

**Министерство науки и высшего образования Российской
Федерации**

**Федеральное государственное автономное
образовательное учреждение высшего образования**

«Национальный исследовательский университет ИТМО»

Факультет информационных технологий и программирования

Администрирование в ОС Windows Server

Лабораторная работа №1

Выполнили студенты группы № М33091

Фисенко Никита Данилович

Рустамов Марк Самирович

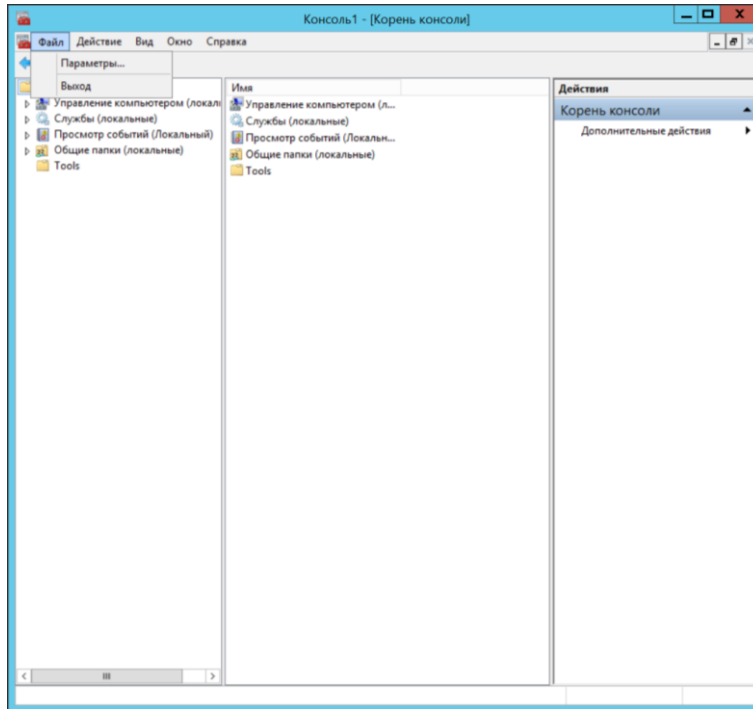
Санкт-Петербург 2023

Цель работы:

