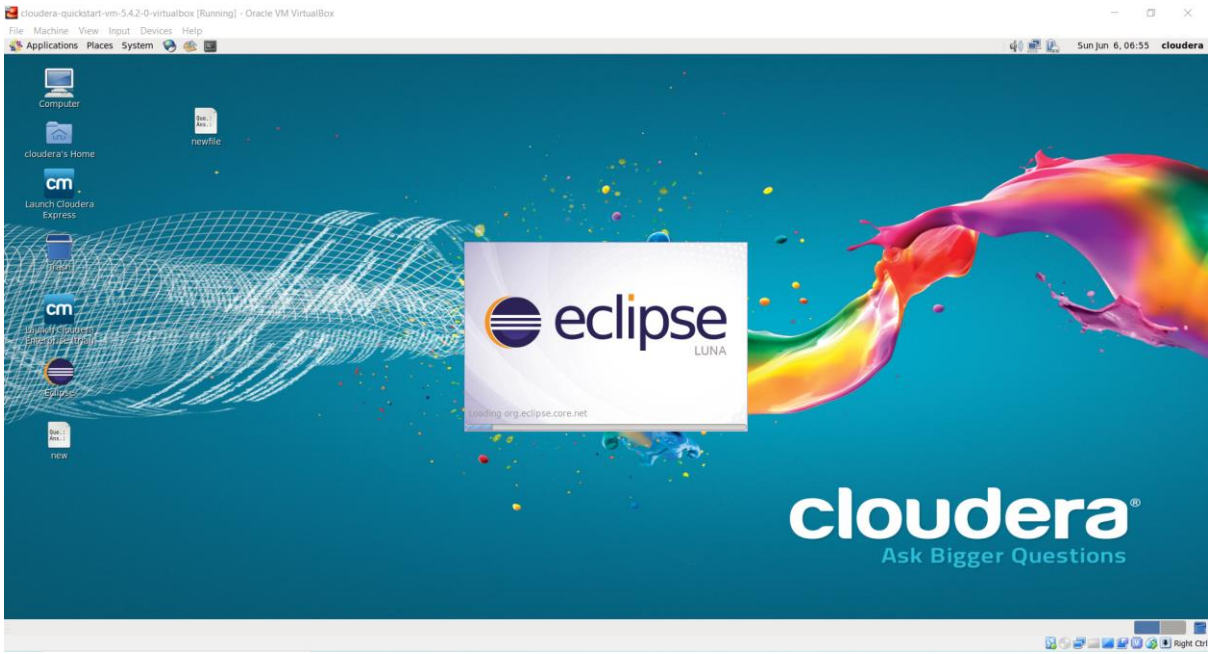
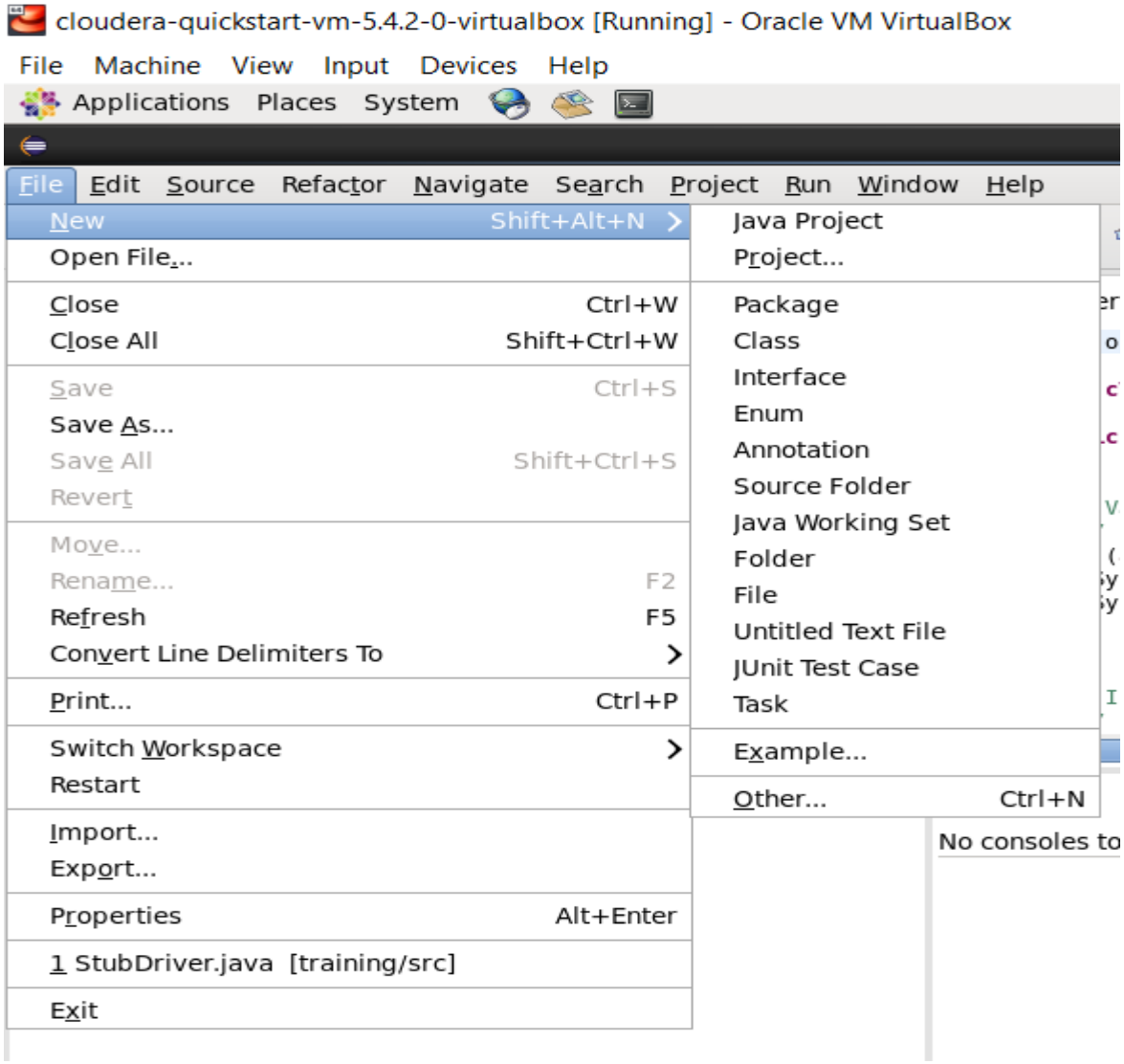
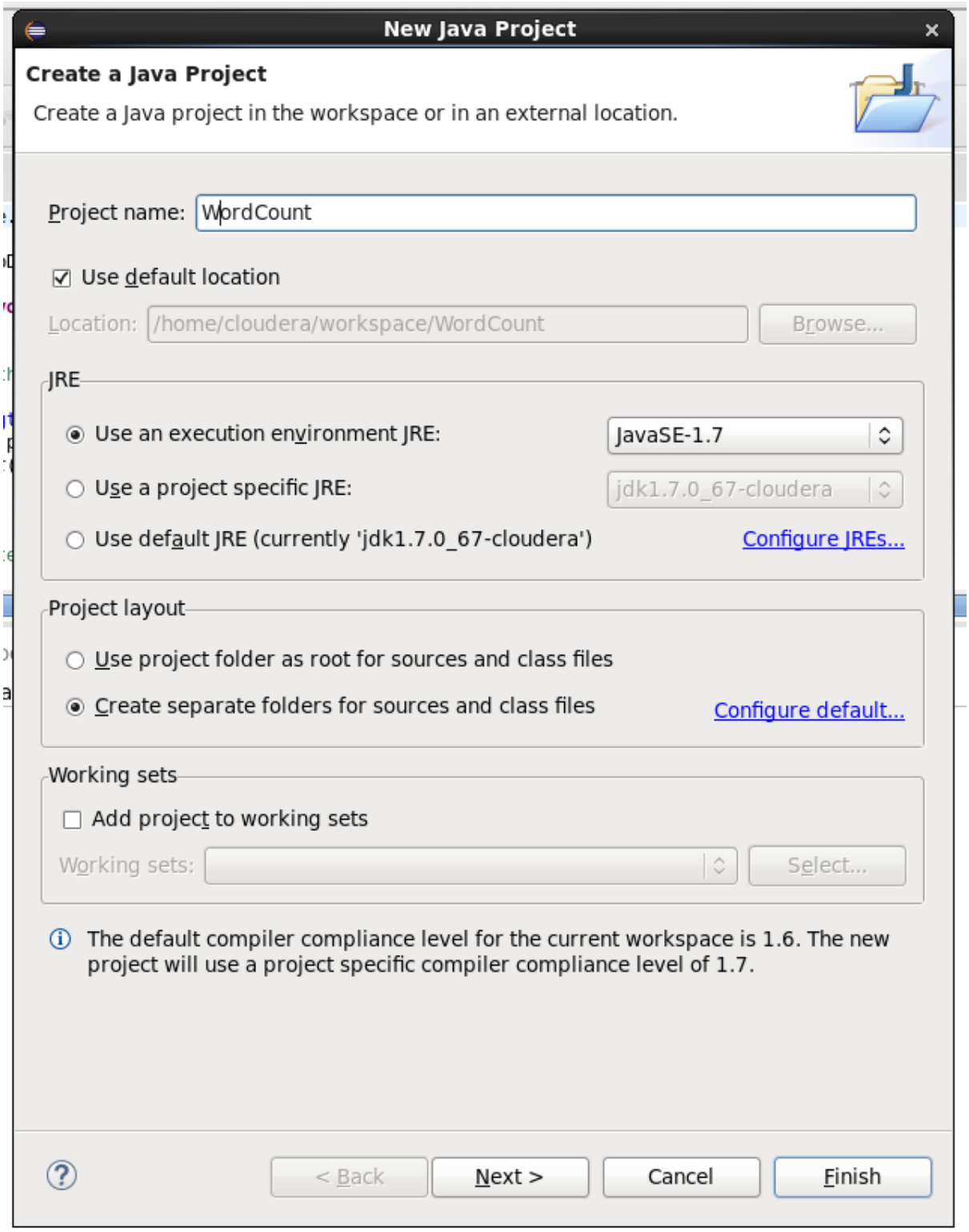
