

În această lecție, ne vom ocupa de tipografia pe care o aplicăm în documentele HTML.

Bazele tipografiei

Tipografia este tehnica și arta prezentării textului într-un mod inteligibil și vizibil.

Este clar că tipografia a apărut odată cu primele cuvinte scrise în istorie, dar că s-a schimbat și a progresat, a evoluat de-a lungul anilor și a secolelor. În ultimii ani, tipografia web a fost separată de întreaga știință, de care ne vom ocupa, cel puțin o dată.

Pentru cei care doresc să afle mai multe despre tipografia web clasică, le recomandăm următoarea adresă: <http://webtypography.net/>, precum și acest articol interesant din revista „A List Apart”: <http://www.alistapart.com/articles/on-web-typography/>.

La fel ca în cazul celorlalte elemente de pe pagină, pagina HTML setează conținutul, iar regulile CSS influențează toate aspectele tipografiei, ceea ce poate avea un efect semnificativ asupra paginii. Dacă textul este scris corect, este lizibil și vizibil, totul va fi în regulă. Pe de altă parte, un text ilizibil va descuraja cu siguranță vizitatorii. Nu uitați, chiar dacă toate celelalte elemente sunt excelente, respectiv designul sau imaginile, că vizitatorul nu a venit pe site pentru ele, ci pentru conținut (text, eventual imagini etc.). Tipografia este deosebit de importantă pentru site-urile web moderne, în special, care au depășit perioada elementelor scriitoare, a animațiilor agresive etc. și sunt acum deschise la noi posibilități, care erau inaccesibile înainte.

Fonturile (familiile de fonturi)

Este logic să începem prin a alege fontul pe care îl vom folosi. Fonturile se împart, de obicei, în câteva categorii de bază.

- **Serif** - Aceste fonturi au anumite detalii la capetele liniilor. Detaliile se numesc *serife*, de unde și numele acestui grup de fonturi. În tipografia clasică, fonturile cu serif sunt considerate mai lizibile în comparație cu alte fonturi, în paragrafe și texte. Cele mai frecvent întâlnite fonturi cu serife de pe web sunt Times New Roman, Garamond și Times etc.
- **Sans Serif** - Cuvântul sans provine din limba franceză și înseamnă „fără” și deja putem să presupunem că aceste fonturi sunt caracterizate de lipsa detaliilor serife. Aceste fonturi sunt caracterizate de linii drepte, curate și solide, fără prea multe detalii. Au un aspect mai curat și mai modern. Cele mai frecvent întâlnite fonturi Sans-Serif de pe web sunt Arial, Tahoma, Verdana, Helvetica etc.
- **Monospace** - Fiecare caracter din aceste fonturi ocupă același spațiu pe orizontală și are aceeași lățime. Au fost utilizate prima dată împreună cu apariția mașinilor de scris, când, din cauza limitărilor tehnologice, astfel de fonturi erau necesare. Mai târziu, au fost utilizate pe primele calculatoare și terminale și se folosesc încă în programare. Cel mai comun font monospace de pe web este Courier New (sau Courier).

serif

HTML & CSS

sans serif

HTML & CSS

monospace

HTML & CSS

Imaginea 15.1 - Cele mai des întâlnite categorii de fonturi

Pe lângă categoriile menționate, există multe altele, cum ar fi fonturile **Script** (care imită scrisul de mână), precum și subcategorii ale fonturilor deja menționate, precum fonturile **Slab-Serif** (caracterizate de serife dreptunghiulare). Noi ne vom opri asupra categoriilor menționate mai devreme, deoarece sunt cele mai frecvent întâlnite pe web.

Notă:

În vorbire, folosim adesea cuvântul font în contexte greșite. Fontul este un termen care se referă la stilul și dimensiunea precisă a unei familii de fonturi. De exemplu, **Verdana, 11 pt, bold** este un **font**, în timp ce **Verdana** în sine este o **familie de fonturi** (*font family*). Totuși, în continuarea textului vom folosi termenul font, deoarece este general-acceptat.

Terminologie

Pentru a înțelege mai bine despre ce este vorba, vom trece în revistă câțiva dintre termenii de bază, preluați din tipografia clasifică.