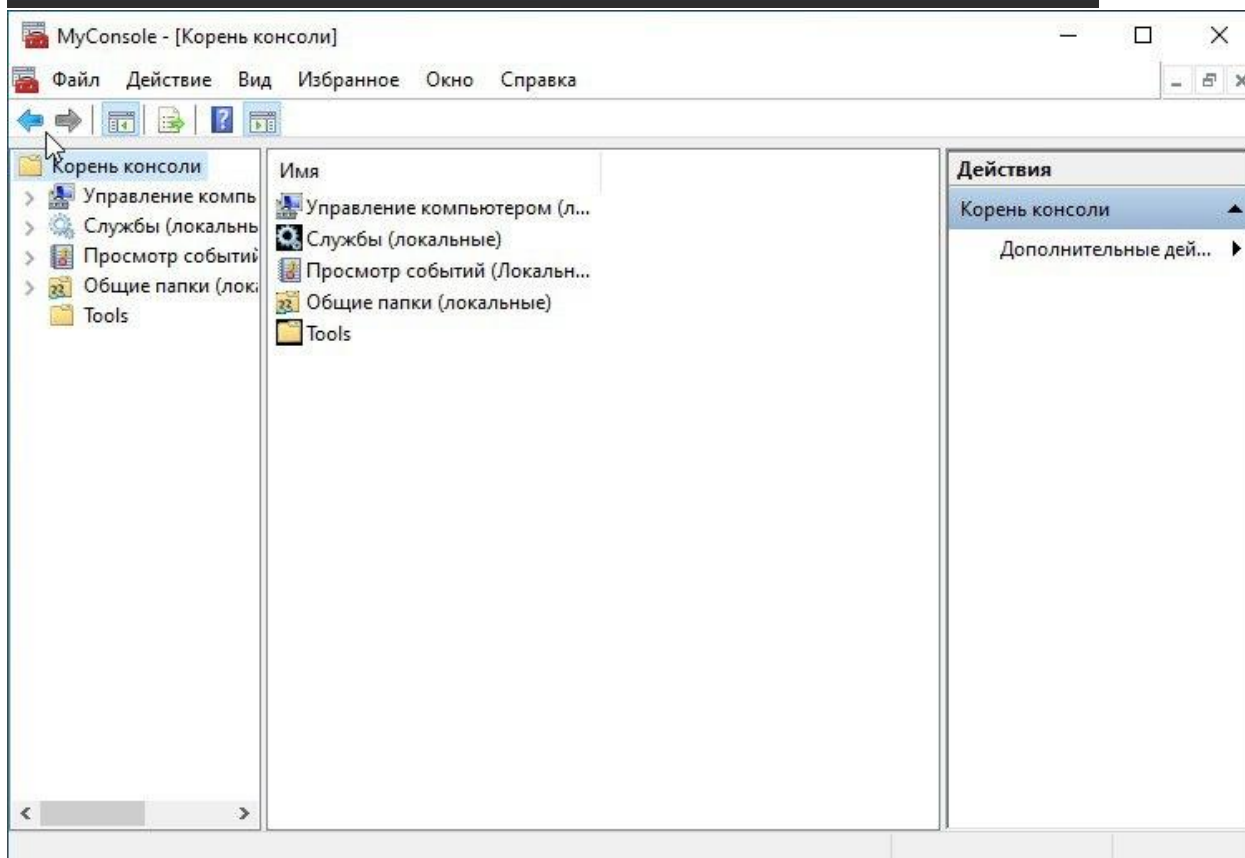
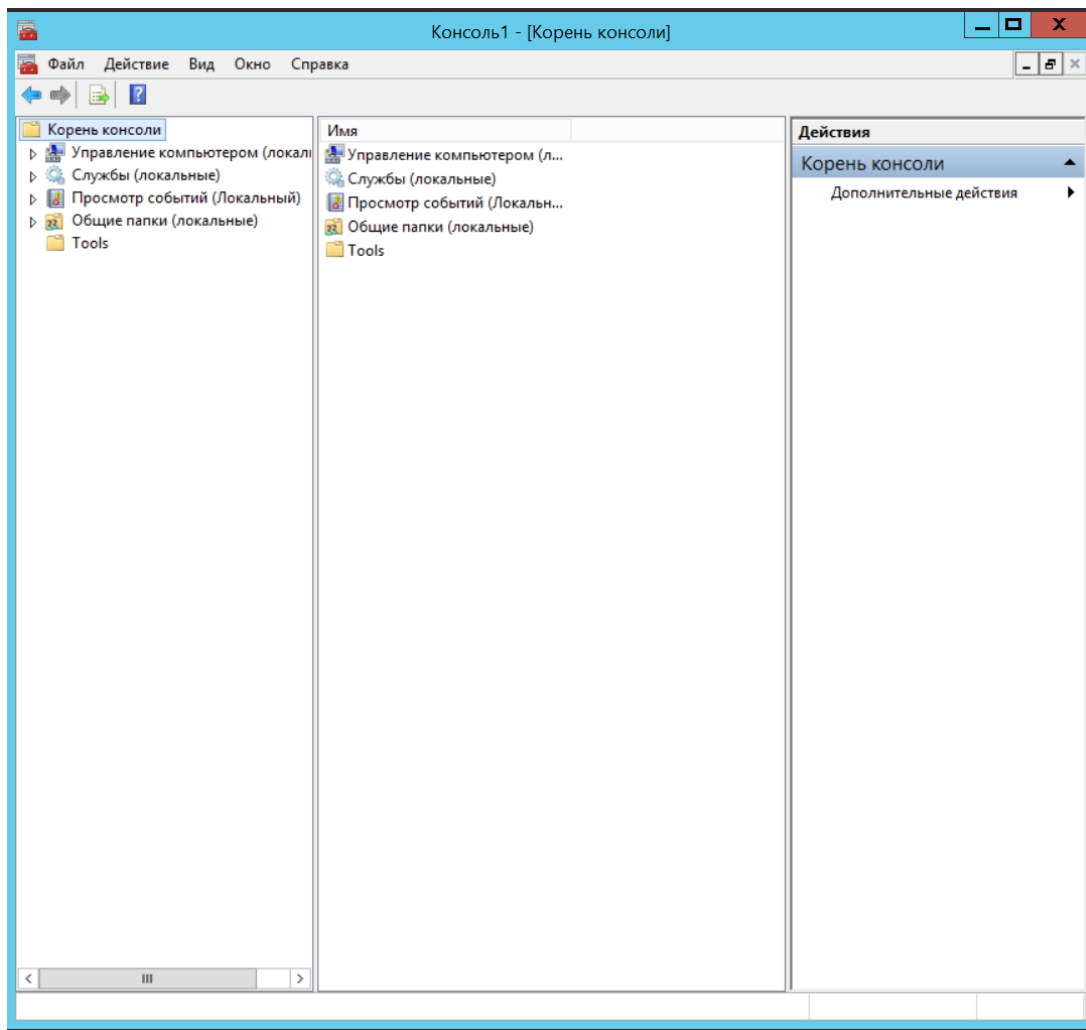
Получить представление и практические навыки работы с основными инструментальными средствами управления Windows Server на примере управления локальными учетными записями и параметрами сетевых интерфейсов.

