**172.1.14.200:81/cclab**

**Instruction Guide**

**JAVA**

1. Verify whether you have JAVA installed in your system using the following command.

**$ javac --version**

1. If not, to install JDK in Linux, use the following command.

**$ sudo apt install default-jdk**

1. Then, the JRE File of Java will be installed using the following command.

**$ sudo apt install default-jre**

1. To verify the installation, the following command you can use. It will prompt the Java version used there.

**$ javac --version**

1. Create a new folder named <1MS23SCS/SCN\*\*> and navigate inside the folder using the following command.

**$ cd <1MS23SCS/SCN\*\*>**

**Hadoop**

1. Now download the hadoop tar file to install using the given url.
2. Once you have downloaded hadoop-3.2.2.tar.gz, extract this file with the below command (make sure to check your tar filename).

**$ tar xvzf hadoop-3.2.2.tar.gz**

**[** *tar: The command to create or extract tar archives.*

*x: Extract files from the archive.*

*v: Verbosely list files processed (optional, but useful for seeing what’s being extracted).*

*z: Decompress the archive using gzip.*

*f: Specifies that a filename follows (in this case, hadoop-3.2.2.tar.gz).*

*Running this command will unpack the contents of the hadoop-3.2.2.tar.gz file into the*

*current directory.*

]

1. Now navigate inside the folder using the below command.

**$ cd hadoop-3.2.2/**

1. Create and open a new ***bash.sh*** file inside the directory.

**$ gedit bash.sh**

1. We configure the file, copy the below command inside this file and save it.
2. **export JAVA\_HOME=$(readlink -f $(which javac) | awk 'BEGIN {FS="/bin"} {print $1}')**
3. **export PATH=$(echo $PATH):$(pwd)/bin**
4. **export CLASSPATH=$(hadoop classpath)**
5. *The command sets the JAVA\_HOME environment variable to point to the directory where the Java compiler (javac) is located. Here’s a breakdown of the command:*
6. ***which javac****: This command finds the path to the javac executable.*
7. ***readlink -f****: This resolves any symbolic links in the path, returning the canonical path.*
8. ***awk 'BEGIN {FS="/bin"} {print $1}'****: This uses awk to split the resolved path at /bin and prints the first part, which is the path to the Java installation directory.*

*Putting it all together, the command effectively sets JAVA\_HOME to the root directory of your Java installation.*

1. *The command modifies the PATH environment variable to include the bin directory located in the current working directory (pwd). Here's a breakdown of how it works:*
2. ***export PATH=****: This command sets the PATH variable for the current session and makes it available to child processes.*
3. ***$(echo $PATH)****: This retrieves the current value of the PATH variable. Using echo and command substitution with $(...) allows you to include the existing PATH in the new assignment.*
4. ***$(pwd)/bin****: The pwd command outputs the current working directory. Appending /bin constructs the path to the bin directory within that current directory.*
5. ***:****: This is a separator used in the PATH variable to separate different directory paths.*

*It appends the path to the bin directory of the current working directory to the existing PATH. This allows you to execute any scripts or binaries located in $(pwd)/bin from anywhere in your terminal session.*

1. *The command sets the CLASSPATH environment variable to the classpath used by Hadoop. Here’s a breakdown of what this does:*
2. ***export CLASSPATH=****: This sets the CLASSPATH variable for the current session and exports it so that it’s available to any child processes.*
3. ***$(hadoop classpath)****: This uses command substitution to execute hadoop classpath, which outputs the paths of the necessary Hadoop libraries and dependencies. The result is assigned to the CLASSPATH variable.*

***CLASSPATH****: In Java, the CLASSPATH variable specifies the location of user-defined classes and packages. By setting it to the output of hadoop classpath, you ensure that Java applications can find all the necessary Hadoop libraries and resources.*

*This command is particularly useful when you're developing or running Java applications that depend on Hadoop, ensuring that all necessary classes are available at runtime.*

1. Execute the bash.sh File using following command

**$ source bash.sh**

1. Verify ***JAVA\_HOME*** variable to be set to Java Path and ***PATH*** variable has your Hadoop Folder.

[ Echo $JAVA-HOME

Echo $PATH

]

1. Verify Hadoop is Installed or not by executing hadoop command. If command gives Information about Hadoop command, then Hadoop is Successfully Installed.

**$ hadoop**

**Spark**

1. Verify that hadoop is installed and running in your system.

**$ hadoop**

1. Download Apache Spark from the given url inside your folder.
2. Now we extract this tar file with the help of below command (make sure to check your tar filename).

**$ tar xvzf spark-3.5.2-bin-hadoop3.tgz**

1. Now navigate inside the folder using the below command.

**$ cd spark-3.5.2-bin-hadoop3/**

1. Create a new file named bash.sh inside your folder by using any text editor.

**$ gedit bash.sh**

1. Copy below code and paste inside bash.sh file

**export PATH=$(echo $PATH):$(pwd)/bin**

1. In terminal, execute bash.sh file using the following command.

**$ source bash.sh**

1. Verify spark version with the below command.

**$ spark-shell --version**

**PIG**

1. Verify that hadoop is installed and running in your system.

**$ hadoop**

1. Download Apache Pig from the given url inside your folder.
2. Now we extract this tar file with the help of below command (make sure to check your tar filename).

**$ tar -xvf pig-0.17.0.tar.gz**

1. Now navigate inside the folder using the below command.

**$ cd pig-0.17.0/**

1. Create and open a new bash.sh file inside the directory.

**$ gedit bash.sh**

1. We configure file, copy the below command inside this file and save it.

**export PIG\_INSTALL=$(pwd)**

**export PATH=$PATH:$(pwd)/bin**

1. Execute the bash.sh File using following command

**$ source bash.sh**

1. You can check your pig version with the below command.

**$ pig -version**

1. Once you get it correct that’s it we have successfully install pig to our Hadoop single node setup, now we start pig with below pig command.

**$ pig**