarar ráða við SSI en kerfisstjóri þarf að kveikja á því í sínum vefmiðlara. HTML skjöl sem nota SSI þurfa að hafa viðskeytið .shtml. Hafðu einnig í huga að erfitt er að vinna SSI vefi öðruvísi en með beinan aðgang að vefsvæði. Það er afturamóti lítið mál að setja upp vefþjón, t.d. apache á vinnuvélinni þinni og vinna vefina þar og senda síðan á endanlegan vefþjón. Dreamweaver getur unnið með SSI að einhverju leyti.

Kostir:

Vefsíða búin til af miðlara, ekki þarf að hafa áhyggjur af stuðningi vafra
Hægt að geyma útlit og texta sitt í hvoru lagi og breyta og bæta hvort um sig án þess að snerta hitt.

Gallar:

Þróaðri not krefjast forritunarþekkingar
Þarf að kveikja á SSI í vefmiðlara, ekki allir kerfisstjórar eru hrifnir af því

Notkun:

Til að líma skrá inni .shtml skjal er notuð eftirfarandi skipun:

```
<!--#include virtual="fotur.ssi"-->
```

Ef þessi lína (að ofan) er sett neðst í skjal, á þann stað sem þú vilt að kóðinn birtist þá getur þú haft eina skrá sem inniheldur þær upplýsingar sem eiga að vera neðst á öllum síðum. Ef þú síðan breytir einhverju í þessari skrá þá breytist það á öllum síðunum sem vísa í hana.

Þetta eru algengustu not SSI en langt í frá þau einu. Það er t.d. hægt að nota SSI til að setja inn dagsetningar ("Þessu skjali var síðast breytt xx.xx.2002") á síður, þær eru þá lesnar úr skráakerfinu og eru því alltaf réttar.

Tengla í upplýsingar um SSI er að finna á where.is/webdesign

Úrehtar og heimskulegar skipanir

Fyrirsagnir - ``

Athugaðu að `` skipunin er ekki studd af XHTML staðlinum, hún virkar þó í öllum vöfrum og er líkleg til að gera það á komandi árum. XHTML Transitional leyfir `font` skipunina en miklu betra er að venja sig strax á að nota CSS í stað ``. Læt hér fylgja með gamlar leiðbeiningar um notkun `font` skipunarinnar en mæli eindregið gegn því að þú notir þessa skipun.

`Stærsti mögulegi texti` Hægt að nota inni miðjum texta. Gildi frá 1-7 möguleg, 1 minnst og 7 stærst, sjálfgefið er 3.

`Setur lit á texta.` Sömu reglur gilda um litina þar einsog þegar settur er litur inni `<body>` skipun. Athugið samt að breytingar á textalit geta ruglað fólk í tenglarýminu.

`Breytir leturgerð` Nauðsynlegt er að skilgreina nokkrar leturgerðir þar sem mismunandi stýrikerfi og tölvur hafa ekki sömu leturgerðir. Gæfuleg gildi eru það sem að ofan er notað auk "**Times,Times New Roman,serif**" og "**Courier New, Courier, monospace**". Ef æskilegt er að fá fram aðrar leturgerðir er hægt að setja nöfn þeirra fremst í aðra hvora romsuna, þá eftir því hvort viðkomandi letur er serif eða sans-serif.

Hægt er að nota `` til að breyta lit, letri og stærð á texta.

Undirstrikaður texti

`<u>` í XHTML og `text-decoration: underline;` í CSS er ekki mjög gáfulegt að nota á texta því þrátt fyrir að tenglar hafi lengi verið án undirstrikunar á mörgum vefjum þá er það mjög fast í vefgestum að undirstrikaður texti sé tengill. Notaðu frekar feitletrun, skáletrun eða jafnvel ýmsar mýkri leiðir CSS til að leggja áherslu á textasvæði.

Breytur á `
` - nota CSS í staðinn

`<br clear="all" />` Ýtir efni niður fyrir allar myndir.

`<br clear="right" />` Ýtir efni niður fyrir myndir á hægri hönd.

`<br clear="left" />` Ýtir efni niður fyrir myndir á vinstri hönd.

Skilyrði fyrir notkun þessarar bókar

Þér er frjálst að afrita, dreifa og sýna þessa bók

Að uppfylltum eftirtöldum skilyrðum:

Tileinkun

Þú verður að tilgreina upphaflegan höfund.

Ekki í hagnaðarskyni

Þú mátt ekki nota bókina til að hagnast sjálfur á. Ríkisrekið skólastarf telst ekki vera í hagnaðarskyni en einstök námskeið teljast vera það.

Engin afleidd verk

Þú mátt ekki breyta, umbreyta eða byggja á þessari bók án samráðs við höfund.

Nota nýjustu útgáfu

Fyrir hverja útprentun eða fjölritun skal athuga hvort komin er nýrri útgáfa og nota hana ef svo er.

Fastar fylgisiður

Eftirtaldar síður fylgi með í útprentun og fjölritun:

Forsíða

Efnisyfirlit

Inngangur

Þessi síða

Ef þú ert ekki viss um hvort þín notkun standist skilyrðin að ofan eða vilt biðja um undanþágu frá skilyrðum, hafðu þá samband við mig, helst í tölvupósti á gunnar@where.is eða í síma 897-8229.

Höfundur, Gunnar Grímsson, gunnar@where.is á fullan höfundarrétt á öllu efni sem hér er. Bókin er gefin út á Netinu á slóðinni <http://where.is/handbok/>

Þetta leyfi er þýtt og staðfært frá:

<http://creativecommons.org/licenses/by-nd-nc/1.0/>