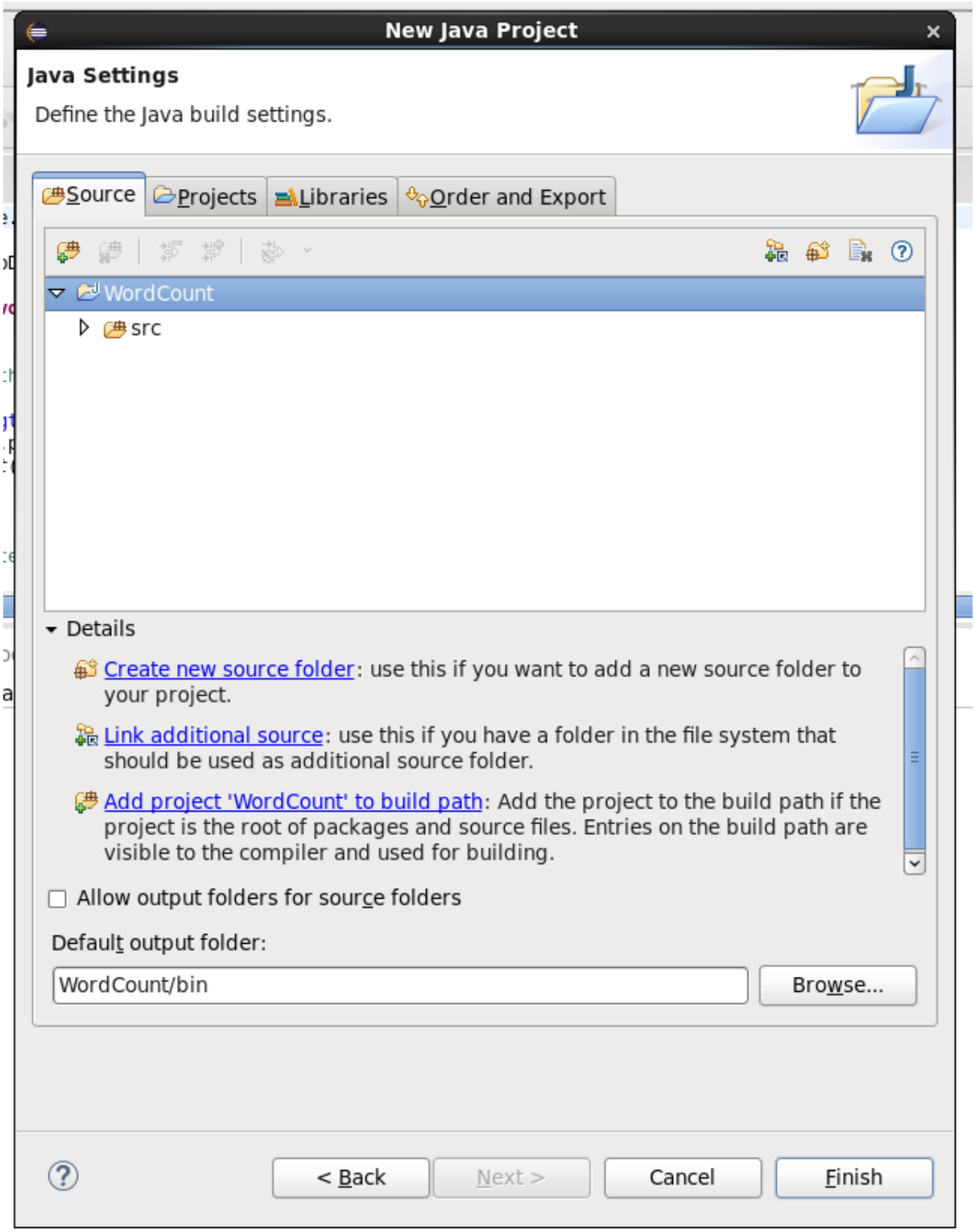
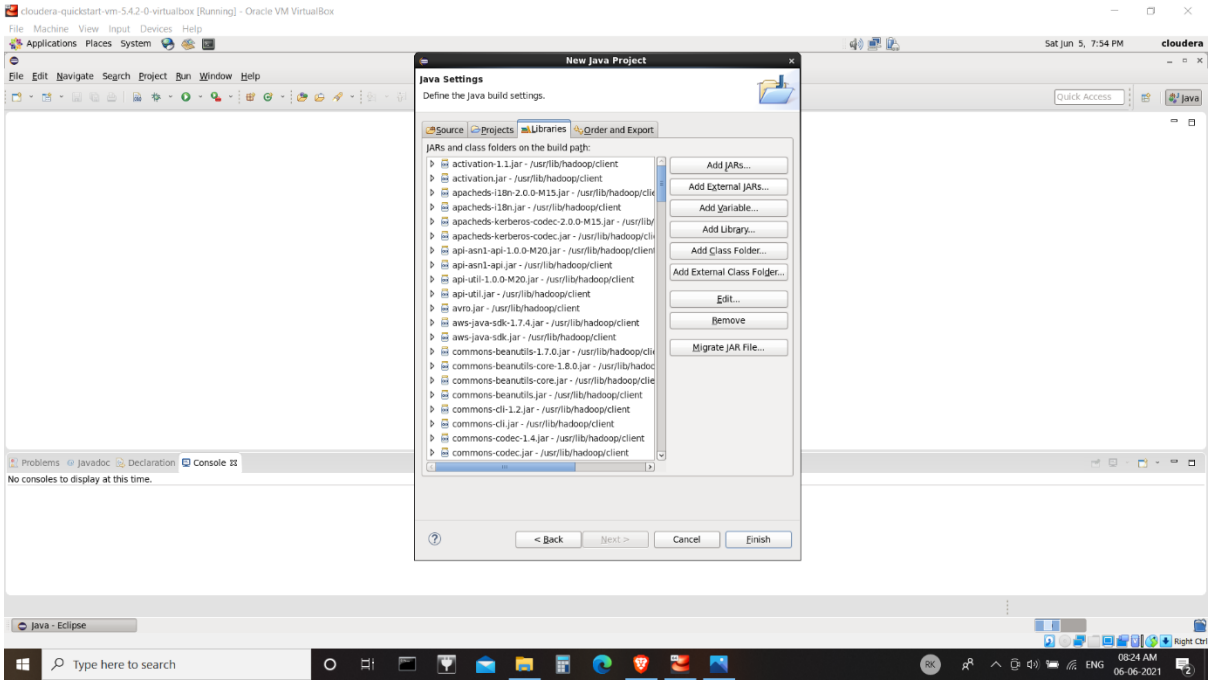
**Experiment 03**

