

Hadoop Assignment Steps (Port No: 9870)

Prerequisites

Ensure Hadoop is installed and running.

Step-by-Step Instructions

1. Start Hadoop Services

```
hadoop user login
stop-all.sh
jps    # (Check running services)
start-all.sh
jps    # (Verify services started)
```

2. HDFS Directory Creation

```
hadoop fs -mkdir /user7
hadoop fs -mkdir /user7/input
```

Now open the browser and go to: <http://localhost:9870>

Create a file `input.txt` using a text editor.

3. Eclipse Setup

- Open Eclipse
 - Create **New Java Project**
 - Right-click on project > **Properties** > **Java Compiler** > **Compliance Level**
 - Set it to 11, click **Apply & Close**
-

4. Create Package and Class

- Right-click on `src` > **New** > **Package** → `hadoopPrac`
- Right-click on package > **New** > **Class** → `wordcount`

Code Template:

```
package hadoopPrac;

public class wordcount {
```

```
// Your MapReduce logic here  
}
```

5. Clean `module-info.java`

- Open `module-info.java` (if any) and **delete contents** or remove the file.
-

6. Add Hadoop Libraries

- Right-click on project > **Build Path** > **Configure Build Path**
 - Add External Archives:
 - Navigate to your Hadoop folder:
 - Add all JARs from:
 - `share/hadoop/common`
 - `share/hadoop/common/lib`
 - `share/hadoop/client`
-

7. Export JAR File

- Right-click on your package > Export > Runnable JAR File
 - Choose destination and **remember the path**
-

8. Upload Input File to HDFS

```
hadoop fs -put input.txt /user7/input
```

9. Run the Hadoop Job

```
hadoop jar example1.jar hadoopPrac.wordcount /user7/input /user7/output
```

Where:

- `example1.jar` → JAR file name
- `hadoopPrac.wordcount` → Package and class name
- `/user7/input` → HDFS input path
- `/user7/output` → HDFS output path

Note: Make sure to delete `/user7/output` directory in HDFS if it already exists:

```
hadoop fs -rm -r /user7/output
```

10. View Output File

To see the contents of the output file generated by your MapReduce job:

```
hadoop fs -cat /user7/output/part-r-00000
```

You can also list the output directory files using:

```
hadoop fs -ls /user7/output
```

This command will display all result files like `part-r-00000` which typically contains your output data.