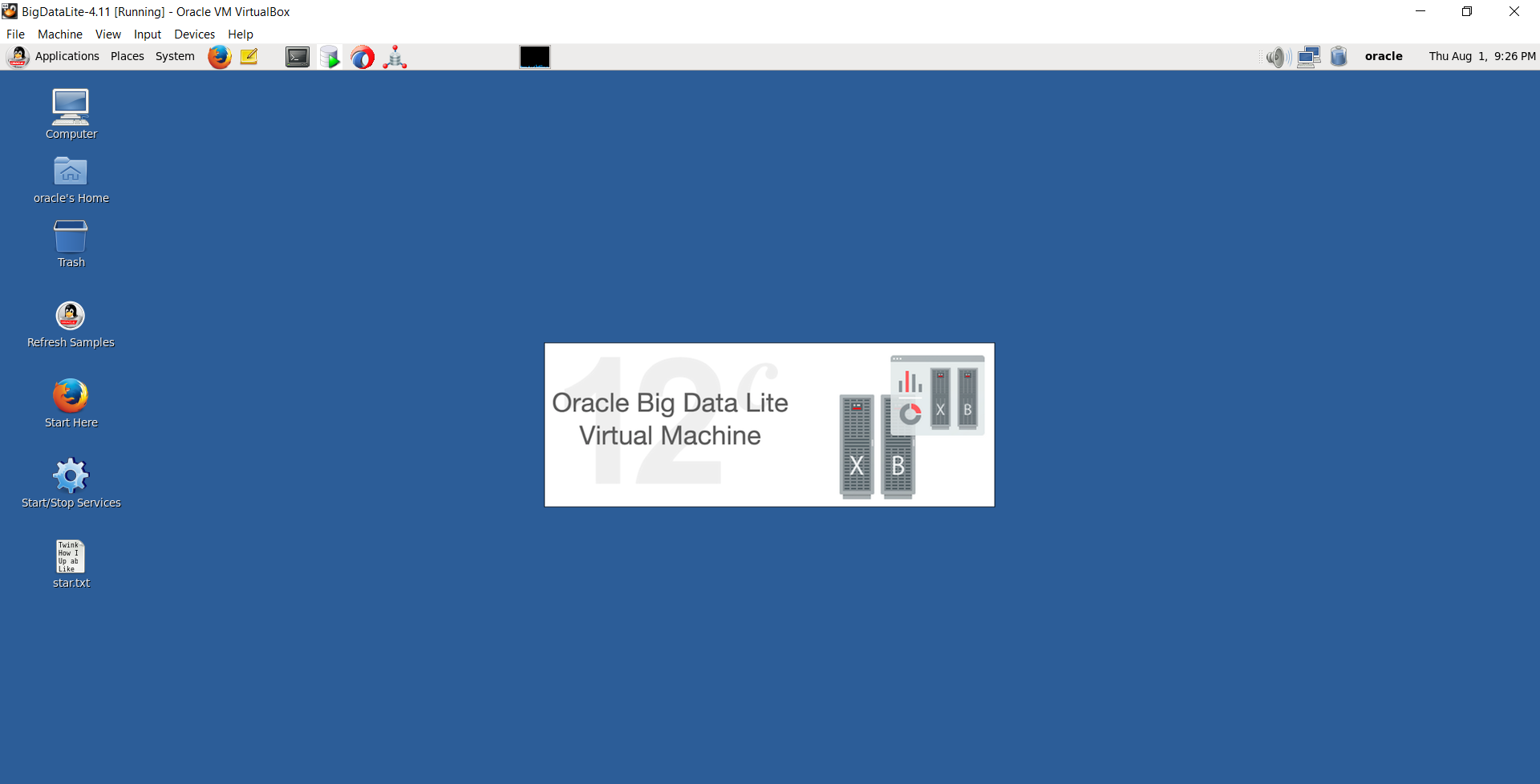
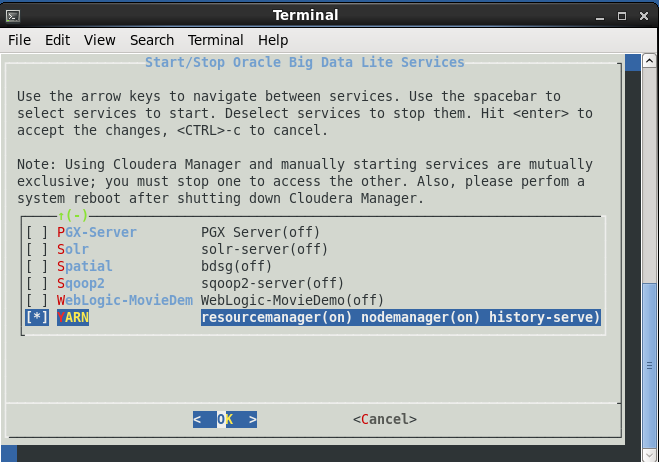
Assignment 6: Use Map Reduce for Word Count

