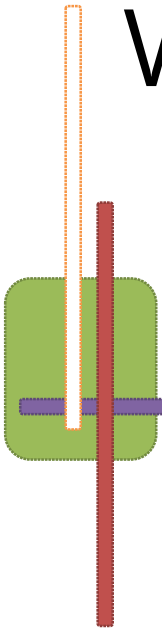


WordCount using Spark in Java

Step-by-Step

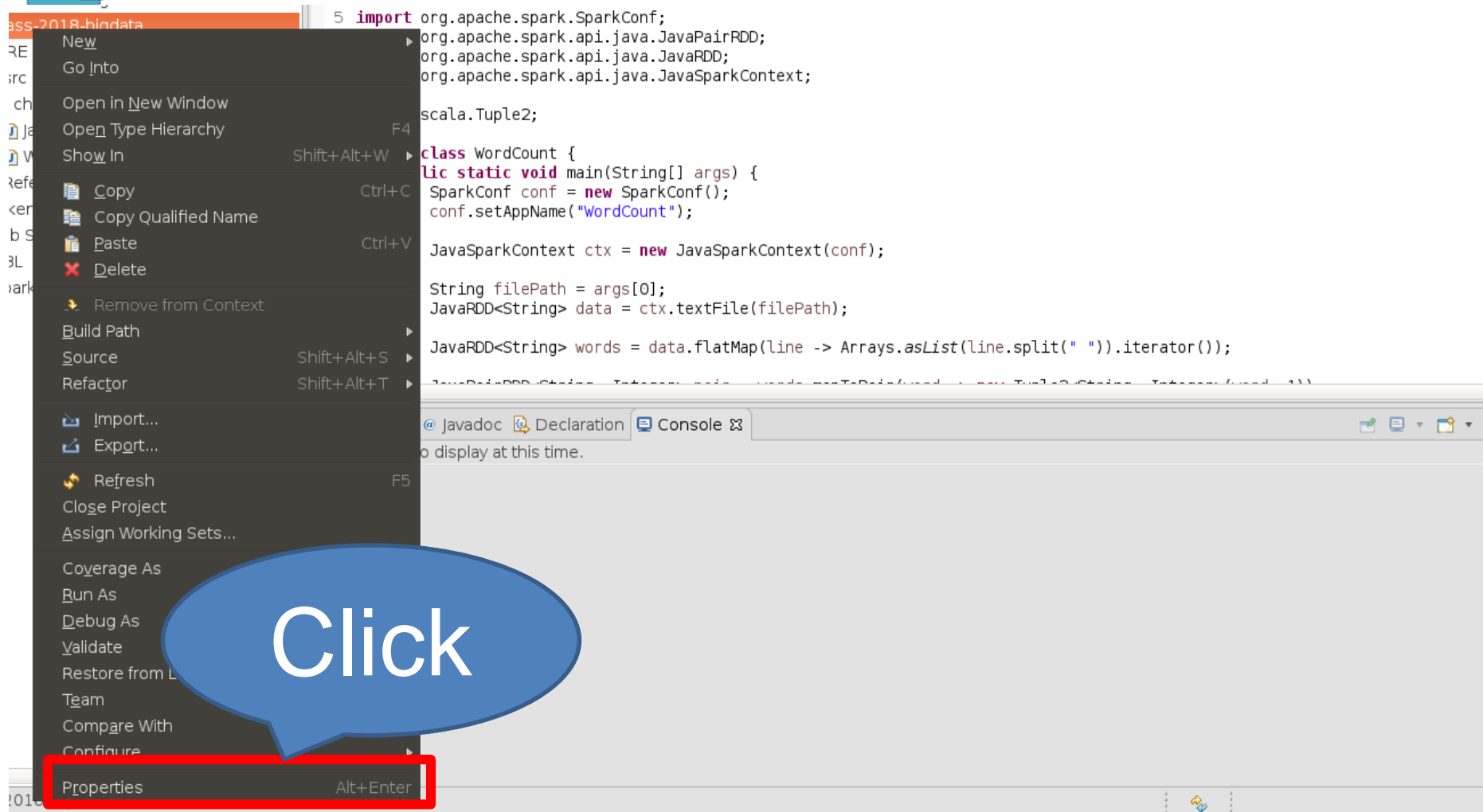
BigDataMiningLab.





