

Digitaalisen median tekniikat
JavaScript_osa2

7.4.2004 Harri Laine 1

JavaScript oliot

- JavaScriptissä voidaan määritellä myös luokkia ja olioita.
- Luokka määritellään konstruktorifunktion avulla

```
function Ympyra(x,y,r) {  
  this.xkoord=x;  
  this.ykoord=y;  
  this.sade=r;  
  this.keha= 2*Math.PI*this.sade;  
  this.pinta_ala= Math.PI*(this.sade * this.sade);  
}
```

eivät muutu, jos sädetä myöhemmin muutetaan

- ja tälle instanssi

```
var y1 = new Ympyra(10,10,3);
```

7.4.2004 Harri Laine 2

JavaScript oliot

- instanssin muuttujiin viitataan piste-notaatiolla tai indeksillä:
 - y1.sade
 - y1['sade']
- Olio voidaan ajatella assosiatiivisena taulukkona

y1	
'xkoord'	10
'ykoord'	10
'sade'	3
'keha'	
'pinta_ala'	

7.4.2004 Harri Laine 3

JavaScript oliot

- Kuten taulukkoonkin oloon voidaan lisätä uusia ominaisuuksia, jos indeksi ei ole käytössä luodaan uusi alkio:
y1.vari='punainen';
- aiheuttaa uuden ominaisuuden lisäyksen

7.4.2004 Harri Laine 4

JavaScript oliot

- Metodin lisääminen:
- Muutetaan ympyrän määrittelyä:

```
function Ympyra(x,y,r) {  
  function laske_keha { return (2*Math.PI*this.sade); }  
  function laske_pinta_ala {  
    return (Math.PI*(this.sade * this.sade));  
  }  
  this.xkoord=x;  
  this.ykoord=y;  
  this.sade=r;  
  this.keha= laske_keha;  
  this.pinta_ala= laske_pinta_ala;  
}
```

FUNKTIOARVOISIA MUUTTUIJA

7.4.2004 Harri Laine 5

JavaScript oliot

- Edellä oleva voidaan antaa myös:

```
function Ympyra(x,y,r) {  
  this.xkoord=x;  
  this.ykoord=y;  
  this.sade=r;  
  this.keha= laske_keha;  
  this.pinta_ala= laske_pinta_ala;  
}  
function laske_keha { return (2*Math.PI*this.sade); }  
function laske_pinta_ala {  
  return (Math.PI*(this.sade * this.sade));  
}
```

FUNKTIOARVOISIA MUUTTUIJA

7.4.2004 Harri Laine 6

JavaScript oliot

- Metodin kutsu: `y1.keha()`
- Metodilla voi olla myös parametreja
- Instanssimetodit määritellään konstruktorissa sijoittamalla arvo instanssimuuttujalle.
- Tällainen metodi on jokaisella instanssilla
- Metodeja voidaan lisätä myöhemmin, mutta silloin ne ovat instanssikohtaisia
- `y1.isonna= kasvata(z);`

```
function kasvata(z) {
  this.sade +=z;
}
```

7.4.2004

Harri Laine

7

JavaScript oliot

- Luokkametodit ovat myös mahdollisia:

```
function laske_halkaisija (sade) {
  return (sade*2);
}
```

`Ympyra.halkaisija = laske_halkaisija;`

7.4.2004

Harri Laine

8

JavaScript valmiit luokat

- **Date** - päiväys
 - mittaa unix-aikaa eli millisekunteja 1.1.1970 alkaen
 - `var nykyhetki =new Date();`
 - osien asetus- ja kyselymetodeja (**katso esimerkki**)

<code>getDate()</code>	the day on the month - numeric value
<code>getDay()</code>	the day of the week 1=Monday .. 7
<code>getMonth()</code>	the month as an integer 0= January .. 11
<code>getFullYear()</code>	the year as a four digit number
<code>getMinutes()</code>	minutes (0-59)
<code>getHours()</code>	integer in 24 hour system. (0-23)