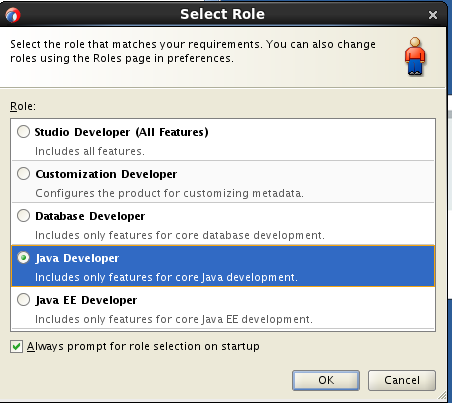
CIS: Big Data Solutions for Business

Karishma Borse 700701566

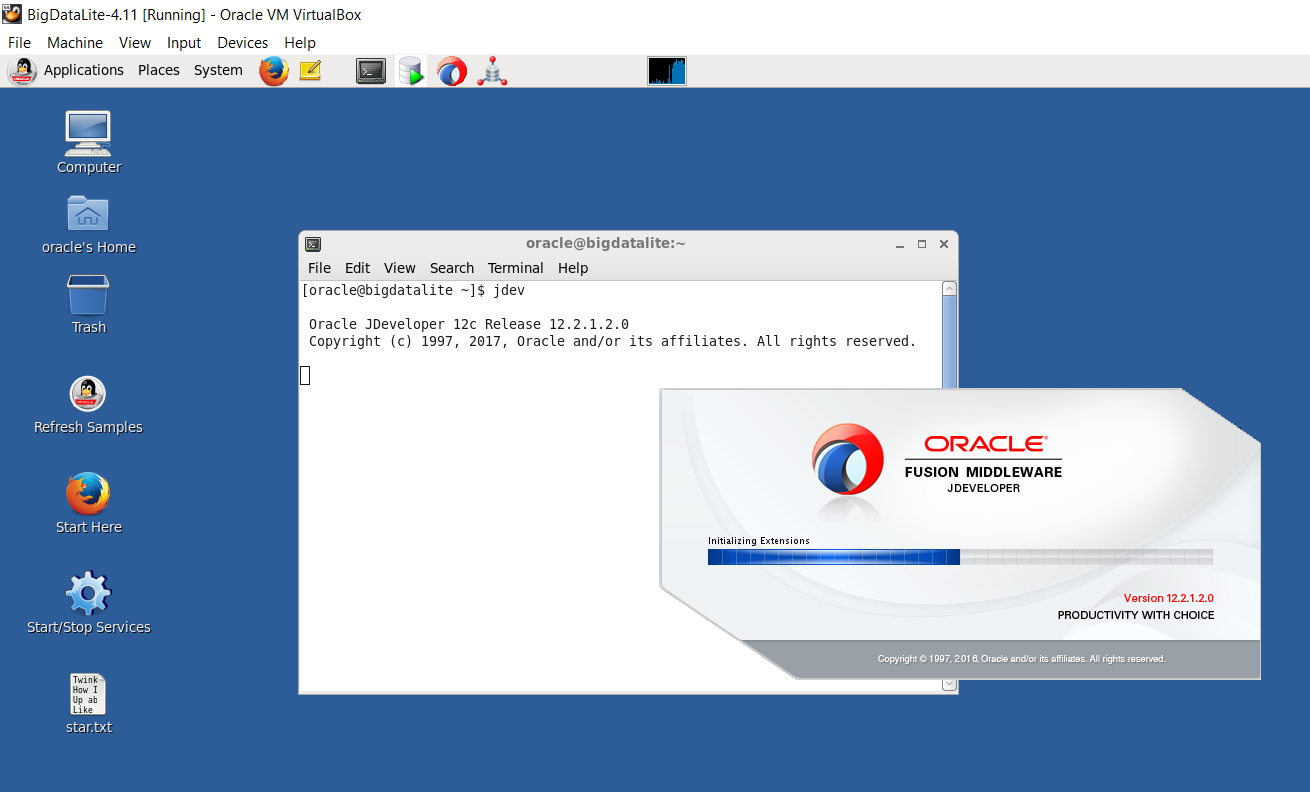
1. **Login to the BigDatalite oracle.**



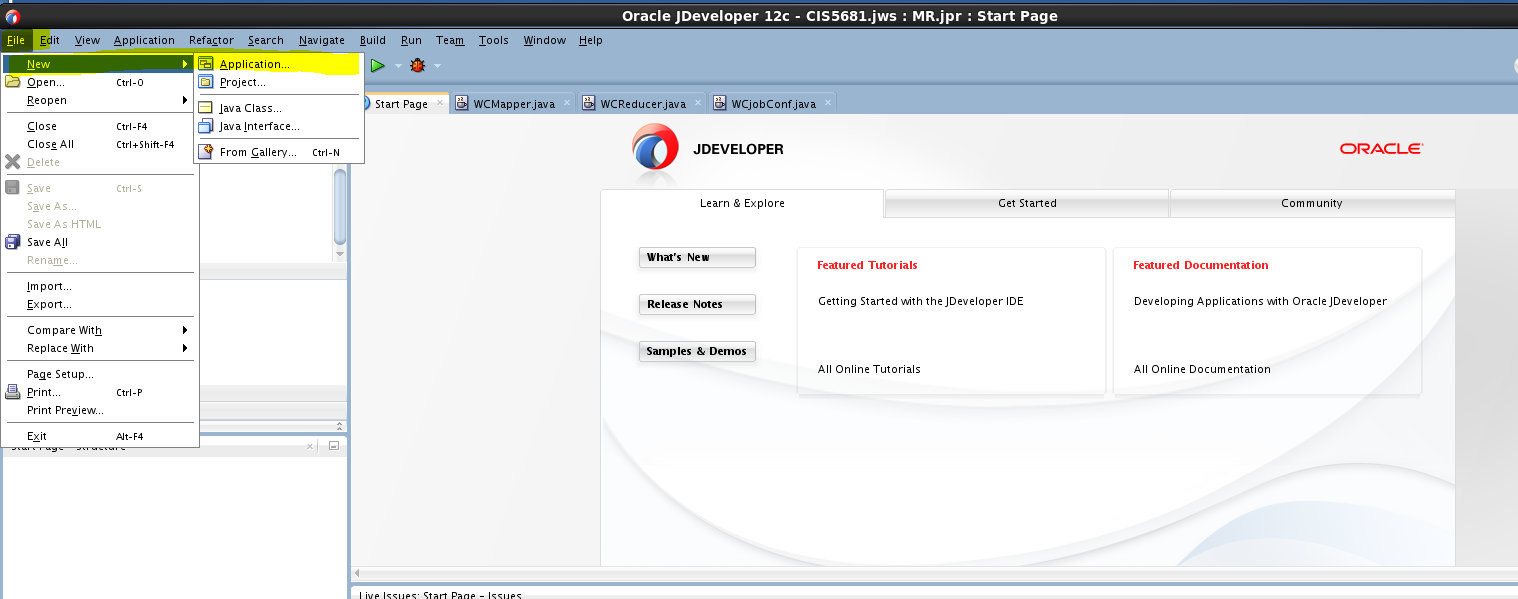
1. **Configure the start stop service. Set HDFS and YARN service on and click okay**
2. **Open the terminal and start jdev**
3. Click ok



