**Реферат на тему «Язык скриптовых сценариев WSH»**

**Назначение и основные свойства WSH**

WSH представляет минимальные требования к объему ОП, а также является удобным инструментов для автоматизации задач юзеров и админов ОС.

С помощью сценариев WSH можно работать с файловой системой ПК, а также управлять работой других приложений, при этом возможности сценариев ограничены средствами, которые предоставляют доступные серверы автоматизации.

Наиболее очевидные задачи, для автоматизации которых подходят сценарии WSH:

* организация резервного копирования на сетевой сервер файлов с локальной машины
* быстрое изменение конфигурации рабочего стола windows
* автоматический запуск программ microsoft office, создание документов, распечатка этих документов и закрытие приложений
* подключение и отключение сетевых ресурсов
* выполнение задач администрирования локальной сети

**Создание и запуск простейших сценариев WSH**

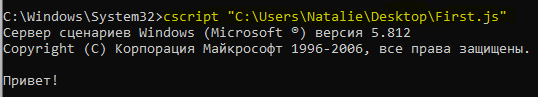
Простейший сценарий WSH – обычный текстовый файл с расширением выбранного языка (js, vbs), создать который можно в любом текстовом редакторе, способном сохранять документы в формате «только текст». Размер сценария может изменятся вплоть до тысяч строк, а предельный размер ограничивается максимальным размеров файла в соответствующей файловой системе.

Создадим сценарий, который выводит диалоговое окно «Привет!». Для этого откроем блокнот и создадим файл First.js, который будет содержать одну строку:

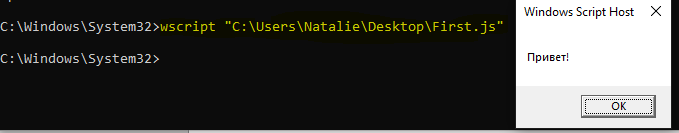
WScript.Echo(“Привет!”);

Запуск сценариев:

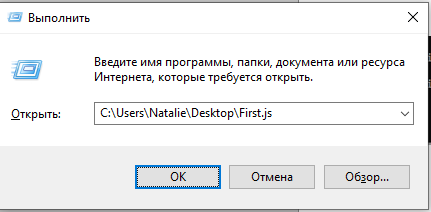
1. из командной строки в консольном режиме

