

Introducció als Llenguatges de Marques. HTML

Índex

| | |
|---|----|
| 1. Introducció als llenguatges de marques..... | 2 |
| 1.1. Definició..... | 2 |
| 1.2. Classificació..... | 2 |
| 2. Introducció als estàndards web/continguts..... | 2 |
| 2.1. Per què són necessaris els estàndard web?..... | 2 |
| 3. El model d'estàndards web: HTML, CSS i JavaScript..... | 3 |
| 3.1. Etiquetat, la base de cada pàgina..... | 3 |
| 3.1.1. Què és XHTML?..... | 3 |
| 3.1.2. Validació..... | 4 |
| 3.2. CSS: afegim un poc d'estil..... | 4 |
| 3.2.1. Redefinició d'un element..... | 5 |
| 3.2.2. Definició d'una ID..... | 5 |
| 3.2.3. Definició d'una classe..... | 6 |
| 3.3. JavaScript: afegir comportament a les pàgines web..... | 6 |
| 3.4. Una pàgina d'exemple..... | 6 |
| 4. Conceptes bàsics de HTML..... | 11 |
| 4.1. Què és l'HTML?..... | 11 |
| 4.2. Aspecte de l'HTML..... | 11 |
| 4.3. Estructura d'un document HTML..... | 11 |
| 4.4. La sintaxi dels elements HTML..... | 12 |
| 4.5. Elements de bloc i en línia..... | 13 |
| 4.6. Referències de caràcter..... | 13 |
| 5. L'element <head> del HTML..... | 14 |
| 5.1. Què és <head>?..... | 14 |
| 5.2. Definir l'idioma principal del document..... | 14 |
| 5.3. Títol del document..... | 15 |
| 5.4. Afegir paraules clau i una descripció..... | 16 |
| 5.5. Afegir estils..... | 17 |
| 5.6. Afegir funcions dinàmiques amb JavaScript..... | 19 |
| 5.7. L'ús de CSS integrat i JavaScript al <head> no és la millor idea!..... | 21 |
| 6. Escollir el doctype correcte per als documents HTML..... | 23 |
| 6.1. El doctype..... | 23 |
| 6.2. La declaració XML (només HTML 4.01 o XHTML 1.0)..... | 24 |

1. Introducció als llenguatges de marques

1.1. Definició

Llenguatges de marques són aquells que combinen la informació que conté el document amb marques o anotacions relatives a l'estructura del text o la forma de representar-lo. Cada llenguatge de marques especifica quines etiquetes són possibles, on s'han de colocar i quin significat tindran.

Normalment aquestes marques no són visibles a l'usuari final, que només li interessa el contingut.

Els llenguatges de marques no es solen considerar pròpiament llenguatges de programació perquè no inclouen funcions aritmètiques o variables.

1.2. Classificació

Es solen classificar en 3 grups;

- **Llenguatges orientats a la presentació:** Usats pels processadors de text com per exemple el Word per mostrar el seu contingut.
- **Llenguatges procedimentals:** Utilitza les etiquetes igualment per a la presentació però està integrat dintre d'un marc procedimental que permet definir scripts i subrutines. Exemples: LaTeX i PostScript (impressores)
- **Llenguatges descriptius:** No defineixen únicament com és el document sinó que indiquen què és la informació que acompanyen. La majoria de llenguatges de marques actuals estan dintre d'aquest grup. HTML, XML, JSON, ...

2. Introducció als estàndards web/continguts

2.1. Per què són necessaris els estàndard web?

La utilització d'estàndards web ofereix els següents avantatges:

1. Eficiència del codi:

- **Reutilització** del codi
- **Separar** el **contingut** HTML de la informació d'**estil** (CSS) i de **comportament** (JavaScript).
- Permet **mantenir una mida petita** dels fitxers i **escriure codi només una vegada**, i poderlo reutilitzar més endavant quan sigui necessari.

2. Facilitat de manteniment:

- **Separar** el **contingut** HTML de la informació d'**estil** (CSS) i de **comportament** (JavaScript).

- Si es necessita fer algun canvi, **només cal ferho en un sol lloc**, i s'aplicarà a tot el lloc web.

3. Accessibilitat:

- **Lloc web accessible per a tot el món**, independentment de qui sigui i de les seves circumstàncies.
- Llocs webs utilitzables per **persones amb discapacitats** com ceguera, problemes de visió i problemes motors, sense cap tipus d'esforç addicional.