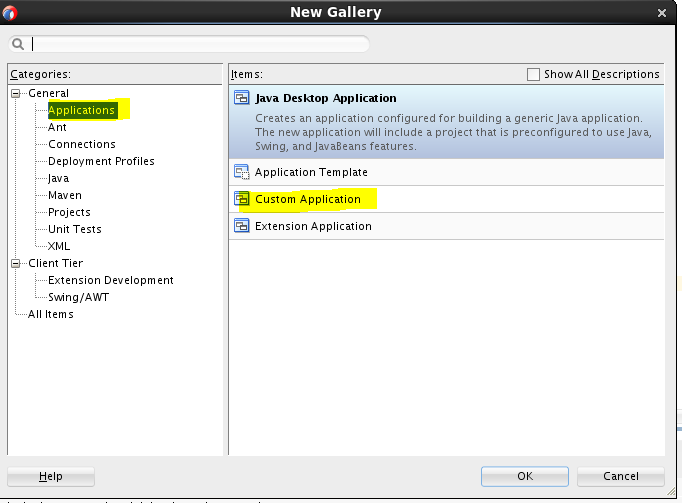
1. Jdeveloper is started

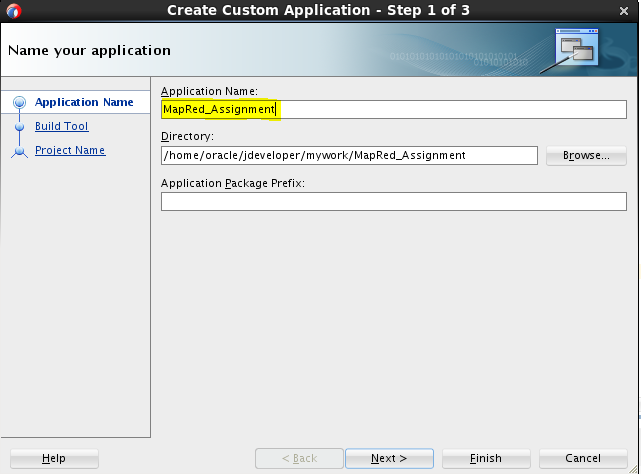
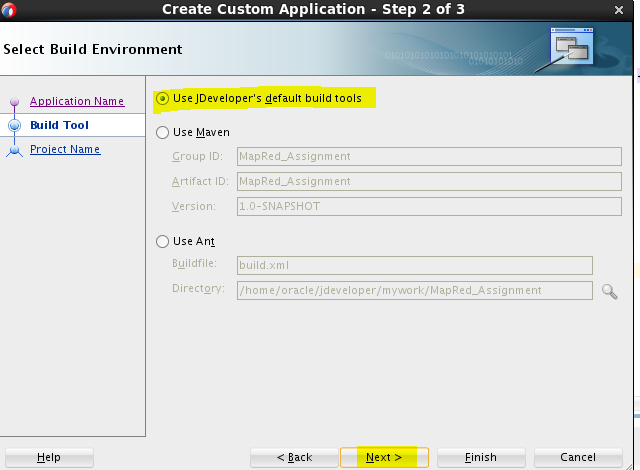
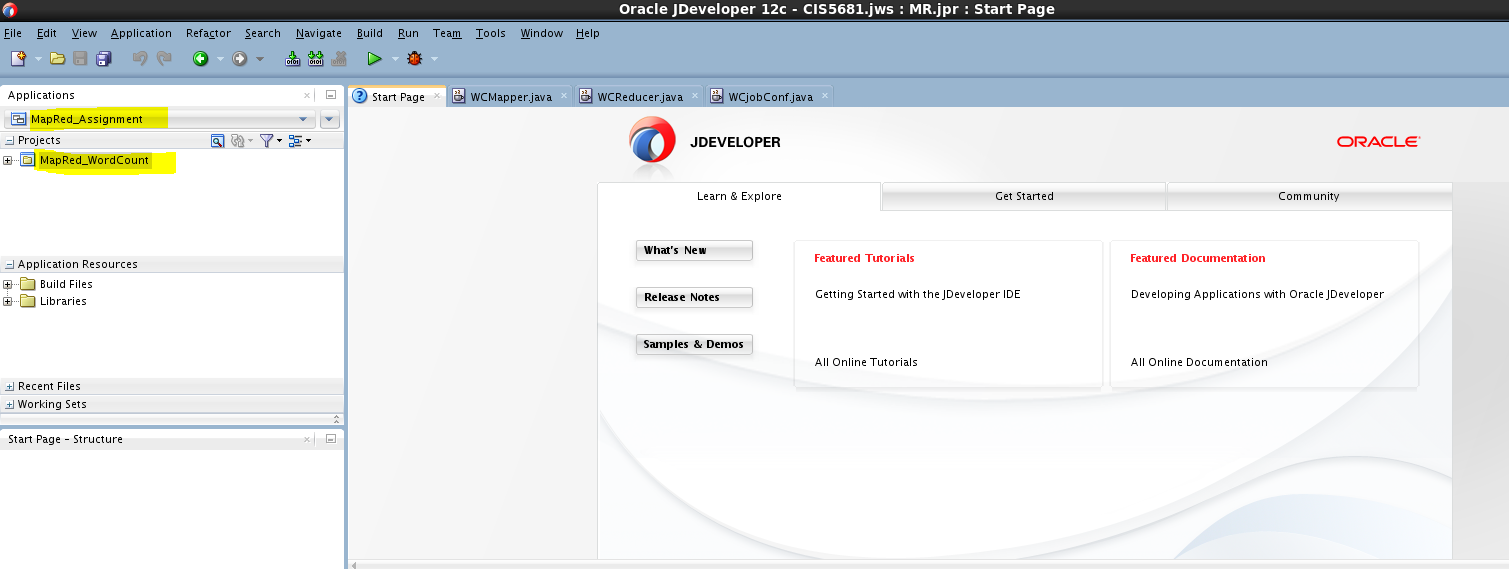
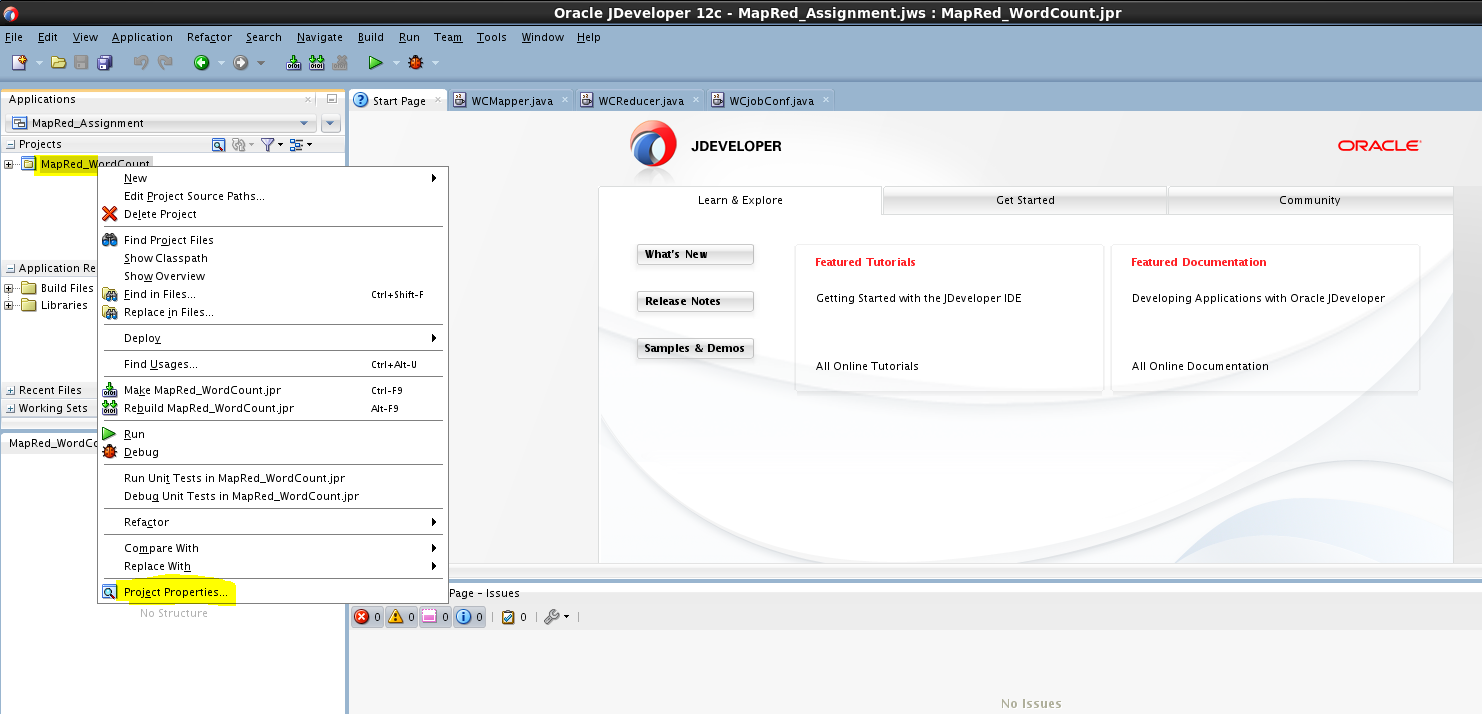
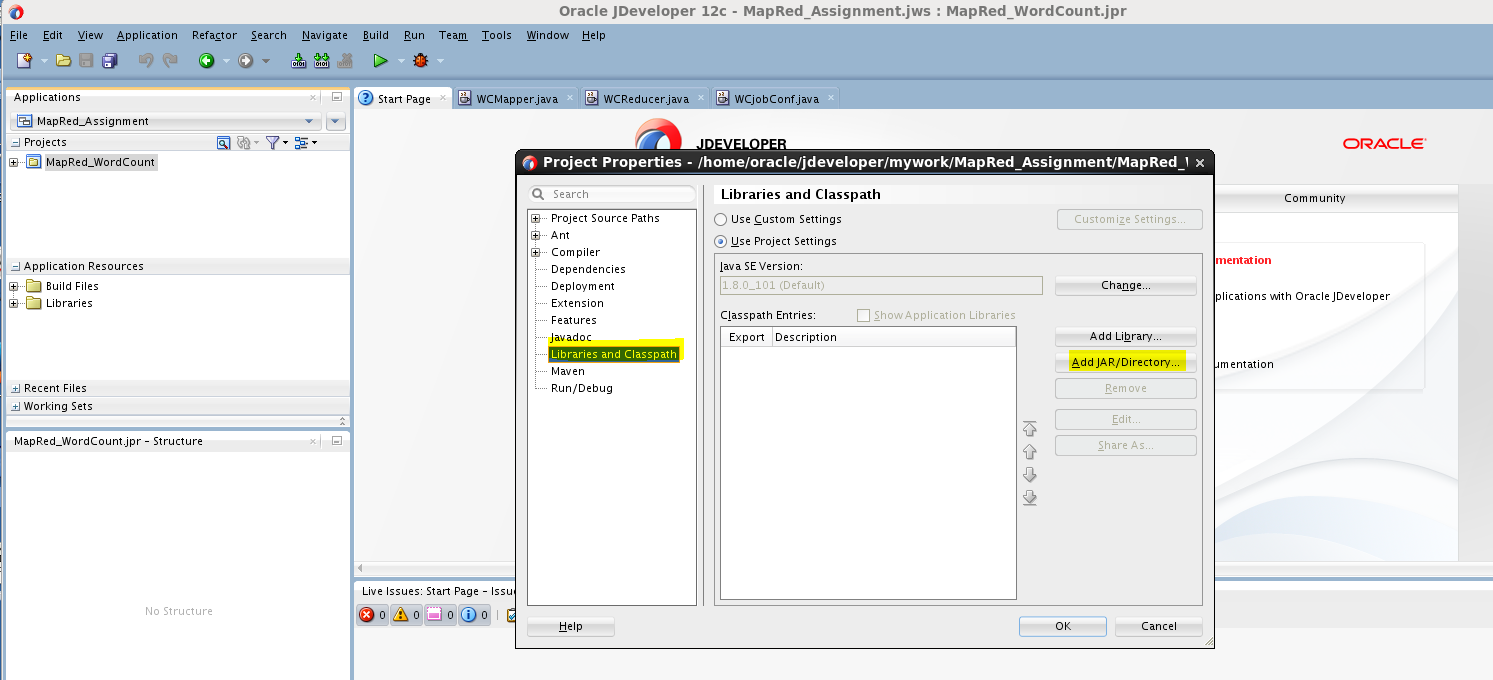
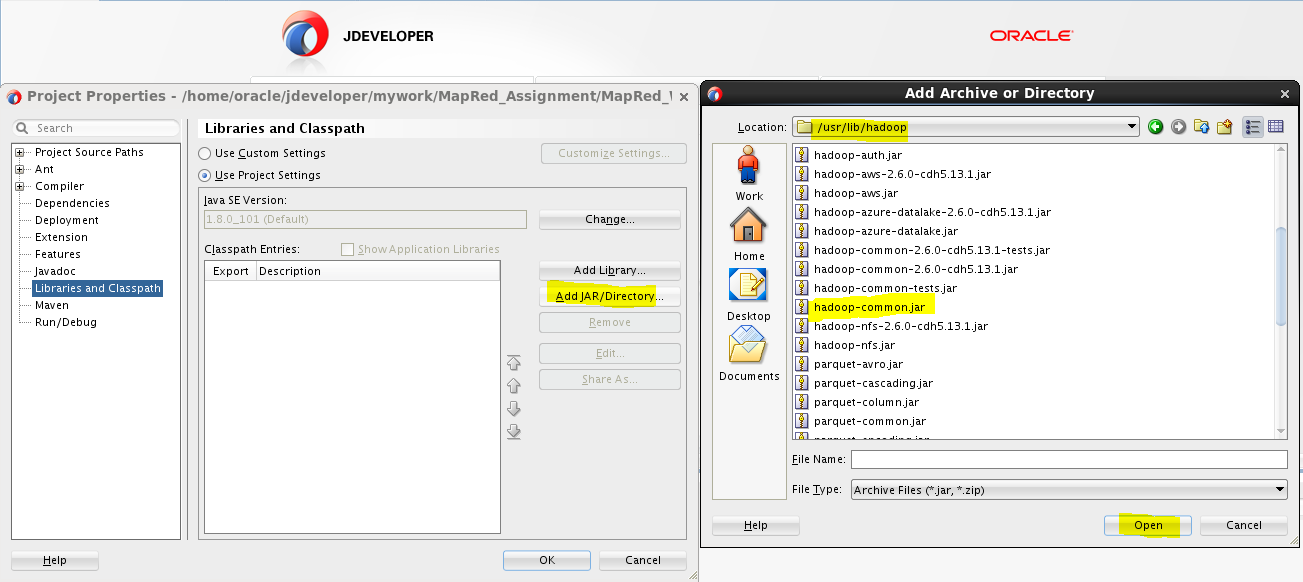


