

Task 1

Step1: Creating the Module

Run these Commands in the PowerShell

```
New-Item -Path "C:\Users\sharm\Desktop\Task\DockerHelper" -ItemType Directory
```

```
New-ModuleManifest -Path "C:\Users\sharm\Desktop\Task\DockerHelper\DockerHelper.psd1" -RootModule  
"DockerHelper.psm1"
```

```
Import-Module C:\Users\sharm\Desktop\Task\DockerHelper\DockerHelper.psd1
```

Note: *DockerHelper.psd1 and DockerHelper.psm1 Both files are available in Zip file.*

Step2: To Build the image

Run these commands in PowerShell

For Local

```
Build-DockerImage -Dockerfile "C:\Users\sharm\Desktop\Task\Docker\Dockerfile" -Tag "myapp:latest" -  
Context "C:\Users\sharm\Desktop\Task\Docker\"
```

Note: Replace C:\Users\sharm\Desktop\Task\Docker\Dockerfile and context as per the actual location accordingly

For Remote

```
Build-DockerImage -Dockerfile "\\54.237.187.240\C$\Docker\Dockerfile" -Tag "myapp:latest" -Context  
"\\54.237.187.240\C$\Docker\" -ComputerName "54.237.187.240"
```

Note: Replace 54.237.187.240\C\$\Docker\MyApp\Dockerfile and context as per the actual remote location accordingly

Step3: To copy files to Remote Machine

Run these commands in PowerShell

```
Copy-Prerequisites -ComputerName "54.237.187.240" -Path "C:\Files\" -Destination "C$\Test\Files"
```

Note: Replace 54.237.187.240 with actual server name and destination too.

Step4: Run-DockerContainer

Run these commands in PowerShell

For Local

```
$containerName = Run-DockerContainer -ImageName "myapp:latest" -DockerParams @("-p", "8000:8000")
```

Note: Enter image name accordingly

For Remote

```
$containerName = Run-DockerContainer -ImageName "myapp:latest" -ComputerName "54.237.187.240" -DockerParams @("-p", "8000:8000")
```

Note: Enter image name and server name accordingly.

Test it on URL:

http://localhost:8000/

Connectivity

Prepare the Remote Server

Run these commands in PowerShell

```
Enable-PSRemoting -Force
```

```
winrm set winrm/config/service '@{AllowUnencrypted="true"}'
```

```
winrm set winrm/config/service/Auth '@{Basic="true"}'
```

Prepare Your Local Machine

Run the following command to add the remote server IP to the TrustedHosts list

```
Set-Item WSMan:\localhost\Client\TrustedHosts "54.237.187.240" -Concatenate -Force
```

```
$cred = Get-Credential -Credential "Administrator"
```

```
Enter-PSSession -ComputerName 54.237.187.240 -Credential $cred -Authentication Basic
```

For more secure connection

We can use https connectivity:

Syntax

```
winrm create winrm/config/Listener?Address=*&Transport=HTTPS @{Hostname="<YourServer'sFQDN>";  
CertificateThumbprint="<CertThumbprint>"}
```

Task 2

Step1: Creating the Module

Run these commands in PowerShell

```
New-Item -Path "C:\Users\sharm\Desktop\Task2\DockerHelper" -ItemType Directory
```

```
New-ModuleManifest -Path "C:\Users\sharm\Desktop\Task2\DockerHelper\DockerHelper.psd1" -  
RootModule "DockerHelper.psm1"
```

```
Import-Module C:\Users\sharm\Desktop\Task2\DockerHelper\DockerHelper.psd1 -Force
```

Note: *DockerHelper.psd1 and DockerHelper.psm1 Both files are available in Zip file.*

Step2: To Build the image

Run these commands in PowerShell

```
Build-DockerImage -Dockerfile "C:\Users\sharm\Desktop\Task2\Docker\Dockerfile" -Tag "fibonacci:latest" -  
Context "C:\Users\sharm\Desktop\Task2\Docker\"
```

Note: Replace C:\Users\sharm\Desktop\Task2\Docker\Dockerfile and context as per the actual location accordingly

Step3: Run-DockerContainer

Run these commands in PowerShell

For Sequence

```
Run-DockerContainer -ImageName "fibonacci:latest"
```

For Specific Number

```
Run-DockerContainer -ImageName "fibonacci:latest" -DockerParams @("-n", "15")
```

Note: Enter image name accordingly