**Московский государственный технический**

**Университет им. Н.Э. Баумана**

**Факультет «Информатика и системы управления»**

**Кафедра ИУ5 «Системы обработки информации и управления»**

Курс «Парадигмы и конструкции языков программирования»

Отчет по Домашнему заданию

«Создание браузерной игры на языке программирования JavaScript»

Выполнил:

студент группы ИУ5Ц-52Б

Мефодьев Илья

Проверил:

Нардид А.Н.

\

2024 г.

**Задание**

Создать браузерную игру, используя стандартные инструменты разработки веб-приложений. Тема игры – Мультфильм «Король лев».

Будет плюсом создание многофайлового проекта.

**Текст программы**

Файл index.html:

*<!--index.html-->*

<!DOCTYPE *html*>

<html *lang*="ru">

    <head>

        <meta *charset*="utf-8">

        <link *rel*="stylesheet" *type*="text/css" *href*="style.css">

        <title>Game King Lion</title>

    </head>

<body>

    <div *class*="container">

        <div *class*="menu" *id*="menu">

            <form *name*="forma" *id*="forma">

                <input *type*="text" *name*="text" *class*="text" *id*="text" *oninput*="valid()" *onChange*="valid()" *placeholder*="введите никнейм">

*<!--<input type="submit" name="submitt" class="submitt" id="submitt" onClick="start()">-->*

                <p>Выберите Игрока</p>

                <div *class*="df"><input *type*="radio" *name*="pers" *id*="Timon" *value*="Timon"><p>Тимон</p></div>

                <div *class*="df"><input *type*="radio" *name*="pers" *id*="Pumba" *value*="Pumba"><p>Пумба</p></div>

            </form>

            <div *class*="submitt1" *id*="submitt1" *name*="submitt" *onClick*="start()">

                <p>start game</p>

            </div>

*<!--div class="instruction">*

*new instruction*

*</div-->*

        </div>

        <canvas *id*="game" *name*="game" *class*="game">not Defined</canvas>

    </div>

    <script *type*="text/javascript" *src*="script.js"></script>

</body>

</html>

Файл style.css

*/\* style.css \*/*

\*{

    margin:0 auto;

}

*.container*{

    width:100%;

    height:100%;

}

*.menu*{

    width:400px;

    height:500px;

    background:#737df1;

    margin-top:200px;;

}

*.text*{

    width:200px;

    height:20px;

    margin-top:50px;

}

*#forma*{

    text-align: center;

}

*.df*{

    display: flex;

    margin: 0 auto;

    width: 100px;

}

*.submitt1*{

    width:100px;

    height:40px;

    text-align:center;

    background:url("Media/red.png");

    margin-top:10px;

    margin-left:150px;

}

*.submitt2*{

    width:100px;

    height:40px;

    text-align:center;

    background:red;

    margin-top:10px;

    margin-left:150px;

}

*.instruction*{

    width:350px;

    height:200px;

    font: 20px arial;

}

*.invis*{

    display: none;

}

*#pers*{

    width:100px;

    height:100px;

    padding-left:2000px;

    margin-left:2000px;

}

*#nik*{

    padding-top: 20px;

    padding-left: 100px;

}

Файл script.js

*// script.js*

var canvas=document.getElementById('game');

var ctx=canvas.getContext('2d');

var leftPressed=false;

var rightPressed=false;

var isEnemy=false;

var mobHp;

var power;

var thisEnemy;

var interval;

var timerInterval;

var ss=0;

var mm=0;

var upPressed=false;

var pausePressed=false;

var downPressed=false;

var name;

var holmX=randomInt(0, 1500);

var holmX1=randomInt(0, 1500);

var isGuz=true;

var isGuz1=true;

var score=0;

var pers;

var persona;

*//var audio = new Audio("Media/Music/fon.mp3");*

var guzEat=0;

document.addEventListener('keydown', keyDownHandler, false);

document.addEventListener('keyup', keyUpHandler, false);

document.addEventListener("keydown", keyDownHandler, false);

document.addEventListener("keyup", keyUpHandler, false);

function keyDownHandler(e){

    if(e.keyCode==37){

        leftPressed=true;

