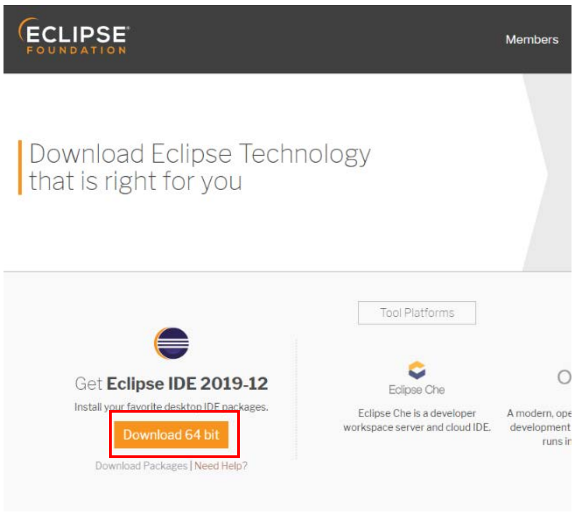
Hadoop Eclipse Map Reduce 練習

1.在 windows 上下載 eclipse

https://www.eclipse.org/downloads/



ECLIPSE FOUNDATION

Members

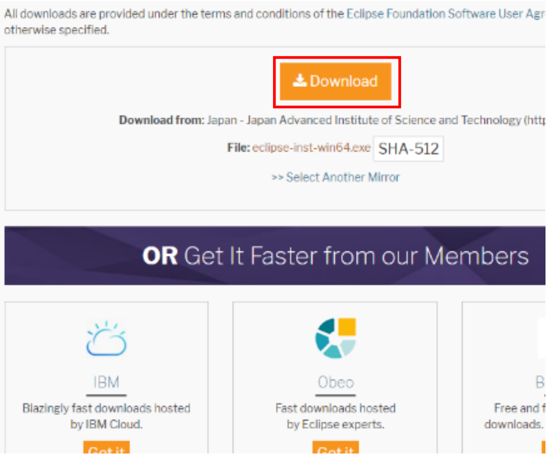
Download Eclipse Technology
that is right for you

Tool Platforms

Get **Eclipse IDE 2019-12**
Install your favorite desktop IDE packages.
Download 64 bit
Download Packages | Need Help?

Eclipse Che
Eclipse Che is a developer workspace server and cloud IDE.

A modern, ope
development
runs i



All downloads are provided under the terms and conditions of the Eclipse Foundation Software User Agr
otherwise specified.

Download

Download from: Japan - Japan Advanced Institute of Science and Technology (htt

File: eclipse-inst-win64.exe SHA-512

>> Select Another Mirror

OR Get It Faster from our Members

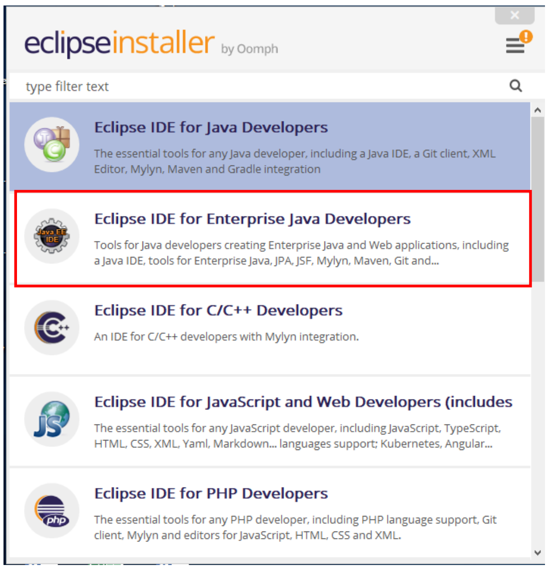
IBM
Blazingly fast downloads hosted by IBM Cloud.

Obeo
Fast downloads hosted by Eclipse experts.

Free and f
downloads.

2.安裝 eclipse

點擊下載下來的檔案(eclipse-inst-win64.exe)



eclipseinstaller by Oomph

type filter text

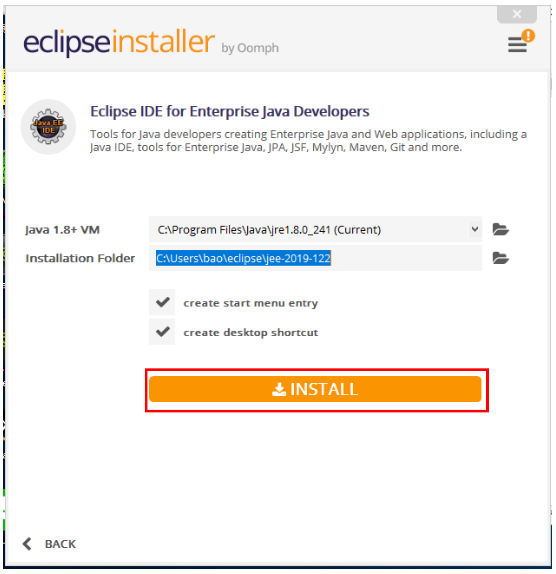
Eclipse IDE for Java Developers
The essential tools for any Java developer, including a Java IDE, a Git client, XML Editor, Mylyn, Maven and Gradle integration

Eclipse IDE for Enterprise Java Developers
Tools for Java developers creating Enterprise Java and Web applications, including a Java IDE, tools for Enterprise Java, JPA, JSF, Mylyn, Maven, Git and...

