PROGRAM NO:3

Develop a Map Reduce program that mines weather data and displays appropriate messages indicating the weather conditions of the day.

Step1:goto Search button run cmd prompt as administrator

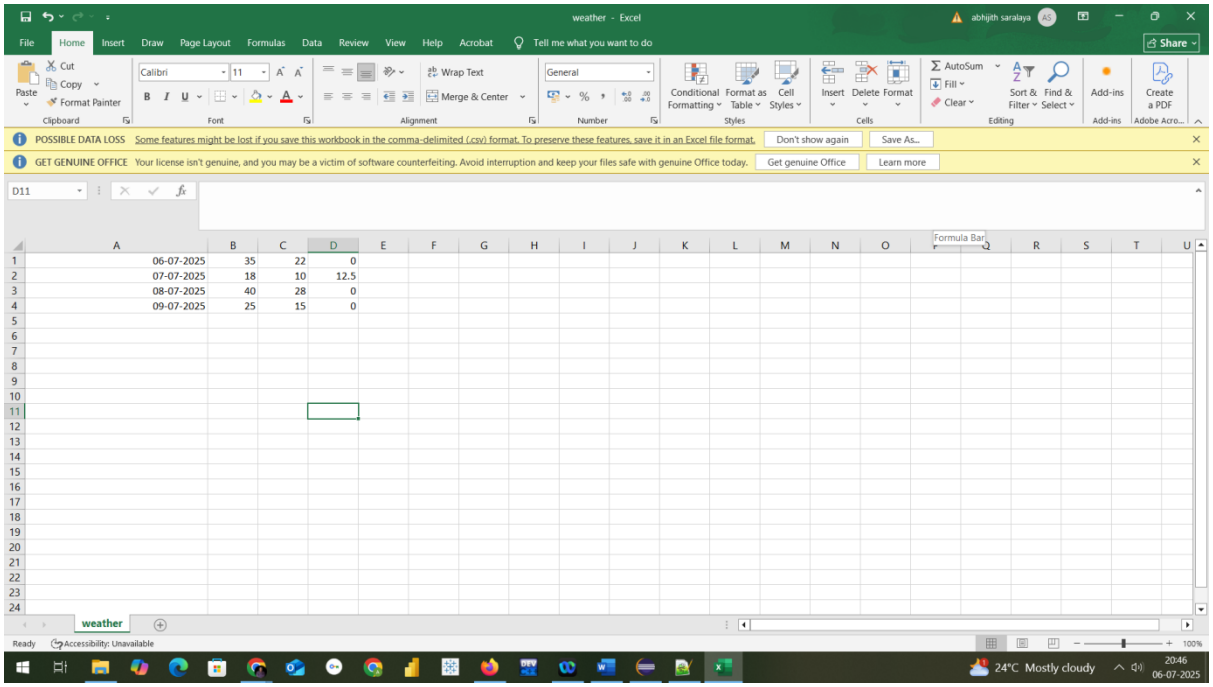
Step2: Initiate all required component of hadoop by command

start-all.cmd

Step3: Create a weather.csv file where we have folder running hadoop java program in my case I have created in Desktop folder D:\hadoopprograms\weatherinput

U can create anywhere as u wish with weather and content looks as in below

06-07-2025	**35**	**22**	**0**
07-07-2025	**18**	**10**	**12.5**
08-07-2025	**40**	**28**	**0**
09-07-2025	**25**	**15**	**0**



