Операционные системы

Отчет по лабораторной работе № 2

Цель работы

1) Изучение объектов WSH.

2) Изучение приемов программирования с использованием объектов WSH.

3) Получение навыков разработки сценариев.

Примеры

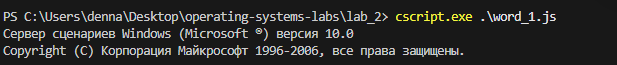
Пример 1

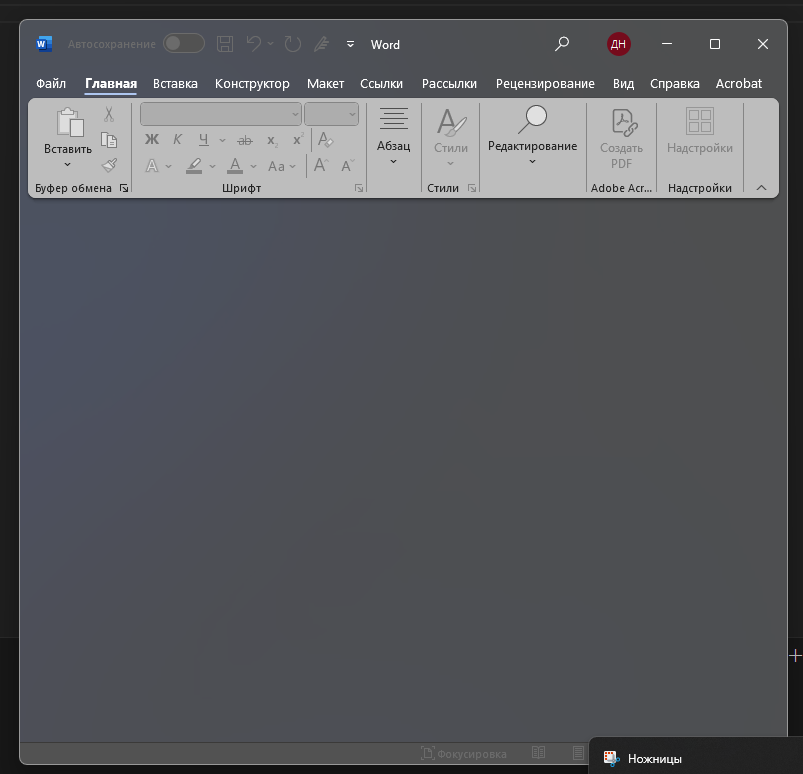
**Листинг программы:**

var wdApp = new ActiveXObject("Word.Application");

wdApp.Visible = true;

**Пример работы программы:**

****

****

Пример 2

**Листинг программы:**

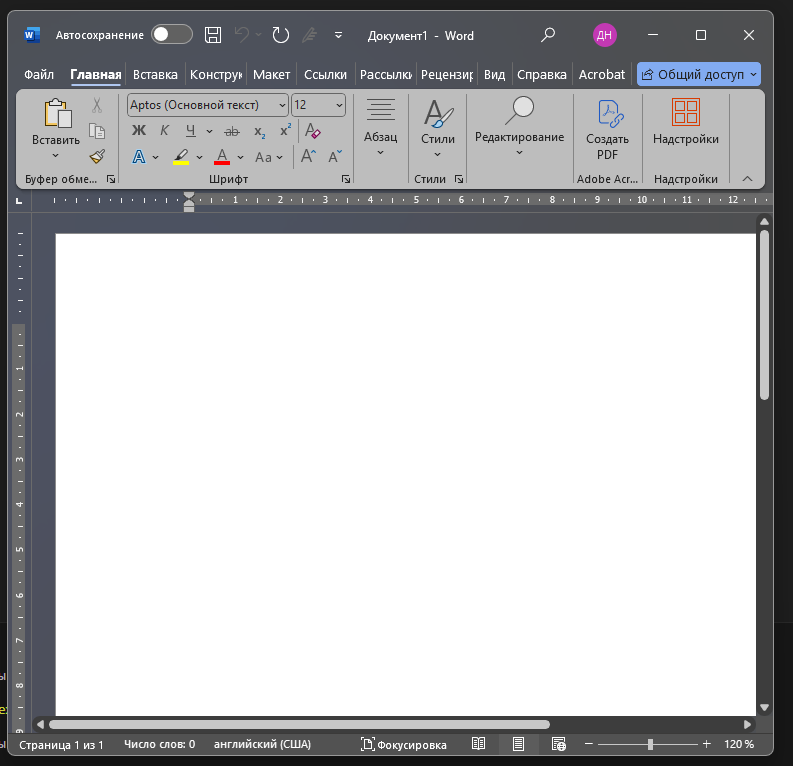
var wdApp = new ActiveXObject("Word.Application");

wdApp.Visible = true;

wdApp.Documents.Add()