4. Compatibilitat de dispositius:

- Diferents plataformes: Windows, Mac, Linux, etc.
- Dispositius de navegació alternatius: telèfons mòbils, televisors, consoles de joc, etc.
- Limitacions: mida de la pantalla, potència de processament, mecanismes de control disponibles, etc.
- Utilitzant estàndards web i les millors pràctiques, es garanteix que els llocs web funcionaran en la majoria d'aquests dispositius.
- Hi ha més telèfons mòbils en el món que ordinadors. No es poden deixar de banda aquests clients potencials del lloc web.

5. Motors de recerca:

- "Optimització de motors de recerca": els llocs siguin tan visibles com sigui possible per als cercadors web que rastregen la web i indexen llocs web.
- Objectiu d'oferir millors posicions de cerca en llocs com Google. Això és bo per als negocis.

Alguns llocs web no segueixen els estàndards web, i alguns desenvolupadors realitzen pràctiques incorrectes i obsoletes. Alguns dels motius poden ser:

1. Falta de formació
2. Polítiques d'empresa
3. "No necessito aprendreho"
4. "Costa massa aprendreho"
5. Compatibilitat dels estàndards en els navegadors

2. El model d'estàndards web: HTML, CSS i JavaScript

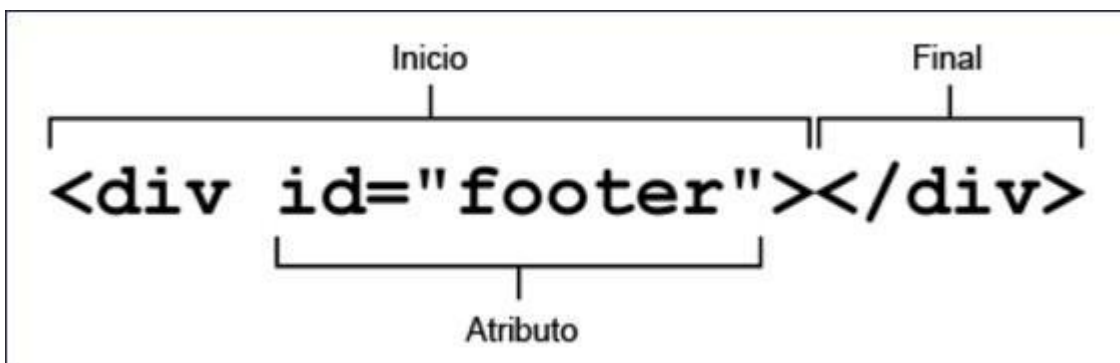
2.1. Etiquetat, la base de cada pàgina

HTML i XHTML són llenguatges d'etiquetat formats per elements que contenen atributs (alguns opcionals i altres obligatoris). Aquests elements s'utilitzen per etiquetar els diferents tipus de continguts en els documents.

Els **elements** defineixen el tipus de contingut actual, mentre que els **atributs** defineix informació addicional sobre aquests elements (com per exemple una ID per identificar l'element o una ubicació perquè senyali un enllaç).

L'etiquetat és el més semàntic possible, ha de descriure la funció del contingut de la manera més acurada possible.

La sintaxi de tot element HTML/XHTML és la següent:



Llavors, quina és la diferència entre HTML i XHTML?

2.1.1. Què és XHTML?

La "X" de XHTML vol dir "extensible", és a dir, ampliable. La principal diferència entre HTML i XHTML radica en l'estructura.

La següent taula mostra les diferències bàsiques entre HTML i XHTML:

| HTML | XHTML |
|--|--|
| <p>Els elements i atributs no distingeixen entre majúscules i minúscules. <code><h1></code> és el mateix que <code><H1></code></p> <p>Alguns elements no necessiten una etiqueta de tancament (per exemple, <code><p></code>).</p> <p>Altres (elements buits) prohibeixen l'etiqueta de tancament, com <code></code>.</p> <p>Alguns valors d'atributs poden estar escrits sense estar entre cometes: <code></code></p> <p>Alguns atributs es poden abreviar: <code><option selected></code></p> <p>Els servidors haurien de servir/enviar l'HTML al client amb un tipus de medi <code>text/html</code>.</p> | <p>Els elements i atributs distingeixen entre majúscules i minúscules. Tots apareixen en minúscules.</p> <p>Tots els elements s'han de tancar: <code><p>A paragraph</p></code>.</p> <p>Els elements sense contingut es poden tancar utilitzant una barra inclinada en l'etiqueta d'obertura: <code><hr></hr></code> <code><hr /></code></p> <p>Els valors dels atributs s'han d'incloure entre cometes: <code></code></p> <p>S'ha d'utilitzar la forma d'atribut sencera per a tots els atributs: <code><option selected="selected"></code></p> <p>El XHTML hauria d'utilitzar el tipus de medi <code>application/xhtml+xml</code>, però es pot utilitzar <code>application/xml</code>, <code>text/xml</code> o <code>text/html</code>.</p> |

