

Лабораторна робота № 11

Рух і перетягування

Мета роботи: вивчити методи створення сценаріїв для автоматичного руху об'єктів по заданій траєкторії і для перетягування об'єктів мишкою.

Рух

Для управління рухом об'єкта використовуються:

- метод ім'я процесу = `setInterval` (вираз, період) об'єкта `window`, що запускає процес циклічного виконання виразу (функції);
- метод `clearInterval` (ім'я процесу) об'єкта `window`, що зупиняє процес.

Методом `setInterval` (вираз, період) запускається процес, що складається в циклічному виконанні заданого в першому параметрі вираження до періоду, заданого в другому параметрі. Процес зупиняється методом `clearInterval` (ім'я процесу). Як вираження зазвичай використовується функція. Скрипт, що задає рух фотографії по діагоналі екрану, приведений в прикладі 1.

Приклад 1

```
<HTML>
<HEAD> <TITLE> Рух фотографії </ TITLE>
</ HEAD>
<BODY>
<BUTTON ID = "B1" onclick = Start_stop ()
STYLE = "position: absolute; left: 300; top: 5">
Запустити </ button>
<IMG ID = "lv" src = "LV.jpg"
STYLE = "position: absolute; top: 0; left: 20">
</ BODY>
<SCRIPT>
dx = 8;
dy = 3
st = false // зупинити процес
function Start_stop ()
{If (st)
  {Window.clearInterval (proc) // зупинитися
   st = false // замінити напис на кнопці з ім'ям B1
   document.all.B1.innerHTML = "Запустити"
  }
  else // рухатися (запустити процес з ім'ям proc)
  {Proc = window.setInterval ( "move ()", 100)
   st = true
   // замінити напис на кнопці з ім'ям B1
   document.all.B1.innerHTML = "Стоп"
  }
}
function move ()
{// ордината верхнього краю фотографії (переведено
// в тип INTEGER)
var y = parseInt (document.all.lv.style.top)
// абсциса лівого краю фотографії
var x = parseInt (document.all.lv.style.left)}
```

```

document.all.lv.style.top = y + dy // Переміщення по OY
document.all.lv.style.left = x + dx // Переміщення по OX
}
</ Script>
</ HTML>

```

У початковому стані фотографія знаходиться у верхньому лівому куті екрану. Для запуску і зупинки руху служить кнопка. Напис на кнопці змінюється. Якщо фотографія нерухома, то на кнопці написано Запустити, при русі -Стоп.

У браузері Internet Explorer напис на кнопці можна змінити за допомогою властивості innerText:

```

document.all.B1.innerText = "Запустити"
document.all.B1.innerText = "Стоп"

```

Браузер Mozilla Firefox не підтримує властивість innerText, тому в прикладі використано властивість innerHTML, підтримуване обома браузерами. При натисканні кнопки виконується функція Start_stop, що служить для запуску або зупинки процесу ргос. Якщо ргос запущений, то через кожні 100 мсек викликається функція move, що переміщає фотографію на 8 пікселів вправо і на 3 вниз.

Завдання 1. Напишіть сценарій переміщення кольорового квадрата по колу. Траєкторію зручно описувати параметричними рівняннями:

$$y = R * \sin(t),$$

де: R- радіус кола, $0 \leq t \leq 2$

Квадратом може служити контейнер <DIV> ... </ div> з кольоровим фоном.

Перетягування

У сценаріях для перетягування будь-якого предмета по вікну браузера використовується подія onmousemove. Розглянемо два варіанти сценарію, в яких використовується подія onmousemove. В обох сценаріях користувач може мишкою пересувати фотографію по всьому вікну браузера. Перший варіант сценарію (Приклад 2) застосовується лише в браузері Internet Explorer. Другий варіант (Приклад 3) правильно виконується в браузерах Internet Explorer 6.0 і MozillaFirefox 2.0, але він складніше для розуміння.

Приклад 2.

```

<HTML>
<HEAD> <TITLE> ПЕРЕМІЩЕННЯ зображення мишкою
</ TITLE> </ HEAD>
<BODY id = "body1" STYLE = "font-size: 12pt">
<H2 align = "center" style = "color: # ff00ff">
Перетягування фотографії мишкою </ h2>
<P align = "center">
Фотографія починає рухатися при попаданні на неї курсора мишки
<BR> При захопленні і переміщенні фотографії не натискайте кнопки
мишки. <BR> Для припинення руху клацніть лівою кнопкою мишки </ p>
<P align = right> Примітка. Сторінка правильно відображається
только в браузері <i> InternetExplorer 6.0 </ i> </ p> <img id = "IM1"
src = "LV.jpg" "STYLE =" position: absolute; top: 200; left: 10 ">
</ BODY>
<SCRIPT> flag = false // заборона на переміщення
// Функція запускається при пересеченні
// курсором мишки над ділянкою
function IM1.onmouseover ()

```

```

{Flag = true}
// Перетягування дозволено //
function IM1.onmousemove ()
{If (flag)
  {// пересчёткоординаті зображення
document.all.IM1.style.top = event.clientY-170
document.all.IM1.style.left = event.clientX-120
  }
} // заборона на переміщення. Запускається при відпуску кнопки
мишки
function IM1.onclick ()
{Flag = false}
</ SCRIPT>
<SCRIPT language = "JavaScript" src = "ProvBrows.js">
</ Script>
</ HTML>

```

У прикладі 2 потрапляння курсора мишки на фотографію служить сигналом до початку переміщення. При подальшому переміщенні курсора мишки фотографія пересувається разом з ним. Користувач при переміщенні фотографії не повинен натискати на кнопки мишки, так як натискання на ліву кнопку зупинить рух і курсор можна буде прибрати з фотографії. Натискання правої кнопки крім припинення руху викликає спливання меню.