Eclipse IDE for C/C++ Developers
An IDE for C/C++ developers with Mylyn integration.

Eclipse IDE for JavaScript and Web Developers (includes
The essential tools for any JavaScript developer, including JavaScript, TypeScript, HTML, CSS, XML, Yaml, Markdown... languages support; Kubernetes, Angular...

Eclipse IDE for PHP Developers
The essential tools for any PHP developer, including PHP language support, Git client, Mylyn and editors for JavaScript, HTML, CSS and XML.



eclipseinstaller by Oomph

Eclipse IDE for Enterprise Java Developers
Tools for Java developers creating Enterprise Java and Web applications, including a Java IDE, tools for Enterprise Java, JPA, JSF, Mylyn, Maven, Git and more.

Java 1.8+ VM: C:\Program Files\Java\jre1.8.0_241 (Current)

Installation Folder: C:\Users\bao\eclipse\jee-2019-122

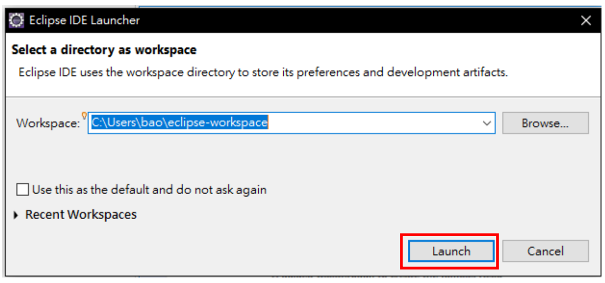
✓ create start menu entry

✓ create desktop shortcut

INSTALL

BACK

3. 開始使用 Eclipse IDE



Eclipse IDE Launcher

Select a directory as workspace

Eclipse IDE uses the workspace directory to store its preferences and development artifacts.

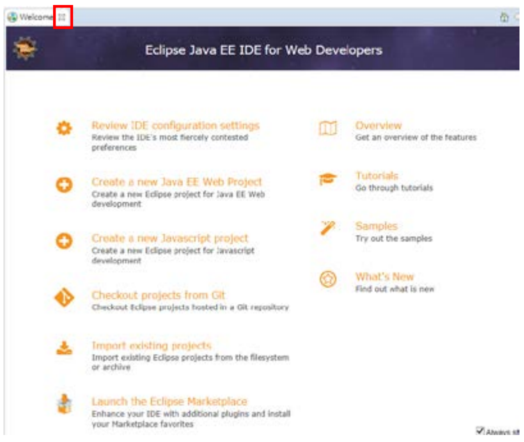
Workspace: C:\Users\bao\eclipse-workspace

Browse...

☐ Use this as the default and do not ask again

Recent Workspaces

Launch Cancel



Welcome

Eclipse Java EE IDE for Web Developers

Review IDE configuration settings
Review the IDE's most fiercely contested preferences

Create a new Java EE Web Project
Create a new Eclipse project for Java EE Web development

Create a new Javascript project
Create a new Eclipse project for Javascript development

Checkout projects from Git
Checkout Eclipse projects hosted in a Git repository

Import existing projects
Import existing Eclipse projects from the filesystem or archive

Launch the Eclipse Marketplace
Enhance your IDE with additional plugins and install your Marketplace favorites

Overview
Get an overview of the features

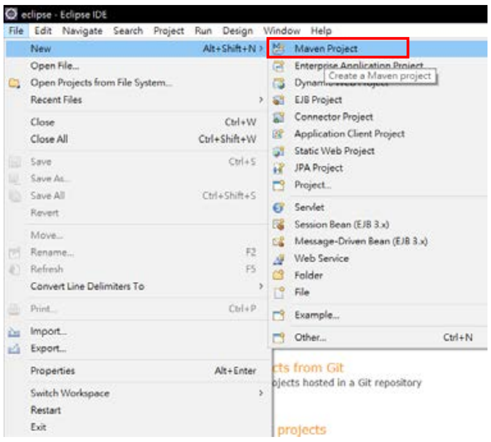
Tutorials
Go through tutorials

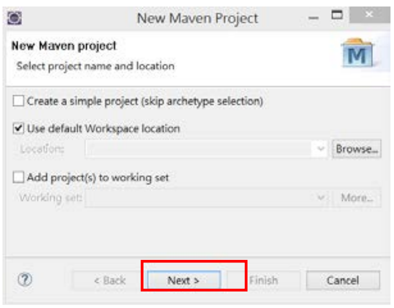
Samples
Try out the samples

What's New
Find out what is new

✓ Always s

4. File – New – Maven Project –Next - quickstart - Next

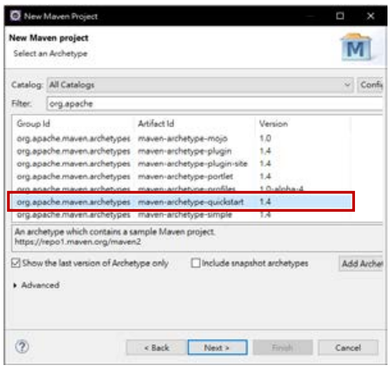




New Maven Project

New Maven project
Select project name and location

☐ Create a simple project (skip archetype selection)
☑ Use default Workspace location
Location:
☐ Add project(s) to working set
Working set:



New Maven Project

New Maven project
Select an Archetype

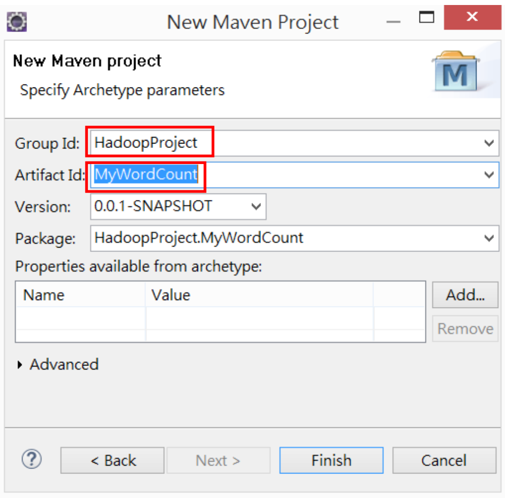
Catalog: All Catalogs
Filter: org.apache

Group Id	Artifact Id	Version
org.apache.maven.archetypes	maven-archetype-mojo	1.0
org.apache.maven.archetypes	maven-archetype-plugin	1.4
org.apache.maven.archetypes	maven-archetype-plugin-site	1.4
org.apache.maven.archetypes	maven-archetype-portlet	1.4
org.apache.maven.archetypes	maven-archetype-profiles	1.0-alpha-4
org.apache.maven.archetypes	maven-archetype-quickstart	1.4
org.apache.maven.archetypes	maven-archetype-simple	1.4

An archetype which contains a sample Maven project.
https://repo1.maven.org/maven2

☑ Show the last version of Archetype only ☐ Include snapshot archetypes

5. 設定 Group Id(package，專案群組識別)、Artifact Id(專案名稱) - Finish

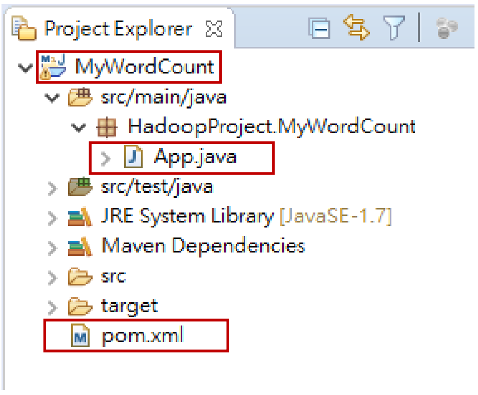


New Maven Project

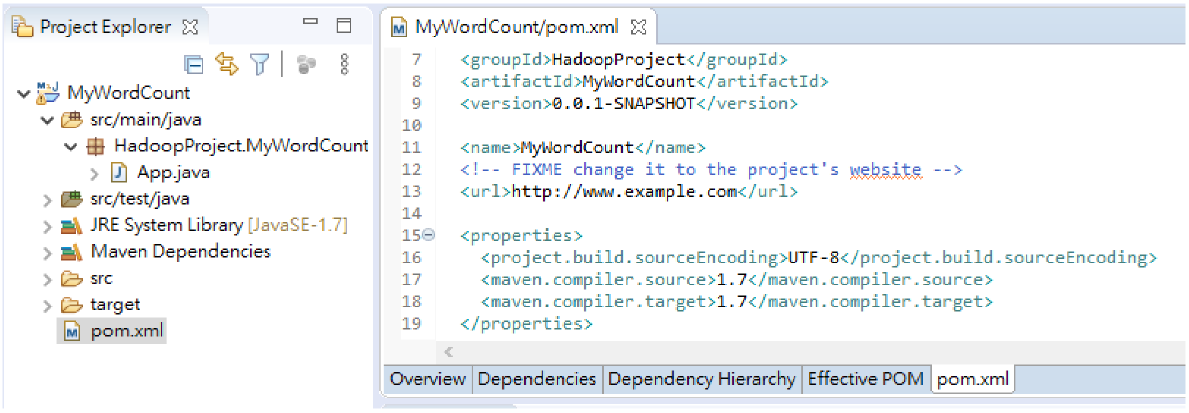
New Maven project
Specify Archetype parameters

Group Id: HadoopProject
Artifact Id: MyWordCount
Version: 0.0.1-SNAPSHOT
Package: HadoopProject.MyWordCount
Properties available from archetype:

Name	Value



6. 點選左邊 pom.xml，再點選下面 pom.xml



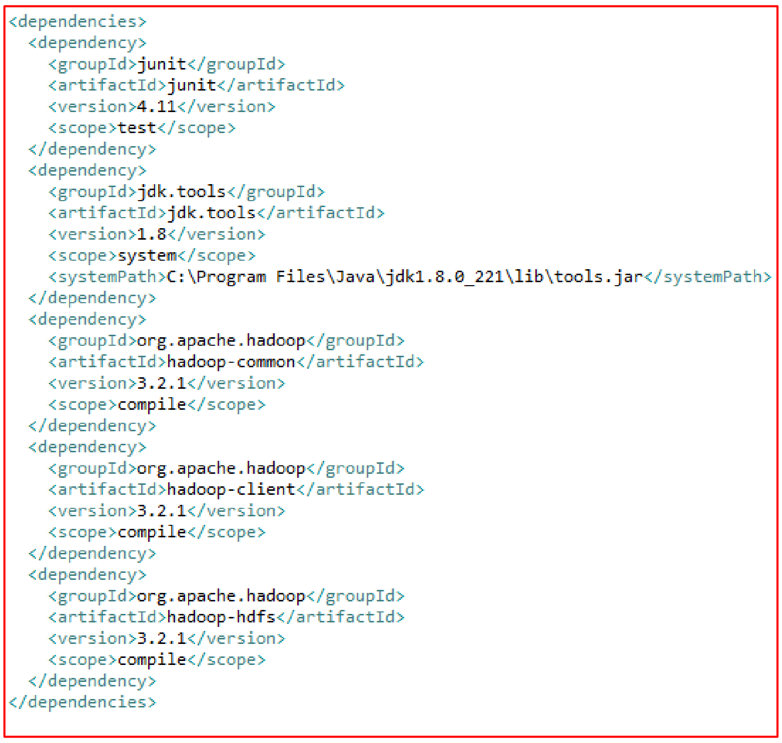
```xml
  <groupId>HadoopProject</groupId>
  <artifactId>MyWordCount</artifactId>
  <version>0.0.1-SNAPSHOT</version>

  <name>MyWordCount</name>
  <!-- FIXME change it to the project's website -->
  <url>http://www.example.com</url>

  <properties>
    <project.build.sourceEncoding>UTF-8</project.build.sourceEncoding>
    <maven.compiler.source>1.7</maven.compiler.source>
    <maven.compiler.target>1.7</maven.compiler.target>
  </properties>
```

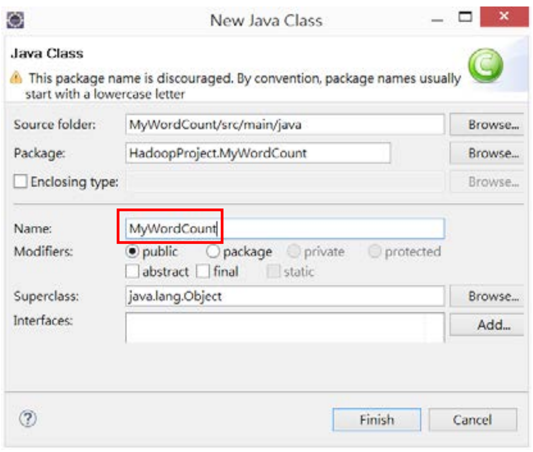
7.新增以下套件資料，存檔。
其中，<systemPath>須改為自己的路徑。<jdk 的安裝路徑>\lib\tool.jar

```
    <dependency>
      <groupId>jdk.tools</groupId>
      <artifactId>jdk.tools</artifactId>
      <version>1.8</version>
      <scope>system</scope>
      <systemPath>C:\Program Files\Java\jdk1.8.0_221\lib\tools.jar</systemPath>
      <!--systemPath要改為自己的jdk版本-->
    </dependency>
    <dependency>
      <groupId>org.apache.hadoop</groupId>
      <artifactId>hadoop-common</artifactId>
      <version>3.2.1</version>
      <scope>compile</scope>
    </dependency>
    <dependency>
      <groupId>org.apache.hadoop</groupId>
      <artifactId>hadoop-client</artifactId>
      <version>3.2.1</version>
      <scope>compile</scope>
    </dependency>
    <dependency>
      <groupId>org.apache.hadoop</groupId>
      <artifactId>hadoop-hdfs</artifactId>
      <version>3.2.1</version>
      <scope>compile</scope>
    </dependency>
```



```
<dependencies>
  <dependency>
    <groupId>junit</groupId>
    <artifactId>junit</artifactId>
    <version>4.11</version>
    <scope>test</scope>
  </dependency>
  <dependency>
    <groupId>jdk.tools</groupId>
    <artifactId>jdk.tools</artifactId>
    <version>1.8</version>
    <scope>system</scope>
    <systemPath>C:\Program Files\Java\jdk1.8.0_221\lib\tools.jar</systemPath>
  </dependency>
  <dependency>
    <groupId>org.apache.hadoop</groupId>
    <artifactId>hadoop-common</artifactId>
    <version>3.2.1</version>
    <scope>compile</scope>
  </dependency>
  <dependency>
    <groupId>org.apache.hadoop</groupId>
    <artifactId>hadoop-client</artifactId>
    <version>3.2.1</version>
    <scope>compile</scope>
  </dependency>
  <dependency>
    <groupId>org.apache.hadoop</groupId>
    <artifactId>hadoop-hdfs</artifactId>
    <version>3.2.1</version>
    <scope>compile</scope>
  </dependency>
</dependencies>
```

8. 刪除 App.java，New – class – MyWordCount.java

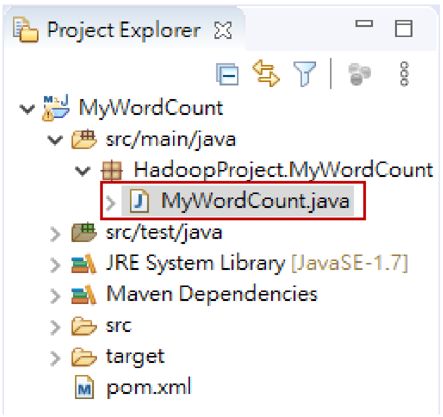


New Java Class

Java Class

This package name is discouraged. By convention, package names usually start with a lowercase letter

Source folder: MyWordCount/src/main/java
Package: HadoopProject.MyWordCount
Enclosing type:
Name: MyWordCount
Modifiers: [✓ public] package private protected
abstract final static
Superclass: java.lang.Object
Interfaces:



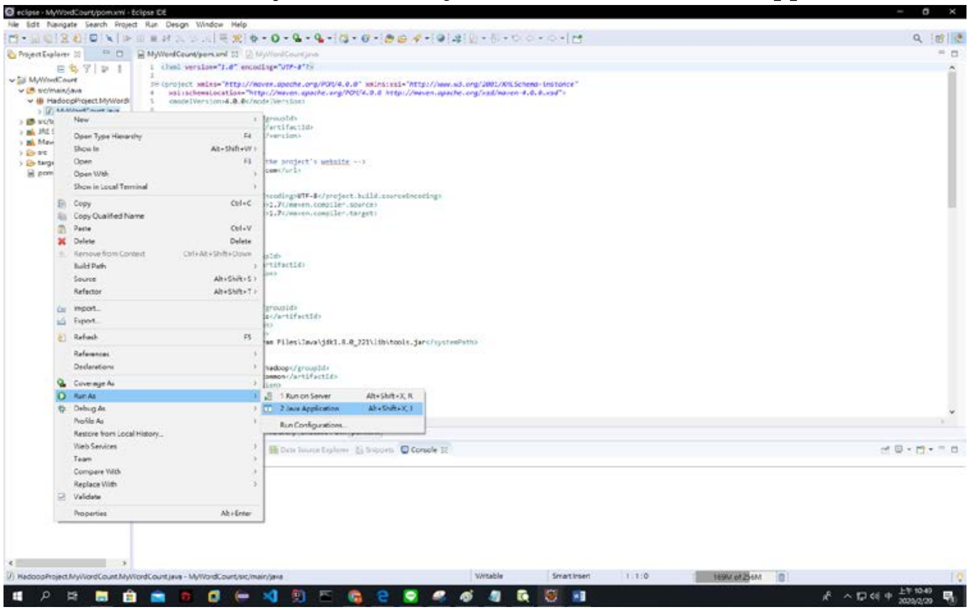
Project Explorer

```
MyWordCount
  src/main/java
    HadoopProject.MyWordCount
      MyWordCount.java
  src/test/java
  JRE System Library [JavaSE-1.7]
  Maven Dependencies
  src
  target
  pom.xml
```