- **Linia de bază (*baseline*)** este o linie imaginară, pe care sunt așezate literele dintr-un text.
- **Linia mediană (*median*)** este o linie imaginară, care determină înălțimea literelor mici, a minusculelor.
- **Înălțimea fontului (*x-height*)** este spațiul dintre linia de bază și mediană. În mod tradițional, depinde de litera mică x din alfabetul latin.
- **Ascendența (*Ascender*)** este acea parte a literei, care se ridică deasupra medianei.
- **Descendența (*Descender*)** este acea parte a literei, care coboară sub linia de bază.



Imaginea 15.2 - Detaliile de bază ale tipografiei

Lizibilitatea depinde de alegerea fontului. Este important ca literele, adică anumite caractere, să fie lizibile și separate vizual. Toate acestea sunt aspecte care depind de font și de caracteristicile sale. De aceea, avem opțiunea de a alege corect familia de fonturi și de a ordona logic pagina, ceea ce conferă textului și fluxului său **vizibilitate** și o înțelegere ușoară.

Lungimea rândului influențează, de asemenea, vizibilitatea, iar opinia generală este că un singur rând de text ar trebui să aibă între 47 și 75 de caractere, deși aceasta nu este o regulă și există excepții. Unele site-uri și portaluri, care sunt alcătuite în principal din text, au o lungime a rândurilor de 100 de caractere, ceea ce este acceptabil. În cazul în care textul este împărțit pe coloane, ar trebui să se aleagă un număr mai mic de caractere. Dacă se folosește alinierea *justify*, rândurile mai lungi se comportă mai bine în browser-e.

Înălțimea rândului sau **Leading (Line height)** marchează spațiul dintre două rânduri de bază ale textului. Valoarea implicită în browser-ele web este de 1.2. Se recomandă un minimum de 1.2 em sau mai mare. Exemplu: dacă dimensiunea fontului este de 10 px, o valoare de 1.4 em pentru înălțimea rândului ne dă 14 px.

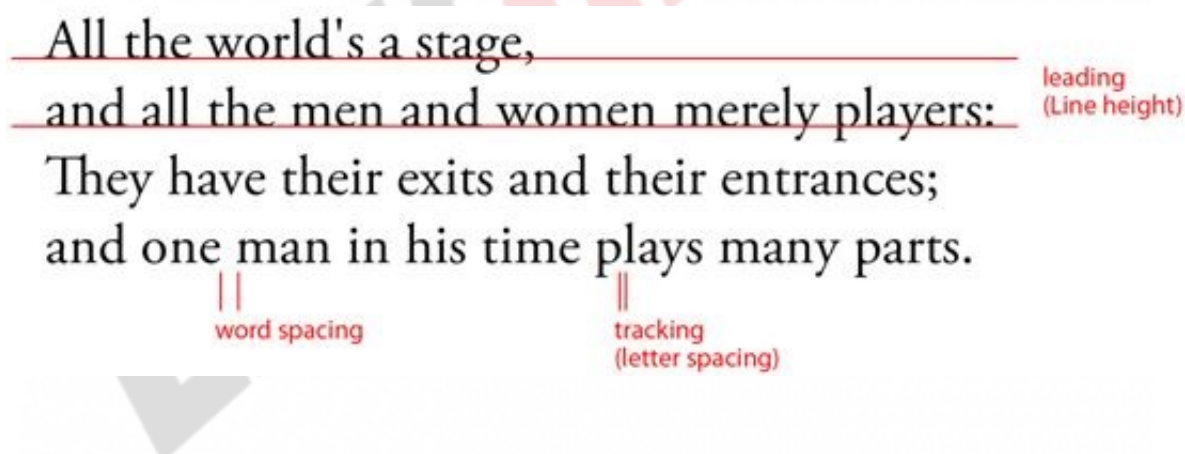
Tracking (letter spacing) este spațiul dintre literele cuvintelor din text. Poate avea și o valoare negativă. Ar trebui să acordați atenție anumitor perechi de litere, precum ft, rn, vv, deoarece pot părea unite

și creează confuzie, făcând textul ilizibil.

În tipografie, se folosește și termenul **kerning**, care, de asemenea, definește spațiul dintre caracterele individuale din text. Deseori, se confundă tracking și kerning, deși nu reprezintă același lucru.

Tracking este spațiul dintre caractere (litere), dar la nivelul întregului text (al paragrafului, propoziției), în timp ce *kerning* este spațiul între două caractere (litere) concrete în text.