2.1.2. Validació

HTML i XHTML són estàndards establerts, així com CSS. El World Wide Web Consortium (W3C) ha creat una eina anomenada **validador**:

- Comprova automàticament les pàgines
- Assenyala qualsevol problema/error que pugui contenir el codi, com ara:
 - Falta d'etiquetes de tancament
 - Falta de cometes als atributs

Validador HTML: <http://validator.w3.org/> (detectarà automàticament si s'utilitza HTML o XHTML, així com el tipus de document que s'utilitzi).

Validador CSS: <http://jigsaw.w3.org/cssvalidator/>.

2.2. CSS: afegim un poc d'estil

Els **fulls d'estil en cascada (CSS)** permeten un control precís sobre el format i la disposició del document.

Es poden canviar o afegir colors, fons, tipus de lletra, mides i estils de tipus de lletra, la posició dels elements de la pàgina web, etc.

Hi han **tres formes** d'aplicar estils amb CSS:

- redefinició d'un element
- aplicació d'un estil a una ID
- aplicació d'un estil a una classe

2.2.1. Redefinició d'un element

Es pot canviar la forma en que qualsevol element de HTML/XHTML es mostra mitjançant la redefinició d'una regla d'estil.

Per exemple, si es desitja que tots els paràgrafs siguin a doble espai i en verd:

Document HTML:

```
<p>Exemple de paràgraf</p>
```

Document CSS:

```
p {  
  line-height: 2;  
  color: green;  
}
```

Ara, tot allò inclòs entre `<p>` i `</p>` al document HTML tindrà una altura de línia doble i serà de color verd.

2.2.2. Definició d'una ID

Es pot donar a un element un atribut `id` per identificarlo de manera única en una pàgina. Cada `id` es pot utilitzar només una vegada per pàgina.

Per exemple, `<ul id="navigation_menu">`.

Això permet tenir un control més precís sobre el format d'una pàgina.

Per exemple, si es desitja que un paràgraf determinat d'una pàgina sigui a doble espai i destacat amb text verd, es pot assignar una ID:

Document HTML

```
<p id="highlight">Contingut del paràgraf</p>
```

Document CSS:

```
#highlight {  
  line-height: 2;  
  color: green;  
}
```

Aquesta regla CSS només s'aplicarà al paràgraf amb un atribut `id` del tipus `highlight`.

`#` és una convenció CSS per indicar que es tracta d'una ID.

2.2.3. Definició d'una classe

Les classes són com les ID, excepte que es poden tenir més d'un element de la mateixa classe en una pàgina.

Per exemple, si es desitja que els dos primers paràgrafs d'una pàgina siguin a doble espai i estiguin destacats en verd, hauríem de definir:

Document HTML:

```
<p class="highlight">Contingut del paràgraf</p>
<p class="highlight">El contingut del segon paràgraf</p>
```

Document CSS:

```
.highlight {
  line-height: 2;
  color: green;
}
```

En aquest cas, `highlight` és una classe, no una ID.

El punt (.) és una convenció de CSS per indicar que es tracta d'una classe.

2.3. JavaScript: afegir comportament a les pàgines web

JavaScript és un llenguatge de *script* que s'utilitza per afegir comportament a les seves pàgines web.

JavaScript es pot utilitzar per:

- Validar dades que s'introdueixen en un formulari (discriminar si estan en el format correcte o no).
- Oferir funcionalitat d'arrastrar i soltar.
- Canviar estils sobre la marxa.
- Animar elements de les pàgines com els menús
- Tractar les funcions amb els botons.
- Etc.

JavaScript funciona localitzant un element HTML de destí, i després fentli alguna cosa, igual que el CSS, però la manera de funcionament, sintaxi, etc. és bastant diferent.