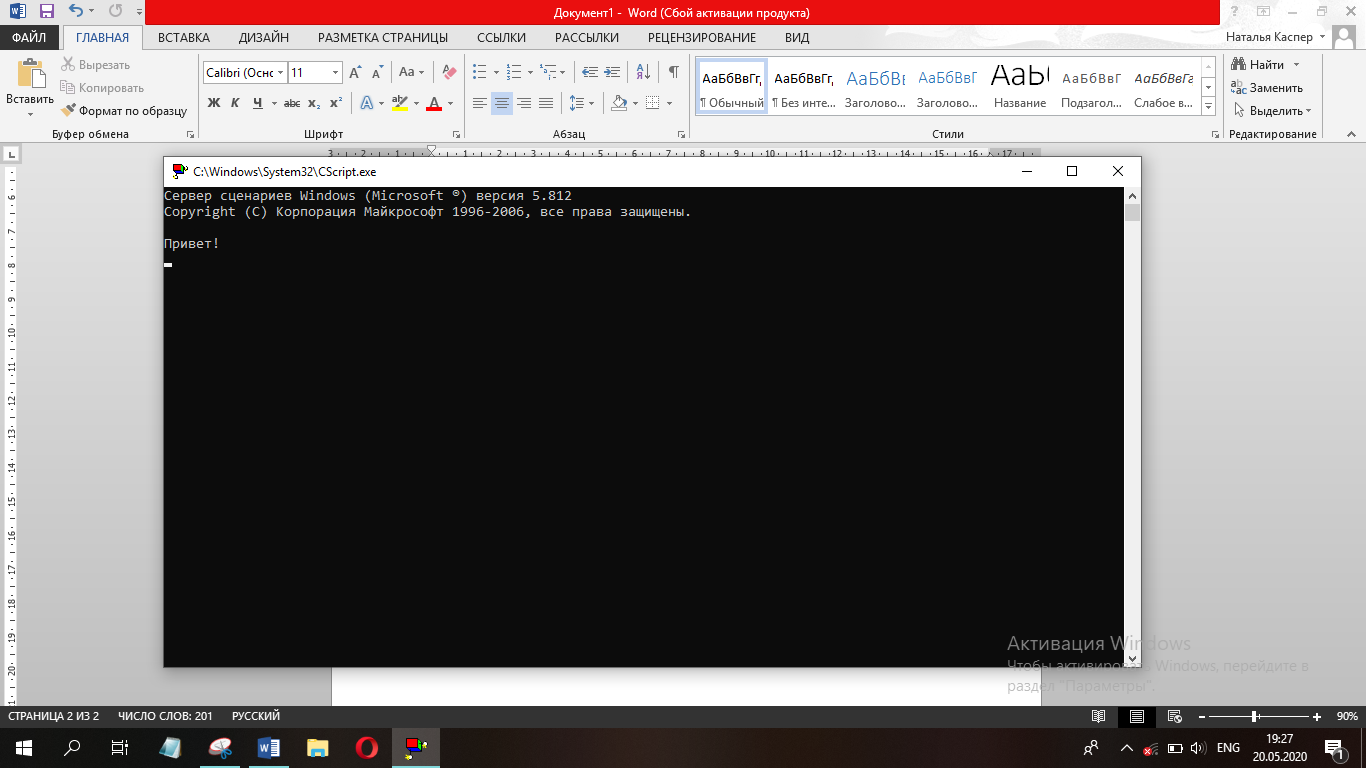


1. из командной строки в графическом режиме

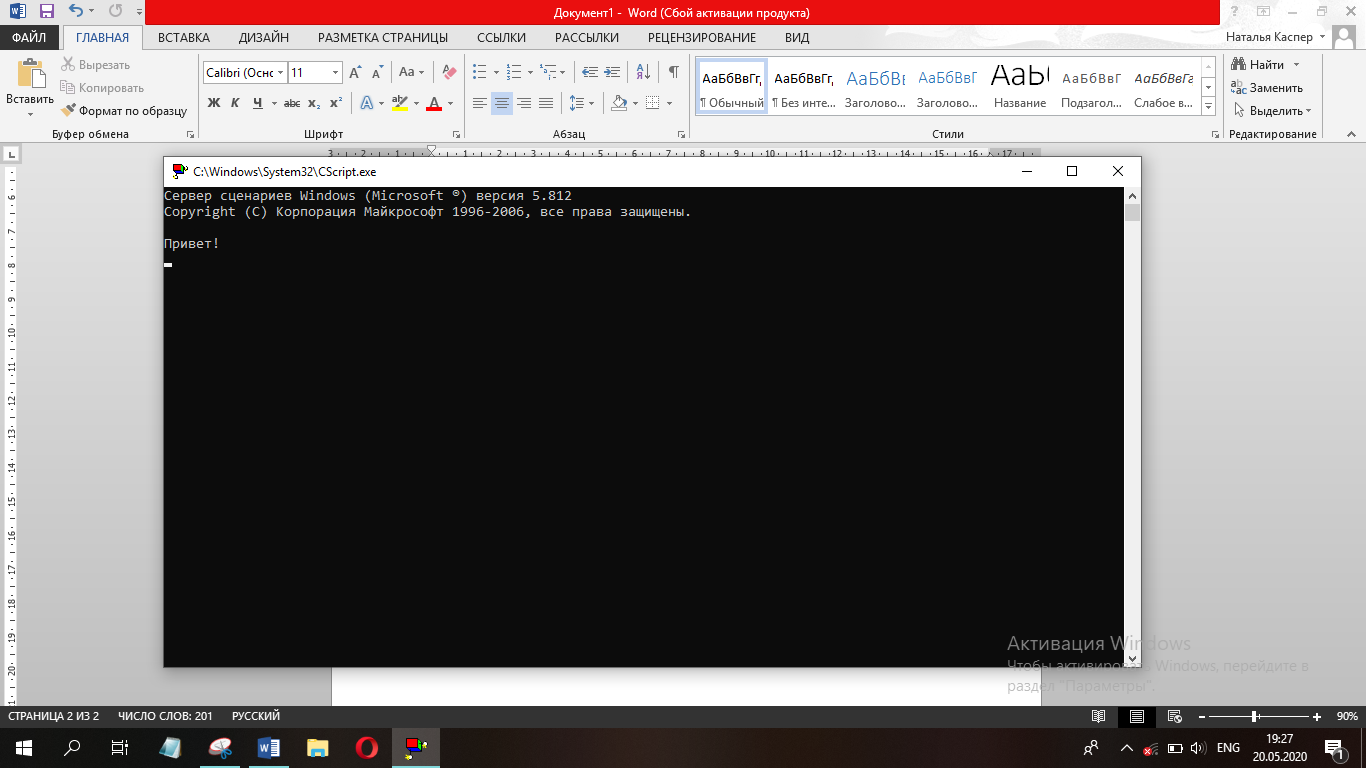


1. с помощью меню Пуск

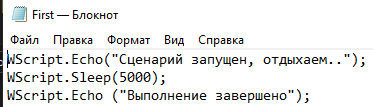


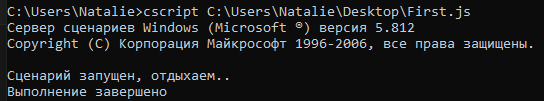


1. с помощью проводника Windows (Windows Explorer)



**Метод Sleep**





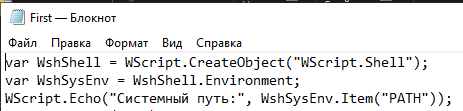
**Объект WshShell**

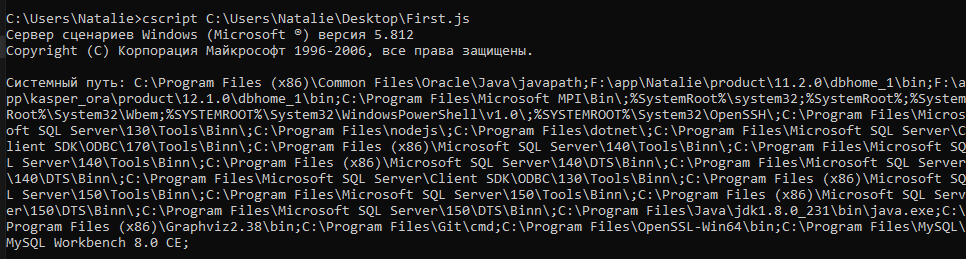
С помощью этого объекта можно получить новый процесс, создавать ярлыки, работать с системным реестром, получать доступ к переменным среды и специальным папкам Windows.

var WshShell = WScript.CreateObject("WScript.Shell");

**Объект WshEnvironment**

Позволяет получить доступ к коллекции, содержащей переменные среды заданного типа. Приведем пример для получения значений переменной среды PATH:





**Получение сведений о диске**