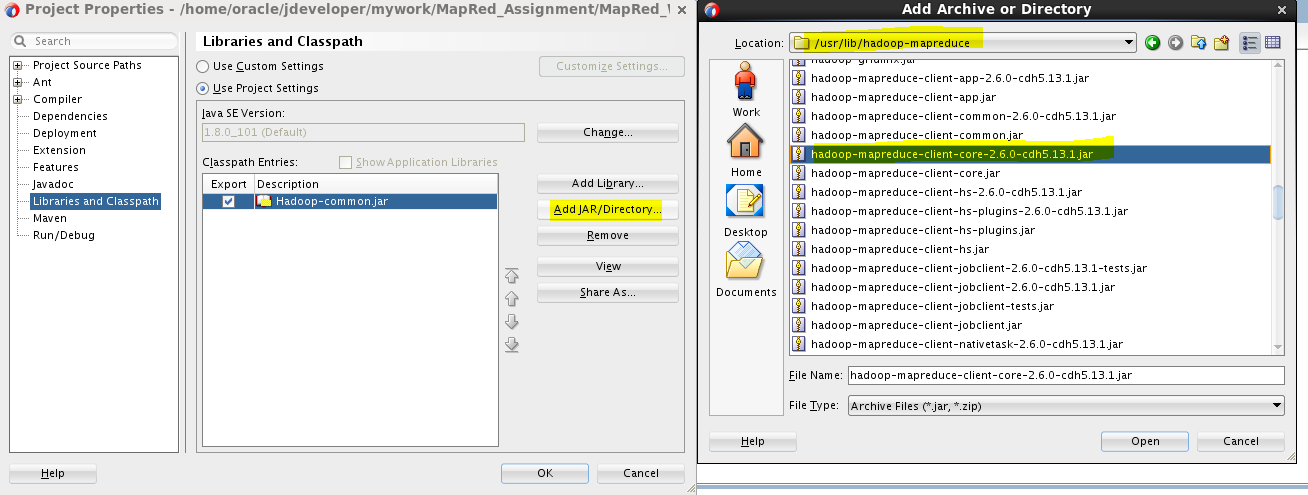
**4. Go to file. Select Application**

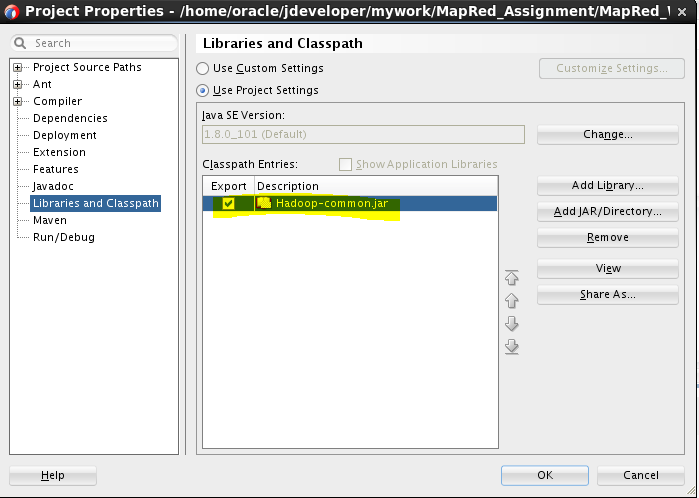


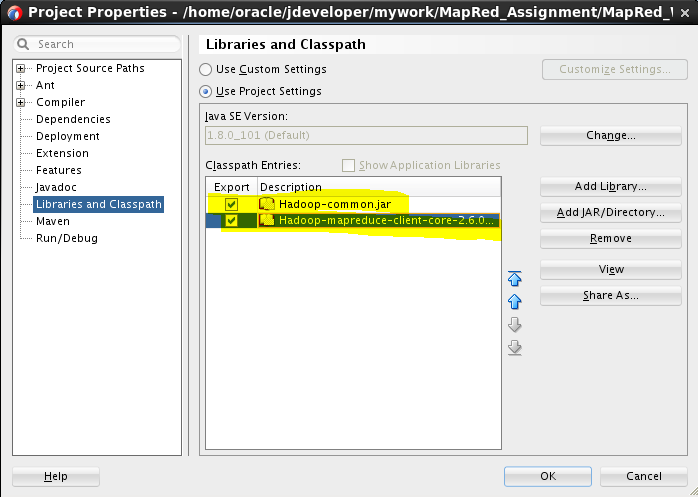
**5.Select custom application**



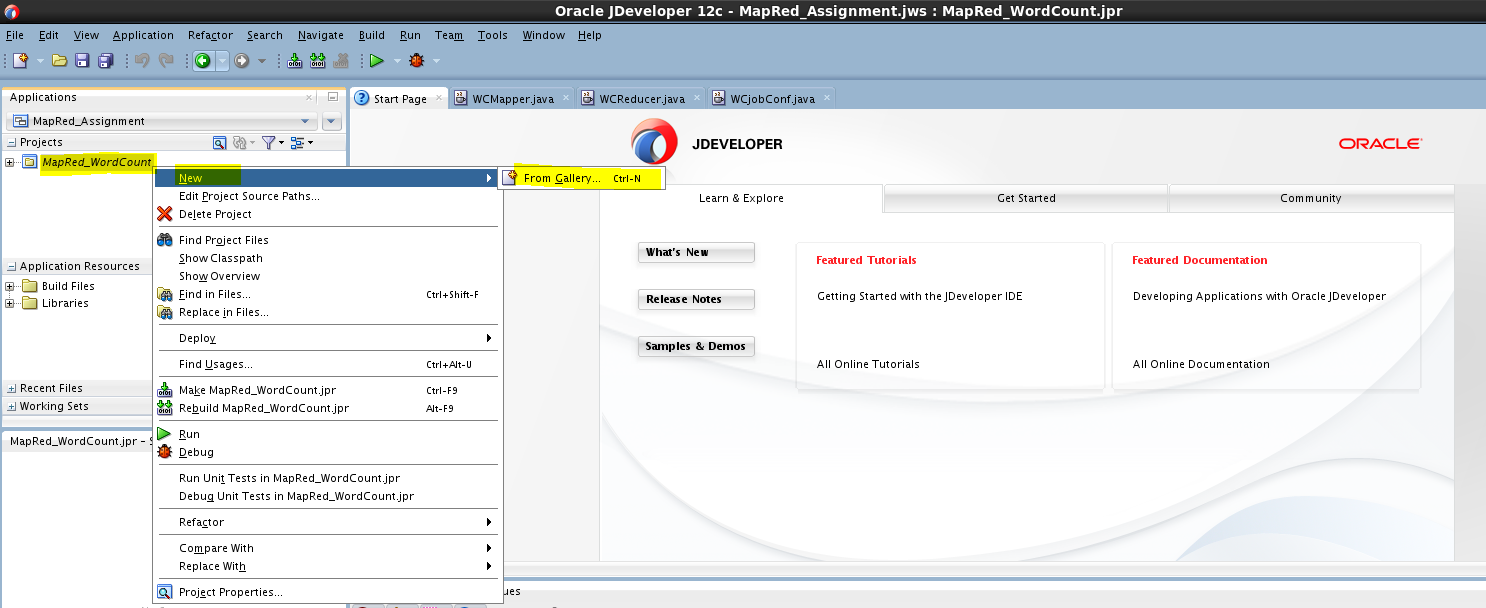
1. **Give application name and click next**
2. Select JDeveloper’s default build tools and click next
3. Enter the Project Name and click Finish
4. the Application and Project is created
5. **Set libraries and classpath**
6. Right click on the project name and select Project Properties
7. Select Libriaries and classpath 
8. On the right pane, click Add JAR/Directory