Word spacing este spațiul dintre cuvintele din text.



Imaginea 15.3 - Detaliile de bază ale tipografiei

Alegerea fontului

Deși poate părea ciudat, **pe web nu putem folosi orice font**. În alte medii, de exemplu, când creăm un text în Word sau când realizăm un poster în Adobe Illustrator, avem la dispoziție toate fonturile care există. În ceea ce privește paginile web, situația este diferită, deoarece fontul depinde de calculatorul utilizatorului. Mai exact, **nu încorporăm fontul în pagină, ci doar informația cu privire la ce font am**

utilizat.

De exemplu, dacă introducem textul și specificăm prin CSS să se folosească fontul Verdana, browser-ul utilizatorului va primi textul și informația despre font, odată cu celelalte informații și stiluri. Atunci, browser-ul afișează textul, dar utilizând fontul încărcat din calculatorul utilizatorului. Dacă utilizatorul nu are instalat acest font (în exemplul nostru, Verdana), textul va fi afișat cu fontul implicit, care la sistemele Windows este Times New Roman. Bineînțeles, acesta nu este rezultatul dorit și reprezintă o limitare. În plus, utilizatorii folosesc sisteme diverse, precum Windows, Mac, Linux, Android, iOS, Symbian etc., care au instalate fonturi diferite. De exemplu, dacă textului de pe pagină i-am fi aplicat fontul Helvetica, browser-ele de pe sistemele Mac l-ar fi reprezentat în mod normal (deoarece acesta este unul dintre fonturile Mac implicite), în timp ce în Windows sau Linux ar fi apărut o problemă pentru toți cei care nu și-au instalat singuri Helvetica.

Deci, ce este de făcut dacă intervine o astfel de problemă? Avem câteva fonturi comune pentru majoritatea sistemelor, pe care le putem folosi fără probleme, deși nu sunt complet identice pe calculatoarele cu Windows și Mac, așadar trebuie să le definim de două ori/dual, după cum vom vedea mai târziu.

Fonturile sigure sunt (pe Windows/Mac): Arial/Helvetica, Times New Roman/Times, Tahoma (Verdana)/Geneva, Courier New/Courier, Georgia, Comic Sans MS. Folosindu-le pe acestea, puteți fi siguri că textul va fi afișat cu fontul ales.

Numărul fonturilor este restrâns, lucru care poate fi frustrant, dar, dacă analizați site-urile de pe web, veți vedea că peste 90% dintre ele sunt concepute utilizând unul dintre aceste fonturi standard. Înainte, anumiți autori foloseau tehnici specifice care permiteau utilizarea altor fonturi, cum ar fi sIFR sau Cufón. Totuși, utilizarea lor nu este recomandată, deoarece se bazează pe JavaScript, Flash și alte limbaje, ceea ce încetinește încărcarea. Putem spune că aceste tehnici au fost rudimentare, dar au pus bazele pentru ce a urmat.

Recent, odată cu apariția opțiunii @font-face, lucrurile au început să se

schimbe și acum, practic, putem folosi orice font pe paginile web (mai multe despre aceasta la sfârșitul lecției). Această opțiune se folosește foarte mult de ceva timp, dar din cauza lipsei unui standard (fiecare browser are propriile reguli) și a suportului în browser-ele mai vechi (Internet Explorer), trebuie folosită cu precauție.

Valorile absolute și relative ale fontului

În tipografia web, putem folosi valori relative sau absolute, pentru a defini dimensiunea caracterelor sau a întregului text. Toate aceste unități se folosesc pentru tipografie, dar unele se pot aplica și altor elemente, precum marginile, dimensiunile lățimii sau înălțimii etc.

Valorile absolute sunt determinate cu exactitate și nu depind de alte elemente sau valori.

Valorile relative depind de valoarea moștenită sau de valoarea implicită (în absența celei precedente), după cum puteți vedea în exemple.

Unele dintre unitățile disponibile sunt:

- Unități absolute:
 - in – inch (2.54 mm),
 - pt – point, punct tipografic (1 point este 1/72 inchi),
 - pc – pica (1 pica este 12 puncte),
 - cm – centimetru,
 - mm – milimetru,
 - px – pixel (monocromatic, punct indivizibil de pe ecran/monitor).