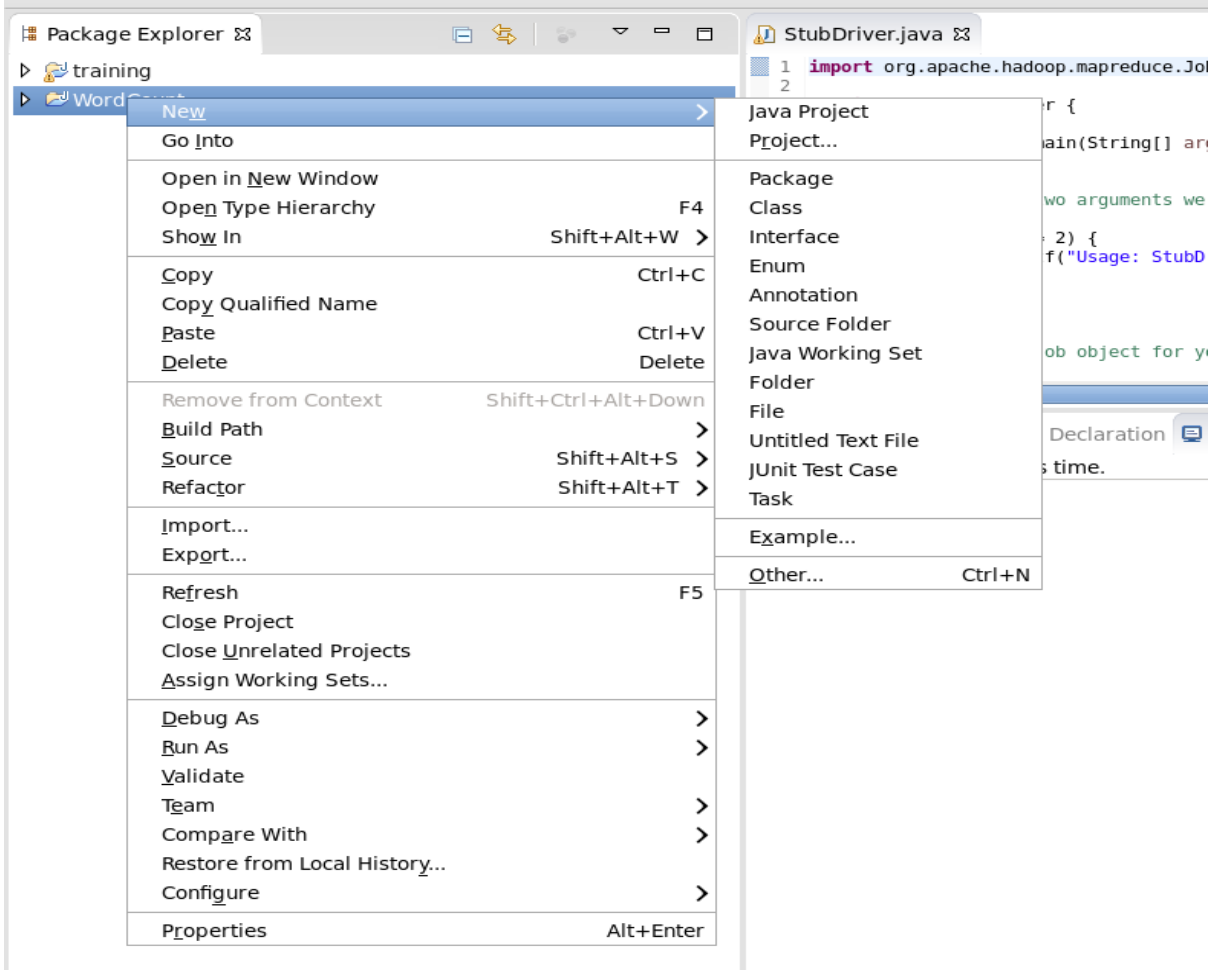


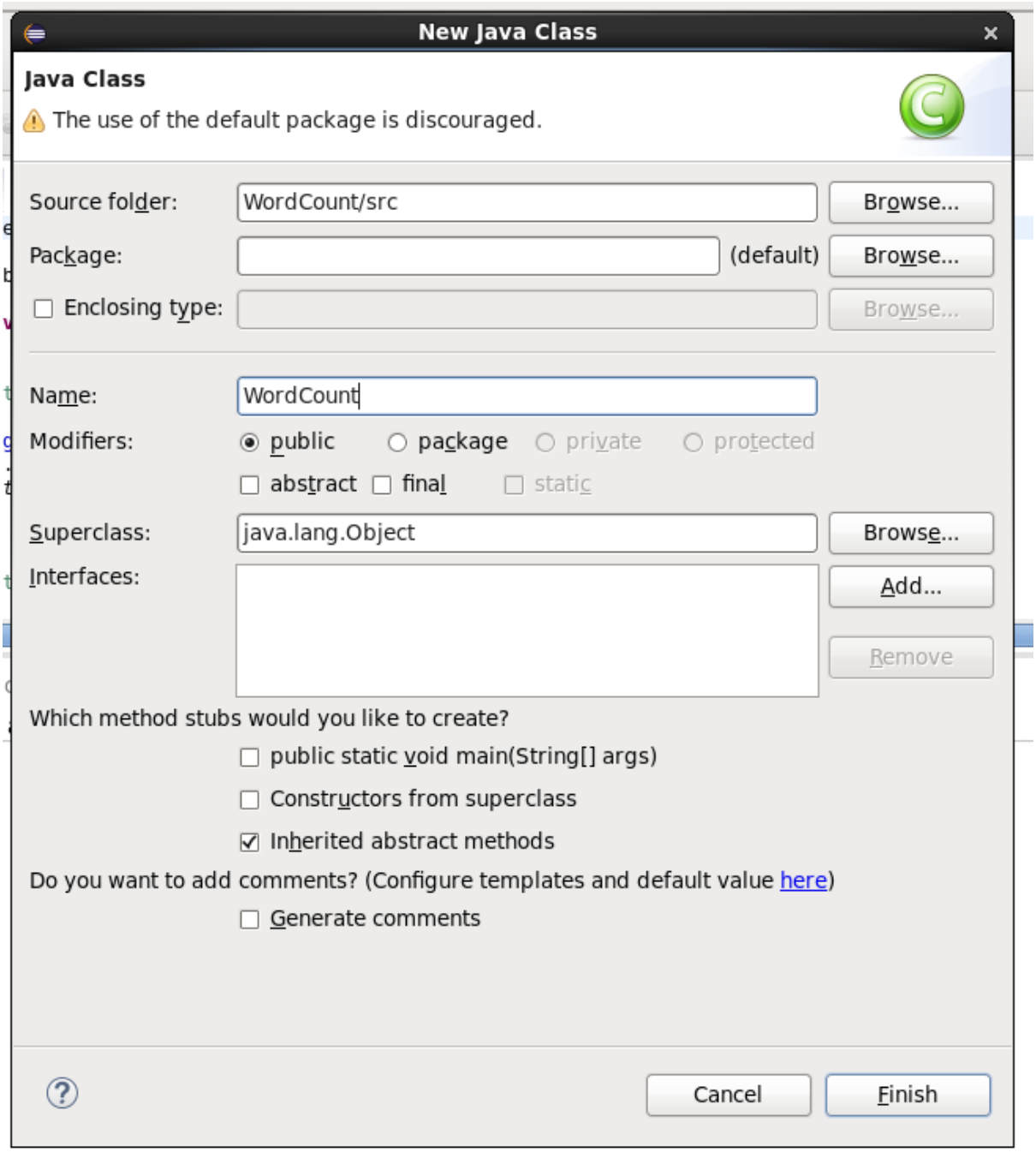




New Java Project
Create a Java Project
Create a Java project in the workspace or in an external location.
Project name: WordCount
Use default location
Location: /home/cloudera/workspace/WordCount
Browse...
JRE
Use an execution environment JRE:
JavaSE-1.7
Use a project specific JRE:
jdk1.7.0_67-cloudera
Use default JRE (currently 'jdk1.7.0_67-cloudera')
Configure JREs...
Project layout
Use project folder as root for sources and class files
Create separate folders for sources and class files
Configure default...
Working sets
Add project to working sets
Working sets:
Select...
The default compiler compliance level for the current workspace is 1.6. The new project will use a project specific compiler compliance level of 1.7.
< Back
Next >
Cancel
Finish


New Java Project
Java Settings
Define the Java build settings.
Source
Projects
Libraries
Order and Export
WordCount
src
Details
Create new source folder: use this if you want to add a new source folder to your project.
Link additional source: use this if you have a folder in the file system that should be used as additional source folder.
Add project 'WordCount' to build path: Add the project to the build path if the project is the root of packages and source files. Entries on the build path are visible to the compiler and used for building.
Allow output folders for source folders
Default output folder:
WordCount/bin
Browse...
< Back
Next >
Cancel
Finish