2.4. Una pàgina d'exemple

index.html

```
<!DOCTYPE html>

<html>
  <head>
```

```
<meta charset="utf-8" />
<title>References</title>
<style type="text/css">
    @import
        url("styles.css");
</style>
</head>
<body>
    <div id="bggraphic"></div>
    <div id="header">
        <h1>References</h1>
    </div>
    <div id="references">
        <cite class="article">Adams, J. R. (2008). The Benefits of Valid Markup:
        A Post-Modernistic Approach to Developing Web Sites. <em>The Journal
        of Awesome Web Standards, 15:7,</em> 57-62.</cite>
        <cite class="book">Baker, S. (2006). <em>Validate Your Pages....
        Or Else!.</em> Detroit, MI: Are you out of your mind publishers.</cite>
        <cite class="article">Lane, J. C. (2007). Dude, HTML 4, that's like
        so 2000. <em>The Journal that Publishes Genius, 1:2, </em>
        12-34.</cite>
        <cite class="website">Smith, J. Q. (2005). <em>Web Standards and
        You.</em>
        Retrieved May 3, 2007 from <a
        href="http://www.webstandardsandyou.com/"> Web standards and
        you</a>.</cite>
    </div>
    <div id="footer">
        <p>The content of this page is copyright © 2007
        <a href="mailto:jonathanlane@gmail.com">J. Lane</a></p>
    </div>
```

Tipus de document o doctype

```
<!DOCTYPE html>
```

Declaració de tipus de document o **doctype**. Aquesta és la de HTML5.

El tipus de document especifica una sèrie de regles que ha de seguir l'etiquetat, i gràcies a les quals es pot validar.

Importació d'un arxiu CSS a la pàgina

```
<style type="text/css">
  @import url("styles.css");
</style>
```

Els estils continguts en l'arxiu "styles.css" s'aplicaran als diferents elements de la pàgina. Un poc més endavant es mostra el contingut d'aquest arxiu.

Assignació de classes

S'ha assignat una classe diferent a cadascun dels tipus de referència. Així es podrà aplicar un estil diferent a cada tipus de referència.

```
<cite class="article">Adams, J. R. (2008). The Benefits of Valid Markup:
A Post-Modernistic Approach to Developing Web Sites. <em>The Journal of
Awesome Web Standards, 15:7,</em> 57-62.</cite>
<cite class="book">Baker, S. (2006). <em>Validate Your Pages....
Or Else!.</em> Detroit, MI: Are you out of your mind publishers.</cite>
<cite class="article">Lane, J. C. (2007). Dude, HTML 4, that's like so
2000. <em>The Journal that Publishes Genius, 1:2, </em> 12-34.</cite>
<cite class="website">Smith, J. Q. (2005). <em>Web Standards and
You.</em>
Retrieved May 3, 2007 from <a href="http://www.webstandardsandyou.com/">
Web standards and you</a>.</cite>
```

Assignació de IDs

S'ha assignat una id diferent a cada element <div> (contenedor). Així es podrà aplicar un estil diferent a cada contenidor.

```
<div id="bggraphic"></div>
<div id="header">
  <h1>References</h1>
</div>
<div id="references">
  <cite class="article">Adams, J. R. (2008). The Benefits of Valid Markup:
```

```

A Post-Modernistic Approach to Developing Web Sites. <em>The Journal
of Awesome Web Standards, 15:7,</em> 57-62.</cite>

<cite class="book">Baker, S. (2006). <em>Validate Your Pages....
Or Else!.</em> Detroit, MI: Are you out of your mind publishers.</cite>

<cite class="article">Lane, J. C. (2007). Dude, HTML 4, that's like
so 2000. <em>The Journal that Publishes Genius, 1:2, </em>
12-34.</cite>

<cite class="website">Smith, J. Q. (2005). <em>Web Standards and
You.</em>

Retrieved May 3, 2007 from <a
href="http://www.webstandardsandyou.com/"> Web standards and
you</a>.</cite>
</div>
<div id="footer">
    <p>The content of this page is copyright © 2007
    <a href="mailto:jonathanlane@gmail.com">J. Lane</a></p>
</div>

```

styles.css

```

body{
    background: #fff url('images/gradbg.jpg') top left repeat-x; color:
    #000;
    margin: 0;
    padding:0;
    border: 0;
    font-family: Verdana, Arial, sans-serif; font-size: 1em;
}
div {
    width: 800px;
    margin: 0 auto;
}
#bggraphic {
    background: url('images/pen.png') top left no-repeat;
    height: 278px;
    width: 362px;
    position: absolute;
    left: 50%;
    z-index: 100;
}

```

```
h1 {
  text-align: center;
  text-transform: uppercase;
  font-size: 1.5em;
  margin-bottom: 30px;
  background: url('images/headbg.png') top left repeat; padding:
  10px 0;
}
#references cite {
  margin: 1em 0 0 3em;
  text-indent: -3em;
  display: block;
  font-style: normal;
  padding-right: 3px;
}
.website {
  border-right: 5px solid blue;
}
.book {
  border-right: 5px solid red;
}
.article {
  border-right: 5px solid green;
}
#footer {
  font-size: 0.5em;
  border-top: 1px solid #000;
  margin-top: 20px;
}
#footer a {
  color: #000;
  text-decoration: none;
}
#footer a:hover{
  text-decoration: underline;
}
```