- *Unități relative:*

- em – ems,
- % – procent,
- ex – x-height.

Unitățile absolute preluate din presa scrisă, precum **in**, **pt**, **pc**, **cm** și **mm**, ar trebui **evitate** complet pe web, cu excepția cazului în care pregătim o regulă CSS specială pentru tipărire.

Cu ajutorul **pixelilor (px)**, determinăm dimensiunea exactă și absolută. Dacă setăm valoarea dimensiunii la 12 px, aceasta va fi întotdeauna 12 px. Dimensiunea reală depinde însă de rezoluția monitorului etc. Alături de valoarea em, pixelul este poate cea mai frecvent utilizată unitate.

Unitatea procentuală **em** a obținut această denumire deoarece în trecut, în zilele imprimării analoage, depindea de majuscula M în font, deoarece aceasta crea forma dreptunghiulară. Astăzi, în fonturile digitale moderne, **valoarea em este valoarea procentuală a fontului utilizat**. Cu alte cuvinte, dimensiunea se moștenește de la elementul părinte. Practic, aceasta înseamnă să înmulțim dimensiunea rezultată cu valoarea em.

Cel mai simplu este să explicăm printr-un exemplu. În mod implicit, browser-ele moderne atribuie dimensiunea de 16 px unui text normal în elementul <body> (pe întreaga pagină). Dacă setăm dimensiunea la 1 em, dimensiunea ar fi 16 px. Dacă setăm dimensiunea la 2 em, aceasta ar fi de **16px*2em=32px**. Haideți să analizăm încă un exemplu. Valoarea implicită este de 16 px, după cum am spus. Vom seta un div și textului său îi vom atribui dimensiunea de 3 em. În cadrul său, vom scrie încă un div cu dimensiunea de 0,5 em. Vom obține **16px*3em*0.5em=24px**. După cum putem concluziona, valoarea em depinde în mod direct de elementul parent, de elementul de lângă el.



Imaginea 15.4 - Raportul dintre dimensiuni

Unitatea **Ex** este relativă și depinde de înălțimea caracterului **x** în fontul selectat; se comportă la fel ca em, dar ar trebui evitată.

Manipularea tipografiei prin CSS

Alegerea familiei de fonturi

Alegerea fontului pentru afișarea textului într-un element se face prin setarea proprietății *font-family*, urmată de valoarea referitoare la unul sau mai multe fonturi (familii). Să vedem următorul exemplu:

```
body {  
  font-family:Georgia;  
}
```

În codul de mai sus, am definit ca întreaga pagină (body element) să fie afișată cu fontul *Georgia*. Dar, dacă nu suntem siguri că fontul menționat este instalat pe calculatorul utilizatorului, putem adăuga mai multe valori, separate prin virgulă, pentru a ajuta browser-ul; acesta va încerca prima dată să afișeze textul cu fontul menționat, iar dacă nu reușește, va încerca să facă asta cu al doilea font etc. Haideți să vedem următorul exemplu:

```
h1 {
```

```
font-family:Arial, Helvetica, sans-serif;  
}
```

După cum putem vedea din exemplul de mai sus, browser-ul va încerca să afișeze textul cu fontul Arial, iar dacă nu îl găsește pe calculator, va încerca cu Helvetica. Dacă nici acest font nu este accesibil, va alege orice font sans-serif, deoarece am introdus denumirea generică (există și *serif*).

Dacă numele fontului este alcătuit din mai multe cuvinte, scriem valoarea între ghilimele:

```
body {  
  font-family: "Times New Roman", Times, Georgia, serif;  
}
```

Alegerea dimensiunii textului (*font-size*)

Proprietatea `font-size` ne permite să alegem dimensiunea textului, despre care am mai vorbit. Vă recomandăm să folosiți valorile `px`, `em` sau `%`.

```
body {  
  font-family: Arial, Helvetica, sans-serif;  
  font-size: 12px;  
}
```

```
h1 {  
  font-size: 200%;  
}
```

```
h2 {  
  font-size: 1.3em;  
}
```

În mod implicit, textul de pe paginile web are dimensiunea de 16 px (cu excepția titlurilor - tag-urile h). Majoritatea autorilor optează pentru una dintre aceste variante. Putem seta fontul principal la 12 px, dar atunci trebuie să mărim titlurile în mod proporțional. Sau putem lăsa textul principal la 16 px, ceea ce poate părea prea mare la prima vedere, dar, dacă ne obișnuim astfel, revenim cu greu la valori mai mici.