Step3: Now we need a jar file to execute mapreduce program fr wordcount

So open Eclipse

Create a java project with name MapReduceWeather

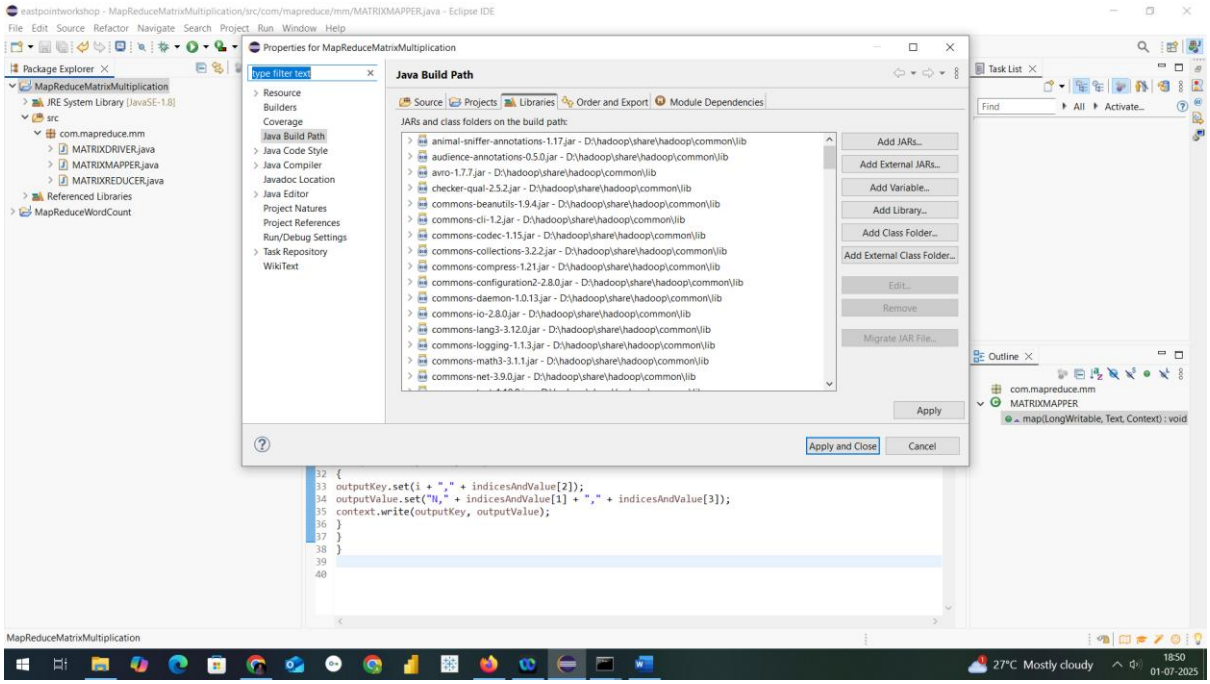
Select execution environment as JavaSE-1.8

And click on next&finish

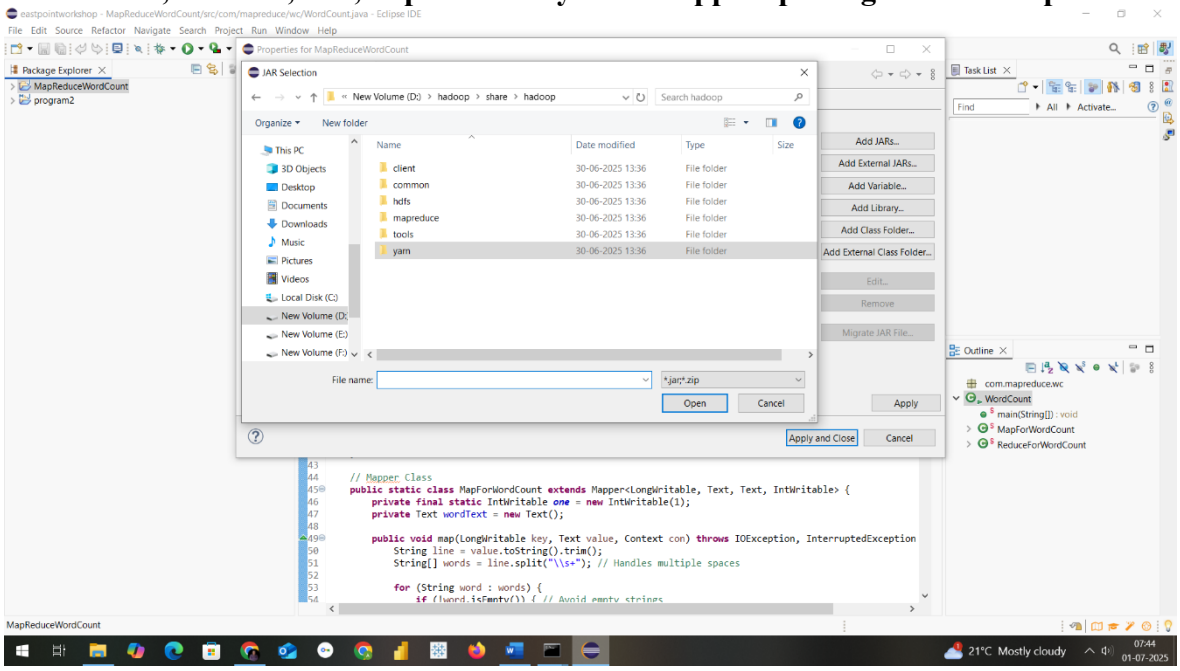
Step4: Right Click on Projectname---→ Click on New---→create a package with name com.mapreduce.mm and click on Finish

Step5: Add required libraries to support hadoop by navigating as in below

Right Click on Project-------→Right Click on BuildPath-------→ Configure Buildpath Then goto Libraries as in screen below



Step6: Add necessary External jar file from D:\hadoop\share\hadoop
Like clients,common,hdfs,mapreduce & yarn to support packages for hadoop