3. Conceptes bàsics de HTML

3.1. Què és l'HTML?

El HTML (HyperText Markup Language) és un llenguatge per a descriure el contingut dels documents de la web. Utilitza una sintaxi especial que conté marcadors ("elements") que envolten al text que hi ha dins del document per indicar als agents d'usuari com han d'interpretar aquesta part del document.

3.2. Aspecte de l'HTML

HTML és només una representació textual del contingut i del seu significat general.

Per exemple, el codi per l'encapçalament "Aspecte de l'HTML" és el següent:

```
<h2 id="htmllooks">Aspecte de l'HTML</h2>
```

- `<h2>` és un **marcador** o **etiqueta**. El text que li segueix es considera com un títol de segon nivell.
- `</h2>` és l'etiqueta de tancament. És una etiqueta que indica on s'acaba el títol de segon nivell.
- L'etiqueta d'obertura `<h2>`, l'etiqueta de tancament `</h2>` i tot el que hi ha entre les dues (`<h2 id="htmllooks">Aspecte de l'HTML</h2>`) s'anomena **element**.
- `id="htmllooks"` és un atribut.

La majoria dels navegadors incorporen l'opció "**Codi font**" o "**Veure el codi font**".

3.3. Estructura d'un document HTML

El document HTML vàlid més petit possible seria alguna cosa similar a:

```
<!DOCTYPE html>
<html lang="ca">
  <head>
    <meta charset="utf-8">
    <title>Pàgina d'exemple</title>
  </head>
  <body>
    <h1>Hola, món!</h1>
  </body>
</html>
```

Explicació:

- `<!DOCTYPE html>`: tipus de document o doctype. Descriu el tipus de HTML que s'utilitza perquè els agents d'usuari (navegadors o altres) puguin determinar com s'interpretarà el document. També és utilitzat pels validadors per saber quines normes aplicaran.
- `<html> ... </html>`: inclou tot el document. `</html>` és sempre la última etiqueta en tot document HTML.
- `<head> ... </head>`: apareix dins de `<html>`. Conté informació sobre el propi document (metadades). Dins del `<head>` apareix sempre `<title>`, que defineix el títol del document que apareix a la vorera superior del navegador.
- `<body> ... </body>`: element que inclou el contingut real del document. En aquest exemple només apareix un element d'encapçalament de nivell 1, `<h1>`, que conté el text "Hola, món!".

Si aprofundim una mica més en l'estructura del document html5, al body podríem trobar les següents etiquetes.



3.4. La sintaxi dels elements HTML

- Un element bàsic d'HTML consisteix en dos marcadors a l'inici i al final d'un bloc de text.
- Hi ha elements que no envolten al text (`
`, `<hr>`, etc.).
- Generalment els elements contenen subelements (html > head, body).

- Els elements poden tenir atributs: modifiquen el comportament de l'element i introdueixen un significat addicional.

Exemple:

```
<div id="masthead">
  <h1>Conceptes bàsics del
    <abbr title="llenguatge de marques d'hipertext">HTML</abbr>
  </h1>
</div>
```

Explicació:

- `<div>`: divisió de pàgina, manera de partir un document en blocs lògics.
- `id="masthead"`: atribut `id` afegit que té el valor de "masthead".
- `<h1>`: element d'encapçalament de primer nivell o més important, que conté text.
- `<abbr>`: s'utilitza per l'expansió de les sigles (HTML). Conté l'atribut `title="llenguatge de marques d'hipertext"`.

3.5. Elements de bloc i en línia

En HTML hi ha **dos grans categories generals d'elements**, que són dos tipus de contenidors i estructures:

- **Elements de bloc**: són elements de nivell superior i normalment defineixen l'estructura del document. Són aquells que comencen en una nova línia i que trenquen amb allò que hi havia abans. Exemples: `<div>`, `<p>`, `<h1>`, `<h2>`, etc.
- **Elements en línia**: són elements que es troben inclosos en els elements estructurals de bloc, i que inclouen només parts petites del contingut del document, i no paràgrafs sencers ni grups de contingut. Exemples: `<a>`, ``, `<abbr>`, una paraula, una cita breu, etc.