    }

    if(e.keyCode==39){

        rightPressed=true;

    }

    if(e.keyCode==38){

        upPressed=true;

    }

    if(e.keyCode==27){

        pausePressed=true;

    }

    if(e.keyCode==40){

        downPressed=true;

    }

}

function keyUpHandler(e){

    if(e.keyCode==37){

        leftPressed=false;

    }

    if(e.keyCode==39){

        rightPressed=false;

    }

    if(e.keyCode==38){

        upPressed=false;

    }

    if(e.keyCode==40){

        downPressed=false;

    }

}

function randomInt(min, max){

    let rand=min+Math.random()\*(max+1-min);

    return Math.floor(rand);

}

function backGround(){

    this.x=0;

    this.y=0;

    this.bgImg=new Image();

    this.bgImg.src="Media/fones/fon.png";

    this.draw=function(){

        ctx.beginPath();

        ctx.drawImage(this.bgImg, this.x, this.y, 5000, canvas.height);

        ctx.closePath();

        return;

    }

}

function player(){

    this.hp=100;

    this.mp=100;

    this.x=100;

    this.y=400;

    this.p=new Image();

    this.p.src=pers;

    this.draw=function(){

        ctx.beginPath();

        ctx.drawImage(this.p, this.x, this.y, 100, 100);

        ctx.closePath();

    }

    persona=this.p;

}

function enemy(mobHp,power){

    this.hp=mobHp;

    this.power=power;

    this.x=0;

    this.y=400;

    this.p=new Image();

    this.p.src="Media/Huena/huena.png";

    this.spawn=function(){

        let random=randomInt(0, canvas.width);

        this.x=random;

    }

    this.draw=function(){

        if(this.x<thisPlayer.x-10){

            this.x+=4;

        }else if(this.x>thisPlayer.x+10){

            this.x-=4;

        }else{

            if(this.y===thisPlayer.y){

                thisPlayer.hp-=this.power;

            }

        }

        ctx.beginPath();

        ctx.drawImage(this.p, this.x, this.y, 100, 100)

        ctx.closePath();

    }

}

function holm(){

    this.y=400;

    this.x=holmX;

    this.p=new Image();

    this.p.src="Media/guz/holm.png";

    this.draw=function(){

        ctx.beginPath();

        ctx.drawImage(this.p, this.x, this.y, 240, 100);

        ctx.closePath();

    }

}

function holm1(){

    this.y=400;

    this.x=holmX1;

    this.p=new Image();

    this.p.src="Media/guz/holm.png";

    this.draw=function(){

        ctx.beginPath();

        ctx.drawImage(this.p, this.x, this.y, 240, 100);

        ctx.closePath();

    }

}

function guz(){

    this.y=350;

    this.x=holmX+100;

    this.p=new Image();

    this.p.src="Media/guz/guz.png";

    this.draw=function(){

        ctx.beginPath();

        ctx.drawImage(this.p, this.x, this.y, 50, 50);

        ctx.closePath();

    }

}

function guz1(){

    this.y=350;

    this.x=holmX1+100;

    this.p=new Image();

    this.p.src="Media/guz/guz.png";

    this.draw=function(){

        ctx.beginPath();

        ctx.drawImage(this.p, this.x, this.y, 50, 50);

        ctx.closePath();

    }

}

function start(){

    if(document.getElementById('text').value==""){

        alert('введите никнейм');

    }else{

        name=document.getElementById('text').value;

        if(document.getElementById('Timon').checked){

            pers="Media/Timon/p.png";

        }else if(document.getElementById('Pumba').checked){

            pers="Media/Pumba/Pumba.png";

        }else{

            alert("Выбери перса");

            location.reload();

        }

        document.getElementById('menu').style.display="none";

        canvas.style.display="block";

        canvas.height=document.documentElement.clientHeight;

        canvas.width=document.documentElement.clientWidth;

        interval=setInterval(draw, 30);

        timerInterval=setInterval(timerr, 1000);

        window.thisPlayer=new player();

        window.thisBackGround=new backGround();

        window.thisHolm=new holm();

        window.thisHolm1=new holm1();

        window.thisGuz=new guz();

        window.thisGuz1=new guz1();