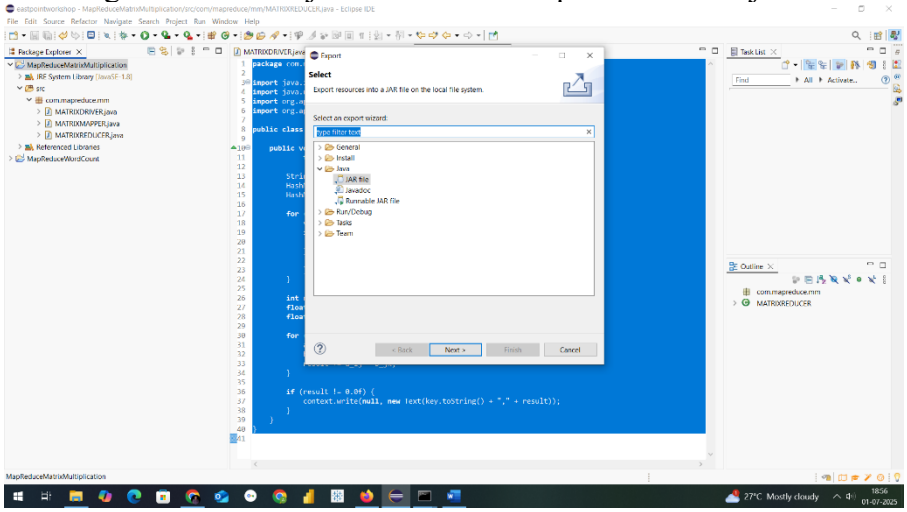
Step7: Create a class within package com.mapreduce.we with name WeatherAnalysis and paste this code

```
package com.mapreduce.we
import java.io.IOException;
import org.apache.hadoop.conf.Configuration;
import org.apache.hadoop.fs.Path;
import org.apache.hadoop.io.*;
```

```
import org.apache.hadoop.mapreduce.Job;
import org.apache.hadoop.mapreduce.Mapper;
import org.apache.hadoop.mapreduce.Reducer;
import org.apache.hadoop.mapreduce.lib.input.FileInputFormat;
import org.apache.hadoop.mapreduce.lib.output.FileOutputFormat;

public class WeatherAnalysis {

   public static class WeatherMapper extends Mapper<LongWritable, Text, Text, Text> {
      public void map(LongWritable key, Text value, Context context)
            throws IOException, InterruptedException {

         String line = value.toString();
         String[] fields = line.split(",");

         if (fields.length == 4) {
            String date = fields[0];
            try {
               float tmax = Float.parseFloat(fields[1]);
               float tmin = Float.parseFloat(fields[2]);
               float prcp = Float.parseFloat(fields[3]);

               String message = "";

               if (tmax > 35) {
                  message += "Hot day ";
               }
               if (tmin < 15) {
                  message += "Cold day ";
               }
               if (prcp > 10) {
                  message += "Rainy day ";
               }

               if (message.isEmpty()) {
                  message = "Normal weather";
               }

               context.write(new Text(date), new Text(message.trim()));
            } catch (NumberFormatException e) {
               // Ignore malformed rows
            }
         }
      }
   }

   public static class WeatherReducer extends Reducer<Text, Text, Text, Text> {
      public void reduce(Text key, Iterable<Text> values, Context context)
            throws IOException, InterruptedException {
         for (Text val : values) {
```

```
                context.write(key, val);  // One value per date
            }
        }
    }

    public static void main(String[] args) throws Exception {
        Configuration conf = new Configuration();
         conf.set("mapreduce.framework.name", "local");
        conf.set("fs.defaultFS", "file:///");
        conf.set("n", "3");

        Job job = Job.getInstance(conf, "Weather Analysis");

        job.setJarByClass(WeatherAnalysis.class);
        job.setMapperClass(WeatherMapper.class);
        job.setReducerClass(WeatherReducer.class);

        job.setOutputKeyClass(Text.class);
        job.setOutputValueClass(Text.class);

        FileInputFormat.addInputPath(job, new Path(args[0]));  // Input file path
        FileOutputFormat.setOutputPath(job, new Path(args[1])); // Output directory path

        System.exit(job.waitForCompletion(true) ? 0 : 1);
    }
}
```

Step8: Procedure to create jar file from eclipse project
Now Right click on Project and click on export and select jar as in screen



Mention the path where u want to save jar with name of Jar File in my case I have saved in D:\hadoopprograms as Weather.jar

Final step: Run the jar file with command as in below
hadoop jar D:\hadoopprograms\Weather.jar com.mapreduce.we.WeatherAnalysis D:\hadoopprograms\weatherinput\weather.csv D:\hadoopprograms\weatheroutput

THE OUTPUT FORMAT WILL BE CREATED THIS TIME IN LOCALFOLDER WHERE MENTIONED
----→ D:\hadoopprograms\weatheroutput
In part-r-00000 text file

NOW TO CHECK THE RESULT OF A PROGRAM