3.6. Referències de caràcter

En HTML els caràcters `<`, `>` i `&` són especials, entre d'altres. Aquests caràcters inicialitzen o finalitzen parts del document HTML, i no representen els caràcters "menor que", "major que", "and", etc.

A <http://ascii.cl/es/codigoshtml.htm> trobareu la taula completa dels caràcters especials que es poden utilitzar.

4. L'element <head> del HTML

4.1. Què és <head>?

Estructura HTML

```
<!DOCTYPE html>
<html>
  <head>
    ...
    <title>Títol de la pàgina</title>
    ...
  </head>
  <body>
    ...
  </body>
</html>
```

A banda del <title>, tot allò que s'especifica al <head> no serà visible per als usuaris que visiten la pàgina.

Però important:

- <head> és el lloc on estan la major part d'instruccions per al navegador.
- En <head> és on s'emmagatzema informació addicional (metadades > <meta>).

4.2. Definir l'idioma principal del document

Definició

```
<html lang="en">
  ...
</html>
```

L'idioma es refereix a un idioma humà: francès, tailandès, anglès, etc. És important definir aquest atribut, i més si el públic de la pàgina ha de ser internacional.

Això és molt important per als:

- Lectors de pantalla.
- Motors de recerca.

També es pot definir l'idioma només d'una secció de la pàgina:

```
<span lang="fr">Bonjour</span>
```

Codis de 2 lletres: https://en.wikipedia.org/wiki/ISO_6391

Alguns exemples més sobre l'idioma: <https://www.w3.org/International/articles/language-tags/>

4.3. Títol del document

Consideracions a tenir per al títol del document HTML:

- El títol apareix en la majoria dels agents d'usuari o navegadors dins la barra de títol de l'aplicació.
- És el primer que veuen els usuaris de la web quan el visiten, i per tant, és molt important.
- Els lectors de pantalla també el llegeixen, per donar una primera idea al visitant.
- Molts motors de recerca funcionen d'una manera similar.
- Cal escollir un bon títol, llegible per a les persones i que contingui les paraules clau adequades.

Exemple:

```
<!DOCTYPE html>
<html>
  <head>
    <title>I am a title example</title>
  </head>
  <body>
  </body>
</html>
```

Al obrir-ho al navegador, apareix el títol a la barra de títol de l'aplicació:



4.4. Afegir paraules clau i una descripció

Cal afegir una descripció de la pàgina, i una llista de paraules clau dins de l'element <head>, dins d'elements <meta>.

Exemple:

```
<!DOCTYPE html>
<html>
<head>
  <title>Yahoo! UK & Ireland Eurosport-Sports News | Live Scores | Sport</title>
  <meta name="description" content="Latest sports news and live scores
    from Yahoo! Eurosport UK. Complete sport coverage with Football results,
    Cricket scores, F1, Golf, Rugby, Tennis and more.">
  <meta name="keywords" content="eurosport,sports,sport,sports news, live
    scores,football,cricket,fl,golf,rugby,tennis,uk,yahoo">
</head>
<body>
</body>
</html>
```

Els motors de recerca indexen els documents HTML de la xarxa Internet gràcies a aquests elements:



Alguns navegadors mostren la descripció com informació addicional quan s'afegeix el document a la llista d'adreces d'interès:



4.5. Afegir estils

Un altre element que es pot afegir a <head> són les normes per als estils, que es defineixen en fulls d'estil en cascada (CSS). Es poden integrar directament en <head> utilitzant un element <style>:

```
<!DOCTYPE html>
<html>
<head>
  <title>Breeding Dogs-Tips about Alsatians</title>
  <meta name="description" content="How to breed Alsatians, tips on proper
    breeding and information about common issues with this breed.">
  <meta name="keywords" content="Dogs,Alsatian,Breeding,Dog,Tips,Free,Pet">
  <style type="text/css">
    body{
      background:#000;
      color:#ccc;
      font-family: helvetica, arial, sans-serif;
    }
  </style>
</head>
<body>
<p>Test!</p>
</body>
</html>
```