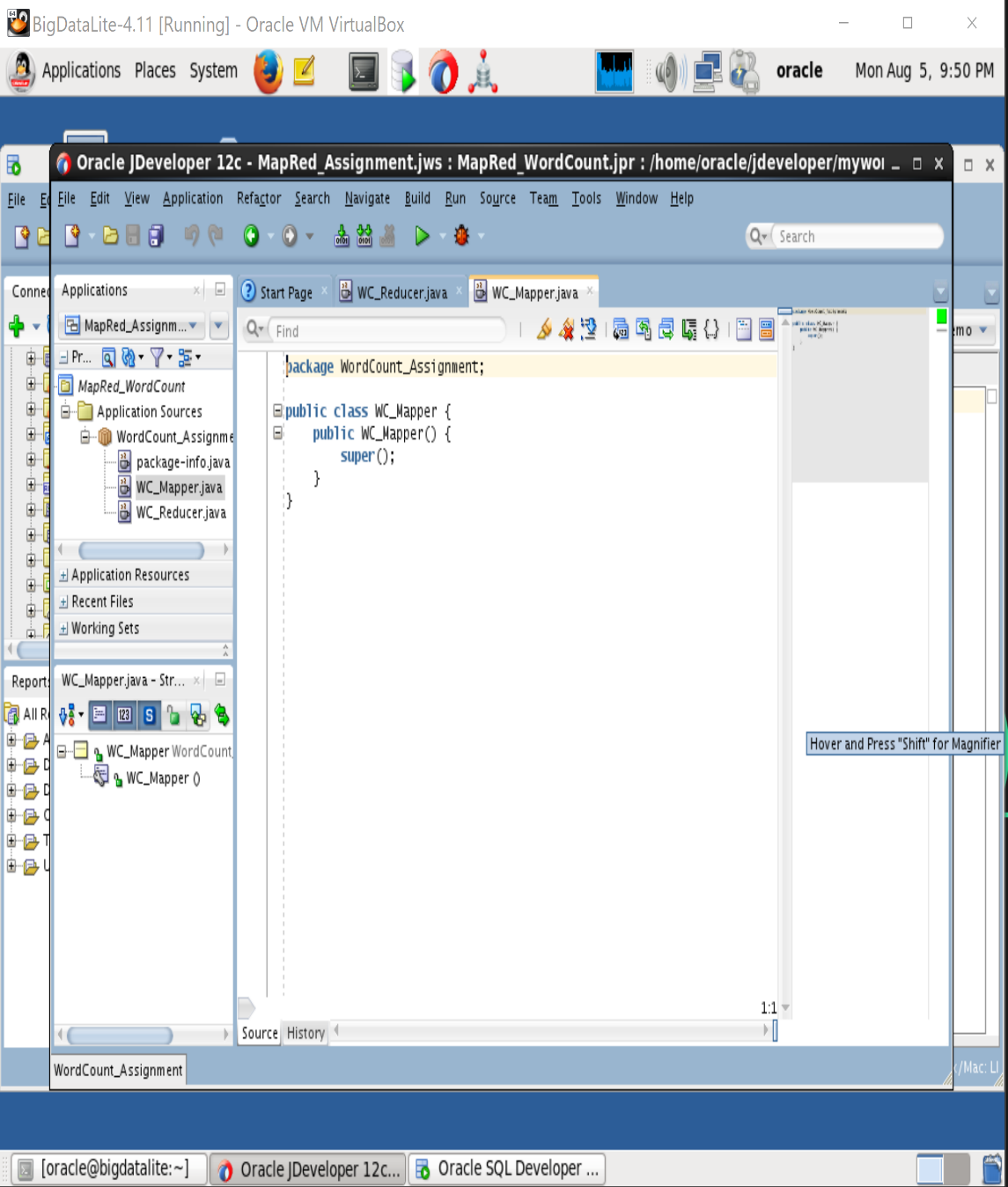
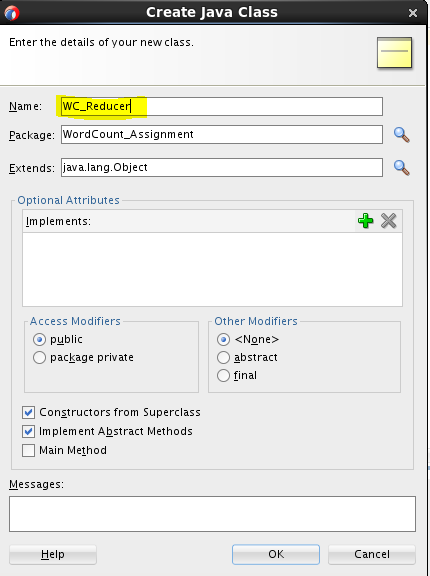
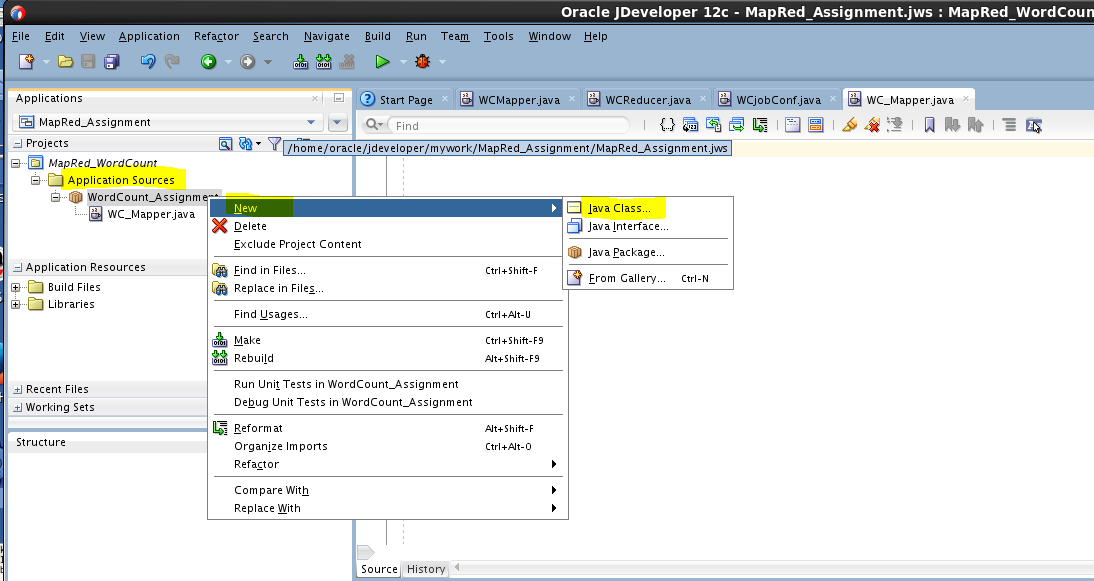
1. Select the jars

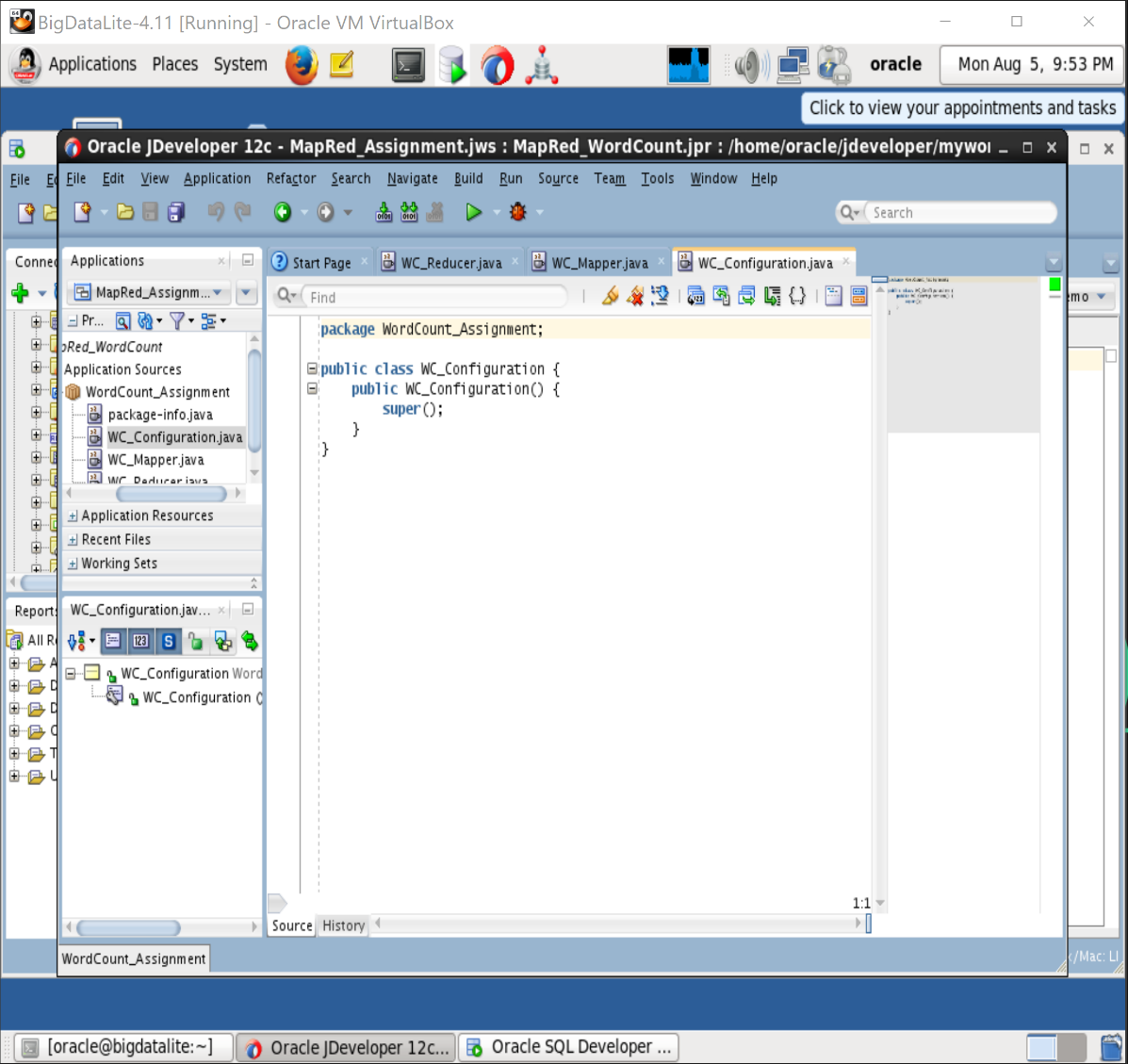


1. **Create and design the mapper**



1. **Create java package**





**10.WC\_Mapper**

import java.io.IOException;

import javax.naming.Context;

import org.apache.hadoop.io.IntWritable;

import org.apache.hadoop.io.LongWritable;

import org.apache.hadoop.io.Text;

import org.apache.hadoop.mapreduce.Mapper;

public class WordCountMapper extends Mapper<LongWritable, Text, Text, IntWritable> {

public void map(LongWritable key, Text value, Context context)

throws IOException, InterruptedException {

String line = value.toString();

for (String word : line.split("\\W+")) {

if (word.length() > 0) {

context.write(new Text(word), new IntWritable(1));