Import Spark Libraries into Project

Import Spark Libraries



The screenshot shows an IDE window with a Java file named `ass-2018-bingdata`. The code contains the following imports and class definition:

```
5 import org.apache.spark.SparkConf;
   org.apache.spark.api.java.JavaPairRDD;
   org.apache.spark.api.java.JavaRDD;
   org.apache.spark.api.java.JavaSparkContext;

scala.Tuple2;

class WordCount {
    public static void main(String[] args) {
        SparkConf conf = new SparkConf();
        conf.setAppName("WordCount");

        JavaSparkContext ctx = new JavaSparkContext(conf);

        String filePath = args[0];
        JavaRDD<String> data = ctx.textFile(filePath);

        JavaRDD<String> words = data.flatMap(line -> Arrays.asList(line.split(" ")).iterator());

        JavaPairRDD<String, Integer> wordsToPairs = new JavaPairRDD(words, (word, _) -> 1);
    }
}
```

A context menu is open over the code, listing various actions. The 'Properties' option at the bottom is highlighted with a red rectangle. A blue speech bubble with the word 'Click' points to this option.

Context Menu Options:

- 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
- Remove from Context
- Build Path
- Source (Shift+Alt+S)
- Refactor (Shift+Alt+T)
- Import...
- Export...
- Refresh (F5)
- Close Project
- Assign Working Sets...
- Coverage As
- Run As
- Debug As
- Validate
- Restore from Library
- Team
- Compare With
- Configure
- Properties (Alt+Enter)**

1

2

3

Java Build Path

Libraries

JARs and class folders on the build path:

- activation-1.1.1.jar - /home/hadoop/spark/jars
- antlr-2.7.7.jar - /home/hadoop/spark/jars
- antlr-runtime-3.4.jar - /home/hadoop/spark/jars
- antlr4-runtime-4.5.3.jar - /home/hadoop/spark/jars
- aopalliance-1.0.jar - /home/hadoop/spark/jars
- aopalliance-repackaged-2.4.0-b34.jar - /home/hadoop/spark/jars
- apache-log4j-extras-1.2.17.jar - /home/hadoop/spark/jars
- apacheds-i18n-2.0.0-M15.jar - /home/hadoop/spark/jars
- apacheds-kerberos-codec-2.0.0-M15.jar - /home/hadoop/spark/jars
- api-asn1-api-1.0.0-M20.jar - /home/hadoop/spark/jars
- api-util-1.0.0-M20.jar - /home/hadoop/spark/jars
- arpack_combined_all-0.1.jar - /home/hadoop/spark/jars
- avro-1.7.7.jar - /home/hadoop/spark/jars
- avro-inc-1.7.7.jar - /home/hadoop/spark/jars

Add External JARs...

Add Variable...

Add Library...

Add Class Folder...

Add External Class Folder...

Edit...

Remove

Migrate JAR File...

