

## Лабораторна робота №12

### Вікна і таблиці

**Мета роботи:** вивчити способи управління вікнами і таблицями.

#### *Створення нового вікна*

Об'єкт window знаходиться в вершині дерева об'єктної моделі W3C DOM і має багато властивостей і методів. У сценаріях, особливо при налагодженні, часто використовується метод alert. Властивості setInterval, clearInterval і setTimeout необхідні для пожвавлення сторінки, організації руху об'єктів. Щоб не захаращувати основне вікно, часто для перегляду додаткової інформації (графіків, таблиць і т.д.) зручно відкрити нове додаткове вікно. Для створення нового вікна використовується метод open, що має такий синтаксис: var windowObject = window.open ([параметри]). Розглянемо сценарій (приклад 1), за яким вікно з описуваних на сторінці графіком з'являється при наведенні курсора мишкою на слова на графіку.

#### **Приклад 1**

```
<HTML>
<HEAD> <TITLE> </ TITLE> </ HEAD>
<BODY>
<H1 align = center> Графік відкривається в новому вікні </ h1>
Залежність  $y = f(x)$  <B onmouseover = "gr ('Графік 1', 'prim1.gif')"
style = "color: blue;"> на графіку 1 </ b> має лінійний характер.
<P> Зміна ціни і обсягів продажів показані
<B onmouseover = "gr ('Графік 2', 'prim2.gif')" style = "color:
green">
на графіку 2 </ b>. Обсяги продажів сірого кольору.
<P> <font size = -1>
Примітка. Для перегляду графіків підвідіть курсор мишкою до слів "на
графіку "<BR> </ p>
<Script>
function gr (txt, fil)
{GrWin = window.open ( "", "", "width = 650, height = 400")
 grWin.document.write ( "<H1 align = center>" + txt +
 "</ H1> <CENTER> <IMG SRC = '" + fil + "'> </ center>")
}
</ Script>
</ BODY>
</ HTML>
```

Перші два параметри, URL і назва вікна в прикладі залишені порожніми. У третьому параметрі задані ширина і висота вікна в пікселях.

**Завдання 1.** Створіть сайт, що складається з двох сторінок. Сайт служить для виведення таблиці значень тригонометричної функції ( $\sin$ ,  $\cos$  або  $\tg$ ) в заданому діапазоні і з заданим кроком. На першій сторінці (рис. 1) користувач задає вихідні дані, а на другий отримує відповідну таблицю. Вікно з новою сторінкою повинно відкриватися методом open () .

## ТРИГОНОМЕТРИЧЕСКИЕ ФУНКЦИИ

Функция .....

Введите диапазон углов в градусах

начало

конец

шаг ...

Рис.1

Вихідні дані повинні перевірятися відразу після введення і після натискання кнопки Обчислити. Таблиця повинна мати такий вигляд:

Кут		sin
в градусах	в радіанах	
30	0.5236	0.5
32	0.5585	0.5299
34	0.5934	0.5592
36	0.6283	0.5878

Для перевірки цифрових символів служить наступна функція:

```
function isNumb (fld)
{T = fld.value
 for (i = 0; i <t.length; i++)
 {Var ch = t.charAt (i)
 if ((ch < "0" || ch> "9") &! (ch == "," || ch == "."))
 {Alert ( "Не цифра") fld.focus (); fld.select ()
 return false
 }
 }
 return true
}
```

Параметр fld - об'єкт, значення якого перевіряється. Приклад звернення до функції isNumb (fld):

```
<INPUT TYPE = "text" NAME = "h" onchange = isNumb (this)>
```

Результат виконання оператора `ch = t.charAt (i)` - i-й символ строкової змінної (об'єкта) t. Функція повідомляє про помилку, якщо введений будь-який символ окрім десяткових цифр, крапки чи коми.

#### Управління таблицею

Для занесення даних в комірка таблиці служить вираз:

```
document.all.tab1.rows [i] .cells [j] .innerHTML = x  
// де tabl - id таблиці
```

Читання даних з комірки в браузері Internet Explorer симетрично занесенню:

```
x = document.all.tab1.rows [i] .cells [j] .innerHTML
```

Браузер Mozilla не сприймає цей оператор. Обидва браузера правильно виконують наступний оператор:

```
x = document.all.tab1.rows [i] .cells [3] .firstChild.nodeValue
```

Вміст комірки таблиці розглядається як дочірній вузол по відношенню до вузла, відповідного тегу <TD>. У прикладі витягаються дані з i-го осередки третього рядка таблиці, що має ID = tab1. Нумерація рядків і комірок починається з нуля. За допомогою сценарію можна міняти структуру таблиці, додаючи або видаляючи рядки та клітинки. Операція вставки рядка в таблицю складається з наступних етапів.

Виклик методу `insertRow ()` і збереження повернутого посилання на новий незаповнений рядок.

Використання посилання на рядок при виклику методу `insertCell ()` для кожного осередку в рядку (при цьому зберігається повернута посилання на кожну нову незаповнену клітинку).

Присвоєння значень властивостям осередки, включаючи її вміст.

Приклад додавання рядка і комірки:

```
NewRow = document.all.tab1.insertRow () // вставка рядка  
NewCell = NewRow.insertCell (0) // вставка осередки
```

Для видалення рядків і стовпців служать методи `deleteRow` (індекс рядка) і `deleteCell` (індекс комірки). Приклад видалення всіх рядків таблиці:

```
while (tableRef.rows.length > 0)  
{TableRef.deleteRow (0);}
```

**Завдання 2.** Додати сторінку (рис. 2) для обліку надходження товарів. Користувач може змінювати в таблиці кількість і ціну обраного товару, вводячи нові значення в поля, розташовані під таблицею. Введеними значеннями замінюються відповідні дані в таблиці і автоматично підраховується сумарна вартість всіх товарів.

### Учёт поступления товаров

Наименование товара	цена за единицу	количество	стоимость
Стол письменный	12000	5	60000
Стол кухонный	8000	10	80000
Стул	20	1200	12000
Шкаф книжный	14200	4	68800
ИТОГО			220800

Изменение цены и количества

Товар .....

Цена .....

Количество

Рис.2