

Javascript dátum, idő, számláló megjelenítő

Használati útmutató

Tartalomjegyzék

1. Bevezető.....	1
2. Néhány szó a képekről.....	2
2.1. A felhasznált fontok.....	2
2.2. A képek magassága.....	2
2.3. A betűköz és a megjelenítés szélessége.....	3
2.4. Az áttetsző képek.....	3
2.5. A képcsomagok.....	3
3. A szkriptet alkotó függvények.....	4
3.1. Az „image_precache” függvény.....	4
Paraméterek.....	4
Forráskód.....	4
Működés.....	4
Példa.....	5
3.2. A „graph_text_output” függvény.....	6
Paraméterek.....	6
Forráskód.....	6
Működés.....	6
Példa.....	7
3.3. A „display_date_time_counter” függvény.....	7
Paraméterek.....	7
Forráskód.....	7
Működés.....	8
Példa.....	9
4. Példaprogram.....	10
4.1. A HTML állomány.....	10
4.2. A javascript állomány.....	10
4.3. A CSS állomány.....	11
4.4. A kimenet a böngészőben.....	12
5. A honlapról letölthető állományok.....	13
Impresszum.....	14

1. Bevezető

A javascript dátum, idő számláló megjelenítő egy néhány függvényből és a számjegyek, elválasztó karakterek helyettesítésére szolgáló képekből álló csomag, ami weboldalakon használható dátum, idő, valamint számlálók számjegyeinek a megjelenítésére.

A szkript különálló függvények segítségével valósítja meg a működését, ami egyszerűbbé teszi a felépítését, de magát a működtetést megnehezíti, mivel külön kell létrehozni a szükséges változókat, megjeleníteni a kívánt információt és gondoskodni az időzítésről. Először egy objektumorientált megoldás készült erre a feladatra, aminek nagy előnye volt, hogy egy lépésben lehetett példányosítani, inicializálni, és az időzítés hozzárendelésével működtetni. Ez leegyszerűsítette az életet és a honlapra való beillesztést, mert egyetlen sor beírásával ott volt és működött a weblapon az óra.

Azonban. Webfejlesztésnél szem előtt kell tartani azt, hogy az a weblapot nem kizárólag asztali gépen fogják megnézni, ezért az eszközök széles skáláján történő megjelenésre alkalmassá kell tenni. Gondolok itt a táblagépekre, TV-k és játékkonzolok operációs rendszerébe integrált böngészőkre, mobiltelefonokra. És mivel például a mobilböngészők némelyikét két vállra fektetik a megjelenítő függvényekhez futásidőben dinamikusan hozzárendelt időzítések, ezért szétszedtem az objektumot különálló változókra és függvényekre.

Tehát a fentebb leírtak miatt itt az egyszerűbb felépítésű, az implementáláskor több figyelmet igénylő, ám böngészők és hardverek széles skáláján használható változat.

Kőrösi József (jzsi1978@windowslive.com)

2. Néhány szó a képekről

A megjeleníteni kívánt óra, dátum vagy számláló méretét és egyéb tulajdonságait CSS segítségével lehet beállítani. A mellékelt „working_samples.zip” és „documentation_sample.zip” állományokban található stíluslapokon változatos CSS beállításokat lehet találni a különböző számlálók, órák, dátummegjelenítők formázásához.

2.1. A felhasznált fontok

Az képcsomagokhoz elkészítéséhez használt fontok mindegyike szabadon felhasználható nem kereskedelmi célra. Az egyes képcsomagok alapjául felhasznált fontok (amelyik fontra alapult) a következők:

