МИНОБРНАУКИ РОССИИ

Федеральное государственное бюджетное образовательное учреждение высшего образования

НИЖЕГОРОДСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ

УНИВЕРСИТЕТ им. Р.Е.АЛЕКСЕЕВА

Институт радиоэлектроники и информационных технологий

Кафедра информатики и систем управления

\_\_\_\_\_\_\_\_ Лабораторная работа №2\_\_\_\_\_\_\_\_\_\_\_

(наименование темы проекта или работы)

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

к курсовой работе

(курсовому проекту, ОТЧЕТ по лабораторной работе)

по дисциплине

Информатика и компьютерные технологии \_

(наименование дисциплины)

РУКОВОДИТЕЛЬ:

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_Савкин А.Е.\_\_ \_

(подпись) (фамилия, и.,о.)

СТУДЕНТ:

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_Смирнов А. А.\_\_\_\_\_

(подпись) (фамилия, и.,о.)

\_\_\_24-ИСз\_\_\_\_\_\_\_\_\_\_\_

(шифр группы)

Работа защищена «\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_

С оценкой \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Задание к выполнению лабораторной работы № 2**

**Реализация вебсайта**

**Задание на лабораторную работу:**

Сделать вебсайт используя HTML, CSS, JS

**Цель работы:**

Научиться использовать современные технологии веб разработки.

**Ход работы:**

Идея: написать сайт с карточной игрой пасьянс Клондайк (Косынка).

Цель игры проста ­- разложить карты по мастям в порядке от туза до короля в четыре стопки. Карту можно перекладывать на другую рангом выше, но другого цвета . В каждую из четырёх стопок, по которым необходимо разложить все карты, сначала кладутся тузы, затем двойки, тройки и так далее до короля. Карты можно сдавать из оставшейся от раздачи колоды (в правом верхнем углу). В свободную ячейку игрового поля можно положить только короля. Игра заканчивается, когда все карты разложены.

Таким образом, стол делится на три поля: раздача колоды, игровое поле и стопки кард. Реализуем их через Canvas:

//canvas1

var ctx = document.getElementById("canvas1").getContext("2d");

ctx.fillStyle = '#c0d0f0';

ctx.strokeStyle = 'green';

ctx.fillRect(0,0,document.getElementById("canvas1").width,document.getElementById("canvas1").height);

for (var i=0;i<52;i++)

{

if ((cards[i].p1==-1)&&(cards[i].p2>=offs1)&&(cards[i].p2<=offs1+2))

ctx.drawImage(imgs[i],(cards[i].p2-offs1)\*dist\_W1,0);

}

ctx.drawImage(imgs2[0],3\*dist\_W1,0);

if (lastclickp == 1)

{

ctx.strokeRect(lastclickn\*dist\_W1,5,dist\_W1,190);

}

//canvas2

ctx = document.getElementById("canvas2").getContext("2d");

ctx.fillStyle = '#c0d0f0';

ctx.strokeStyle = 'green';

ctx.fillRect(0,0,document.getElementById("canvas2").width,document.getElementById("canvas2").height);

for (var i=0;i<52;i++)

{

var p1 = cards[i].p1, p2 = cards[i].p2;

if ((p1>=1)&&(p1<=7)&&(p2>0))

ctx.drawImage(imgs2[0],

0,10,card\_W,10,

(cards[i].p1-1)\*dist\_W1,70-10\*cards[i].p2,card\_W,10);

if ((p1>=1)&&(p1<=7)&&(p2>fieldcardco[1][p1-1])&&(p2<=0))

ctx.drawImage(imgs[i],

0,0,card\_W,30,

(p1-1)\*dist\_W1,70-30\*p2,card\_W,30);

if ((p1>=1)&&(p1<=7)&&(p2<=fieldcardco[1][p1-1]))

ctx.drawImage(imgs[i],

0,0,card\_W,card\_H,

(p1-1)\*dist\_W1,70-30\*p2,card\_W,card\_H);

}

if (lastclickp == 2)

ctx.strokeRect((lastclickn-1)\*dist\_W1,75,card\_W,190-30\*fieldcardco[1][lastclickn-1]);

