

Webbprogrammering

Föreläsning 2

Repetition

- Internet
- Hypertext
- Server och klient
- HTTP
- HTML

Internet

- Många sammankopplade nät
- Gemensamma adresser
 - T ex 10.10.0.1
- Gemensamma protokoll
 - Protokoll = regler och algoritmer för kommunikation
 - IP-protokollet
 - TCP-protokollet
 - UDP-protokollet

Hypertext

- Text försedd med länkar till andra texter
- Filer försedda med länkar till andra filer

Server och klient

- En server
 - Har en fast adress på internet
 - Är alltid på (i bästa fall...)
 - Kör ett serverprogram (Apache, IIS, ...)
- En klient
 - Ändrar adress
 - Är inte alltid på
 - Kör ett klientprogram, en webbläsare

HTTP

- HyperText Transmisson Protocol
 - Regler för hur en webbserver och en webbläsare kommunicerar
- Kommandon från webbläsare till webbserver
 - GET är viktigast
- Webbservern svarar på kommandon
 - Om allt går väl får webbläsaren en fil som kan visas
 - Det är ofta en html-fil
 - Kan vara en bild, vanlig textfil eller något annat

HTML

- HyperText Markup Language
 - Anger hur en webbsida ser ut
 - Innehåller taggar
 - <html> ... </html> börjar och avslutar filen
 - <head> ... </head> headern innehåller information
 - <body> ... </body> här finns själva innehållet
 - <p> ... </p> omsluter ett stycke (paragraph)
 - ... omsluter en punktlista
 - <a> ... omsluter en länk
 - ... omsluter en bild
 - Plus många fler !!

HTML

- En tagg kan ha ett eller flera attribut
- Ett attribut ändrar t ex färg, font, storlek etc. eller var man hittar en fil
- Exempel:
 - <p color:"red"> ... </p> ger röd text
 - <p background:"yellow"> ... </p> ger gul bakgrund
 - <p font-family:"verdana"> ... </p> ger fonten Verdana
 - <a href=<http://www.fi.se>> Finansinspektionen ger en länk till finansinspektionen

En html-fil

```
<!DOCTYPE html>
<html>
<body>

<h2>Spectacular Mountain</h2>


</body>
</html>
```

Nerladdning

1. Man skriver www.fi.se och slår return
2. Ett GET skickas till servern www.fi.se
3. Man får tillbaka index.html från servern
4. Webbläsaren kollar om det finns andra objekt som behövs t ex bilder
5. Webbläsaren skickar GET för att få de andra objekten
6. När alla objekt har kommit kan hela sidan visas

Bilder

- Rastergrafik
 - Pixlar är grundelement
 - Ser pixelerade ut vid förstoring
 - Format: JPG, GIF
- Vektorgrafik
 - Elementära geometriska former beskriver bilden
 - Behåller skärpan vid förstoring
 - Format: SVG, EPS

Bilder

- Raster- och vektorgrafik:



Och nu exempel 1...

Att ändra utseende

- Kan göras med attribut i en tagg
- Gäller då på ett ställe

```
<!DOCTYPE html>
<html>
<body>

<h2 style="color:red">I am Red</h2>
<h2 style="color:blue">I am Blue</h2>

</body>
</html>
```

Att ändra utseende

- Kan göras i headern

```
<!DOCTYPE html>
<html>
<head>
<style>
body {background-color:lightgray}
h1 {color:blue}
p {color:green}
</style>
</head>
<body>

<h1>This is a heading</h1>
<p>This is a paragraph.</p>

</body>
</html>
```

Och nu exempel 2 ...

Problem

- En stor webbplats med enhetlig design
- Ändring av designen => många filer måste ändras
- Lägg designen i en fil, "importera denna"
- Kallas CSS-filer!
- CSS = Cascading Style Sheets

Och nu exempel 3 ...

Var finns CSS?

- I en extern fil
 - Lätt att ändra många filer
- I headern på en html-fil
 - Används bara i just den html-filen
- I taggen i ett element i en html-fil
 - Används bara i just det elementet

Prioritetsordning

1. Det som finns i elementet (högst prioritet)
2. Det som finns i headern
3. Det som finns i css-filen

Och nu exempel 4 ...

Elementselektor

- Elementselektor

```
p {
  text-align: center;
  color: red;
}
```

- Sedan blir alla `<p> ... </p>` röda och centrerade

Id-selektor

- Exempel:

```
#para1 {
  text-align: center;
  color: red;
}
```

- Sedan blir `<p id="para1">Hello World!</p>` röd och centrerad

Klass-selektor

- Exempel:

```
.center {
  text-align: center;
  color: red;
}
```

- Då blir `<h1 class="center">Red and center</h1>` röd och centrerad

Och nu exempel 5 ...

Placering av element

- Hittills blir det bara en linjär text på hela sidan
- Om man vill göra menyfält, ha saker bredvid varandra etc. etc.

En viktig tagg...

- `<div> ... </div>`
- Omsluter en del av ett html-dokument
- Används ofta tillsammans med css

Användning av div

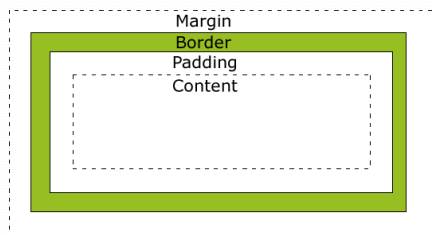
Exempel:

```
<div style="color:blue">
  <h3>This is a heading</h3>
  <p>This is a paragraph.</p>
</div>
```

All text inom div-taggen får blå text.

Boxmodellen

- Alla HTML-element finns i en box



Exempel på en box

```
<!DOCTYPE html>
<html>
<head>
<style>
div {
  background-color: lightgrey;
  width: 300px;
  padding: 25px;
  border: 25px solid navy;
  margin: 25px;
}
</style>
</head>
<body>
<div>I How that personage haunted my dreams, I need scarcely tell you.
On stormy nights, when the wind shook the four corners of the house
and the surf roared along the cove and up the cliffs, I would see him in a
thousand forms, and with a thousand diabolical expressions. </div>
</body>
</html>
```

Placering av en box

- Static, normal placering i flödet
- Relative, i förhållande till sin normala placering
- Fixed, alltid på samma plats
- Positionerad i förhållande till den närmaste föregångaren
- Överlappning, något ovanför något annat

Och nu exempel 6 ...

Slut för i dag!