



Digitaalisen median tekniikat

Palvelinohjelmointi_2

Palvelinohjelmointi (PHP)

- PHP:n syntaksista ja operaattoreista
 - Lause lopetetaan puolipisteeseen
 - Sijoitus ($\$a=\b) on lauseke, jonka arvona on sijoitettu arvo
 - $\text{if } (\$a=\$b) \{c\}$
 - c suoritetaan, jos \$a:lle sijoitetaan nollasta, null:sta tai FALSE:sta poikkeava arvo
 - Operaattorit ovat pääpiirteissään C:n ja Javan kaltaisia, Eroja:
 - merkkijonojen katenointi -operaattori on piste (.)
 - $\$jono3 = \$jono1.\$jono2;$
 - $\$jono4 .= \$jono5;$
 - Operaattoria ++ ja – voi käyttää myös merkkijonoihin
 - $\$jono1 = 'apua';$
 - $\$jono++;$ (arvo nyt 'apub')
 - Taulukko-operaatioita: yhdiste: $\$taulukko1+\$taulukko2$

Palvelinohjelmointi (PHP)

- Vertailut

- Verrattaessa merkkijonoa numeroon (tai käytettäessä merkkijonoa laskennassa) muunnetaan merkkijono ensin numeroksi. Jos merkkijono alkaa kirjaimella on muunnoksen tulos 0, muuten otetaan jonon alusta numero-osuus (kuten JavaScript)
 - ('Hei'>10) on siis sama kuin (0>10)
- Yhtäsuuruus (==) ja identtisyys (===) vertailut tarjolla kuten JavaScriptissä. Identtisyyttä tutkittaessa ei tehdä tyyppimuunnosta.
- Myös taulukkoja voi vertailla
 - yhtäsuuruus = samat alkiot
 - identtisyys = samat alkiot samassa järjestyksessä

Palvelinohjelmointi (PHP)

- Kielen perusrakenteita:
 - Ohjelmakontrollin perusrakenteet kuten C ja Java
 - if, else, while, do..while, for, switch, break, return
 - tarjolla vaihtoehtoinen syntaksi
 - Toistolauseena **foreach** – taulukon läpikäyntiin
 - `foreach(taulukkolauseke as $arvo)` tai
 - `foreach(taulukkolauseke as $indeksi=>$arvo)`
 - muuttujat \$indeksi ja \$arvo saavat kullakin kierroksella arvoikseen vuorossa olevan alkion indeksin ja arvon.
 - hyvin kätevä tapa taulukon läpikäyntiin

Palvelinohjelmointi (PHP)

- Taulukon läpikäynti (vaihtoehto `foreach` rakenteelle)
 - funktio `each(taulukko)` palauttaa 2-alkioisen taulukon, jossa on parametritaulukon vuorossa olevan alkion indeksi ja arvo, jos vuoro menee ulos taulukosta palautetaan FALSE
 - funktio `list(muuttuja1, muuttuja2,...,muuttujaN)` kokoaa yhteen joukon muuttujia esim. sijoitusoperaation kohteeksi
 - taulukon käsittelyn voi perustaa sisäiseen vuoro-osoittimeen
 - `reset(taulukko)` asettaa vuoro-osoittimen taulukon alkuun

```
foreach($taulukko as $key=>$value) { ....}
```

vastaa rakennetta

```
reset($taulukko);
```

```
while (list($key,$value)=each($taulukko)) { .... }
```

sillä erolla, että jälkimmäinen vie vuoro-osoittimen taulukon loppuun ja edellinen ei vaikuta mitään vuoro-osoittimeen

Palvelinohjelmointi (PHP)

- Dokumentin kokoaminen osista – ulkoisen tiedoston liittäminen:
 - `include(filename)`
 - käsittely jatkuu, jos tiedostoa ei löydy
 - `require(filename)`
 - käsittely päättyy, jos tiedostoa ei löydy
- Lisättävät osat voivat sisältää php-koodia tai html-koodia tai jotain muuta. Tiedoston tyyppitunnus voi olla mitä tahansa.

Palvelinohjelmointi (PHP)

- Valmiit funktiot:
 - PHP:ssä on satoja valmiita funktioita eri tarkoituksiin (manuaalissa 112 funktioryhmää ja yhdessä ryhmässä voi olla kymmeniä funktioita) ja omia voi tehdä, jos valmiit eivät riitä.
 - **esimerkkejä:**
 - c-tyyppiset merkkijonovertailufunktiot strcmp,...
 - merkkijonomuunnokset strtoupper, ...
 - tiedostonkäsittely
 - tietokantakäsittely (sekä tkhj-kohtaisia funktioita, että yleiskäyttöisiä odbc-funktioita)
 - taulukkokäsittely,
 - sähköposti,
 - päiväysten käsittely
 - html-muunnokset
 - säännölliset lausekkeet

Esimerkkejä

- Kutsuparametrien listaus
- Globaalien taulukoiden alkiot
- Konfigurointitiedoston lukeminen
- Lomakkeen käsittely
 - elementtien käsittely
 - tarkistukset
 - virheistä ilmoittaminen
 - JavaScriptillä voidaan tehdä tarkistuksia selaimessa, mutta tiedot on aina tarkistettava uudelleen palvelimessa
 - lomake kannattaa pitää kootuna
 - esim erillinen funktio tuottaa lomakkeen
 - virheistä ilmoittaminen pitäisi suunnitella lomaketta suunniteltaessa.
- XML:n lukeminen