*//audio.play();*

    }

}

function valid(){

    if(document.getElementById("text").value[0]==undefined){

        document.getElementById("submitt1").style.background='url("Media/red.png");';

        document.getElementById("submitt1").style.cursor="default";

    }else{

        document.getElementById("submitt1").style.background='url("Media/green.png");';

        document.getElementById("submitt1").style.cursor="pointer";

    }

}

function timerr(){

    ss+=1;

    if(ss==60){

        ss=0;

        mm+=1;

    }

    thisPlayer.hp-=1;

}

function drawTimer(){

    if(ss<10){

        drawSs="0"+ss;

    }else{

        drawSs=ss;

    }

    ctx.beginPath();

    ctx.font="30px arial white";

    ctx.fillText("0"+mm+":"+drawSs, 100, canvas.height-100);

    ctx.closePath();

}

function gameOver(){

    clearInterval(interval);

    clearInterval(timerInterval);

    canvas.style.display="none";

    document.getElementById("menu").style.display="block";

    canvas=document.getElementById('game');

    ctx=canvas.getContext('2d');

    leftPressed=false;

    rightPressed=false;

    isEnemy=false;

*// mobHp;*

*// power;*

*// thisEnemy;*

    upPressed=false;

    pausePressed=false;

    downPressed=false;

    holmX=randomInt(0, 1500);

    holmX1=randomInt(0, 1500);

    isGuz=true;

    isGuz1=true;

    document.getElementById('invis').style.display="block";

    document.getElementById('text1').value=name;

    document.getElementById('num').value=score;

    name="";

    score=0;

*//audio.pause();*

*//audio.currentTime = 0;*

    guzEat=0;

}

window.onresize=function(){

    canvas.height=document.documentElement.clientHeight;

    canvas.width=document.documentElement.clientWidth;

}

*//thisPlayer.y<301 помогло!*

function tryDown(){

    if (thisPlayer.x+100>thisHolm.x && thisPlayer.x<thisHolm.x+240 && thisPlayer.y<301 ||

    thisPlayer.x+100>thisHolm1.x && thisPlayer.x<thisHolm1.x+240 && thisPlayer.y<301){

        thisPlayer.y=300;

    }else{

        thisPlayer.y=400;

        clearInterval(window.tryDownInt);

    }

}

function tryUp(){

    if(!downPressed){

        thisPlayer.y=400;

    }else{

        thisPlayer.y=500;

    }

}

function draw(){

    ctx.clearRect(0, 0, canvas.width, canvas.height);

    thisBackGround.draw();

    drawUI();

    thisHolm.draw();

    thisHolm1.draw();

    if(isGuz){

        thisGuz.draw();

        if (thisPlayer.x+10>thisGuz.x && thisPlayer.x<thisGuz.x+10){

            isGuz=false;

            guzEat+=1;

            score++;

            thisPlayer.hp+=20;

*//alert(thisPlayer.hp);*

        }

    }

    if(isGuz1){

        thisGuz1.draw();

        if (thisPlayer.x+10>thisGuz1.x && thisPlayer.x<thisGuz1.x+10){

            isGuz1=false;

            guzEat+=1;

            score++;

            thisPlayer.hp+=20;

        }

    }

    if(thisPlayer.hp<=0){

        gameOver();

    }

    if(thisPlayer.x>=canvas.width){

        gameOver();

    }

    if (!isEnemy){

        thisEnemy=new enemy(20,1);*//30*

        thisEnemy.spawn();

        isEnemy=true;

    }else{

        thisEnemy.draw();

        if(thisEnemy.hp<=0){

            isEnemy=false;

        }

    }

    if(leftPressed){

        if(!rightPressed && !upPressed  && !downPressed){

            thisPlayer.x-=5;

        }else{

            leftPressed=true;

            rightPressed=false;

            upPressed=false;

            downPressed=false;

        }

    }

    if(rightPressed){

        if(!leftPressed && !upPressed && !downPressed){

            thisPlayer.x+=5;