- counter_background_lighted_1: A Gimp eszközeivel rajzolva
- counter_background_lighted_2: A Gimp eszközeivel rajzolva
- counter_background_lighted_3: A Gimp eszközeivel rajzolva
- counter_background_lighted_4: A Gimp eszközeivel rajzolva
- counter_black_background: A Windows Monospace fontja
- counter_pixelated_big: A Gimp eszközeivel rajzolva
- counter_pixelated_simple: A Gimp eszközeivel rajzolva
- counter_white_background: A Windows Monospace fontja
- destructo_beam_normal: Destructo Beam font (licenc információ mellékelve)
- destructo_beam_shadowed: Destructo Beam font (licenc információ mellékelve)
- digital_dream_fat_fix_width, digital_dream_fat_fix_width_shadowed_less, digital_dream_fat_fix_width_shadowed_more, digital_dream_fat_normal, digital_dream_fat_normal_shadowed_less, digital_dream_fat_normal_shadowed_more, digital_dream_license.txt, glowing_blue_fix_width, glowing_blue_normal, glowing_green_fix_width, glowing_green_normal, glowing_pink_fix_width, glowing_pink_normal, glowing_red_fix_width, glowing_red_normal, glowing_turquoise_fix_width, glowing_turquoise_normal, glowing_yellow_fix_width, glowing_yellow_normal: DigitalDream font (licenc információ mellékelve)
- transparent_black_fix_width: A Gimp eszközeivel rajzolva
- transparent_black_normal: A Gimp eszközeivel rajzolva
- transparent_white_fix_width: A Gimp eszközeivel rajzolva
- transparent_white_normal: A Gimp eszközeivel rajzolva

A Destructo Beam és a DigitalDream fontok szabadon felhasználhatók nem kereskedelmi célra, a hozzájuk tartozó licenc információ a képcsomagok könyvtárában található.

2.2. A képek magassága

Az egyes képcsomagokhoz tartozó képek mindig azonos magasságúak, a méretüket is ezzel lehet

szabályozni a hozzárendelt stíluslapon beállítva. Ez azért van így, mert vannak, amelyek fix és vannak, amelyek változó szélességű képekből állnak.

2.3. A betűköz és a megjelenítés szélessége

A betűközt a képek jobb vagy bal margójának beállításával lehet szabályozni. Amelyik képcsomag képein nagyobb árnyék van, annak nagyobb negatív margót érdemes adni.

A teljes óra vagy számláló szélességét is így lehet szabályozni a magasságával összhangban.

2.4. Az áttetsző képek

Vannak kifejezetten olyan képcsomagok, amelyek az azt tartalmazó HTML elem háttérétől, vagy a saját háttérszínétől kapja meg a színét, például a „counter_background_lighted_2” nevezetű. Ilyenből csak fix szélességűek vannak, mert ezeket szorosan egymás mellé kell illeszteni a megfelelő megjelenítéshez.

Ezeknek nagy előnye, hogy tetszőleges színben megjeleníthetők. Egy ilyen képcsomag szerepel a 4. fejezetben bemutatott példaprogramban is.

2.5. A képcsomagok

A képcsomagokat alkotó képek a Gimp 2.8-as verziójával készültek, egyrészt a rendelkezésre álló betűtípusok felhasználásával, másrészt a beépített eszközökkel megrajzolva.

A letölthető „javascript_date_time_counter_images.zip” állományban 32 képcsomag van. Vannak benne világos és sötét háttérhez illeszkedő képek, valamint olyanok is, amik kifejezetten számláló megjelenítéséhez készültek.

És van benne rózsaszín is. Lányoknak :)

3. A szkriptet alkotó függvények

A szkript a következő függvények segítségével jeleníti meg a dátumot, időt, vagy a számláló számjegyeit:

- ***image_precache*** - ez gondoskodik a számjegyek képeinek előzetes letöltéséről
- ***graph_text_output*** - szövegből grafikus kimenetet készít
- ***display_date_time_counter*** - dátumot, időt vagy számlálót jelenít meg az előző két eljárás segítségével

3.1. Az „*image_precache*” függvény

```
image_precache(image_path, image_style, character_count)
```

Paraméterek

- ***image_path***: a képcsomagokat tartalmazó könyvtár neve
- ***image_style***: a képcsomag neve
- ***character_count***: a karakterszám, ennyi karakterhez veszi fel tömbbe az egyes számjegyek, elválasztó karakterek képeit

Forráskód

```
function image_precache(image_path, image_style, character_count)
{
    var digits_array=new Array();
    for(var i=0; i<=(character_count-1); i++)
    {
        digits_array[i]=new Array();
        for(var j=0; j<=9; j++)
        {
            digits_array[i][j]=new Image();
            digits_array[i][j].src=image_path+image_style+'/'+image_style+'_'+j+'.png';
        }
        digits_array[i][10]=new Image();
        digits_array[i][10].src=image_path+image_style+'/'+image_style+'_selector_1.png';
        digits_array[i][11]=new Image();
        digits_array[i][11].src=image_path+image_style+'/'+image_style+'_selector_2.png';
        digits_array[i][12]=new Image();
        digits_array[i][12].src=image_path+image_style+'/'+image_style+'_selector_3.png';
    }
    return digits_array;
}
```

Működés

Az ***image_path*** változó tartalmazza a képcsomagokat tartalmazó könyvtár nevét. Ebben a könyvtárban vannak a képcsomagok neveivel megegyező nevű könyvtárakban az egyes képcsomagokat alkotó képek.