cloudera-quickstart-vm-5.4.2-0-virtualbox [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Applications Places System
Sat Jun 5, 7:54 PM
cloudera
File Edit Navigate Search Project Run Window Help
Quick Access
Java
New Java Project
Java Settings
Define the Java build settings.
Source
Projects
Libraries
Order and Export
JARs and class folders on the build path:
activation-1.1.jar - /usr/lib/hadoop/client
activation.jar - /usr/lib/hadoop/client
apacheds-i18n-2.0.0-M15.jar - /usr/lib/hadoop/clie
apacheds-i18n.jar - /usr/lib/hadoop/client
apacheds-kerberos-codec-2.0.0-M15.jar - /usr/lib/
apacheds-kerberos-codec.jar - /usr/lib/hadoop/cli
api-asn1-api-1.0.0-M20.jar - /usr/lib/hadoop/client
api-asn1-api.jar - /usr/lib/hadoop/client
api-util-1.0.0-M20.jar - /usr/lib/hadoop/client
api-util.jar - /usr/lib/hadoop/client
avro.jar - /usr/lib/hadoop/client
aws-java-sdk-1.7.4.jar - /usr/lib/hadoop/client
aws-java-sdk.jar - /usr/lib/hadoop/client
commons-beanutils-1.7.0.jar - /usr/lib/hadoop/cli
commons-beanutils-core-1.8.0.jar - /usr/lib/hadoc
commons-beanutils-core.jar - /usr/lib/hadoop/clie
commons-beanutils.jar - /usr/lib/hadoop/client
commons-cli-1.2.jar - /usr/lib/hadoop/client
commons-cli.jar - /usr/lib/hadoop/client
commons-codec-1.4.jar - /usr/lib/hadoop/client
commons-codec.jar - /usr/lib/hadoop/client
Add JARs...
Add External JARs...
Add Variable...
Add Library...
Add Class Folder...
Add External Class Folder...
Edit...
Remove
Migrate JAR File...
< Back
Next >
Cancel
Finish
Problems
Javadoc
Declaration
Console
No consoles to display at this time.
Java - Eclipse
Type here to search
ENG
08:24 AM
06-06-2021