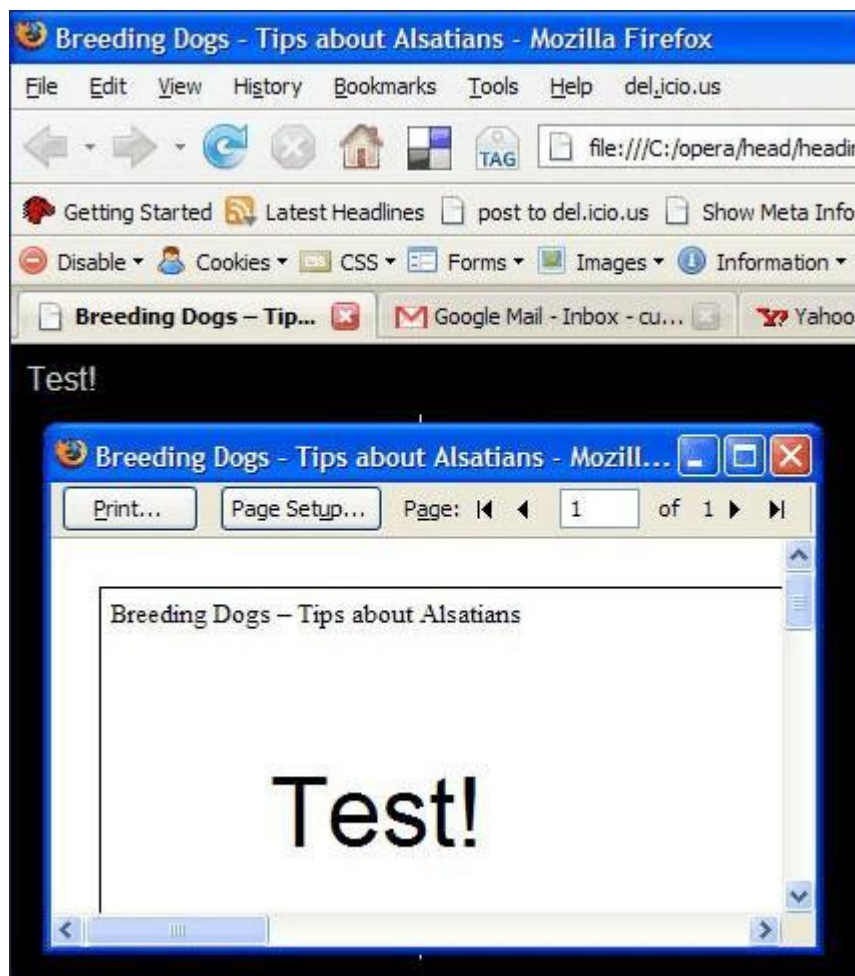
<style> pot contenir un atribut anomenat media, que defineix quin tipus de suport utilitzarà aquests estils:

- <style media="screen">: per a la visualització per pantalla.
- <style media="print">: aspecte que tindrà el document imprès.
- <style media="handheld">: en dispositius mòbils i altres dispositius portàtils.
- <style media="projection">: per presentacions en HTML (Opera Show).

Exemple:

```
<!DOCTYPE html>
<html>
<head>
  <title>Breeding Dogs-Tips about Alsatians</title>
  <meta name="description" content="How to breed Alsatians, tips on proper
    breeding and information about common issues with this breed.">
  <meta name="keywords" content="Dogs,Alsatian,Breeding,Dog,Tips,Free,Pet">
  <style type="text/css" media="screen">
    body{
      background:#000;
      color:#ccc;
      font-family: helvetica, arial, sans-serif;
    }
  </style>
  <style type="text/css" media="print">
    body{
      background:#fff;
      color:#000;
      font-family: helvetica, arial, sans-serif;
      font-size:300%;
    }
  </style>
</head>
<body>
<p>Test!</p>
</body>
</html>
```

Visualització per pantalla o a la impressora:



4.6. Afegir funcions dinàmiques amb JavaScript

Un altre element que es pot afegir a `<head>` són scripts que executarà el navegador (scripts de client) escrits en JavaScript.

JavaScript afegeix un comportament dinàmic a l'HTML estàtic, com per exemple, efectes d'animació, validació de dades de formularis o altres coses que passen quan l'usuari realitza determinades accions.

Per afegir JavaScript a un document HTML, cal afegir un element `<script>` dins de `<head>`.

Exemple:

```
<!DOCTYPE html>
<html>
<head>
  <title>Breeding Dogs-Tips about Alsatians</title>
  <meta name="description" content="How to breed Alsatians, tips on proper
    breeding and information about common issues with this breed.">
  <meta name="keywords" content="Dogs,Alsatian,Breeding,Dog,Tips,Free,Pet">
  <style type="text/css" media="screen">
    body{
      background:#000;
      color:#ccc;
      font-family: helvetica, arial, sans-serif;
    }
    a {color:#fff}
  </style>
  <style type="text/css" media="print">
    body{
      background:#fff;
      color:#000;
      font-family: helvetica, arial, sans-serif;
      font-size:300%;
    }
  </style>
  <script>
    function leave(){
      return confirm("This will take you to another site,\n are you sure you
        want to go?")
    }
  </script>
</head>
<body>
Test!
<a href="http://dailypuppy.com" onclick="return leave()">The Daily Puppy</a>
</body>
</html>
```

4.7. L'ús de CSS integrat i JavaScript al <head> no és la millor idea!

Millor arxius separats per HTML, CSS i JavaScript. Com els unim?