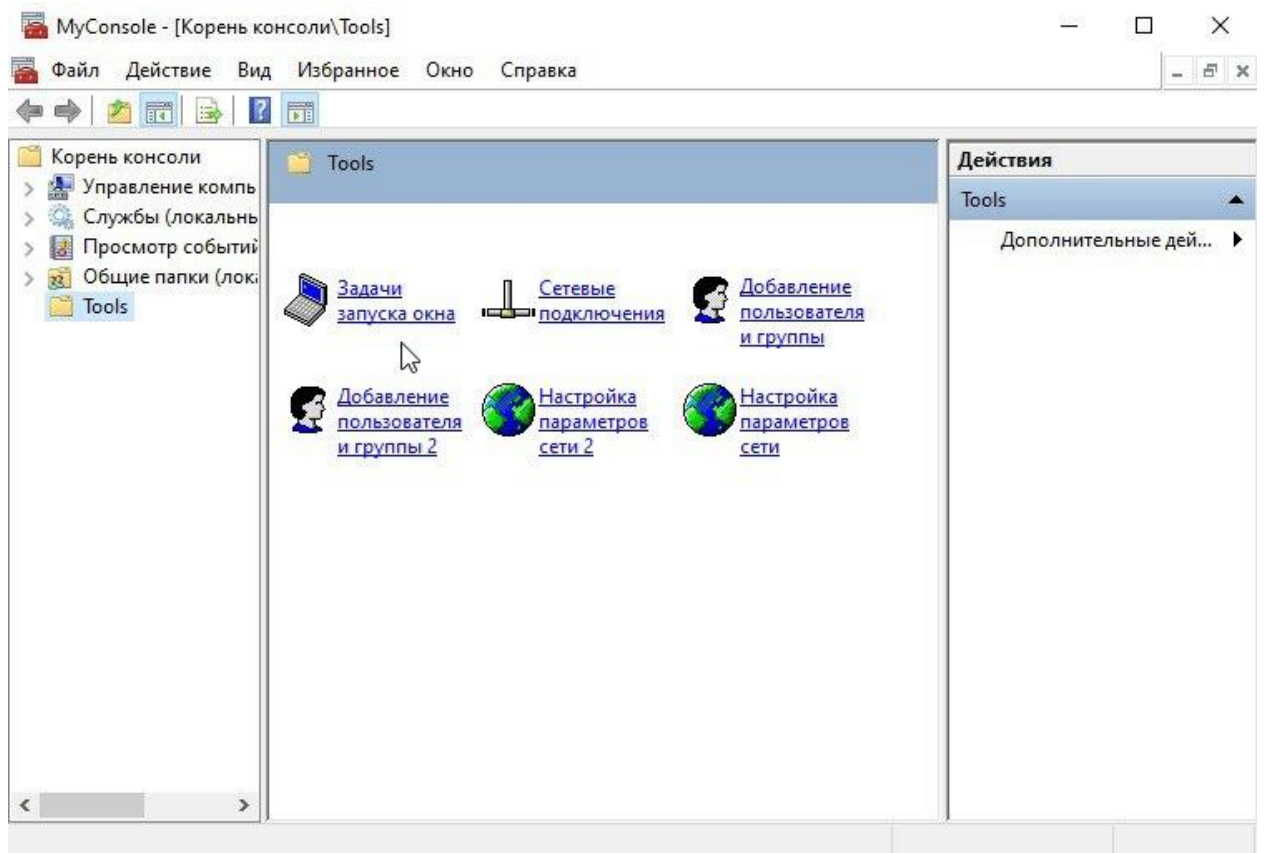
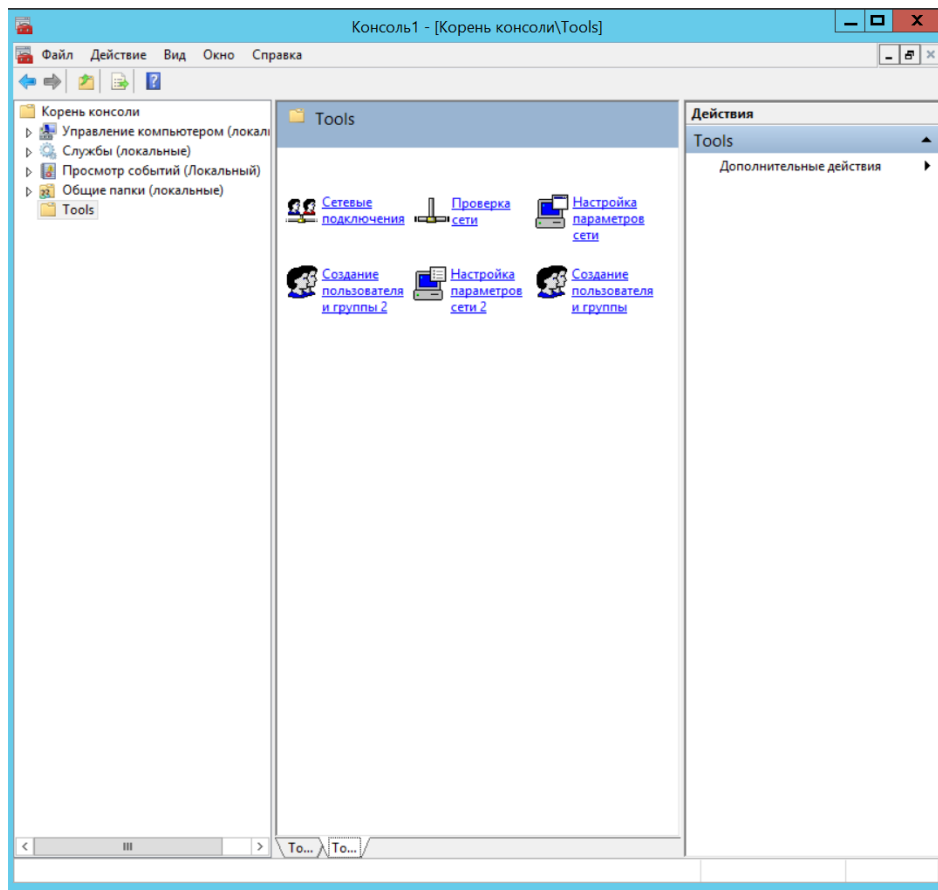
Артефакты:

1. Скриншот финальной версии консоли



Консоль нельзя редактировать.





2. Тексты скриптов из части 2 и части 3.

script21.bat

```
@echo off
set /p str="Enter your string: "
setlocal enabledelayedexpansion
set i=0
for /l %%i in (0, 1, 255) do (
    set c=!str:~%%i,1%!
    if defined c (
        if "!c!" equ " " (
            echo The string must not contain spaces
            exit /b
        )
        set /a i+=1
    )
)
if "!i!" neq "4" (
    echo The string must contain 4 symbols, you have !i!
    exit /b
)
net user UPart2!str! /add

net localgroup GPart2!str! /add
net localgroup GPart2!str! UPart2!str! /add
net user UPart2!str! /active:yes
endlocal
```

script22.bat

```
@echo off

set interfaceName=Ethernet

echo Choose auto or manual. If the user selects Auto, then configures the
receipt of the IP address, mask, gateway and DNS automatically, otherwise
sets the standart parameters.

choice /c "am" /m "yours input: "

if %errorlevel%==1 (
    echo Automatic setup is running
    netsh interface ip set address name=%interfaceName% dhcp
    netsh interface ip set dns %interfaceName% dhcp
    echo Done!
    pause
) else if %errorlevel%==2 (
    echo Manual setup is running
    netsh interface ip set address name="%interfaceName%" static 192.168.1.10
255.255.255.0 192.168.1.1
    netsh interface ip set dns "%interfaceName%" static 8.8.8.8
    echo Done!
    pause
) else (
    echo Incorrect input
    pause
)

exit
```

script22.bat(2)

```
@echo off
set /p opt="Enter the parameter 1 - auto, 2 - manual: "
if "%opt%" equ "1" (
    netsh interface ipv4 set address name="Ethernet" source=dhcp
    echo Success
    exit /b
)
if "%opt%" equ "2" (
    netsh interface ipv4 set address name="Ethernet" source=static
    address=192.168.1.10 mask=255.255.255.0 gateway=192.168.1.1
    netsh interface ipv4 set dnsservers name="Ethernet" source=static
    address=8.8.8.8
    echo Success
    exit /b
)
echo The parameter must be 1 either 2
```

script31.ps1

```
$name = Read-Host "Enter username (without spaces, 4 letters)"
if ($name.Contains(" ")){
    Write-Host "Username must not contain spaces."
    pause
    exit
}

if ($name.Length -ne 4){
    Write-Host "Username must contain 4 characters"
    pause
    exit
}

$username = 'UPart3' + $name
$groupname = 'GPart3' + $name
$pass = Read-Host "Enter password"
$password = ConvertTo-SecureString -String $pass -AsPlainText -Force

try {
    New-LocalUser -Name $username -Password $password -ErrorAction stop
    New-LocalGroup -Name $groupname
    Add-LocalGroupMember -Group $groupname -Member $username
    Write-Host "Done!"
    pause
}
catch{
    Write-Host "Strengthen the password!"
    pause
}

exit
```

script31.ps1(2)

```
$str = Read-Host "Enter your string: "
if ( $str.Contains(" ") ) {
    "The string must not contain spaces, please try again"
    exit
}
```

```

}
if ( $str.Length -ne 4 ) {
    "The string must contain 4 symbols, please try again"
    exit
}
$username="UPart3"+$str
$groupname="GPart3"+$str
New-LocalUser -Name $username -NoPassword
New-LocalGroup -Name $groupname
Add-LocalGroupMember -Group $groupname -Member $username
Enable-LocalUser -Name $username