Package Explorer
training
StubDriver.java
import org.apache.hadoop.mapreduce.Jo
New
Go Into
Open in New Window
Open Type Hierarchy F4
Show In Shift+Alt+W
Copy Ctrl+C
Copy Qualified Name
Paste Ctrl+V
Delete Delete
Remove from Context Shift+Ctrl+Alt+Down
Build Path
Source Shift+Alt+S
Refactor Shift+Alt+T
Import...
Export...
Refresh F5
Close Project
Close Unrelated Projects
Assign Working Sets...
Debug As
Run As
Validate
Team
Compare With
Restore from Local History...
Configure
Properties Alt+Enter
Java Project
Project...
Package
Class
Interface
Enum
Annotation
Source Folder
Java Working Set
Folder
File
Untitled Text File
JUnit Test Case
Task
Example...
Other... Ctrl+N
Declaration



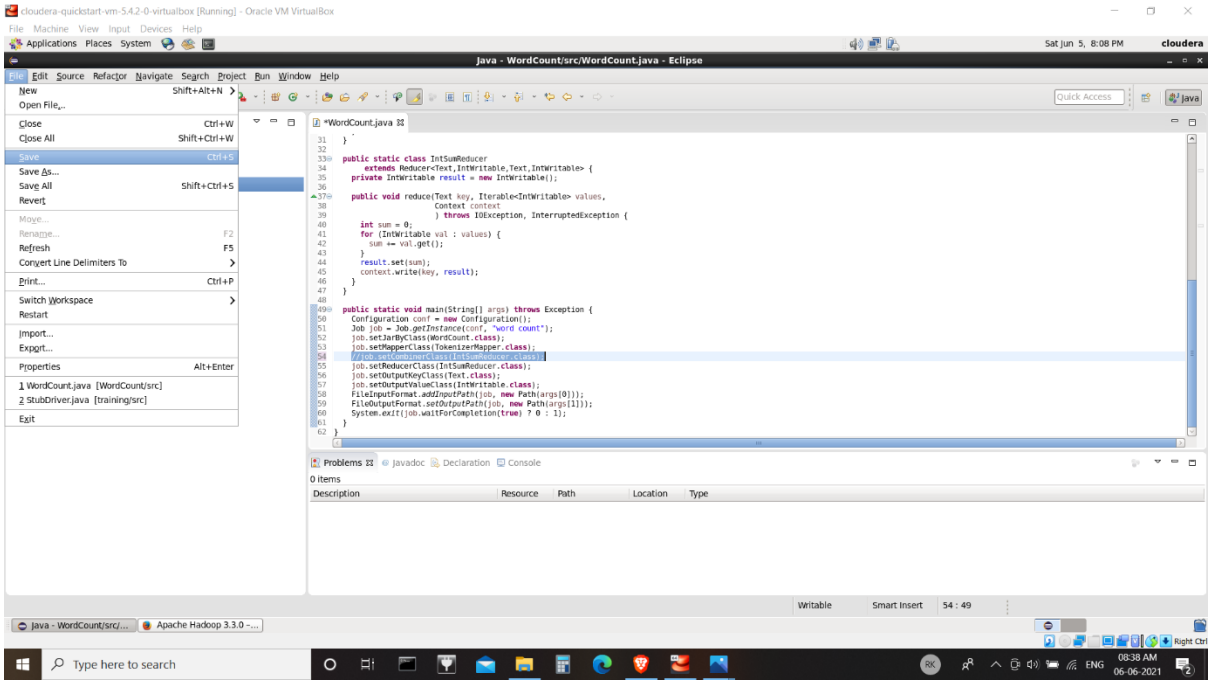

- **In this class file Word Count code will be added, after code is added save the file.**

  Code:

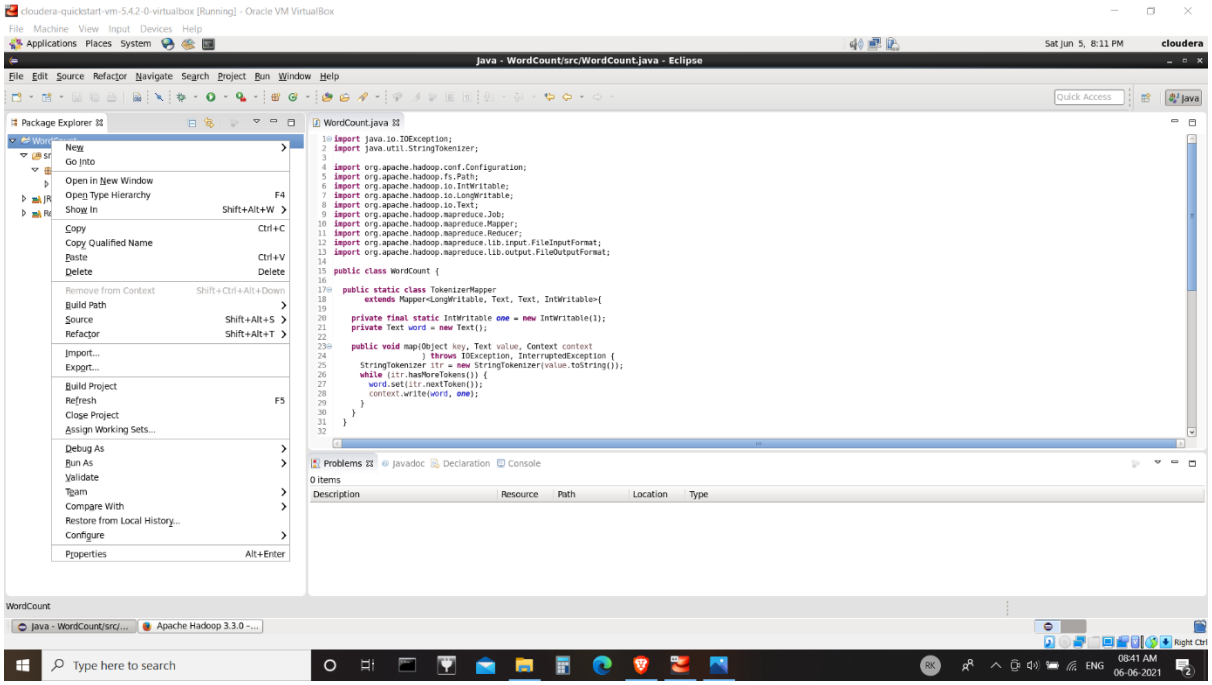
```
import java.io.IOException;

import java.util.StringTokenizer;

import org.apache.hadoop.conf.Configuration;

import org.apache.hadoop.fs.Path;

import org.apache.hadoop.io.IntWritable;
```

```
import org.apache.hadoop.io.LongWritable;
import org.apache.hadoop.io.Text;
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 WordCount {
 public static class TokenizerMapper
      extends Mapper<LongWritable, Text, Text, IntWritable>{
  private final static IntWritable one = new IntWritable(1);
  private Text word = new Text();
  public void map(LongWritable key, Text value, Context context
                  ) throws IOException, InterruptedException {
    StringTokenizer itr = new StringTokenizer(value.toString());
    while (itr.hasMoreTokens()) {
      word.set(itr.nextToken());
      context.write(word, one);
    }
  }
 }

 public static class IntSumReducer
      extends Reducer<Text,IntWritable,Text,IntWritable> {
  private IntWritable result = new IntWritable();
  public void reduce(Text key, Iterable<IntWritable> values,
                     Context context
                     ) throws IOException, InterruptedException {
```