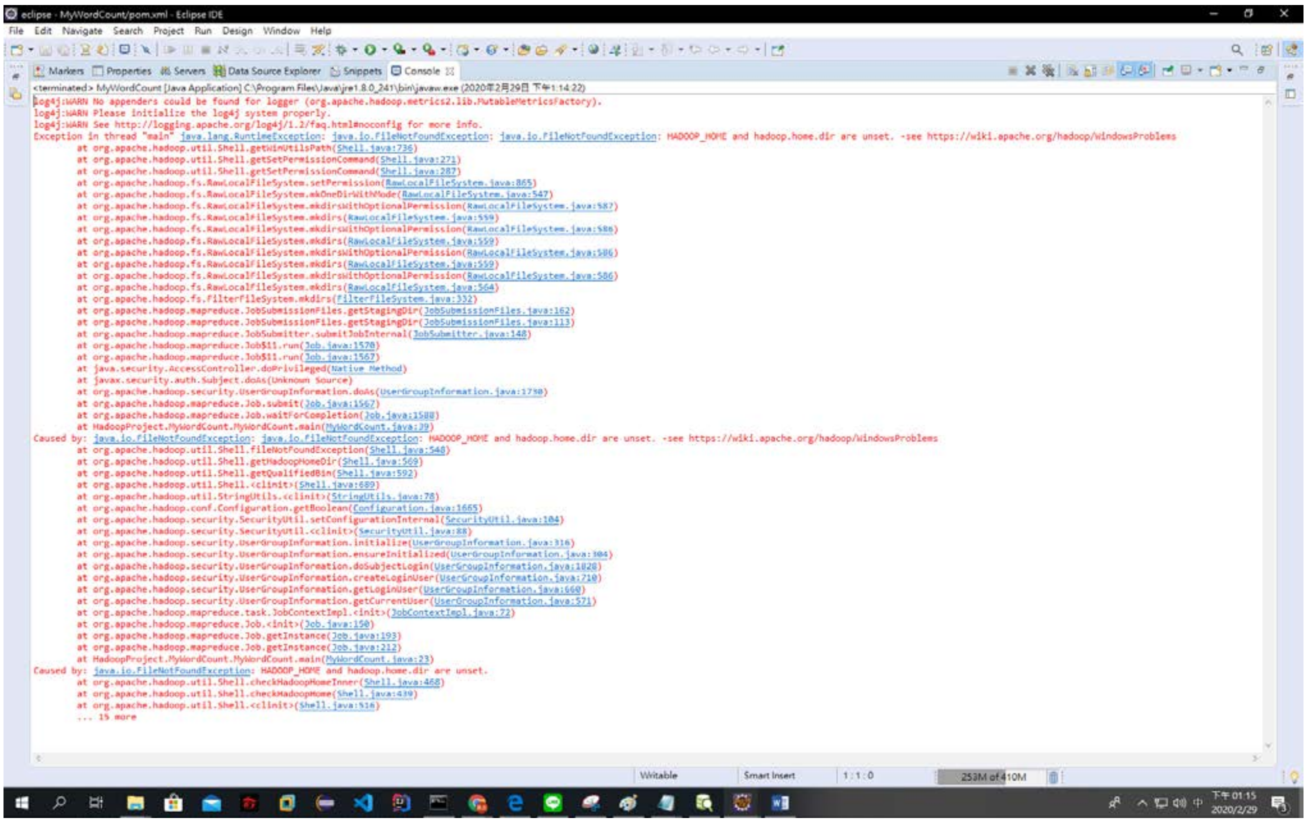
9. MyWordCount.java 內容修改為以下程式碼，存檔

```
package HadoopProject.MyWordCount;
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 MyWordCount {
  public MyWordCount() {
    // TODO Auto-generated constructor stub
  }
  public static void main(String[] args) throws IOException, ClassNotFoundException, InterruptedException {
    /*  初始化  */
    Configuration conf = new Configuration();
    /*  建立 MapReduce Job,  該 job 的名稱為 MyWordcount */
    Job job = Job.getInstance(conf,"MyWordCount");
    /*  啟動 job 的 jar class  為 MyWordcount */
    job.setJarByClass(MyWordCount.class);
    /*  啟動 job 的 map class  為 MyMapper */
    job.setMapperClass(MyMapper.class);
    /*  啟動 job 的 reduce class  為 MyReducer */
    job.setReducerClass(MyReducer.class);
    /*  輸入資料的 HDFS 路徑  */
    FileInputFormat.addInputPath(job, new Path("input01"));
    /*  輸出資料的 HDFS 路徑  */
    FileOutputFormat.setOutputPath(job, new Path("output01"));
    /*  輸出 Key 的型別  */
    job.setOutputKeyClass(Text.class);
    /*  輸出 Value 的型別  */
    job.setOutputValueClass(IntWritable.class);
    /*  啟動 Job 並回傳是否成功執行完畢  */
    boolean isSuccess = job.waitForCompletion(true);
    System.exit(isSuccess ? 0 : 1);
  }
  /* Mapper adapter: input key(LongWritable) , input value(Text) ,
    *    output key(Text) , output value(IntWritable) */
  static class MyMapper extends Mapper
<LongWritable, Text, Text , IntWritable>{
    private final static IntWritable one = new IntWritable(1);
    private Text word = new Text();
    protected void map(LongWritable key, Text value, Context context) throws IOException, InterruptedException
    {
      /*  將每一行的字串存到 lineValue */
      System.out.println("map content:"+key.get() +    "and" + value.toString());
      String lineValue = value.toString();
      /*  用 StringTokenizer 分割有空白、跳行等字元  */
      StringTokenizer stk = new StringTokenizer(lineValue);
      /*  將每個切割的字串存到 wordValue,  再將 wordValue 設為 Reduce 的 Key,
                    * value 設為整數 1 (one) */
      while(stk.hasMoreTokens()) {
        String wordValue = stk.nextToken();
        word.set(wordValue);
        context.write(word, one);
      }
    }
  }
/* Reducer adapter: input key(Text) , input values(IntWritable) ,
  *    output key(Text) , output value(IntWritable) */
static class MyReducer extends Reducer
      <Text, IntWritable, Text , IntWritable>{
    private IntWritable result = new IntWritable();
  protected void reduce(Text key, Iterable
            <IntWritable> values, Context context) throws IOException, InterruptedException
  {
    /*  累加該單字的數量  */
    int sum = 0;
    /*  在 Iterable 變數 values 用迴圈方式,將每個值(整數 1)取出並累加  */
    for(IntWritable value: values)      {
      sum += value.get();
    }
    /*  將累加的結果存到 result */
    result.set(sum);
    /*  輸出計算的結果  */
    context.write(key,result);
  }
 }
}
```

