

## 8. Java, JavaScript, VRML, CGI, Flash, CSS, PHP, MySQL

Az Internet korai korszakában készített html oldalak fix információkat tároltak. A statikus formától való eltérést csupán a CGI programok biztosították, ami a hálózati forgalom terhelésével járt. Idővel tehát igény jelentkezett az olyan dinamikusan változó látványos oldalak megjelenésére is, amely nem jár együtt túlzott hálózati forgalom növekedéssel. Ma már az Internet egyre szépül oldalain, számos olyan oldallal is találkozunk, amely az egérpozíciótól függően megváltozik, adatbázisokkal operál, illetve egyéb speciális funkciókat rejt. E lehetőségek megvalósítására azonban a html nyelv már nem ad kellő lehetőséget. Megvalósításukhoz a html nyelv oldalba beillesztett Java, vagy JavaScript nyelven írt programok, VRML elemek, php szkriptek és MySQL adatbázisok hívhatók segítségül. E programnyelvekkel a lehetőségek olyan tárháza nyílik meg előttünk, amelyek korábbiakban elképzelhetetlenek voltak. E nyelvek részletes leírása azonban e könyv kategóriájába nem férnek be, hiszen több száz oldallal gyarapítanák terjedelmét, s ráadásul megismerésük komoly programozói ismereteket is igényel. Létezésükről és lehetőségeikről nem árt azonban némi információval rendelkezni.

Igen fontos azonban még az elején leszögezni, hogy a Java és a JavaScript nyelvnek semmi köze sincs egymáshoz. A két nyelv – bár mindegyik html környezetben is alkalmazható – teljesen más, utasítás-készletük és programozási módszerük eltér. Bár mind a két esetben a program a kliensgépen van végrehajtva, de az a Java esetében a kiszolgálón, a JavaScript esetében a kliensgépen van lefordítva is, s a Java esetében a program sokkal inkább elválik a html oldaltól.

### 8.1. Java

Mivel az Internet a hálózati lehetőségeket addig soha nem látott mértékben kitágította, megnövekedett az igény az e lehetőségeket kihasználó objektumorientált nyelvekre. Ezt az igényt felismerve fejlesztette ki a Java nyelvet a Sun Microsystems, amely egy hálózatos környezetben összekapcsolható nyílt rendszerek készítésére alkalmas programozási nyelv,

így messze túlmutat az egyéni, gépspecifikus saját gépünkön futó egyfelhasználós programokon. Segítségével olyan programok készíthetők, amelyek az Internet lehetőségeit kihasználva objektumorientált környezetben képesek problémák kezelésére, publikus, vagy korlátozott adatbázisok és rendszerek használatára. A nyelv gazdag utasításkészlettel rendelkezik, inkább a problémamegoldáshoz áll közelebb, mint a gépközei utasításhalmazhoz.

Az html felülethez ott kapcsolódhat, hogy segítségével készíthetünk olyan appleteket (kisebb alkalmazásokat), amelyek hívásai beágyazhatók a html oldalba. (A forrás külön tárolódik.) Mivel ezen appletek egy adott méretű grafikus területet kapnak az oldalból, e területen belül bármilyen grafikus vagy szöveges feladatot elláthatnak. Az oldalba ilyen alkalmazást az `<APPLET>` utasítással illeszthetünk be. Az utasítás paraméterei közt meg kell adnunk az applet nevét `CODE=appletnév.class` formában, valamint a használható ablakterület méretét `WIDTH=szélesség` `HEIGHT=magasság` formában. A html oldalról lehet segítségünk van az appletnek további paramétereket is átadni a `<PARAM NAME="paramé-ternév" VALUE="érték">` utasítások segítségével. Például:

```
<APPLET CODE="CrazyText.class" WIDTH=200 HEIGHT=48>
<PARAM NAME="text" VALUE="BBS-E Bt.">
<PARAM NAME="clear" VALUE="true">
<PARAM NAME="shadowDepth" VALUE="3">
</APPLET>
```

Mivel egyes html szerkesztők támogatják a Java nyelvet is, lehet segítségünk van arra, hogy néhány gombnyomással kész Java appleteket illeszszünk dokumentumainkba, amely már tartalmazza a megfelelő kódot a kívánt effektus, animáció vagy egyéb szolgáltatás megvalósításához.

## 8.2. JavaScript

A JavaScript nyelvet elsősorban a html oldalak interaktív funkcióinak kiszolgálására használhatjuk. Kifejlesztése a Netscape nevéhez fűződik, így sajnos kisebb eltérések lehetnek a különböző böngészők aláírt programok tekintetében. A JavaScript esetében nem szükséges megismerni a teljes nyelvet, nem szükséges megtanulni programozni ahhoz, hogy egy-egy lehetőséget oldalunkon felhasználhassunk. Segítségével könnyedén vihetünk be oldalunkra számos interaktív lehetőséget, könnyedén rendelhetünk egérműveletekhez különböző tevékenységeket, stb.