Az ***image_style*** változó a karakterstílus neve, ami a stílushoz tartozó képeket tartalmazó könyvtár neve, az ***image_path*** változóban található könyvtár alkönyvtára. Ezt a két változó az egyszerűbb stílusmódosítás miatt van különválasztva, amit egyrészt a mellékelt ***samples.html*** állomány

összetettsége is indokol, mivel itt minden stílust szerepeltetni kellett egyetlen oldalon. Másrészt így egyszerűbben lehet hozzáadni, kipróbálni saját készítésű képeket tartalmazó stílusokat az eredeti szerkezet megbolygatása nélkül.

A `character_count` változó az a maximális karakterszám, amit fel akarunk használni a megjelenítéshez. Ezt előre meg kell tervezni, dátum esetén (például 2012.09.12.) legyen 11 vagy több. Kevesebb ne legyen, ez esetben hibát fog jelezni a felhasználás során, mivel nem létezik a tömbem, amire hivatkozunk.

A függvény a szöveg minden egyes karakteréhez felveszi tömbbe a 13 írásjelet, amit a képek jelenítenek meg. Ezek az írásjelek: „0123456789.:’”. A képfájlok elnevezése a következő szerkezetet követi (ez a példa a „digital_dream_fat_normal” képstílus képeire vonatkozik):

```
0 - digital_dream_fat_normal_0.png
1 - digital_dream_fat_normal_1.png
2 - digital_dream_fat_normal_2.png
3 - digital_dream_fat_normal_3.png
4 - digital_dream_fat_normal_4.png
5 - digital_dream_fat_normal_5.png
6 - digital_dream_fat_normal_6.png
7 - digital_dream_fat_normal_7.png
8 - digital_dream_fat_normal_8.png
9 - digital_dream_fat_normal_9.png
. - digital_dream_fat_normal_selector_1.png
: - digital_dream_fat_normal_selector_2.png
/ - digital_dream_fat_normal_selector_3.png
```

A pont, a kettőspont és a perjel képei rendre selector_1, selector_2 és selector_3 utótagot kaptak, a számok pedig a számjegyet, amit ábrázolnak.

Tehát a függvény a példában szereplő 11 karakterből álló szöveghez egy 11x13-as, azaz 143 elemből álló tömböt tölt fel a képekkel. Ez a pazarlásnak tűnő megoldás a további server irányába történő kéréseket hivatott megelőzni, ami e nélkül a függvény nélkül a megjelenített óra vagy számláló minden frissítésekor megtörténne (még akkor is, ha „not modified” választ kap vissza).

Ezzel a módszerrel a szkript betöltődése után nincsenek további server irányú kérések. Ha egy másodpercenként frissülő órát veszünk alapul, akkor már 13 másodperc után ennek az eljárásnak a javára billen a mérleg. Ezért tehát ez a kicsit durvának látszó kétdimenziós tömb a képek tárolására.

A tömb felhasználásakor az első index a karakter szövegben elfoglalt helye (nullától a szöveg hossza mínusz 1-ig), a második index az itt megjeleníteni kívánt kép a karakter alapján.

Példa

```
var digits=image_precache('images/digits','digital_dream_fat_normal',8);
```

Ez az utasítás létrehozza a `digits` nevű tömböt a képeknek, a képcsomagokat az `images/digits` könyvtárban keresi majd és itt pedig a `digital_dream_fat_normal` könyvtárban lévő képeket veszi fel. A tömbnek 8*13 eleme lesz, mivel 8 karakterhez veszi fel a képeket.