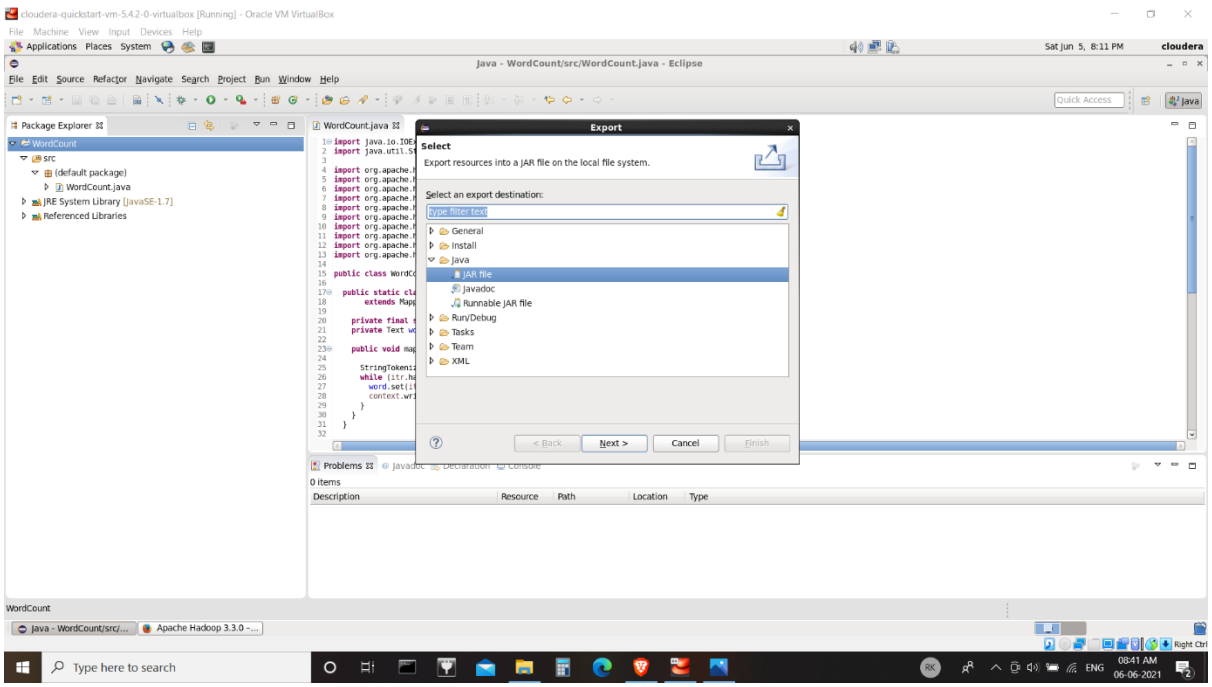
```
    int sum = 0;
    for (IntWritable val : values) {
      sum += val.get();
    }
    result.set(sum);
    context.write(key, result);
  }
 }
 public static void main(String[] args) throws Exception {
  Configuration conf = new Configuration();
  Job job = Job.getInstance(conf, "word count");
  job.setJarByClass(WordCount.class);
  job.setMapperClass(TokenizerMapper.class);
  //job.setCombinerClass(IntSumReducer.class);
  job.setReducerClass(IntSumReducer.class);
  job.setOutputKeyClass(Text.class);
  job.setOutputValueClass(IntWritable.class);
  FileInputFormat.addInputPath(job, new Path(args[0]));
  FileOutputFormat.setOutputPath(job, new Path(args[1]));
  System.exit(job.waitForCompletion(true) ? 0 : 1);
 }
}
```

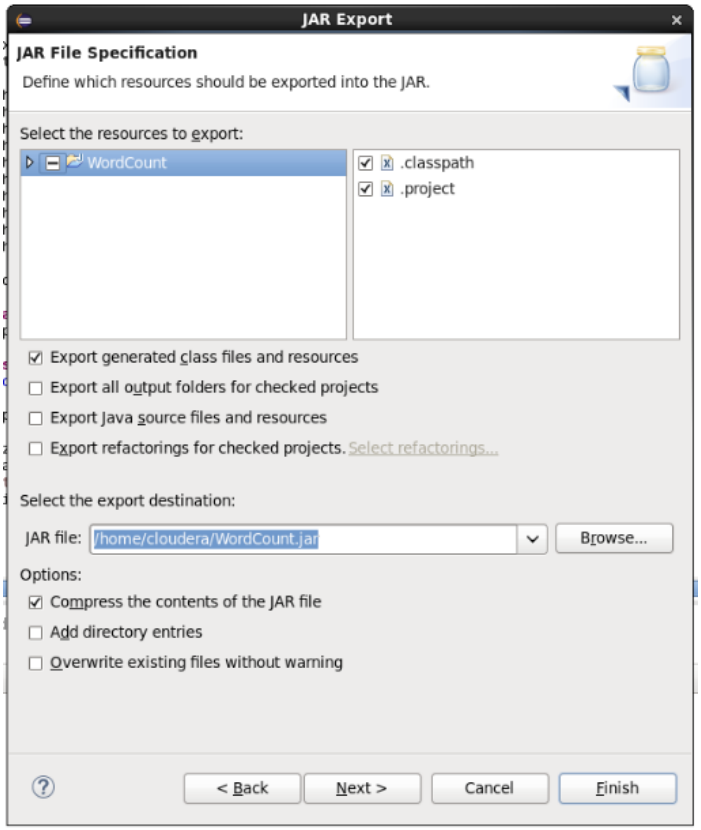



- **Once class file is created in order to run the project it will be exported in order to do so right click on WordCount Project file and select export.**