7.4.2004

Harri Laine

9

JavaScript valmiit luokat

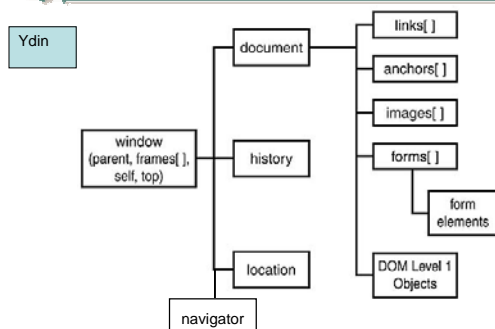
- **Math**: matemaattisia funktioita luokkametodeina
- **String**:
 - merkkijono on luokkakin vaikka sijoitukset ja vertailut toimivat kuten perustietotyypillä
 - ominaisuuksia ja metodeja esim. (**katso esimerkki**)
 - `length`
 - `substring(alku,loppu)` `substring(0,1)` =ensimmäinen merkki =`charAt(0)`
 - `substr(alku,pituus)`
 - `indexOf(merkkijono, alku)`
 - `toLowerCase(),toUpperCase(),`
 - `RegExp, Object,`

7.4.2004

Harri Laine

10

Selainluokat ja oliot



7.4.2004

Harri Laine

11

Selainluokat ja oliot

- **window**
 - selainikkunaan liittyviä palveluja
 - `window` globaali olio, muu rakenne sen alapuolella
 - `open` (uusi ikkuna), `close`
 - pop-upit: `alert`, `prompt`, `confirm` (**katso esimerkki**)
 - kymmeniä metodeja
- **navigator**
 - tietoa selaimesta (**katso esimerkki**)
 - selaimen tunnistus voidaan tehdä `navigator`:in tietojen perusteella - usein kuitenkin kysytään jotain yleisiä ominaisuuksia, jonka tiedetään olevan vain jollain selaimella, esimerkiksi: `if (document.all)` tunnistaa uudehkon IE:n

7.4.2004

Harri Laine

12

Selainluokat ja oliot

- **history**
 - tietoa selailureitistä
 - voidaan esimerkiksi toteuttaa selaimen back- ja forward- näppäimet sivulle
- **location**
 - tietoa ladatusta dokumentista mm href ominaisuus=nykyisen dokumentin URL, vaihtamalla tämän arvo vaihtuu ikkunan sisältö
 - `location.replace(url)` vaihtaa ikkunan sisällön ja korvaa historiassa edellisen `url:n` uudella. Edelliseen ei siten pääse takaisin back-näppäimellä

7.4.2004

Harri Laine

13

Selainluokat ja oliot

- **document**
 - varsinaisen dokumentin pääolio
 - eri tyyppiset elementit erillisissä taulukoissa
 - `anchors`, `applets`, `forms`, `images`, `links`,...
 - rakenteita joilla voidaan käydä läpi eri tyyppisiä elementtejä
 - `childNodes`= solmun lapsielementit, `all` (IE only)
 - dokumentin ominaisuuksia
 - `referrer`, `cookie`, `title`, `attributes`-taulukko, ...

7.4.2004

Harri Laine

14

Selainluokat ja oliot

- **document** metodeja, mm:
 - `open()`, `close()`, `write(string)`, `writeln(string)`
 - navigointimetodit
 - `getElementById(string)` –id:n perusteella
 - `getElementsByTag(string)` – tagin perusteella
 - `getElementsByName(string)` – name attribuutin perusteella

```
tabs=document.getElementsByTag("table");  
alkioihin viitataan tabs[i] tai tabs.item(i)
```

- aineiston luonti
 - `createElement(tag)`
 - `createTextElement(string)`
- arvomuutokset
 - `setAttribute(name,value)`

7.4.2004

Harri Laine

15

Selainluokat ja oliot

- **element** yliokka dokumentin elementeille
 - kaikille yhteisiä rakenteita ja metodeja, tärkeimpiä:
 - `attributes` - attribuuttitaulukko
 - `childNodes` - lapsielementit
 - `id` - tunniste
 - `class` - elementin luokitus (class-määre)
 - `innerHTML` - sisältö tageineen
 - `style` - tyylimäärittys
 - JavaScriptissä tyyliattribuuteille eri nimet kuin CSS:ssä tähän tapaan: `borderTop =CSS:border-top`

7.4.2004

Harri Laine

16

Selainluokat ja oliot

- **element** metodeja
 - `getAttribute(name)`, `setAttribute(name,value)`, `hasAttribute(name)`
 - `getElementsByTagName(tag)`

<http://www.mozilla.org/docs/dom/domref/>

http://msdn.microsoft.com/library/default.asp?url=/workshop/author/dhtml/reference/dhtml_reference_entry.asp