3.2. A „*graph_text_output*” függvény

```
graph_text_output(Containing_element, text, alt, title, image_array)
```

Paraméterek

- ***Containing_element***: az a html elem, amiben megjelenik az óra, dátum, számláló
- ***text***: a megjelenítendő szöveg
- ***alt***: a képekhez rendelt alternatív szöveg
- ***title***: a képekhez rendelt cím
- ***image_array***: a képek számára létrehozott tömb

Forráskód

```
function graph_text_output(containing_element, text, alt, title, image_array)
{
    var characters=text.split('');
    var text_as_graphic=new Array();
    var digit='';
    var count=0;

    for(var i=0;i<characters.length;i++)
    {
        digit=-1;
        switch(characters[i])
        {
            case '0':{digit= 0;}break;
            case '1':{digit= 1;}break;
            case '2':{digit= 2;}break;
            case '3':{digit= 3;}break;
            case '4':{digit= 4;}break;
            case '5':{digit= 5;}break;
            case '6':{digit= 6;}break;
            case '7':{digit= 7;}break;
            case '8':{digit= 8;}break;
            case '9':{digit= 9;}break;
            case '.':{digit=10;}break;
            case ':':{digit=11;}break;
            case '/':{digit=12;}break;
        }
        if(digit!=-1)
        {
            text_as_graphic[count]=image_array[i][digit];
            text_as_graphic[count].alt=alt;
            text_as_graphic[count].title=title;
            count++;
        }
    }
    $(containing_element).empty();
    for (var i=0;i<=count;i++)
    {
        $(containing_element).append(text_as_graphic[i]);
    }
}
```

Működés

A ***Containing_element*** változó tartalmazza a html elem nevét, amelyikbe bele fognak kerülni a megjelenítendő képek. Ezt szöveggént kell megadni és css stílusú kijelölésnek kell lennie például: ***'#counter_area #visitor_count'***.

A ***text*** változó a megjelenítendő szöveget tartalmazza. Az ***alt*** a képek helyén megjelenő alternatív szöveget, a ***title*** változó pedig a képekhez rendelt címet tartalmazza, ezek az órát, dátumot alkotó

képek mindegyikén azonosak lesznek.

Az `image_array` a szöveget alkotó karakterek számára már létrehozott tömb neve.

Amennyiben a szövegben a függvény által meg nem jeleníthető karakterek vannak, tehát a „0123456789.:/” karaktereken kívül bármilyen más karakter, akkor ott nem jelenít meg képet.

Példa

```
graph_text_output('#counter', '0041265', 'Látogatószám', 'Látogatószám', digits);
```

Ez az utasítás a „0041265” szöveget megjeleníti a `digits` nevű tömb alapján képekkel helyettesítve a karaktereket a `counter` azonosítóval rendelkező html elembe. A képekhez rendelt alternatív szöveg és cím a „Látogatószám” lesz.

3.3. A „display_date_time_counter” függvény

```
display_date_time_counter(text, display_format, containing_element, alt, title, image_array)
```

Paraméterek

- `text`: a megjelenítendő szöveg
- `display_format`: a kiválasztott megjelenítési formátum
- `containing_element`: az a html elem, amiben megjelenik az óra, dátum, számláló
- `alt`: a képekhez rendelt alternatív szöveg
- `title`: a képekhez rendelt cím
- `image_array`: a képek számára létrehozott tömb

Forráskód

```
function display_date_time_counter(text, display_format, containing_element, alt, title, image_array)
{
    var actual_date      =new Date();
    var year             =actual_date.getFullYear();
    var month            =actual_date.getMonth()+1;
    var day              =actual_date.getDate();
    var hour             =actual_date.getHours();
    var minute           =actual_date.getMinutes();
    var second           =actual_date.getSeconds();
    var millisecond      =actual_date.getMilliseconds();

    if(month<10)
    {
        month='0'+month;
    }
    if(day<10)
    {
        day='0'+day;
    }
    if(hour<10)
    {
        hour='0'+hour;
    }
}
```

```

if(minute<10)
{
    minute='0'+minute;
}
if(second<10)
{
    second='0'+second;
}
if(millisecond<10)
{
    millisecond='0'+millisecond;
}
if(millisecond<100)
{
    millisecond='0'+millisecond;
}

switch(display_format)
{
    case 'hourmin':
        {formatted_text=hour+':'+minute; }
        break;
    case 'hourminsecond':
        {formatted_text=hour+':'+minute+':'+second; }
        break;
    case 'hourminsecondmillisecond':
        {formatted_text=hour+':'+minute+':'+second+'.'+millisecond; }
        break;
    case 'yearmonthday_1':
        {formatted_text=year+'.'+month+'.'+day+'.'; }
        break;
    case 'yearmonthday_2':
        {formatted_text=year+'/' +month+'/' +day; }
        break;
    case 'raw_text':
        {formatted_text=text; }
        break;
}

graph_text_output(containing_element,formatted_text,alt,title,image_array);
}

