

Practical Works TP 1: Installation and Setup

Important

You have to download Docker Desktop first.

Step 1: Create the project folder

Open your **PowerShell terminal** in your working directory (in normal or administrator mode) and run the following commands:

```
powershell  
New-Item -ItemType Directory -Name "hadoop-docker-cluster"  
cd .\hadoop-docker-cluster
```

Step 2: Create the docker-compose.yml file

On Windows, you can generate the file directly from **PowerShell** using the following command:

```
@'  
version: '3.8'  
  
services:  
  namenode:  
    image: apache/hadoop:3.3.6  
    container_name: namenode  
    hostname: namenode  
    user: "root"  
    command: ["hdfs", "namenode"]  
    ports:  
      - "9870:9870"  
      - "9000:9000"  
    env_file:  
      - ./hadoop.env  
    volumes:  
      - namenode_data:/opt/hadoop/dfs/name  
  
  datanode1:  
    image: apache/hadoop:3.3.6  
    container_name: datanode1  
    hostname: datanode1  
    user: "root"  
    command: ["hdfs", "datanode"]  
    ports:  
      - "9864:9864"  
    env_file:  
      - ./hadoop.env  
    volumes:  
      - datanode1_data:/opt/hadoop/dfs/data
```

```

    depends_on:
      - namenode

datanode2:
  image: apache/hadoop:3.3.6
  container_name: datanode2
  hostname: datanode2
  user: "root"
  command: ["hdfs", "datanode"]
  ports:
    - "9865:9864"
  env_file:
    - ./hadoop.env
  volumes:
    - datanode2_data:/opt/hadoop/dfs/data
  depends_on:
    - namenode

volumes:
  namenode_data:
  datanode1_data:
  datanode2_data:
'@ | Out-File -FilePath "docker-compose.yml" -Encoding utf8

```

Step 3: Create the hadoop.env configuration file

Generate the configuration file with the environment variables in the same way:

```

Powershell
@'
CORE-SITE.XML_fs.defaultFS=hdfs://namenode:9000
HDFS-SITE.XML_dfs.replication=2
HDFS-SITE.XML_dfs.namenode.name.dir=file:///opt/hadoop/dfs/name
HDFS-SITE.XML_dfs.datanode.data.dir=file:///opt/hadoop/dfs/data
'@ | Out-File -FilePath "hadoop.env" -Encoding utf8

```

Step 4: Initialize and start on Windows

(Note: Make sure Docker Desktop is running on your PC)

1. Format the NameNode (Essential step to initialize HDFS):

```

powershell
docker compose run --user "root" --rm namenode hdfs namenode -format

```

2. Start the containers in the background:

```

powershell
docker compose up -d

```

3. Check the status of your nodes :

```

powershell
docker compose ps

```

Step 5: Practice HDFS commands

1. Interface Web (UI)

The interface remains accessible via your Windows browser at <http://localhost:9870>.

2. Execute commands from PowerShell

To send your instructions to the Hadoop cluster, connect to the main container:

```
powershell
docker exec -it namenode bash
```

Once your PowerShell prompt displays `root@namenode:/opt/hadoop#`, run these test commands:

- **Create a test folder:**

```
bash
hdfs dfs -mkdir /my_folder
```

- **Create a text file inside the container:**

```
bash
echo "Hello from Windows PowerShell and Docker" > test.txt
```

- **Copy this file into HDFS:**

```
bash
hdfs dfs -put test.txt /my_folder/
```

- **Verify that the file is present:**

```
bash
hdfs dfs -ls /my_folder
```

- **Read the contents of the file stored on HDFS:**

```
bash
hdfs dfs -cat /my_folder/test.txt
```

To exit the container and return to your standard Windows command line, enter:

```
bash
exit
```

- **Stop the services**

To temporarily shut down the cluster without losing your files stored on HDFS:

```
powershell
docker compose down
```