Magának a JavaScript nyelvnek a részletes bemutatására természetesen nem térünk ki, hiszen ahhoz mégiscsak meg kellene tanulni a programozási ismereteket, ez pedig nem biztos, hogy az első néhány weblap elkészítéskor praktikus lenne. Sok esetben nincs is szükség rá, hiszen a 7.17. fejezetben leírtak segítségével más módon is vihetünk interaktivitást oldalunkra, így gyakran megspórolhatjuk ezeket. Természetesen a ko-

molyabb lehet ségek kihasználásához szükséges lehet a JavaScript ismeretekre is.

A JavaScript programok – ellentétben a Javával – magukon a html lapon találhatóak. Elhelyezhetjük a fejlécben vagy az oldaltörzsben is `<script>` és `</script>` utasítások közt.

Nézzük meg egy példával, hogyan oldhatjuk meg JavaScript programmal a nyomógombok oly módon való megjelenítését, hogy azok rájuk mutatva más képet adjanak, illetve kattintásra a megfelelő oldal hívják be. (Mindezt természetesen praktikus módon az elbbi keretes oldalunk bal oldali keskenyebb keretébe helyezve célszerű megvalósítani, így a fájlt `tocframe.htm` néven mentjük.)

```
<HTML>
<HEAD>
<BASE TARGET="MainFrame">
<script language="JavaScript1.2">
<!--//
csalad_on = new Image(128,32); család_on.src = "n1x.gif";
csalad_off = new Image(128,32); család_off.src = "n1.gif";
aktualis_on = new Image(128,32); aktualis_on.src = "n2x.gif";
aktualis_off = new Image(128,32); aktualis_off.src = "n2.gif";
ceg_on = new Image(128,32); ceg_on.src = "n3x.gif";
ceg_off = new Image(128,32); ceg_off.src = "n3.gif";
ugyf_on = new Image(128,32); ugyf_on.src = "n4x.gif";
ugyf_off = new Image(128,32); ugyf_off.src = "n4.gif";
link_on = new Image(128,32); link_on.src = "n5x.gif";
link_off = new Image(128,32); link_off.src = "n5.gif";
//-->
</script>
<script language="JavaScript">
<!--//
function act (imagename)
{
if (document.images)
document[imagename].src = eval(imagename + "_on.src");
}
function deact (imagename)
{
if (document.images)
document[imagename].src = eval(imagename + "_off.src");
}
//-->
</script>
</HEAD>
<BODY BACKGROUND="hatter.gif" lang="HU">
<CENTER>
MENÜ
<BR><BR>
<A HREF="csalad.htm" onMouseOver="act('csalad')" onMouseOut="deact('csalad')"><IMG
SRC="n1.gif" name="csalad" ALT="Családom bemutatása" VSPACE=1 BORDER=0 HEIGHT=32
WIDTH=128></a><br>
<A HREF="aktualis.htm" onMouseOver="act('aktualis')" onMouseOut="deact('aktualis')"><IMG
SRC="n2.gif" name="aktualis" ALT="Aktuális információk" VSPACE=1 BORDER=0 HEIGHT=32
WIDTH=128></A><BR>
<A HREF="cegism.htm" onMouseOver="act('ceg')" onMouseOut="deact('ceg')"><IMG SRC="n3.gif"
name="ceg" ALT="Cégem bemutatása" VSPACE=1 BORDER=0 HEIGHT=32
WIDTH=128></A><BR>
```

```

<A HREF="ugyf.htm" onmouseover="act('ugyf')" onmouseout="deact('ugyf')"><IMG SRC="n4.gif"
name="ugyf" ALT="Ügyfeleink listája" VSPACE=1 BORDER=0 HEIGHT=32
WIDTH=128></A><BR>
<A HREF="link.htm" onmouseover="act('link')" onmouseout="deact('link')"><IMG SRC="n5.gif"
name="link" ALT="Hivatkozások kedvenc oldalakra" VSPACE=1 BORDER=0 HEIGHT=32
WIDTH=128></A><BR>
<HR size=2 width="100%" align=center>
<A HREF="fo.htm"><IMG SRC="n0.gif" ALT="Vissza a főoldalra" BORDER=0 HEIGHT=32
WIDTH=128></A>
<BR><HR>
<IMG SRC="counter.gif" BORDER=0>
<BR>
</BODY>
</HTML>

```

Ajánlott irodalom: Sikos László: JavaScript kliens oldalon – BBS-INFO kiadó, 2004.