10.點選專案目錄 MyWordCount.java 右鍵-Run As-Java Application

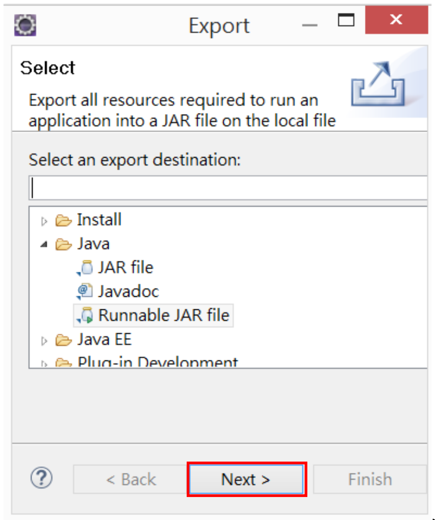


忽略下圖錯誤


eclipse - MyWordCount/pom.xml - Eclipse IDE

```
<terminated> MyWordCount [Java Application] C:\Program Files\Java\jre1.8.0_241\bin\javaw.exe (2020年2月29日 下午1:14:22)
log4j:WARN No appenders could be found for logger (org.apache.hadoop.metrics2.lib.MutableMetricsFactory).
log4j:WARN Please initialize the log4j system properly.
log4j:WARN See http://logging.apache.org/log4j/1.2/faq.html#noconfig for more info.
Exception in thread "main" java.lang.RuntimeException: java.io.FileNotFoundException: java.io.FileNotFoundException: HADOOP_HOME and hadoop.home.dir are unset. -see https://wiki.apache.org/hadoop/WindowsProblems
	at org.apache.hadoop.util.Shell.getWinUtilsPath(Shell.java:736)
	at org.apache.hadoop.util.Shell.getSetPermissionCommand(Shell.java:271)
	at org.apache.hadoop.util.Shell.getSetPermissionCommand(Shell.java:287)
	at org.apache.hadoop.fs.RawLocalFileSystem.setPermission(RawLocalFileSystem.java:865)
	at org.apache.hadoop.fs.RawLocalFileSystem.mkOneDirWithMode(RawLocalFileSystem.java:547)
	at org.apache.hadoop.fs.RawLocalFileSystem.mkdirsWithOptionalPermission(RawLocalFileSystem.java:587)
	at org.apache.hadoop.fs.RawLocalFileSystem.mkdirs(RawLocalFileSystem.java:559)
	at org.apache.hadoop.fs.RawLocalFileSystem.mkdirsWithOptionalPermission(RawLocalFileSystem.java:586)
	at org.apache.hadoop.fs.RawLocalFileSystem.mkdirs(RawLocalFileSystem.java:559)
	at org.apache.hadoop.fs.RawLocalFileSystem.mkdirsWithOptionalPermission(RawLocalFileSystem.java:586)
	at org.apache.hadoop.fs.RawLocalFileSystem.mkdirs(RawLocalFileSystem.java:559)
	at org.apache.hadoop.fs.RawLocalFileSystem.mkdirsWithOptionalPermission(RawLocalFileSystem.java:586)
	at org.apache.hadoop.fs.RawLocalFileSystem.mkdirs(RawLocalFileSystem.java:564)
	at org.apache.hadoop.fs.FilterFileSystem.mkdirs(FilterFileSystem.java:332)
	at org.apache.hadoop.mapreduce.JobSubmissionFiles.getStagingDir(JobSubmissionFiles.java:162)
	at org.apache.hadoop.mapreduce.JobSubmissionFiles.getStagingDir(JobSubmissionFiles.java:113)
	at org.apache.hadoop.mapreduce.JobSubmitter.submitJobInternal(JobSubmitter.java:148)
	at org.apache.hadoop.mapreduce.Job$11.run(Job.java:1570)
	at org.apache.hadoop.mapreduce.Job$11.run(Job.java:1567)
	at java.security.AccessController.doPrivileged(Native Method)
	at javax.security.auth.Subject.doAs(Unknown Source)
	at org.apache.hadoop.security.UserGroupInformation.doAs(UserGroupInformation.java:1730)
	at org.apache.hadoop.mapreduce.Job.submit(Job.java:1567)
	at org.apache.hadoop.mapreduce.Job.waitForCompletion(Job.java:1588)
	at HadoopProject.MyWordCount.MyWordCount.main(MyWordCount.java:39)
Caused by: java.io.FileNotFoundException: java.io.FileNotFoundException: HADOOP_HOME and hadoop.home.dir are unset. -see https://wiki.apache.org/hadoop/WindowsProblems
	at org.apache.hadoop.util.Shell.fileNotFoundException(Shell.java:548)
	at org.apache.hadoop.util.Shell.getHadoopHomeDir(Shell.java:569)
	at org.apache.hadoop.util.Shell.getQualifiedBin(Shell.java:592)
	at org.apache.hadoop.util.Shell.<clinit>(Shell.java:689)
	at org.apache.hadoop.util.StringUtils.<clinit>(StringUtils.java:78)
	at org.apache.hadoop.conf.Configuration.getBoolean(Configuration.java:1665)
	at org.apache.hadoop.security.SecurityUtil.setConfigurationInternal(SecurityUtil.java:104)
	at org.apache.hadoop.security.SecurityUtil.<clinit>(SecurityUtil.java:88)
	at org.apache.hadoop.security.UserGroupInformation.initialize(UserGroupInformation.java:316)
	at org.apache.hadoop.security.UserGroupInformation.ensureInitialized(UserGroupInformation.java:304)
	at org.apache.hadoop.security.UserGroupInformation.doSubjectLogin(UserGroupInformation.java:1828)
	at org.apache.hadoop.security.UserGroupInformation.createLoginUser(UserGroupInformation.java:710)
	at org.apache.hadoop.security.UserGroupInformation.getLoginUser(UserGroupInformation.java:660)
	at org.apache.hadoop.security.UserGroupInformation.getCurrentUser(UserGroupInformation.java:571)
	at org.apache.hadoop.mapreduce.task.JobContextImpl.<init>(JobContextImpl.java:72)
	at org.apache.hadoop.mapreduce.Job.<init>(Job.java:150)
	at org.apache.hadoop.mapreduce.Job.getInstance(Job.java:193)
	at org.apache.hadoop.mapreduce.Job.getInstance(Job.java:212)
	at HadoopProject.MyWordCount.MyWordCount.main(MyWordCount.java:23)
Caused by: java.io.FileNotFoundException: HADOOP_HOME and hadoop.home.dir are unset.
	at org.apache.hadoop.util.Shell.checkHadoopHomeInner(Shell.java:468)
	at org.apache.hadoop.util.Shell.checkHadoopHome(Shell.java:439)
	at org.apache.hadoop.util.Shell.<clinit>(Shell.java:516)
	... 15 more
```

