

Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Сибирский государственный университет телекоммуникаций и информатики»
(СибГУТИ)

Кафедра прикладной математики и кибернетики

ОТЧЕТ
по лабораторной работе №2.4
по дисциплине «Интернет-технологии»

Выполнил:
ст. гр. ИВ-921
Ярошев Р.А.

Проверил:
доц. каф. ПмиК
Приставка П.А.

Оценка «_____»

Новосибирск 2020

Оглавление

Задание.....3

Выполнение.....4

Листинг.....7

Задание

Первый пример - это часы, отображающие время загрузки страницы (т.е. момент вызова скрипта). Используются методы объекта `Date` (`getHours`, `getMinutes`) и графические файлы, отображающие полученное время (имена этих файлов соответствуют цифрам 0-9).





Во втором примере, отображающем текущую дату, также используются методы объекта `Date` (`getDate`, `getMonth`, `getFullYear`) и графические файлы, отображающие полученную дату.

11:45 am 20.10.2020

Задание - сделать идущие часы - [см. пример в статус строке](#).

Картинки:

									
dg0.gif	dg1.gif	dg2.gif	dg3.gif	dg4.gif	dg5.gif	dg6.gif	dg7.gif	dg8.gif	dg9.gif

			
dgam.gif	dgpm.gif	dgc.gif	dgp.gif

Выполнение

Вступление

Поскольку необходимо реализовать «часы» и «дату», создадим для удобства два блока, в которых реализуем соответствующие функции:

```
function setClock() - часы
```

```
function setDate() - дата
```

В функциях «setClock» и «setDate» необходимо указать шаблон, по которому будет осуществляться поиск изображения:

```
var openImage = "<IMG SRC=\"dg"
```

```
var closeImage = ".gif\">"
```

объявив переменные open/closeImage, будем загружать изображения таким образом:

```
for (var i = 0; i < «var».length; ++i) {  
    text += openImage + «var».charAt(i) + closeImage  
}
```

Сравниваем значение i со значением метода на индексах от 0 до n-1.

Метод charAt() вернет, вычисленный в цикле, символ (i).

Функция setClock

Для получения текущего времени воспользуемся методами объекта Date «getHours» и «getMinutes»:

```
var now = new Date()
```

```
var hour = now.getHours()
```

```
var minute = now.getMinutes()
```

По умолчанию время задается в 24 — часовом формате.

В функции «setClock» необходимо определять время до полудня и после:

```
if (hour >= 12) {  
    ampm = "pm" //после полудня
```

```

    }
    else
        ampm = "am" //до полудня

```

далее, после загрузки изображений для часов, загружаем изображения dgpm/am.gif:

```
text += openImage + ampm + closeImage
```

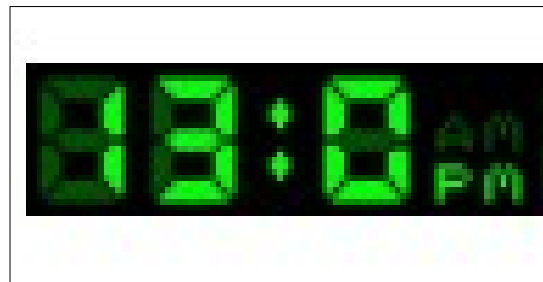
Важно указать условие для минут, когда их значение менее 10, то есть когда на нулевом индексе стоит 0 (01 минута, например):

```

if (minute < 10) {
    minute = "0" + minute
}

```

В противном случае вывод на экране будет осуществляться некорректно:



Функция setDate

Для получения текущей даты воспользуемся методами объекта Date «getDate» «getMonth» «getFullYear»:

```

var now = new Date()
var date = now.getDate()
var month = now.getMonth()
var year = now.getFullYear()

```

Важно указать условие для дня, когда его значение менее 10, то есть когда на нулевом индексе стоит 0 (01, например):

```
if (date < 10){
```

```
date = "0" + date  
}
```

Важно использовать FullYear для отображения года в 4 — символьном виде.

Загрузка изображений происходит методом, описанным во *вступлении*.

Функция rec

В этой функции происходит открытие документа для записи (document.open()), запись содержимого функций setClock и setDate (document.write(...)) и закрытие документа (document.close):

```
function rec(){  
    document.open()  
    document.write(setClock())  
    document.write(setDate())  
    document.close()  
}
```

Для того, чтобы часы «шли», вызовем метод setInterval (вызывает функцию регулярно через определённый интервал времени) с параметром rec и интервалом вызова этой функции 1000 мсек (1 сек).

Получилось



ЛИСТИНГ

Lab2.4.html

```
1 <!DOCTYPE HTML PUBLIC "-//IETF//DTD HTML 3.2//RU">
2 <HTML>
3
4 <HEAD>
5   <TITLE>Лабораторная 2.4</TITLE>
6   <META HTTP-EQUIV="Content-Type" CONTENT="text/html;
7 charset=UTF-8">
8 </HEAD>
9
10 <BODY>
11
12 <SCRIPT>
13
14   function setClock() {
15     var text = ""
16     var openImage = "<IMG SRC=\"dg"
17     var closeImage = ".gif\">"
18     var now = new Date()
19     var hour = now.getHours()
20     var minute = now.getMinutes()
21     var second = now.getSeconds()
22
23     if (hour >= 12) {
24       ampm = "pm" //после полудня
25     }
26     else
27       ampm = "am" //до полудня
28
29     hour += ""
30     for (var i = 0; i < hour.length; ++i) {
31       text += openImage + hour.charAt(i) + closeImage
32     }
33
34     text += openImage + "c.gif\">"
35
36     if (minute < 10) {
37       minute = "0" + minute
38     }
39   }
```

```

40
41     minute += ""
42     for (var i = 0; i < minute.length; ++i) {
43         text += openImage + minute.charAt(i) + closeImage
44     }
45
46     text += openImage + ampm + closeImage
47
48     return text
49 }
50 }
51
52 function setDate() {
53     var text = ""
54     var openImage = "<IMG SRC=\"dg"
55     var closeImage = ".gif\">"
56     var now = new Date()
57     var date = now.getDate()
58     var month = now.getMonth()
59     var year = now.getFullYear()
60
61     if (date < 10){
62         date = "0" + date
63     }
64
65     date += ""
66     for (var i = 0; i < date.length; ++i) {
67         text += openImage + date.charAt(i) + closeImage
68     }
69
70     text += openImage + "p.gif\">"
71
72     month++
73     month += ""
74     for (var i = 0; i < month.length; ++i) {
75         text += openImage + month.charAt(i) + closeImage
76     }
77
78     text += openImage + "p.gif\">"
79
80     year += ""
81     for (var i = 0; i < year.length; ++i) {
82         text += openImage + year.charAt(i) + closeImage

```



```
83     }
84
85     return text
86
87 }
88
89 function rec(){
90
91     document.open()
92     document.write(setClock())
93     document.write(setDate())
94     document.close()
95 }
93
97 setInterval(rec, 1000);
98
99 </SCRIPT>
100 </BODY>
```