PIXELS		PERCENTAGES		EMS
TWELVE PIXEL SCALE				
h1 24px	=	h1 200%	=	h1 1.5em
h2 18px		h2 150%		h2 1.3em
h3 14px		h3 117%		h3 1.17em
body 12px		body 75%		body 100%
				p 0.75em
SIXTEEN PIXEL SCALE				
h1 32px	=	h1 200%	=	h1 2em
h2 24px		h2 150%		h2 1.5em
h3 18px		h3 133%		h3 1.125em
body 16px		body 100%		body 100%
				p 1em

Imaginea 15.5 - Exemplu de dimensiuni de fonturi posibile

Greutatea fontului (*font-weight*)

Cu ajutorul proprietății *font-weight*, stabilim dacă fontul va fi afișat formatat cu bold sau nu. Valorile disponibile sunt *bold* și *normal*.

```
.boldTekst {  
  font-weight:bold;  
}
```

Poate vă întrebați de ce avem nevoie de valoarea *normal*. Putem seta valoarea **bold** unui element care are în cadrul său un alt element. Acest al doilea element va moșteni proprietatea **bold**, apoi putem adăuga *font-weight:normal*. În plus, titlurile (tag-urile *h*) au, în mod implicit, valoarea **bold**. Poate vrem să fie normale.

Stilul fontului (*font-style*)

Dacă vrem să adăugăm proprietatea *italic* unui font, folosim proprietatea *font-style*. Poate avea valorile *italic*, *oblique* sau *normal*. *Italic* reprezintă versiunea italică încorporată în font, în timp ce *oblique* este doar o versiune puțin înclinată a fontului normal. Totuși, practic nu sunt deosebiri în ceea ce privește dimensiunea fontului. Situația cu valoarea *normal* este aceeași ca și la *font-weight* menționat mai devreme.

```
.citat {  
  font-style:italic;  
}
```

Dacă ați folosit sau folosiți Microsoft Word, atunci probabil că v-ați obișnuit să priviți **bold** și *italic* ca elemente conexe, dar, în tipografia web, acestea sunt complet independente unul de altul.

Majuscule și minuscule (*text-transform*)

Dacă vrem să setăm majusculele și minusculele în afișarea noastră, putem folosi *text-transform* și una dintre cele trei valori ale sale: *uppercase*, *lowercase* și *capitalize*. Uppercase face toate literele majuscule, lowercase, invers, în timp ce *capitalize* face fiecare literă inițială a cuvântului majusculă.

```
h1 {  
  text-transform: uppercase;  
}
```

```
h2 {  
  text-transform: lowercase;  
}
```

```
.potpis {  
  text-transform: capitalize;  
}
```

Trebuie să menționăm că în documentul HTML textul rămâne neschimbat, schimbarea apărând doar în previzualizarea pentru utilizator. Această opțiune este utilă, de exemplu, atunci când nu știți sigur ce text va furniza baza de date, dar vreți să fiți siguri cu privire la previzualizare.

Text-decoration

Proprietatea *text-decoration* se folosește pentru a adăuga o linie sub text (valoarea *underline*), deasupra textului (valoarea *overline*), prin text (valoarea *line-through*) sau pentru a face textul să clipească (valoarea *blink*). Toate valorile sunt utile, cu excepția celei din urmă, pe care vă recomandăm să o evitați complet.

```
h1 {  
  text-decoration: underline;  
}
```

Acest exemplu pune o linie sub titlu.

Această proprietate se utilizează des la stilizarea linkurilor. De fapt, linkurile au deja, în mod implicit, o linie sub text, iar dacă vrem, putem să o înlăturăm cu această opțiune.

Priviți exemplul:

```
a, a:link {  
  color: #e31b23;  
  text-decoration:none;  
}  
  
a:visited  
{  
  color: #e31b23;  
  text-decoration:none;  
}  
  
a:hover  
{  
  color: #555555;  
  text-decoration:underline;  
}
```

Efectul este următorul: linkurile normale și vizitate deseori nu vor fi subliniate, iar linia apare doar atunci când ținem mouse-ul peste link (starea hover). Aici, selectorii au pseudo-clase, despre care vom discuta în detaliu mai târziu.