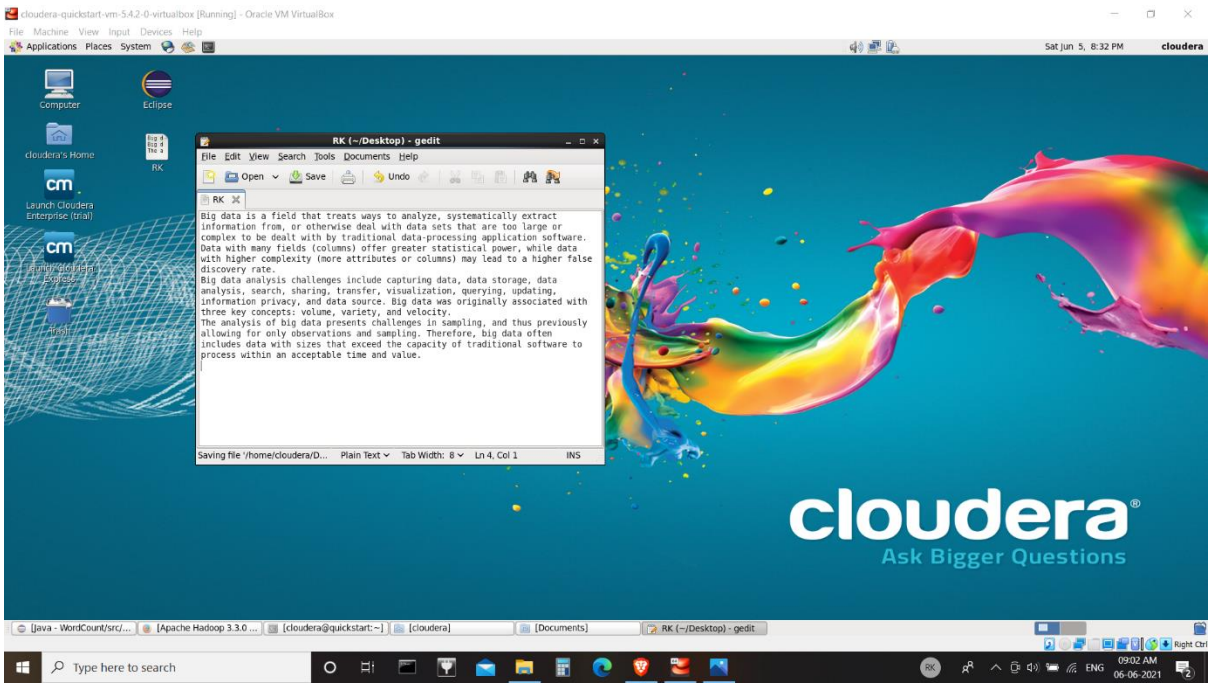


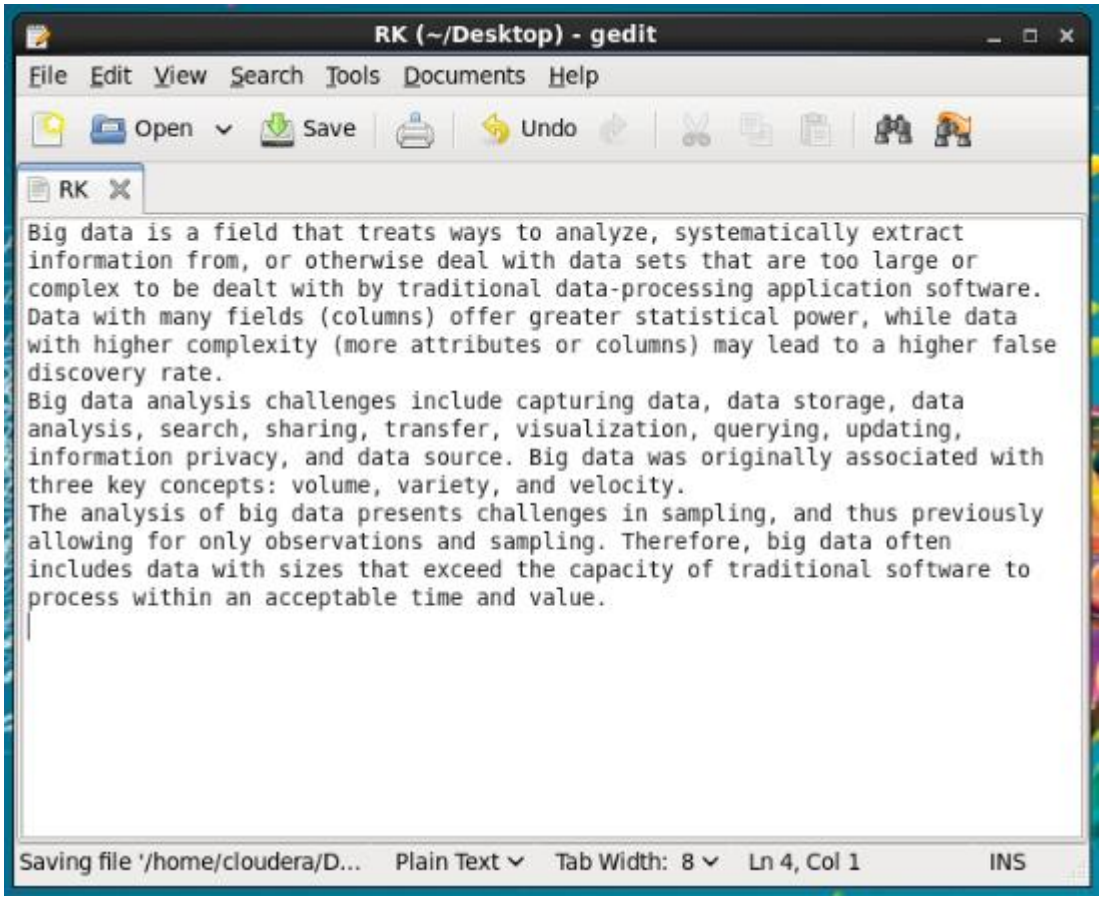
- **In export window select jar file and then click on next, in jar file specification browse WordCount.jar and then click on Finish.**



- **Now in order to run the jar file we will need an input file. Create a new file then name it, enter text in it and save it.**