### 8.3. VRML

Manapság egyre többet lehet hallani a virtuális valóság modellezéséről, annak lehet ségeiről. Internetes környezetben a VRML (Virtual Reality Modeling Language) segítségével van lehet ségünk arra, hogy kialakítsuk ezt a „világot”. A VRML nyelv felépítését tekintve hasonlít a html-hez, hiszen itt is egy sima ascii szövegbe kell az utasításokat rögzíteni. Ezt az állományt lehet aztán behívni a böngészőbe.

A VRML egy objektumorientált nyelv, utasításai olyan tereket, formákat, elemeket leíró utasítások, melyek segítségével egy háromdimenziós koordináta-rendszerben tudunk meghatározni különböző alakzatokat. Az utasítások azonban nem csupán az alakzatok formájának és tulajdonságainak leírására szorítkoznak, meghatározhatók fényforrások, kameraállások és még igen sok hasznos lehet ség is. A tereken belül pedig ugyanúgy hivatkozásokat helyezhetünk el, mint a html oldalakon. A tereket így aztán könnyedén el állíthatjuk, hiszen az egyszerűbb alakzatokból bonyolultabbakat, azokból pedig hivatkozásokkal bejárható térreszeket állíthatunk össze.

Természetesen ahhoz, hogy mindez élethűen jelenjen meg, nem csupán gyors letöltési lehet ségre, hanem viszonylag nagyobb teljesítményű számítógépre is szükség van, hiszen a leírásokban definiált terek látványát a saját gépünknek kell el állítani.

### 8.4. CGI

A CGI egy olyan felület, amely a html oldalak segítségével lehet séget biztosít a kliens szerver-kommunikációra. A CGI program a szerveren fut, és lényegében a html lapokról bárki lefuttathatja. A CGI nem egy programozási nyelv, hiszen CGI programot bármilyen olyan nyelven írhatunk, amely az adott szerveren elérhető (C, C++, Perl, stb.)

Amikor egy html oldalról hivatkozás történik egy CGI programra, úgy az lefut, majd kimenete a böngésző által megjelenített lapra kerül. A bemeneti adatokat az oldalaknál megismert módon szolgáltatathatjuk a CGI programoknak.

CGI programokat készíthetünk adatbázisok kezeléséhez, adatfeldolgozásokhoz, kérdésekhez, közvélemény-kutatásokhoz, oldallátogatói információk gyűjtéséhez, számlálókhoz, stb.

## 8.5. Flash

A Flash a Macromedia cég egy olyan animáció készítő programja, amely vektoros elemekből állítja el és mozgatja a képeket. A vele készített grafikák fájlmérete a speciális tömörítésnek köszönhetően hihetetlenül kicsi, így még modem csatlakozás esetén is gyors letöltést tesz lehetővé. A flash animációk megtekintéséhez azonban az azt lejátszó gépen először telepíteni kell egy futtató modult is. Ennek ellenére egyre népszerűbb, hiszen igen látványos animációk készíthetők vele. A flash-sel számos dolog megjeleníthető, készíthetünk segítségével komplett oldalt, de beintegrálhatjuk egy már létező html oldalba is. A flash animációk elkészítése azonban nem egyszerű, hiszen a grafikai készség mellett szükség van egy kis programozói tudásra is. A flash újabb verziói komoly utasításkészletet tartalmazó önálló programnyelvvé rendelkeznek. (Az utóbbi időben azonban már több szoftverben is találkozhatunk Flash exportáló lehetőséggel.)

Ájánlott irodalom: Sikos László: Macromedia Flash – BBS-INFO kiadó, 2005.

## 8.6. CSS

Amennyiben komolyabb oldalszerkezetet építünk fel, óhatatlanul beleütközünk abba a problémába, hogy a különböző oldalakon használt formázások nem teljesen egyformák, illetve egy későbbi változtatás esetén minden fájlban rengeteg helyen, külön-külön kell átírnunk az oldal megjelenését befolyásoló jellemzők paramétereit. Ezt a problémát tudjuk áthidalni a stíluslapok alkalmazásával. Egy összetett portál korrekt kivitelezéséhez az egységesítés és az oldalak kiterjedtsége miatt stíluslapokat célszerű alkalmazni. A weblapok megfelelő, egységes megjelenése stíluslapok nélkül elképzelhetetlen. A stíluslapok közül a leggyakoribb a külső CSS fájl használata, de végső esetben a stílusinformációkat magába a html fájlba is begépelhetjük. Egy CSS stíluslap utasítások felsorolásából áll. Stíluslapokat a régebbi leírónyelven készült dokumentumok ugyanúgy használhatják, mint a jelenlegi legújabb nyelvű weblapok. Stíluslapokkal számos formázást sokkal egyszerűbben egységesen elvégezni, ráadásul

számos olyan lehet ség is kínálkozik használatukkal, amelyekre a szimpla html kódban lehet ségünk sincs.