//canvas3

ctx = document.getElementById("canvas3").getContext("2d");

ctx.fillStyle = '#c0d0f0';

ctx.strokeStyle = 'green';

ctx.fillRect(0,0,document.getElementById("canvas3").width,document.getElementById("canvas3").height);

for (var i=0;i<=3;i++)

{

if (upcardnum[i]==0)

ctx.drawImage(imgs2[0],i\*dist\_W1,0);

else

ctx.drawImage(imgs[i\*13+upcardnum[i]-1],i\*dist\_W1,0);

}

if (lastclickp == 3)

ctx.strokeRect(lastclickn\*dist\_W1,5,dist\_W1,190);

Карты отличаются значением и мастью. Введем карту переменной, имеющей дополнительные параметры: цвет карты, масть, значение, позиция линии, позиция в линии игрового поля :

var tmpcard = {m1:0,m2:0,v:0,p1:0,p2:0};

Задаем игровые условия

var upcardnum = [0,0,0,0]; - Массив количества карт в стопке дома

var fieldcardco = [[0,1,2,3,4,5,6],[0,0,0,0,0,0,0]]; - количество карт на начало игры и минимальное количество карт в игровом поле

Перед началом игры мы должны задать карты и разложить их по местам, сначала инициализируем карты:

function InitCards()

{

for (var i=0;i<=3;i++)

{

for (var j=1;j<=13;j++)

{

var tmp = {m1:Math.floor(i/2),m2:i % 2,v:j, p1:-10, p2:-10};

cards.push(tmp);

}

}

}

Далее мы задаем каждой карте место:

function MakeCardPlace()

{

//1 random values p1

for (var i=0;i<52;i++)

{

var loopi = 0;

do

{

num = Math.floor(52\*Math.random());

if (cards[num].p1 == -10)

cards[num].p1 = i;

else num = -1;

loopi++;

}

while ((num==-1)&&(loopi<1000))

}

//2 make places

for (var i=0;i<52;i++)

{

if ((cards[i].p1>=0)&&(cards[i].p1<=28-1))

{

var j=1, tmp = cards[i].p1;//27-1-2-3-4-5-6-7

do

{

tmp-=j;

j++;

}

while ((j<=7)&&(tmp>=0));

j--;

tmp+=j;

cards[i].p1 = j;

cards[i].p2 = tmp;

}

if ((cards[i].p1>=28)&&(cards[i].p1<=52-1))

{

cards[i].p2 = cards[i].p1 - 28;

cards[i].p1 = -1;