```

Működés

A `text` változóba a megjelenítendő szöveg kerül. Amennyiben a `display_format` változó értéke „raw_text”, akkor figyelembe veszi és ezt a szöveget alakítja át képekké, ellenkező esetben a függvény figyelmen kívül hagyja az értékét és a `display_format` további lehetséges értékei szerint jár el.

A `display_format` változó értéke a megjelenítendő formátumot állítja be. Lehetséges értékei az alábbiak:

- 'hourmin': óra és perc (megjelenés: '18:25')
- 'hourminsecond': óra, perc, másodperc (megjelenés: '18:25:48')
- 'hourminsecondmillisecond': óra, perc, másodperc, ezredmásodperc (megjelenés: '18:25:48.655')
- 'yearmonthday_1': év, hónap, nap pont elválasztóval (megjelenés: '2012.07.18.')
- 'yearmonthday_2': év, hónap, nap per-jel elválasztóval (megjelenés: '2012/07/18')
- 'raw_text': nyers szöveg, számlálóhoz elsősorban (megjelenés: '00162255')

A függvény lekérdezi az aktuális dátumot és időt, átalakítja a `display_format` változó értéke szerinti formátumba és az átalakított szöveget átadja a `graph_text_output eljárásnak`. Amennyiben a `display_format` értékének „raw_text” van beállítva, akkor az átalakítás nélküli szöveget adja át, ami a paraméterlista elején lévő `text` változóban található.

Példa

```
display_date_time_counter('', 'hourmin', '#clock', 'Idő', 'Idő', clock_array);
```

Ez az utasítás megjeleníti az aktuális időt óra:perc formátumban a „clock” azonosítóval ellátott HTML elemben a `clock_array` nevű tömb alapján képekkel helyettesítve a karaktereket. A képekhez rendelt alternatív szöveg és cím az „Idő” lesz.

4. Példaprogram

A lentebbi példaprogramot alkotó állományok szintén letölthetők betömörítve „documentation_sample.zip” néven. Ez egy rövid példa a szkript működtetésére.

4.1. A HTML állomány

A „documentation_sample.html” állomány tartalma:

```
<!DOCTYPE HTML>
<html>
<head>
  <title>Javascript dátum, idő, számláló megjelenítő példa HTML</title>
  <meta http-equiv="content-type" content="text/html; charset=utf-8">
  <meta name="author" content="Kőrösi József">
  <meta name="viewport" content="width=device-width, initial-scale=1.0, user-scalable=yes">
  <link href="documentation_sample.css" rel="stylesheet" type="text/css">
  <script type="text/javascript" src="jquery_1_7_2.js" ></script>
  <script type="text/javascript" src="javascript_date_time_counter_display.js"></script>
  <script type="text/javascript" src="documentation_sample.js" ></script>
</head>
<body>
  <div id="jsdatetimercounter_samples_container">
    <h1>Javascript dátum, idő, számláló megjelenítő példa HTML</h1>
    <div class="noscript_warning">
      <p>(Ennek a megjelenítéséhez javascript szükséges.)</p>
    </div>
    <div id="clock_area">
      <div id="sample_clock"></div>
    </div>
  </div>
</body>
</html>
```

4.2. A javascript állomány

A „documentation_sample.js” állomány tartalma:

```
var array_clock=image_precache('javascript_date_time_counter_images/', 'counter_background_lighted_2', 8);

function show_clock()
{
  display_date_time_counter('', 'hourminsecond', '#sample_clock', 'A pontos idő', 'A pontos idő', array_clock);
}

function tictac()
{
  timer_current_time=setInterval('show_clock()', 1000);
}

$(document).ready
(
  function ()
  {
    $('#noscript_warning').hide();
    $('#jsdatetimercounter_samples_container #clock_area').show();
    show_clock();
    tictac();
  }
);
```