Belső, helyi stílusokat a dokumentumban a STYLE elemmel hozhatunk létre. Például:

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0//EN">
<HTML>
  <HEAD>
    <TITLE>Gipsz Jakab honlapja</TITLE>
    <STYLE type="text/css">
      h1 { color: blue }
      h1 { font-family: Verdana }
      img{
        border-style: outset;
        border-color: black;
      }
    </STYLE>
  </HEAD>
  <BODY>
    <H1> Üdvözöljük Gipsz Jakab honlapján!</H1>
    <CENTER><IMG SRC="emblema.gif"></CENTER>
    <P> Reméljük, jól érzi magát nálunk, és később is ellátogat oldalunkra.</P>
    ...
    <IMG SRC="kep.gif">
  </BODY>
</HTML>
```

A mintában a h1 stílusát állítottuk be kékre, illetve a képek keretezési módját egy olyan fekete keret térbeli kiemelésre, amit sima html kódban nem tudnánk kivitelezni.

Külső stíluslap esetén a fejlécbe egy <LINK rel="stylesheet" href="fájlnev.css" type="text/css"> sort kell gépelelnünk, ahol a fájlnev.css állomány maga a stíluslap css fájlja.

A CSS elemek igen összetettek lehetnek, azok struktúrába szervezhetők, számos tulajdonságot tartalmazhatnak szigorú szintaxisnak megfelelően.

Ajánlott irodalom: Sikos László: Stíluslapok a weben - CSS kézikönyv – BBS-INFO kiadó, 2005.

## 8.7. PHP

Ha dinamikus tartalmú weblapot szeretnénk készíteni, úgy mindenképp igénybe kell vennünk valamely webes programozási eszközt. Bár sok lehet ségünk létezik, mégis webes alkalmazások készítéséhez a PHP az egyik leggyakrabban használt programozási nyelv. Egyszerűen kezelhető és programozható, de könnyen illeszthető szinte bármelyik operációs rendszer kiszolgálójához, és a HTML szerkesztő programok többsége is kezeli a PHP kódot. A programozási nyelv eszközei, a parancsok szintaktikája a C programozási nyelvre hasonlítanak. Mivel a PHP szkriptnyelv, így a programunk kódját szinte bármilyen szövegszerkesztővel meg tudjuk írni. (A mentésnél egyszer szövegfórmátumba és .php kiterjesztéssel mentjük el az állományunkat a webkiszolgálónk megfelelő könyvtá-

rába.) A PHP szkripteket, hasonlóan bármilyen más HTML tag-hoz, nyitó és zárócímkek közé kell zárni. A szkript fordítását minden alkalommal elvégzi a kiszolgáló szerver, amikor meghívjuk a szkriptet. A php képes együttm ködni adatbázisokkal is, így például a MySQL adatbázis-kezel rendszerrel együtt használva igen hatékony fejleszt eszköz lehet.

Példa egy kis php szkriptre, amelyb l betkintést kaphatunk a php nyelv programozásának világába (a fájlt proba.php néven mentsük):

```
<html>
<head>
<title>PHP szorzótábla</title>
</head>
<body>
<?php
print ("Ez itt egy szorzótábla:");
print("<table border= \"1 \"> \n");
for ($i=1;$i<11;$i++)
{
    print("<tr> \n");
    for ($j=1;$j<16;$j++)
    {
        print(" \t<td>");
        print($i*$j);
        print("</td> \n");
    }
    print("</tr> \n");
}
print("</table>");
?>
</body>
```

Ajánlott irodalom: Sági Gábor: Webes adatbázis-kezelés MySQL és PHP használatával – BBS-INFO kiadó, 2005.

## 8.8. MySQL

A dinamikus, változó tartalmú weblapok készítéséhez nélkülözhetelen egy háttérben lévő adatbázis, amelyb l a rendszer a weboldalak felépítéséhez szükséges adatokat kapja. A MySQL a legelterjedtebb internetes adatbázis-kezelő rendszer. A szerveren tárolt adatokhoz a PHP nyelv segítségével férhetünk hozzá a legegyszerűbben. Az SQL adatbázis-lekérdez nyelv az angol nyelvtan alapján került kialakításra, így a használt parancsok is angol nyelvi alapúak. A MySQL adatbázis-kezelő rendszer egy időben több adatbázist is tud kezelni, így a vele kapcsolatos lehetőségek szinte korlátlanok. A MySQL adatbázison végzett műveletek végrehajtásának több módja is létezik, így az nem függ a használt adatbázis-kezelő rendszertől, az utasítások ugyanúgy működnek más relációs adatbázis-kezelő programoknál. A használatához e könyv tartalmánál lényegesen komolyabb szakértelem szükséges, de Interneten is számos kezelő programot tölthetünk le és telepíthetünk fel gépünkre.

Ajánlott irodalom: Sági Gábor: Webes adatbázis-kezelés MySQL és PHP használatával – BBS-INFO kiadó, 2005.