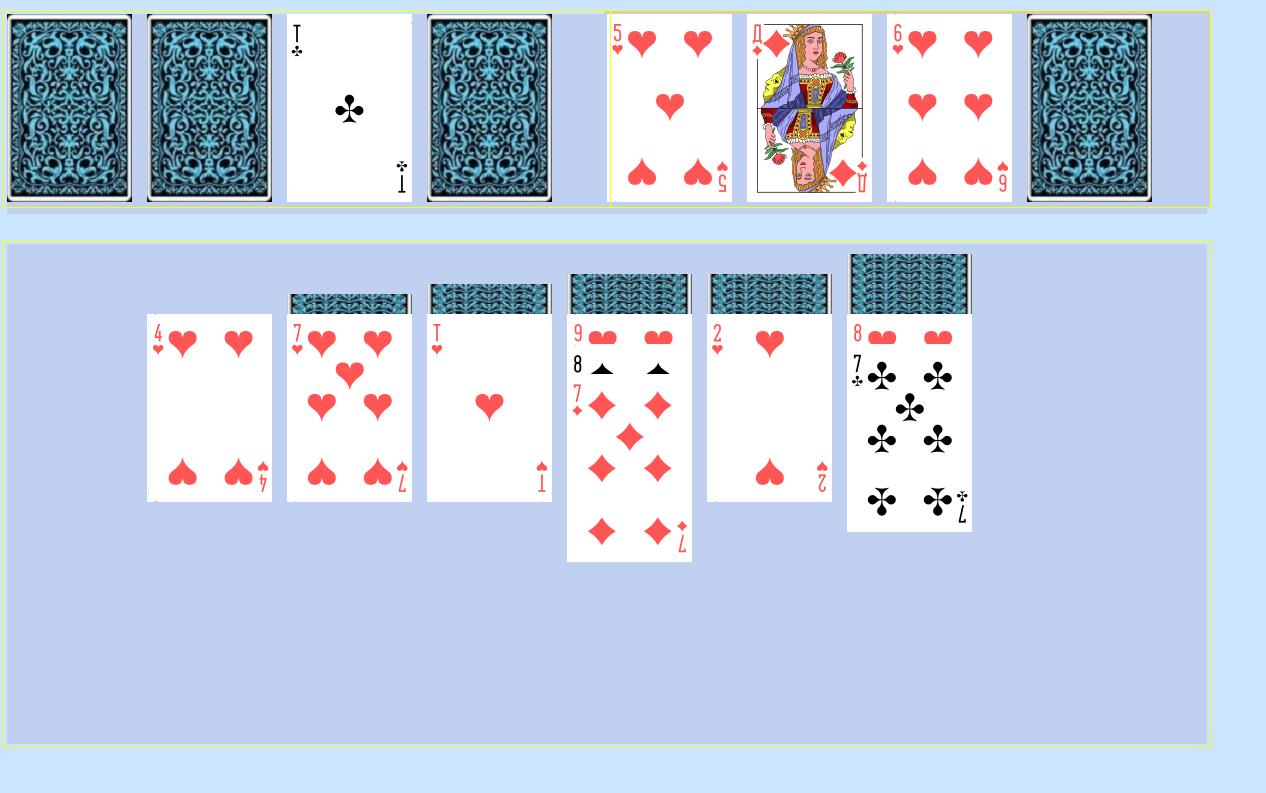
}

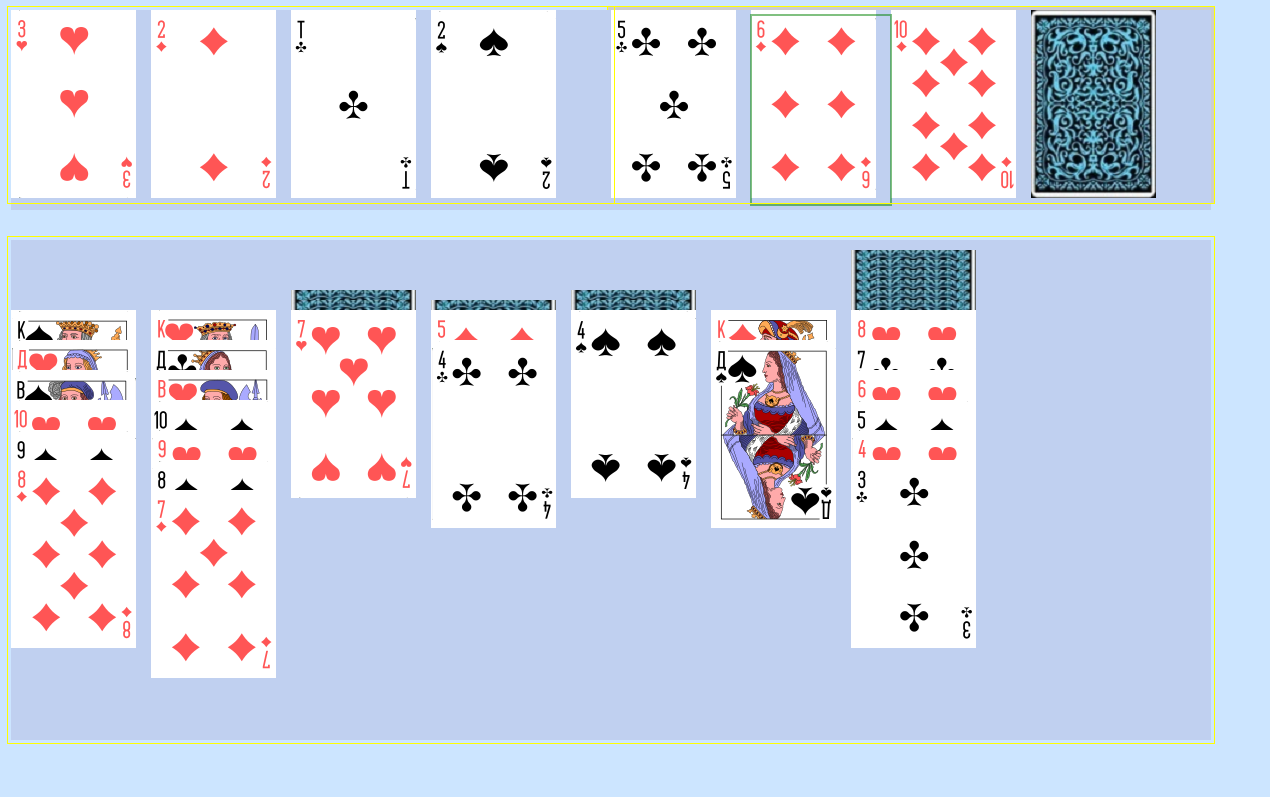
}

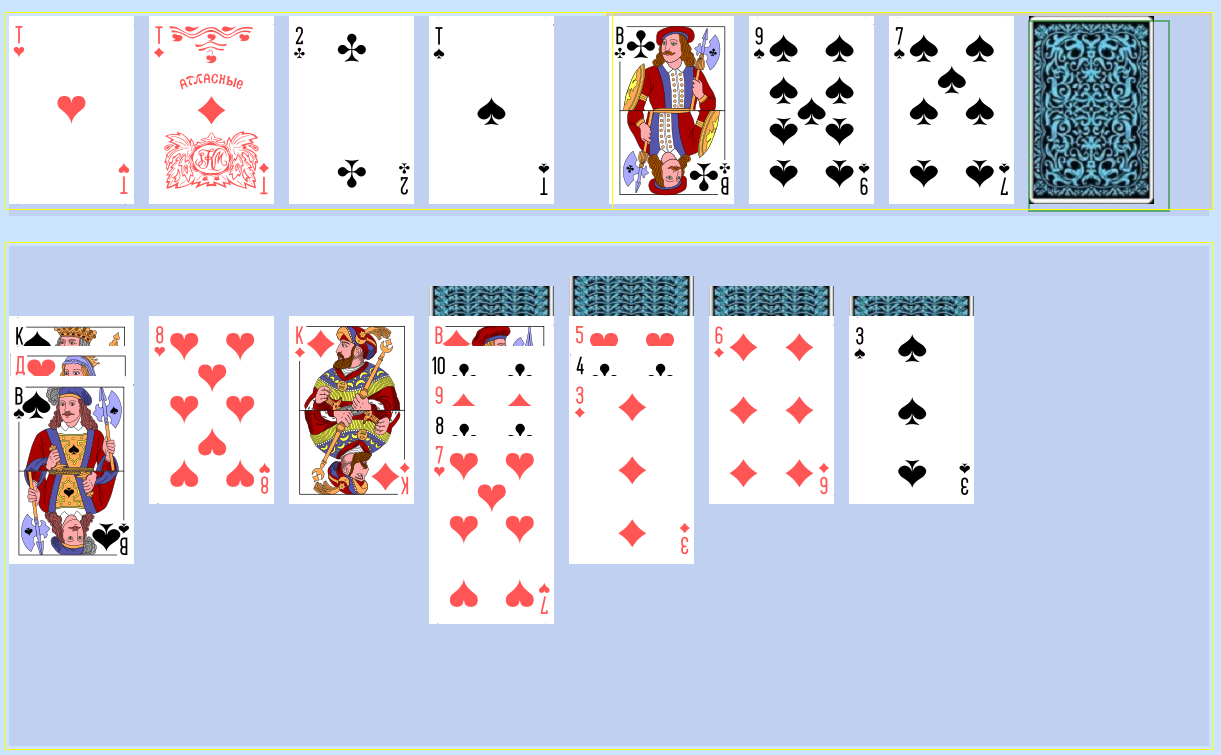
}

В самой игре карта может быть перемещена только на карту, имеющую значение выше и обозначенную другим цветом, карта может перемещаться из по всему столу, но не может возвращаться в колоду, обозначив колоду, игровой стол и стопку зонами 1, 2 и 3 соответственно, задаем перемещение между зонами с соответствующими проверками.









**Вывод:**

В этой лабораторной работе мы научились проектировать и создавать веб-сайты при помощи HTML, CSS и JS.