4.3. A CSS állomány

A „documentation_sample.css” állomány tartalma:

```
.noscript_warning
{
  display:block;
  text-align:center;
}

body
{
  background-color:#cca;
}

h1
{
  font-size:18px;
  font-weight:bold;
  text-align:center;
}

#jsdatetimercounter_samples_container
{
  border:1px solid #000;
  display:block;
  margin:0 auto;
  padding-bottom:20px;
  width:560px;
}

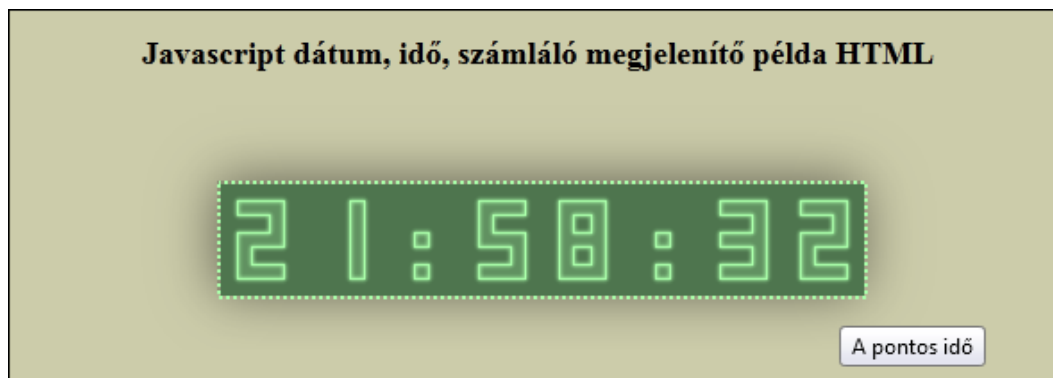
#clock_area
{
  background-image:url('css_images/sample_clock_bg.png');
  display:none;
  height:145px;
  margin:0 auto;
  padding:0;
  position:relative;
  width:460px;
}

#sample_clock
{
  border:2px dotted #afa;
  display:block;
  height:63px;
  left:50px;
  margin:auto auto;
  padding:0;
  position:absolute;
  top:50px;
  width:360px;
}

#sample_clock img
{
  background-color:#afa;
  margin:0;
  padding:0;
  border:none;
}
```

4.4. A kimenet a böngészőben

Ez a kép a fentebb leírt példaoldal HTML, javascript és CSS állományok által létrehozott böngészőben való megjelenését mutatja be. Persze kevésbé csicsás megjelenést is létre lehet hozni...



Az összes, itt felsorolt állomány a „documentation_sample.zip” nevű fájlban van. Kitömörítést követően azonnal megjeleníthető böngészőben, helyi gépen is, nem igényel háttérben futó szerveret.

5. A honlapról letölthető állományok

Ezeket az állományokat lehet a honlapról letölteni tömörített formátumban:

- javascript_date_time_counter_display.zip: a javascript dátum, idő, számláló megjelenítőt működtető függvényeket tartalmazó javascript állomány
- javascript_date_time_counter_images.zip: a szkriptidez készített 32 képcsomag
- jquery_1_7_2.zip: jQuery JavaScript Library 1.7.2
- working_samples.zip: a honlapon látható megjelenítési minták helyi gépen történő megjelenítéséhez átalakított változata
- documentation_sample.zip: a dokumentációban szereplő példaprogram
- javascript_date_time_counter_display_readme.pdf: a szkript dokumentációja (ez a fájl)

Impresszum

Ez a szkripteket és képeket tartalmazó csomag weblapokon használható idő, dátum és számláló megjelenítésére.

Az ehhez tartozó forráskódok, képcsomagok, minta HTML és CSS fájlok letölthetők a <http://www.dreamweaver78.hu> honlapról.

Kérdéseket, észrevételeket, visszajelzéseket a <http://www.dreamweaver78.hu> honlap kapcsolat oldalán, vagy a jozsi1978@windowslive.com e-mail címen várok.

Kőrösi József (jozsi1978@windowslive.com)