- **Open terminal and in it enter the following commands.**
  - **Input:**

    pwd

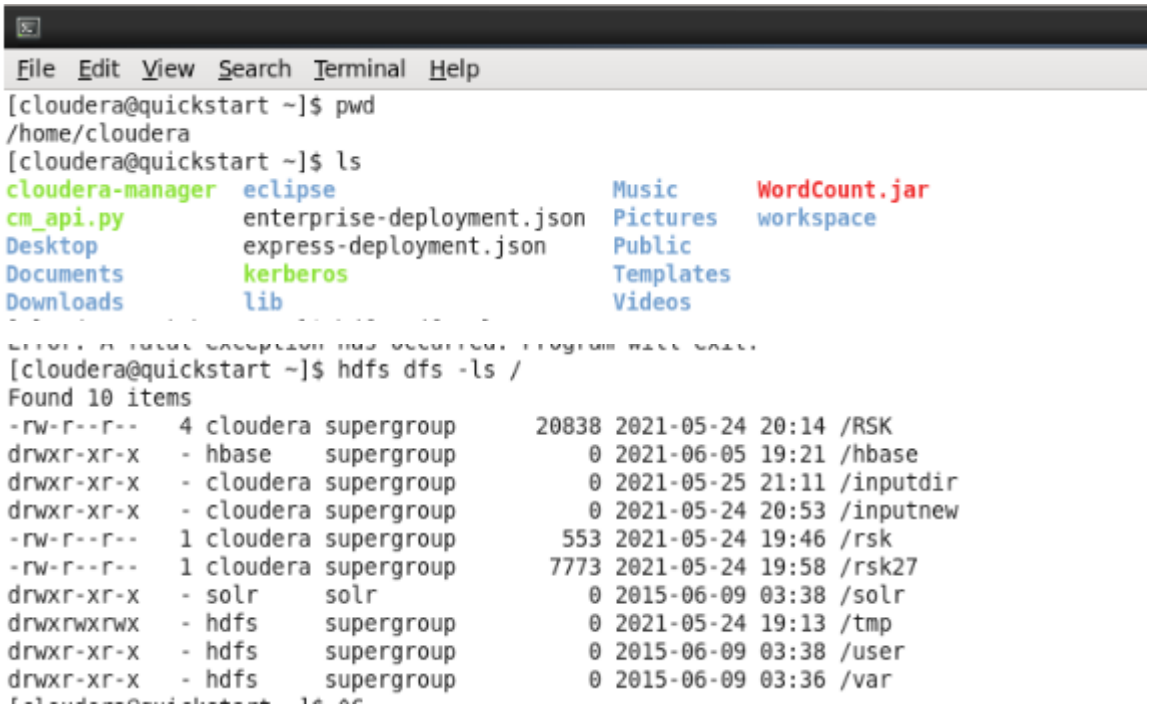
    ls

    hdfs dfs -ls /

    **Description:**

    pwd: check current directory location

    **Output:**




  - **Input:**

    hdfs dfs -mkdir /inputdir

    hdfs dfs -put /home/cloudera/Desktop/RK /inputdir/

    hdfs dfs -ls /inputdir

    **Description:**

    Create a new directory as inputdir and copy file RK from desktop to inputdir the show contents of inputdir

**Output:**

```
[cloudera@quickstart ~]$ hdfs dfs -mkdir /inputdir
[cloudera@quickstart ~]$ hdfs dfs -put /home/cloudera/Desktop/RK /inputdir/
[cloudera@quickstart ~]$ hdfs dfs -ls /inputdir
Found 1 items
-rw-r--r--   1 cloudera supergroup       1910 2021-06-05 20:18 /inputdir/RK
[cloudera@quickstart ~]$ htfs dfs -cat /inputdir/RK
```

- **Input:**

hdfs dfs -cat /inputdir/RK

**Description:**

Display content of RK file

**Output:**

```
[cloudera@quickstart ~]$ hdfs dfs -cat /inputdir/RK
Big data is a field that treats ways to analyze, systematically extract information from,
or otherwise deal with data sets that are too large or complex to be dealt with by traditional data-processing application software
 Data with many fields columns offer greater statistical power, while data with higher complexity more attributes or columns may lead to a higher false discovery rate
Big data analysis challenges include capturing data, data storage, data analysis, search, sharing, transfer, visualization, querying, updating, information privacy, and data source
Big data was originally associated with three key concepts: volume, variety, and velocity.
The analysis of big data presents challenges in sampling, and thus previously allowing for only observations and sampling
Therefore, big data often includes data with sizes that exceed the capacity of traditional software to process within an acceptable time and value.
```

- **Input:**

hadoop jar /home/cloudera/WordCount.jar WordCount /inputdir/RK /outputdir

**Description:**

Run WordCount.jar take input file RK from inputdir and store output file in outputdir

**Output:**

```
[cloudera@quickstart ~]$ hadoop jar /home/cloudera/WordCount.jar WordCount /inputdir/RK /outputdir
21/06/05 20:52:36 INFO client.RMProxy: Connecting to ResourceManager at /0.0.0.0:8032
21/06/05 20:52:37 WARN mapreduce.JobSubmitter: Hadoop command-line option parsing not performed. Implement the Tool interface and exec
21/06/05 20:52:37 INFO input.FileInputFormat: Total input paths to process : 1
21/06/05 20:52:37 INFO mapreduce.JobSubmitter: number of splits:1
21/06/05 20:52:38 INFO mapreduce.JobSubmitter: Submitting tokens for job: job_1622946039261_0006
21/06/05 20:52:38 INFO impl.YarnClientImpl: Submitted application application_1622946039261_0006
21/06/05 20:52:38 INFO mapreduce.Job: The url to track the job: http://quickstart.cloudera:8088/proxy/application_1622946039261_0006/
21/06/05 20:52:38 INFO mapreduce.Job: Running job: job_1622946039261_0006
21/06/05 20:52:50 INFO mapreduce.Job: Job job_1622946039261_0006 running in uber mode : false
21/06/05 20:52:50 INFO mapreduce.Job:  map 0% reduce 0%
21/06/05 20:52:58 INFO mapreduce.Job:  map 100% reduce 0%
21/06/05 20:53:10 INFO mapreduce.Job:  map 100% reduce 100%
21/06/05 20:53:11 INFO mapreduce.Job: Job job_1622946039261_0006 completed successfully
21/06/05 20:53:12 INFO mapreduce.Job: Counters: 49
        File System Counters
                FILE: Number of bytes read=1758
                FILE: Number of bytes written=223789
                FILE: Number of read operations=0
                FILE: Number of large read operations=0
                FILE: Number of write operations=0
                HDFS: Number of bytes read=1042
                HDFS: Number of bytes written=928
                HDFS: Number of read operations=6
                HDFS: Number of large read operations=0
                HDFS: Number of write operations=2
        Job Counters
                Launched map tasks=1
                Launched reduce tasks=1
                Data-local map tasks=1
```

```
		Total time spent by all maps in occupied slots (ms)=6986
		Total time spent by all reduces in occupied slots (ms)=8903
		Total time spent by all map tasks (ms)=6986
		Total time spent by all reduce tasks (ms)=8903
		Total vcore-seconds taken by all map tasks=6986
		Total vcore-seconds taken by all reduce tasks=8903
		Total megabyte-seconds taken by all map tasks=7153664
		Total megabyte-seconds taken by all reduce tasks=9116672
	Map-Reduce Framework
		Map input records=8
		Map output records=137
		Map output bytes=1478
		Map output materialized bytes=1758
		Input split bytes=108
		Combine input records=0
		Combine output records=0
		Reduce input groups=99
		Reduce shuffle bytes=1758
		Reduce input records=137
		Reduce output records=99
		Spilled Records=274
		Shuffled Maps =1
		Failed Shuffles=0
		Merged Map outputs=1
		GC time elapsed (ms)=170
		CPU time spent (ms)=1620
		Physical memory (bytes) snapshot=340320256
		Virtual memory (bytes) snapshot=3007533056
		Total committed heap usage (bytes)=226365440
	Shuffle Errors


		BAD_ID=0
		CONNECTION=0
		IO_ERROR=0
		WRONG_LENGTH=0
		WRONG_MAP=0
		WRONG_REDUCE=0
File Input Format Counters
		Bytes Read=934
File Output Format Counters
		Bytes Written=928
```