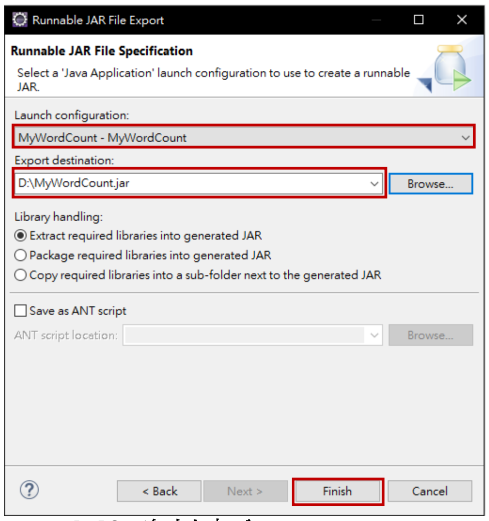
11. 點選專案目錄 MyWordCount.java 右鍵-Export-JAVA:Runnable JAR file，輸出 MyWordCount.jar


Export

Select

Export all resources required to run an application into a JAR file on the local file

Select an export destination:


Runnable JAR File Export

Runnable JAR File Specification

Select a 'Java Application' launch configuration to use to create a runnable JAR.

Launch configuration: MyWordCount - MyWordCount

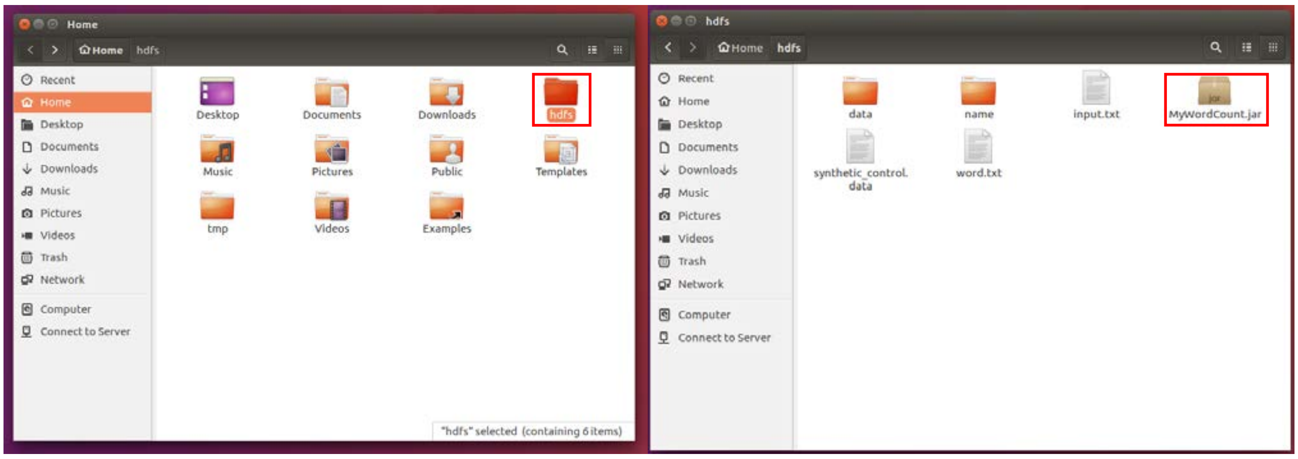
Export destination: D:\MyWordCount.jar

Library handling:
- [✓ Extract required libraries into generated JAR]
- Package required libraries into generated JAR
- Copy required libraries into a sub-folder next to the generated JAR

Save as ANT script

ANT script location:

12. 將 MyWordCount.jar 拖曳至虛擬機的 home/hdfs 資料夾內



13. 開啟 hadoop，編輯 input.txt 放到 HDFS，執行 hadoop MyWordCount

source /opt/hadoop/etc/hadoop/hadoop-env.sh

start-all.sh

hadoop jar /opt/hadoop/share/hadoop/mapreduce/hadoop-mapreduce-examples-3.2.1.jar pi 2 5

cd ~

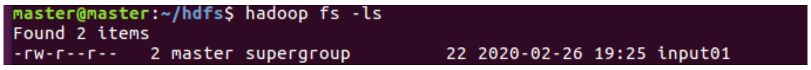
cd hdfs

gedit input.txt

```
hello
world
hello
bug
```

hadoop fs -put input.txt input01

Hadoop fs -ls

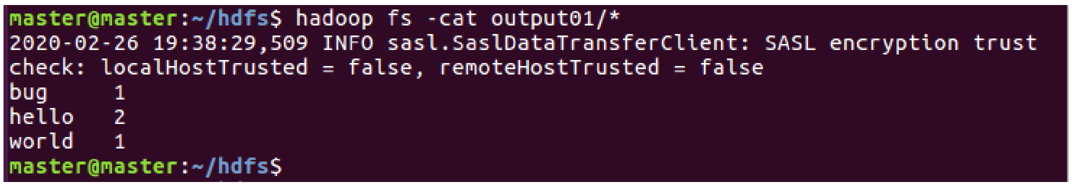


```
master@master:~/hdfs$ hadoop fs -ls
Found 2 items
-rw-r--r--   2 master supergroup         22 2020-02-26 19:25 input01
```

hadoop jar MyWordCount.jar

14. 觀察輸出結果

hadoop fs -cat output01/*



```
master@master:~/hdfs$ hadoop fs -cat output01/*
2020-02-26 19:38:29,509 INFO sasl.SaslDataTransferClient: SASL encryption trust
check: localHostTrusted = false, remoteHostTrusted = false
bug     1
hello   2
world   1
master@master:~/hdfs$
```