7.4.2004

Harri Laine

17

JavaScript tapahtumakäsittelijät

- `onblur` Returns the event handling code for the blur event.
- `onclick` Returns the event handling code for the click event.
- `ondblclick` Returns the event handling code for the dblclick event.
- `onfocus` Returns the event handling code for the focus event.
- `onkeydown` Returns the event handling code for the keydown event.
- `onkeypress` Returns the event handling code for the keypress event.
- `onkeyup` Returns the event handling code for the keyup event.
- `onmousedown` Returns the event handling code for the mousedown event.
- `onmousemove` Returns the event handling code for the mousemove event.
- `onmouseout` Returns the event handling code for the mouseout event.
- `onmouseover` Returns the event handling code for the mouseover event.
- `onmouseup` Returns the event handling code for the mouseup event.
- `onresize` Returns the event handling code for the resize event.
- `onload`, `onunload` usein body elementin yhteydessä

7.4.2004

Harri Laine

18

JavaScript html-dokumentissa

- JavaScriptiä voidaan upottaa HTML-koodiin sekaan:
 - koodia sisältävinä <script> - elementteinä
 - ulkoisia tiedostoja kytkevinä <script> - elementteinä
 - linkin kohteina
 - tapahtuma-käsittelijöinä.

7.4.2004

Harri Laine

19

JavaScript html-dokumentissa

- Tyypillisesti dokumentin head-osaan sisältyy:
 - yksi <script>-elementti, jossa määritellään paikalliset funktiot ja globaalit muuttujat
 - mahdollisesti useita <script>-elementtejä, joissa ladataan ulkoisia JavaScript-tiedostoja (nämä sisältävät tyypillisesti useilla sivuilla tarvittavia funktioita)
- Body-osaan voidaan laittaa:
 - <script>-elementtejä sellaisiin kohtiin, joihin halutaan tuottaa sisältöä JavaScriptillä
 - Linkin kohteeksi voidaan määritellä suoritettava JavaScript-koodi
`OtherFile`
 - Elementin tapahtumakäsittelijäksi voidaan antaa Javascript-koodia
``
(kehysta() on aiemmin määritelty funktio)

7.4.2004

Harri Laine

20

JavaScript html-dokumentissa

- JavaScript koodi suoritetaan siinä vaiheessa kun se tulee vastaan dokumentin käsittelyssä:
- <script>-elementti, jossa koodia

```
<script type="text/javascript" language="JavaScript">
<!-- hide script from older browsers
  var hiThere = 'I am a JavaScript statment'; window.alert(hiThere);
// stop hiding script -->
</script>
```
- XHTML:ssä <script> -elementtiin voi liittää `defer="defer"` attribuutin ilmaisemaan, että koodi suoritetaan vasta kun koko dokumentti on ladattu

7.4.2004

Harri Laine

21

JavaScript html-dokumentissa

- ulkoisen tiedoston lataus:

```
<script type="text/javascript" language="JavaScript"
src="myscript.js">
</script>
```
- Esimerkkejä:
 - Tämän kurssin kurssisivujen valikko
 - <http://jdstiles.com/javamain.html>
 - <http://developer.irt.org/script/script.htm>

7.4.2004

Harri Laine

22

Dynaaminen valikko

- Kurssisivun valikko
 - katso toiminta tarkemmin sivulta <http://www.cs.helsinki.fi/group/vertti/index.html>
 - idea:
 - Sivusto muodostuu joukosta samarakenteisia sivuja
 - Kullakin sivulla on kaksisarakeinen taulukko, jonka vasemmassa sarakkeessa on sivuvalikko.
 - Sivun ollessa valittuna on sitä vastaava valikkoalkio korostettu
 - Valikko voi olla 3-tasoinen
 - alempi taso saadaan näkyviin ja piiloon valikkoalkiota näpdyttämällä

7.4.2004

Harri Laine

23

Dynaaminen valikko

- idea:
 - valikko on toteutettu monitasoisena listana
 - kullekin valikkotasolle on kaksi tasokohtaista luokkaa (class) `mnuX` ($X=0..2$) ja `mactX`
 - `mactX` luokitus annetaan aktiiviselle valikkoalkiolle
 - vain yksi valikkoalkio voi olla samanaikaisesti aktiivinen
 - luokitus tehdään JavaScriptillä, sivun tekijän ei tarvitse tietää siitä mitään
 - sivun tekijä tekee valikon tavallisena monitasoisena linkkilistana, jokaiselle lista-alkiolle on kuitenkin määriteltävä id:
 - alemman tason alkion id ylempään tason id:n alustaan
esim `valinta1` -> `valinta11` -> `valinta111`, `valinta112`,...
 - ylimmällä tasolla ei mikään id saa olla toisen osa
 - id-standardinimeämistä käytetään siihen, että yhdellä läpikäynnillä voidaan avata tarvittavat valikot

7.4.2004

Harri Laine

24