**Пример работы программы:**

****

****

Пример 3

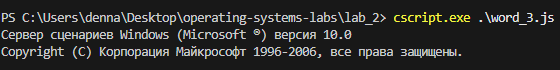
**Листинг программы:**

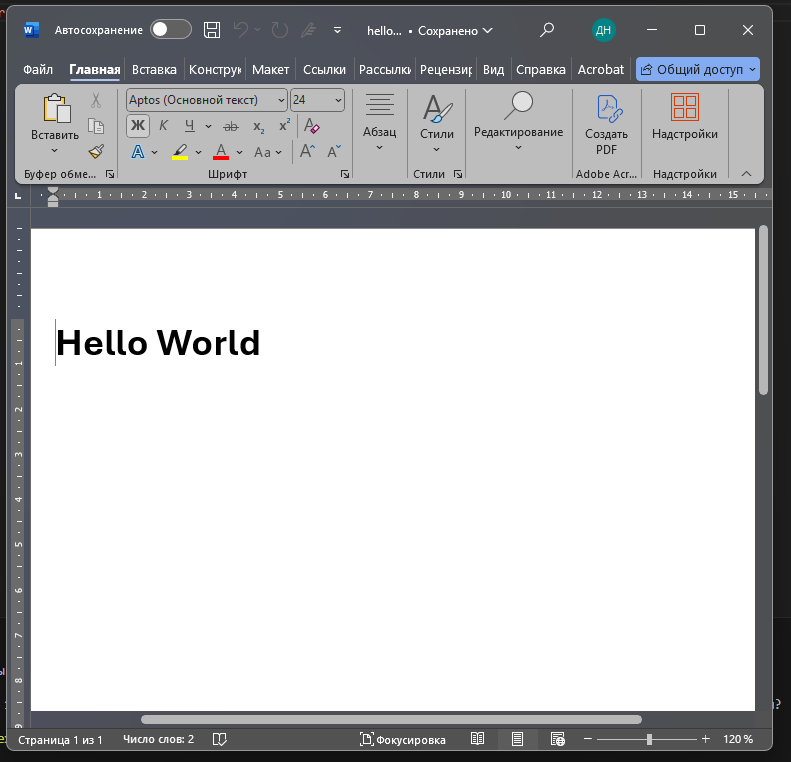
var wdApp = new ActiveXObject("Word.Application");

wdApp.Visible = true;

wdApp.Documents.Open("C:\\Users\\denna\\Desktop\\operating-systems-labs\\lab\_2\\hello\_world.docx")

**Пример работы программы:**

****

****

Задания

Задание 1

Разработать процедуру расчета суммы n первых членов ряда, в соответствии с индивидуальным заданием. Организовать ввод исходных данных в процедуру с помощью параметров командной строки при вызове процедуры. 

**Листинг программы:**

var sum = 0;

var args = WScript.Arguments;

if (args.length < 2) {

    WScript.Echo("Использование: cscript script.js <n> <x>");

    WScript.Quit(1);

}

var n = parseInt(args(0));

var x = parseFloat(args(1));

if (isNaN(n) || n < 1 || n % 1 !== 0) {

    WScript.Echo("Ошибка: n должно быть положительным целым числом.");

    WScript.Quit(1);

}

for (var k = 1; k <= n; k++) {

    var factKMinus1 = 1;

    for (var i = 2; i <= k - 1; i++) {

        factKMinus1 \*= i;

    }

    var fact2K = 1;

    for (var i = 2; i <= 2 \* k; i++) {

        fact2K \*= i;

    }

    var power2Kx = 1;

    for (var i = 0; i < 2 \* k; i++) {

        power2Kx \*= x;

    }

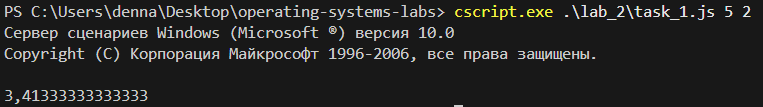
    var term = (factKMinus1 \* factKMinus1 / fact2K) \* power2Kx;

    sum += term;

}

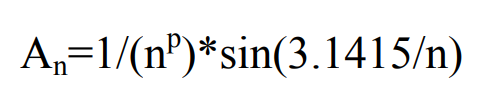
WScript.Echo(sum);

**Пример работы программы:**



Задание 2

Разработать процедуру расчета суммы n первых членов ряда, в соответствии с индивидуальным заданием. Организовать ввод исходных данных в процедуру с помощью параметров командной строки при вызове процедуры.

****

**Листинг программы:**

function calculateAn(n, p) {

    return 1 / (Math.pow(n, p) \* Math.sin(3.1415 / n));

}

var sum = 0;

var args = WScript.Arguments;

if (args.length !== 2) {

    WScript.Echo("Использование: cscript script.js <n> <p>");

    WScript.Quit(1);

}

var n = parseInt(args(0));

var p = parseFloat(args(1));

if (isNaN(n) || n < 1 || n % 1 !== 0) {

    WScript.Echo("Ошибка: n должно быть положительным целым числом.");

    WScript.Quit(1);

}

for (var k = 1; k <= n; k++) {

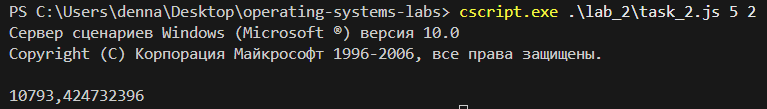
    var term = calculateAn(k, p);

    sum += term;

}

WScript.Echo(sum);

**Пример работы программы:**

****

Задание 3

Задан исходный текст на русском языке. Длина текста - не более NL символов, длина строки - не более NS символов, длина слова – не более NW символов. Исходный текст должен заканчиваться точкой. Составить процедуру, которая в заданном тексте находит слова, оканчиваются заданной буквы. В качестве результата вывести исходный текст, найденные слова и их количество.