}

}

}

}



**11. Reducer**

import java.io.IOException;

import org.apache.hadoop.io.IntWritable;

import org.apache.hadoop.io.Text;

import org.apache.hadoop.mapreduce.Reducer;

public class WordCountReducer extends Reducer<Text,IntWritable,Text,IntWritable>

{

public void reduce(Text key,Iterable<IntWritable> values, Context context)

throws IOException, InterruptedException{

int wordcount = 0;

for (IntWritable value : values){

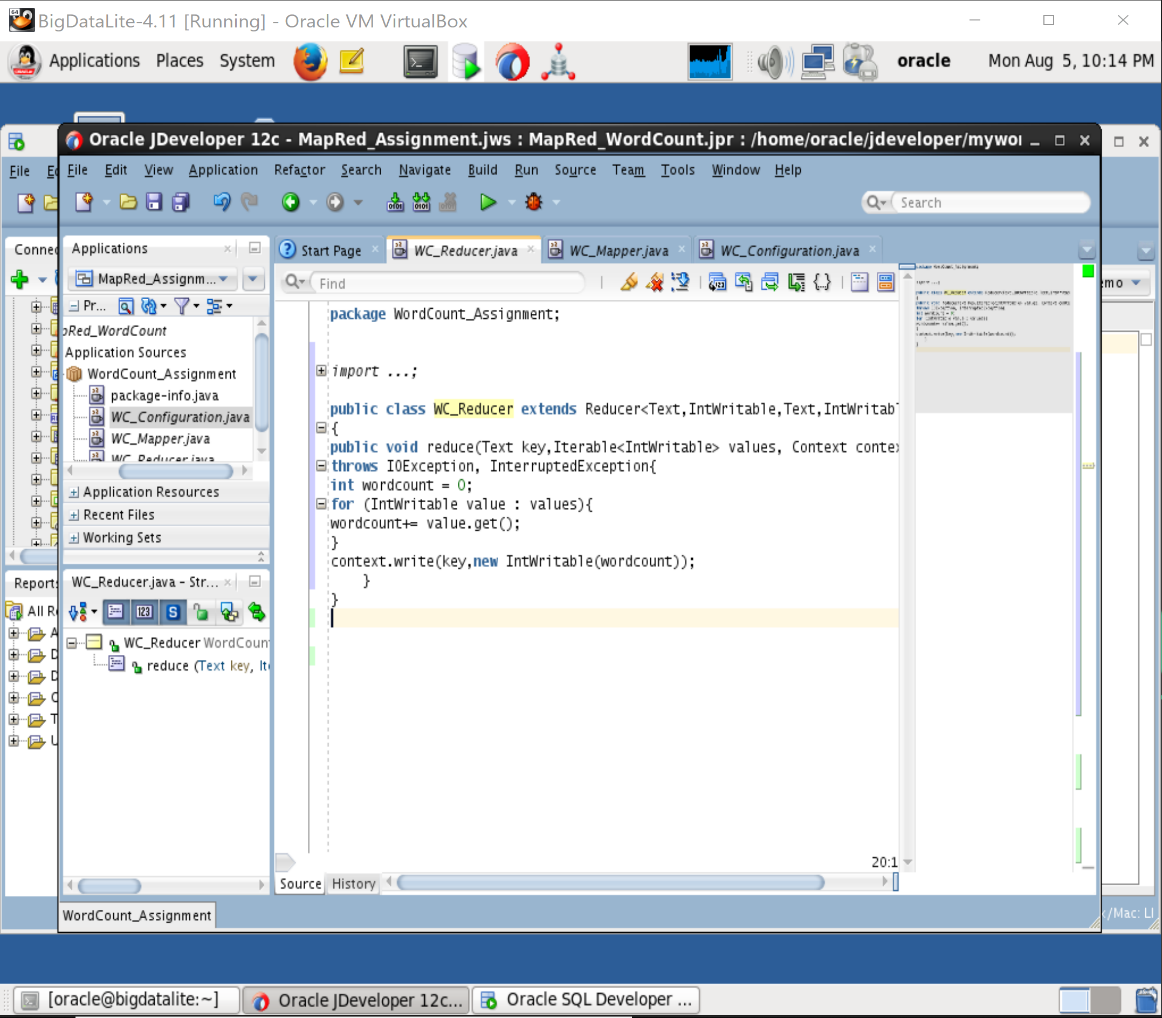
wordcount+= value.get();

}

context.write(key,new IntWritable(wordcount));

}

}



12.Configuration File

**import** java.io.IOException;

**import** org.apache.hadoop.fs.Path;

**import** org.apache.hadoop.io.\*;

**import** org.apache.hadoop.mapreduce.lib.input.FileInputFormat;

**import** org.apache.hadoop.mapreduce.lib.output.FileOutputFormat;

**import** org.apache.hadoop.mapreduce.Job;

**public** **class** WC\_Configuration{

**public** **static** **void** main(String[] args) **throws** IOException, ClassNotFoundException, InterruptedException {

**if**(args.length!=2){

System.***err***.println("Usage:Comment wordcount <input dir><output dir>\n");

System.*exit*(-1);

}

Job job = **new** Job();

job.setJarByClass(WC\_JobConf.**class**);

job.setJobName("WordCount");

FileInputFormat.*addInputPath*(job, **new** Path(args[0]));

FileOutputFormat.*setOutputPath*(job, **new** Path(args[1]));

//Speify the Mapper class

job.setMapperClass(WC\_Mapper.**class**);

//Specify the Reducer Class

job.setReducerClass(WC\_Reducer.**class**);

//Specify the output key of the mapper class

job.setMapOutputKeyClass(Text.**class**);

//Specify the output value of the mapper class

job.setMapOutputValueClass(IntWritable.**class**);

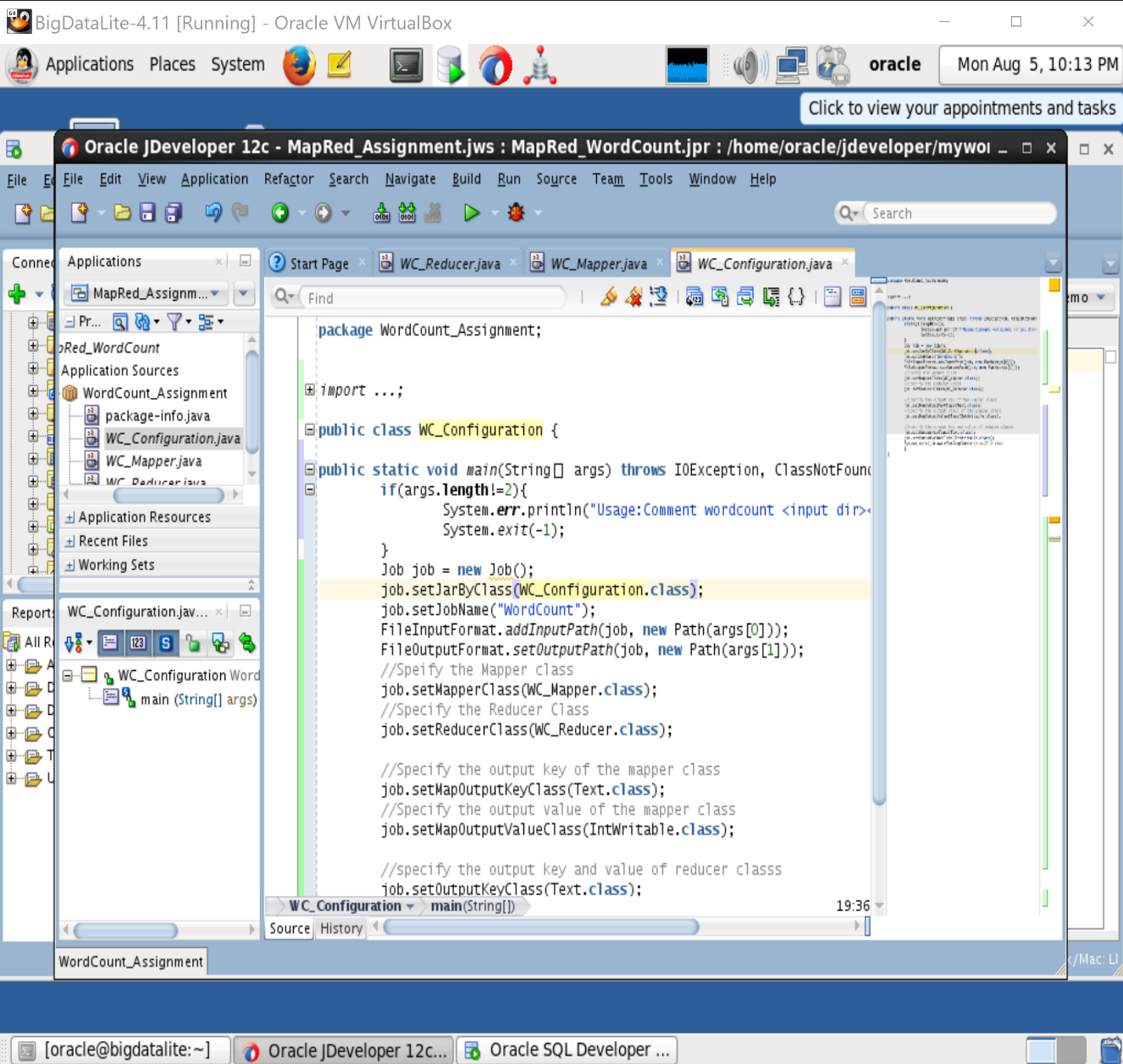
//specify the output key and value of reducer classs