У сценарії використовуються три функції:

- IM1.onmouseover () для запуску процесу перетягування;
- IM1.onmousemove для перетягування;
- IM1.onclick () для припинення перетягування.

Всі три функції мають спеціальні імена. IM1.onmouseover () означає, що ця функція виконується, коли на об'єкті з ID = IM1 відбувається подія onmouseover (). Для функцій зі спеціальними іменами не потрібні оператори виклику.

Функція IM1.onmousemove () виконується циклічно протягом усього часу перебування курсору мишки в площі фотографії. Так як частота виконання функції не нескінченна, то при різкому русі мишкою можна зірвати курсор з фотографії і припинити перетягування, але таке завершення процесу незручно. Для управління процесом перетягування призначена логічна змінна flag. Якщо клацнути кнопкою мишки по фотографії, то виконається функція IM1.onclick (), змінна flag прийме значення false і функція IM1.onmousemove () припинить змінювати координати фотографії. Можна буде прибрати курсор з фотографії. Якщо знову хоча б торкнутися курсором фотографії, то виконається функція IM1.onmouseover (), змінна flag прийме значення true і перетягування відновиться.

У другому варіанті сценарію (приклад 3) не використовуються функції зі спеціальними іменами.

Приклад 3.

```

<html>
<head> <title> Перетягування </ title> </ head>
<body onload = "init ()">
<h2 align = "center" style = "color: # ff00ff">
Перетаскування квадрата мишкою </ h2> <h3 align = "center">
Встановіть курсор всередині червоного квадрата, натисніть ліву
кнопку
мишки і тягніть квадрат. <br> Для звільнення квадрата
від захоплення відпустіть мишку </ h3>

```

```

<P align = right> Примітка. сторінка правильно
відображається в браузерях <i> Mozilla Firefox 2.0 </ i>
i <i> Internet Explorer 6.0 </ i> </ p>
<DIV id = 'kv' style =
  "Position: absolute; left: 10; top: 10; width: 50; height: 50;
background-color: red">
</ Div>
</ Body>
<SCRIPT>
flag = false; // Зв'язування подій з функціями
function init ()
{ // Якщо мишка рухається то виконується ф-ція dragIt
document.all.kv.onmousemove = dragIt
// Якщо кнопка мишки відпускається, то - ф-ція osvobodі
document.all.kv.onmouseup = osvobodі
document.all.kv.onmousedown = zachvat
}
function dragIt (evt)
{Параметр evt створюється самим браузером
sobytie = (evt)? evt: (window.event)? window.event: "";
if (flag)
{Document.all.kv.style.top = sobytie.clientY-25;
  document.all.kv.style.left = sobytie.clientX-25; <
  }
}
function osvobodі ()
{Flag = false; // встановлюється заборона на перетягування
}
function zachvat ()
{Flag = true; // Перетягування дозволяється
}
</ SCRIPT>
</ Html>

```

На відміну від першого варіанта в другому явно викликається одна функція. Рядок
 <Bodyonload = "init ()">

наказує при завантаженні HTML-документа викликати функцію `init ()`. В результаті виконання функції `init ()` події, що відбуваються з об'єктом, що має ідентифікатор `kv`, будуть викликати виконання функцій `dragIt`, `osvobodі` і `zachvat`. Об'єкт `kv` - це контейнер `<DIV>`, що задає червоний квадрат, перетаскується мишкою.

Функція `zachvat ()`, роздільна почне переміщення, виконується при натисканні кнопки мишки на квадраті, тобто коли відбувається подія `onmousedown`. Функція `osvobodі ()` виконується при відпуску кнопки мишки (подія `onmouseup`) і забороняє перетягування.

Функція `dragIt (evt)` пов'язана з подією `onmousemove` і тому циклічно виконується протягом всього часу перебування курсору мишки на фотографії. Параметр `evt` - це об'єкт події, створений у відповідь на дію користувача. В Internet Explorer об'єкт події має позначення `window.event` або просто `event`. У Mozilla Firefox об'єкт події позначається як `evt`. Для сумісності обох випадків служить умовний оператор

```

sobytie = (evt)? evt: window.event;
або в більш звичній формі

```

```
if (evt) sobytie = evt  
else sobytie = window.event
```

Завдання 2. Змініть останній приклад так, щоб червоний квадрат можна було переміщати всередині сірого квадрата розміром 500х500 пікселів.