Leading (line-height)

Am menționat înălțimea rândurilor în partea introductivă a acestei lecții. Pe scurt, dacă mărim proprietatea *line-height*, crește și spațiul

dintre rânduri și invers.

```
p {  
  line-height: 1.4em;  
}
```

În acest exemplu, valoarea este dată în unități em, care sunt și recomandate, deoarece depinde de dimensiunea textului și nu este fixă ca atunci când folosim pixeli.

Text-align

Alinierea pe orizontală a textului poate fi setată cu ajutorul acestei proprietăți. Valorile disponibile pentru *text-align* sunt left, center, right și justify. Cu toții cunoaștem aceste opțiuni din Word și din alte instrumente asemănătoare. Left și right aliniază textul spre stânga, respectiv spre dreapta. Center centrează textul, în timp ce justify îl aliniază de-a lungul marginilor din stânga și dreapta, influențând în același timp și spațierea. Exemplu:

```
#footer {  
  text-align: center;  
}
```

Această proprietate nu influențează doar textul, ci și toate elementele inline.

Vertical-align

Această proprietate poate fi confuză și înțeleasă greșit. Nu este destinată alinierii pe verticală a elementelor block, ci pentru alinierea reciprocă a elementelor inline pe un singur rând, la fel ca în cazul imaginilor și atributului lor HTML valign, care s-a folosit până la apariția lui HTML5 (vezi lecția 7).

```
.textGore {  
  vertical-align: text-top;  
}
```

```
.textBaseline {  
  vertical-align: baseline;  
}
```

Dacă am avea o imagine și un text, browser-ul ar afișa următoarele:



Imaginea 15.6 - Exemple de două valori vertical-align

Valorile disponibile sunt *baseline*, *sub*, *super*, *top*, *text-top*, *middle*, *bottom*, *text-bottom*.

Text-indent

Această proprietate se folosește pentru a introduce textul pe orizontală. În general, se setează în unitățile px sau em.

```
.citat {  
  font-style: italic;  
  text-indent: 40px;  
}
```

Interesant este că valoarea poate fi negativă. Dacă, de exemplu, setăm o valoare de “-9999px”, textul va dispărea din browser, deoarece s-a

mutat cu 9999 pixeli în stânga.

Tracking (letter-spacing)

Această proprietate definește spațiul dintre literele individuale ale cuvintelor din text. Recomandarea este să folosim valoarea em. Creșterea și reducerea acesteia afectează lizibilitatea.

```
p {  
  letter-spacing: 0.4em;  
}
```

Spațiul dintre cuvinte (*word-spacing*)

Această proprietate este identică cu cea precedentă, cu excepția faptului că setează spațiul dintre cuvinte.

```
p {  
  letter-spacing: 0.4em;  
  word-spacing: 0.3em;  
}
```

Text-shadow

După cum putem concluziona și din denumirea sa, proprietatea text-shadow setează umbra pe care textul o aruncă pe elementul aflat sub el. De exemplu, în cod inserăm următoarele:

```
text-shadow: 2px 1px 3px #000000;
```

În proprietate, sunt necesare mai multe valori. În primul rând, introducem poziția orizontală (în exemplu, aceasta este de 2px), apoi cea verticală (în exemplu, 1px), estomparea (3px) și, la sfârșit, adăugăm culoarea.

Font face

În ultimul timp, odată cu folosirea funcționalității @font-face a lui CSS3, putem încorpora un anumit font în paginile noastre. Este important să pregătim fișierele de fonturi într-un anumit mod (în mai multe formate diferite) și să le inițializăm corect prin CSS.

Totuși, având în vedere că sintaxa este relativ complicată, au fost create diferite instrumente care să ne ajute în acest sens. De exemplu, site-ul <http://www.fontsquirrel.com/> poate pregăti fișierele și chiar să creeze codul CSS pentru noi. Tot ceea ce trebuie să facem este să-l încorporăm. Serviciul Google fonts este și mai ușor de utilizat. Este suficient să copiem un rând de cod care este generat pentru noi și în continuare putem folosi fontul dorit în mod regulat. Mult mai multe detalii despre încorporarea fonturilor le puteți găsi în cursul „HTML și CSS avansat”.