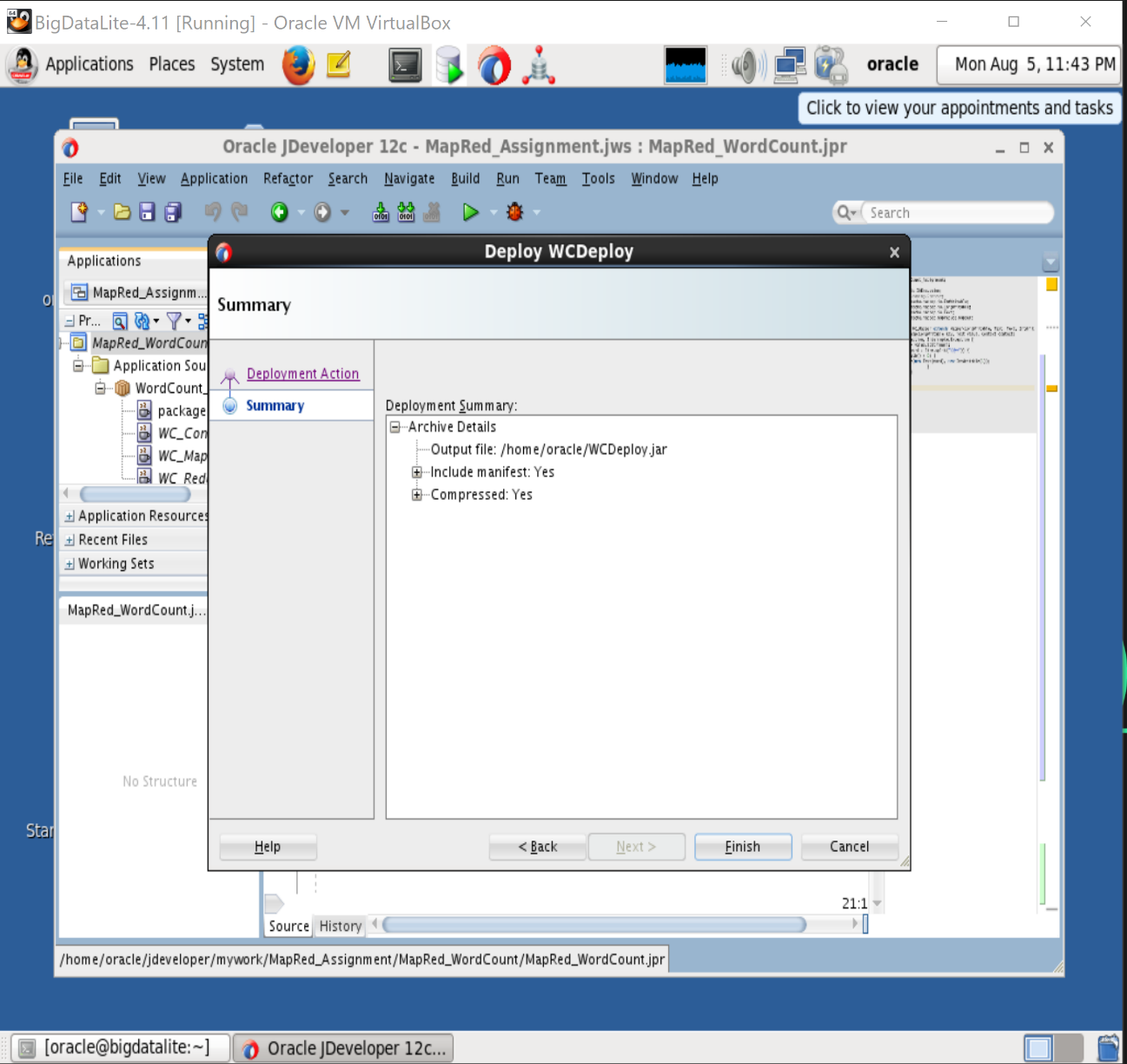
job.setOutputKeyClass(Text.**class**);

job.setOutputValueClass(IntWritable.**class**);

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

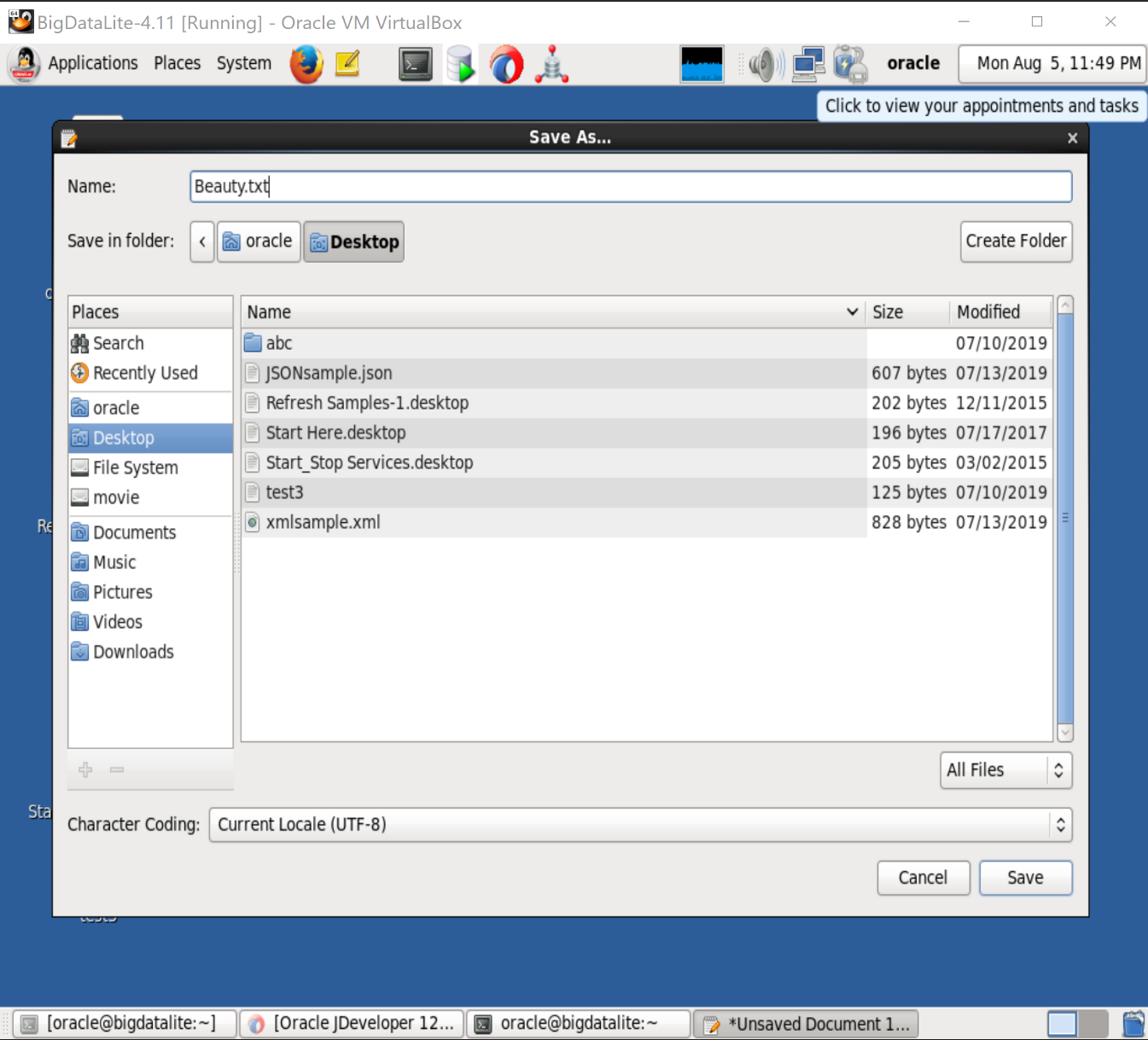
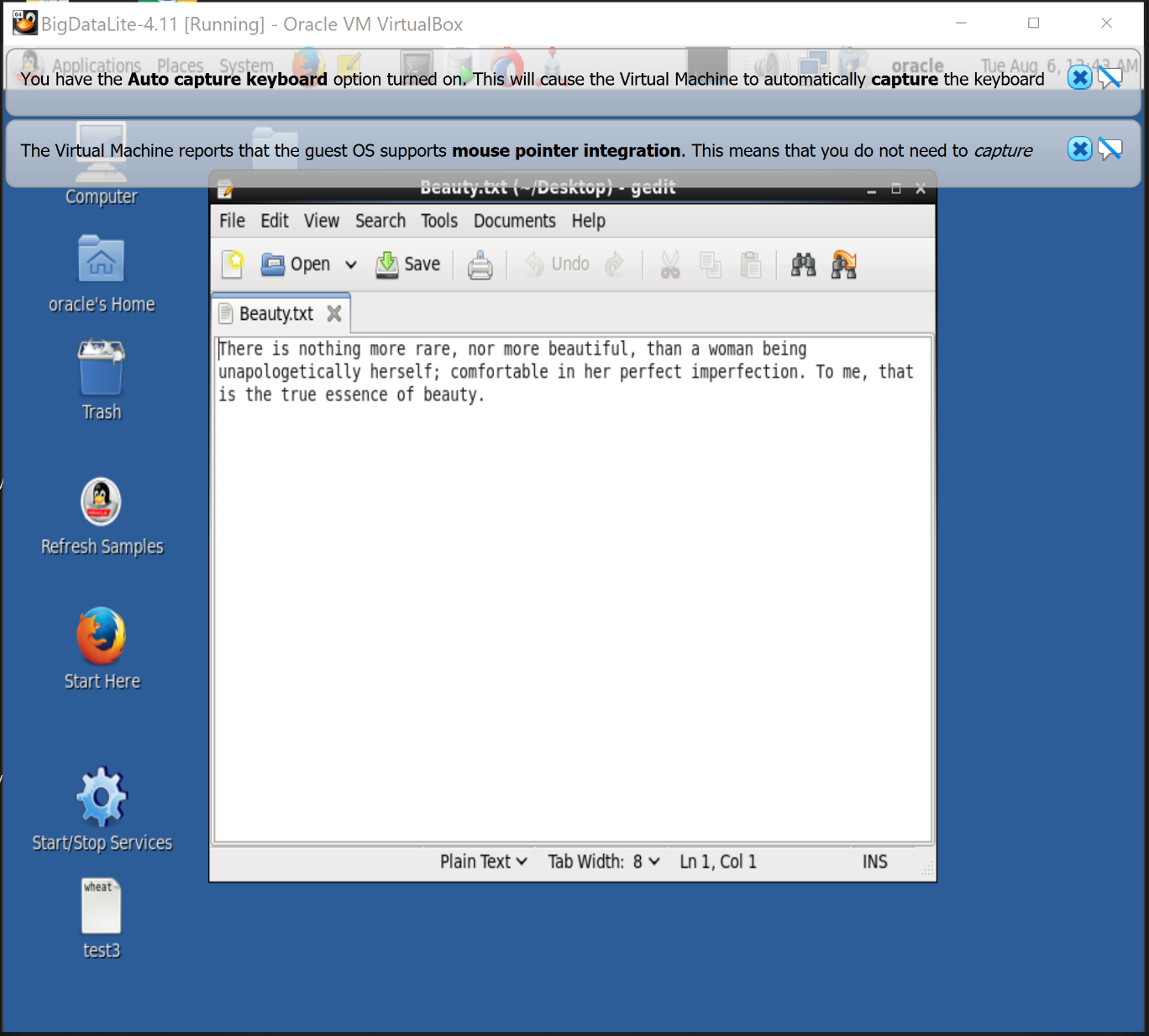
}