**Листинг программы:**

function checkTextLengthValidity(text, maxLength) {

    return text.length <= maxLength;

}

function checkTextEndsWithDot(text) {

    return text.charAt(text.length - 1) === ".";

}

function checkLineLength(line, maxLineLength) {

    return line.length <= maxLineLength;

}

function trimLine(line) {

    var start = 0;

    var end = line.length - 1;

    while (start <= end && line.charAt(start) === " ") start++;

    while (end >= start && line.charAt(end) === " ") end--;

    return line.substring(start, end + 1);

}

function splitIntoWords(trimmedLine) {

    return trimmedLine.split(" ");

}

function isValidWord(word, maxWordLength, targetLetter) {

    if (word.length === 0) return false;

    var wordToCheck = removePunctuationFromEnd(word).toLowerCase();

    targetLetter = targetLetter.toLowerCase();

    if (wordToCheck.length > maxWordLength) return false;

    if (wordToCheck.length === 0) return false;

    return wordToCheck.charAt(wordToCheck.length - 1) === targetLetter;

}

function getCleanWord(word) {

    return removePunctuationFromEnd(word);

}

function findWordsEndingWithLetter(line, maxWordLength, targetLetter) {

    var trimmedLine = trimLine(line);

    var words = splitIntoWords(trimmedLine);

    var foundWords = [];

    for (var j = 0; j < words.length; j++) {

        if (isValidWord(words[j], maxWordLength, targetLetter)) {

            var wordToAdd = getCleanWord(words[j]);

            foundWords.push(wordToAdd);

        }

    }

    return foundWords;

}

function isPunctuation(char) {

    return char === "." || char === "," || char === "!" || char === "?" || char === ";";

}

function removePunctuationFromEnd(word) {

    var cleanedWord = word;

    while (cleanedWord.length > 0 && isPunctuation(cleanedWord.charAt(cleanedWord.length - 1))) {

        cleanedWord = cleanedWord.substring(0, cleanedWord.length - 1);

    }

    return cleanedWord;

}

var args = WScript.Arguments;

if (args.length !== 5) {

    WScript.Echo("Использование: cscript script.js <text> <maxLength> <maxLineLength> <maxWordLength> <targetLetter>");

    WScript.Quit(1);

}

var text = args(0);

var maxLength = parseInt(args(1));

var maxLineLength = parseInt(args(2));

var maxWordLength = parseInt(args(3));

var targetLetter = args(4);

if (isNaN(maxLength) || maxLength <= 0 || maxLength % 1 !== 0) {

    WScript.Echo("Ошибка: maxLength должно быть положительным целым числом.");

    WScript.Quit(1);

}

if (isNaN(maxLineLength) || maxLineLength <= 0 || maxLineLength % 1 !== 0) {

    WScript.Echo("Ошибка: maxLineLength должно быть положительным целым числом.");

    WScript.Quit(1);

}

if (isNaN(maxWordLength) || maxWordLength <= 0 || maxWordLength % 1 !== 0) {

    WScript.Echo("Ошибка: maxWordLength должно быть положительным целым числом.");

    WScript.Quit(1);

}

if (targetLetter.length !== 1) {

    WScript.Echo("Ошибка: targetLetter должно быть одной буквой.");

    WScript.Quit(1);

}

if (!checkTextLengthValidity(text, maxLength)) {

    WScript.Echo("Ошибка: текст превышает NL символов.");

    WScript.Quit(1);

}

if (!checkTextEndsWithDot(text)) {

    WScript.Echo("Ошибка: текст не заканчивается точкой.");

    WScript.Quit(1);

}

var lines = text.split("\n");

var foundWords = [];

var wordCount = 0;

for (var i = 0; i < lines.length; i++) {

    if (!checkLineLength(lines[i], maxLineLength)) {

        WScript.Echo("Ошибка: строка " + (i + 1) + " превышает NS символов.");

        WScript.Quit(1);

    }

    var words = findWordsEndingWithLetter(lines[i], maxWordLength, targetLetter);

    foundWords = foundWords.concat(words);

    wordCount += words.length;

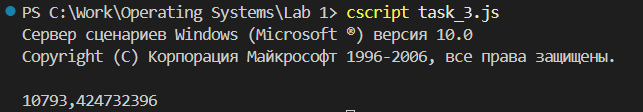
}

WScript.Echo("Исходный текст:\n" + text);

WScript.Echo("Найденные слова: " + foundWords.join(", "));

WScript.Echo("Количество слов: " + wordCount);

**Пример работы программы:**

****