Exemple d'HTML amb JavaScript

Dins del <head>:

```
<script src="leaving.js"></script>
```

src: ruta de l'arxiu on hi ha el codi JavaScript.

```
<!DOCTYPE html>
<html>
<head>
  <title>Breeding Dogs-Tips about Alsatians</title>
  <meta name="description" content="How to breed Alsatians, tips on proper
    breeding and information about common issues with this breed.">
  <meta name="keywords" content="Dogs,Alsatian,Breeding,Dog,Tips,Free,Pet">
  <style type="text/css" media="screen">
    body{
      background:#000;
      color:#ccc;
      font-family: helvetica, arial, sans-serif;
    }
    a {color:#fff}
  </style>
  <style type="text/css" media="print">
    body{
      background:#fff;
      color:#000;
      font-family: helvetica, arial, sans-serif;
      font-size:300%;
    }
  </style>
  <script src="leaving.js"></script>
</head>
<body>
```

```
Test!  
<a href="http://dailypuppy.com" onclick="return leave()">The Daily Puppy</a>  
</body>  
</html>
```

Exemple d'HTML amb CSS

Dins del <head>:

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

- **rel:** tipus de full, en aquest cas "stylesheet" (full d'estils), però pot tenir altres [valors](#).
- **type:** tipus de document, sempre "text/css" en els CSS.
- **media:** pantalla, impressió, etc.
- **href:** fitxer CSS extern per a importarlo.

```
<!DOCTYPE html>  
<html>  
<head>  
  <title>Breeding Dogs-Tips about Alsatians</title>  
  <meta name="description" content="How to breed Alsatians, tips on proper  
    breeding and information about common issues with this breed.">  
  <meta name="keywords" content="Dogs,Alsatian,Breeding,Dog,Tips,Free,Pet">  
  <link rel="stylesheet" type="text/css" media="screen"  
href="screenstyles.css">  
  <link rel="stylesheet" type="text/css" media="print" href="printstyles.css">  
  <script src="leaving.js"></script>  
</head>  
<body>  
Test!  
<a href="http://dailypuppy.com" onclick="return leave()">The Daily Puppy</a>  
</body>  
</html>
```

5. Escollir el doctype correcte per als documents HTML

5.1. El doctype

El primer que hi ha en tot document HTML/XHTML és la declaració DTD o "doctype" (Document Type Definition). Defineix els **elements i atributs que es podran utilitzar** en una versió concreta de l'HTML/XHTML segons el que marca l'estàndard.

S'utilitza amb les següents finalitats:

- Els navegadors web utilitzen el doctype per determinar la manera de representació que hauran d'utilitzar.
- Els validadors d'etiquetat miren el doctype per determinar les normes (estàndard) amb les que han de comprovar el document.

Exemple:

```
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01//EN" "http://www.w3.org/TR/html4/strict.dtd">
```

Les parts més importants són:

- "//W3C//DTD HTML 4.01//EN": indica que és un document DTD publicat pel [W3C](http://www.w3.org/), que la DTD descriu la versió 4.01 de l'HTML, i que l'idioma utilitzat en la DTD és l'anglès.
- "http://www.w3.org/TR/html4/strict.dtd": indica la URL del document DTD utilitzat per a aquest doctype.

Es recomana utilitzar el doctype de HTML5:

```
<!DOCTYPE html>
```

Si per algun motiu el document ha d'utilitzar-se en navegadors antics o incompatibles amb HTML5 (sol ser habitual amb usuaris que encara utilitzen versions antigues de Internet Explorer) cal utilitzar el doctype HTML 4.01:

```
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01//EN" "http://www.w3.org/TR/html4/strict.dtd">
```

Si el document és XHTML 1.0:

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


5.2. La declaració XML (només HTML 4.01 o XHTML 1.0)

És possible que en alguns documents XHTML hi hagi una línia abans del DOCTYPE:

```
<?xml version="1.0" encoding="UTF-8"?>
```

És el que s'anomena declaració XML, y ha d'aparèixer abans del DOCTYPE, però no hi apareix sempre, només en alguns documents XHTML.