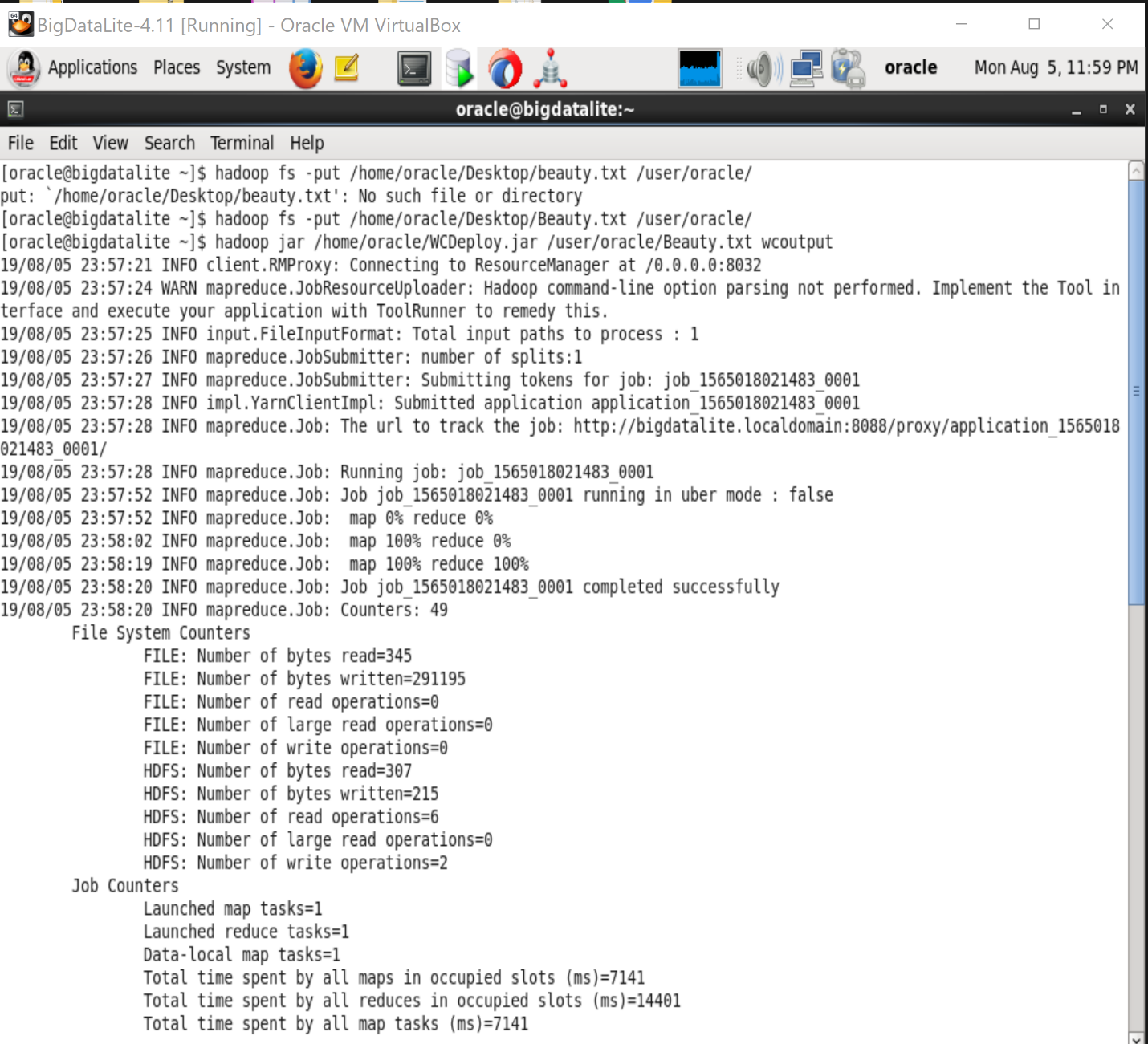
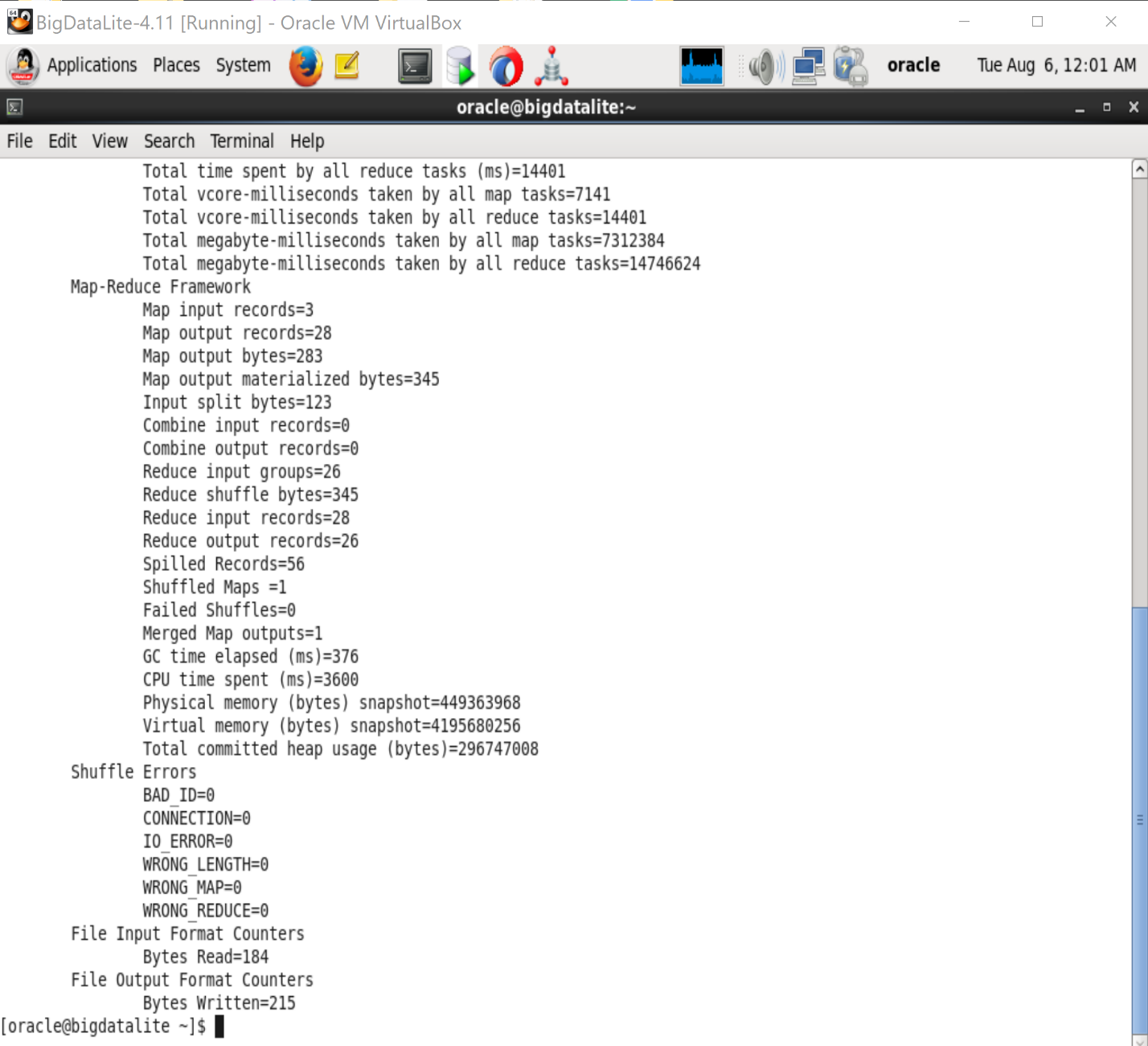
} 

**13**. **Deploy WC\_Configuration** 

**14.Generate source file with text editor(gedit)**



Save in desktop 

**15.Copy Beauty.txt to /user/oracle** 

**16.To see the result type the following command Hadoop fs –cat /user/oracle/wcoutput/part-r-00000** 