        }else{

            rightPressed=true;

            leftPressed=false;

            upPressed=false;

            downPressed=false;

        }

    }

    if(downPressed){

        if(!leftPressed && !upPressed && !rightPressed){

            thisPlayer.y=500;

*//thisPlayer.p="";*

            window.tryUpInt=setInterval(tryUp, 20);

        }else{

            downPressed=true;

            leftPressed=false;

            upPressed=false;

            rightPressed=false;

        }

    }else{

*//thisPlayer.y=400;*

*//thisPlayer.p=pers;*

*//downPressed=false;*

    }

    if(upPressed){

        if(!downPressed && !rightPressed && !leftPressed){

            thisPlayer.y=300;

            window.tryDownInt=setInterval(tryDown, 20);

        }

    }

    if(pausePressed){

        confirm('продолжить?');

        pausePressed=false;

    }

    function drawUI(){

        ctx.beginPath();

        ctx.rect(0, 0, thisPlayer.hp\*2, 40);

        ctx.fillStyle="red";

        ctx.font="40px Arial";

        ctx.fillText("HP", 0, 75);

        ctx.fill();

        ctx.closePath();

        ctx.beginPath();

        ctx.fillStyle="black";

        ctx.font="60px Arial";

        ctx.fillText(name, 700, 60);

        ctx.fill();

        ctx.closePath();

        ctx.beginPath();

        ctx.font="40px Arial";

        ctx.fillText(guzEat, 400, 60);

        ctx.fill();

        ctx.closePath();

    }

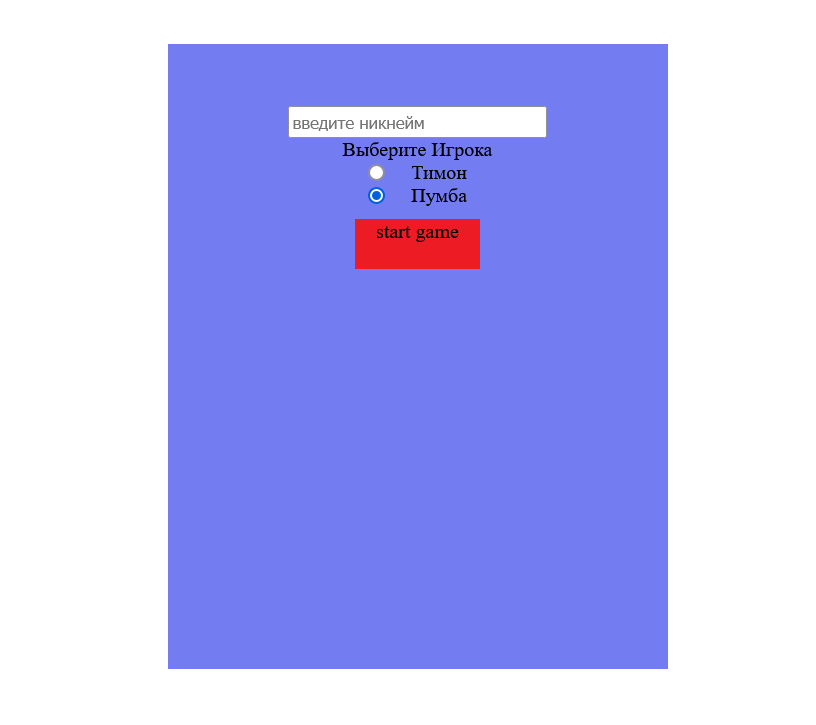
    drawTimer();

    thisPlayer.draw();

}

**Описания и экранные формы**

При запуске программы пользователь попадает на меню входа в игру. В поле ввода игрок вводит свой никнейм, после чего выбирает персонажа и запускает игру. Если же никнейм не был введен – выводится сообщение об ошибке.



Игровой процесс реализован следующим образом: Игрок при помощи клавиш со стрелками перемещается влево и вправо по игровому полю. Цель – дойти до правого края экрана. При этом враг – гиена – будет мешать. Холмы являются безопасной зоной, на них можно забраться, и гиена не сможет наносить игроку урон. Гусеницы на холмах позволяют восстанавливать запас здоровья.