```

script32.ps1

```

$adapter = Get-NetAdapter -Name Ethernet

$choice = Read-Host "Choose <auto> or <manual>. If the user selects Auto,
then configures the receipt of the IP address, mask, gateway and DNS
automatically, otherwise sets the standart parameters. Type auto or manual"

if ($choice.ToLower() -eq "auto"){
    $IPType = "IPv4"
    $interface = $adapter | Get-NetIPInterface -AddressFamily $IPType
    If ($interface.Dhcp -eq "Disabled") {
        If (($interface | Get-NetIPConfiguration).Ipv4DefaultGateway) {
            $interface | Remove-NetRoute -Confirm:$false
        }
        $interface | Set-NetIPInterface -DHCP Enabled
        $interface | Set-DnsClientServerAddress -ResetServerAddresses
    }

    Write-Host "Done!"
    pause
}
elseif ($choice.ToLower() -eq "manual"){
    Remove-NetIPAddress -InterfaceAlias $adapter.InterfaceAlias -
Confirm:$false
    Remove-NetRoute -InterfaceAlias $adapter.InterfaceAlias -Confirm:$false
    New-NetIPAddress -InterfaceIndex $adapter.InterfaceIndex -IPAddress
192.168.1.10 -PrefixLength 24 -DefaultGateway 192.168.1.1 >$nul
    Set-DnsClientServerAddress -InterfaceIndex $adapter.InterfaceIndex -
ServerAddresses 8.8.8.8

    Write-Host "Done!"
    pause
} else{
    Write-Host "Incorrect input"
    pause
}

exit

```

script32.ps1(2)

```

$opt = Read-Host "Enter the parameter 1 - auto, 2 - manual: "
if ( $opt -eq 1 ) {
    Set-NetIPInterface -InterfaceAlias Ethernet -Dhcp Enabled
    Set-DnsClientServerAddress -InterfaceAlias Ethernet -ResetServerAddresses
    Remove-NetRoute -InterfaceAlias Ethernet -Confirm:$false
    Restart-NetAdapter -Name Ethernet
}

```

```

        exit
    }
    if ( $opt -eq 2 ) {
        Remove-NetIPAddress -InterfaceAlias Ethernet -Confirm:$false
        Remove-NetRoute -InterfaceAlias Ethernet -Confirm:$false
        Set-DnsClientServerAddress -InterfaceAlias Ethernet -ResetServerAddresses
        New-NetIPAddress -InterfaceAlias Ethernet -IPAddress 192.168.1.10 -
DefaultGateway 192.168.1.1 -PrefixLength 24
        Set-DnsClientServerAddress -InterfaceAlias Ethernet -ServerAddresses
8.8.8.8
        exit
    }
    "The parameter must be 1 either 2, please try again"

```

Ответы на вопросы:

1. Пользователь оказался в созданной группе GPart1FIO, а также в группе пользователи.
2. Диалог в скриптах PowerShell намного удобнее, чем CMD. В PowerShell существует множество команд (такие как Contains, Length) и другие, которые упрощают разработку скриптов.
3. Чтобы добавить несколько IPv4 адресов на сетевой адаптер, мы можем:
 - Вызвать psra.cpl
 - Открыть свойства необходимого сетевого интерфейса. В списке протоколов выбрать TCP/IP v4 и нажать кнопку Свойства. Нажать на кнопку Advanced (Дополнительно) и в секции «IP адреса» нажать кнопку Добавить.
 - Указать дополнительный IP адрес и маску подсети и нажать Добавить.
 - Сохранить изменения, несколько раз щелкнув на ОК.

(Основной недостаток добавления второго ip адреса таким способом – для него не включается флаг SkipAsSource (SkipAsSource=False). Если флаг SkipAsSource включен (True), то этот IP адрес не будет использоваться системой для исходящих соединений, кроме явного его задействования определенным приложением. Так же при включенном флаге второй IP адрес не регистрируется в DNS (при включенной динамической регистрации). По сути с помощью параметра SkipAsSource вы можете установить основной IP адрес.)

CMD: Netsh int ipv4 add address name=" " IP mask SkipAsSource=True

PowerShell: New-NetIPAddress -IPAddress IP -PrefixLength MASK -InterfaceAlias InterfaceAlias -SkipAsSource \$True

4. Нужно нажать пкм на «Управление компьютером», «Подключиться к другому компьютеру» и ввести его IP-адрес.

Вывод: В ходе лабораторной работы, с помощью cmd, а также powershell мы познакомились с основными инструментальными средствами управления Windows Server на примере управления локальными учетными записями и параметрами сетевых интерфейсов. Написали скрипты, отвечающие за создание пользователя, а также именной группы для этого пользователя, а также скрипты, которые выполняют автоматическую или мануальную настройку сетевых интерфейсов.