Apply

Cancel

Apply and Close

JAR Selection

spark jars

\$SPARK_HOME/jars

EX) ~/spark/jars/

Places

- Home
- Desktop
- Documents
- Downloads
- Music
- Pictures
- Videos
- eclipse

Devices

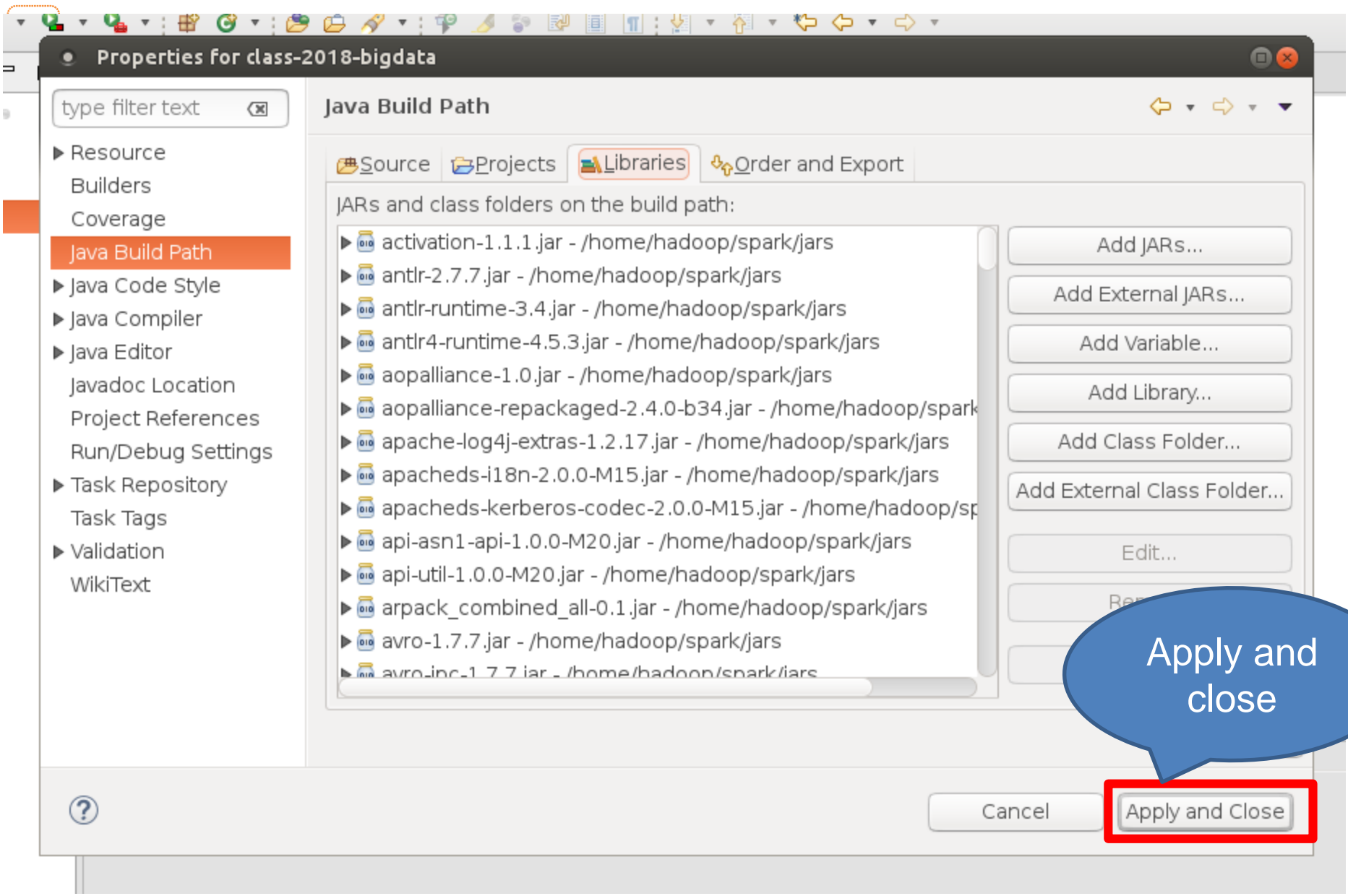
- Computer

Name	Modified
activation-1.1.1.jar	2017년 07월 01일
antlr4-runtime-4.5.3.jar	2017년 07월 01일
antlr-2.7.7.jar	2017년 07월 01일
antlr-runtime-3.4.jar	164.4 kB 2017년 07월 01일
aopalliance-1.0.jar	4.5 kB 2017년 07월 01일
aopalliance-repackaged-2.4.0-b34.jar	14.8 kB 2017년 07월 01일
apacheds-i18n-2.0.0-M15.jar	44.9 kB 2017년 07월 01일
apacheds-kerberos-codec-2.0.0-M15.jar	691.5 kB 2017년 07월 01일
apache-log4j-extras-1.2.17.jar	448.8 kB 2017년 07월 01일
api-asn1-api-1.0.0-M20.jar	16.6 kB 2017년 07월 01일
api-util-1.0.0-M20.jar	79.9 kB 2017년 07월 01일
arpack_combined_all-0.1.jar	1.2 MB 2017년 07월 01일
avro-1.7.7.jar	436.3 kB 2017년 07월 01일
avro-ipc-1.7.7.jar	193.0 kB 2017년 07월 01일
avro-mapred-1.7.7-hadoop2.jar	180.7 kB 2017년 07월 01일
base64-2.3.8.jar	17.0 kB 2017년 07월 01일
bcprov-jdk15on-1.51.jar	2.8 MB 2017년 07월 01일
bonecp-0.8.0.RELEASE.jar	110.6 kB 2017년 07월 01일
breeze_2.11-0.13.1.jar	15.1 MB 2017년 07월 01일
breeze-macros_2.11-0.13.1.jar	
calcite-avatica-1.2.0-incubating.jar	
calcite-core-1.2.0-incubating.jar	
calcite-linq4j-1.2.0-incubating.jar	
chill_2.11-0.8.0.jar	
chill-java-0.8.0.jar	
commons-beanutils-1.7.0.jar	
commons-beanutils-core-1.8.0.jar	
commons-cli-1.2.jar	

Select all file and click "OK"

Cancel

OK



Apply and
close

Cancel

Apply and Close



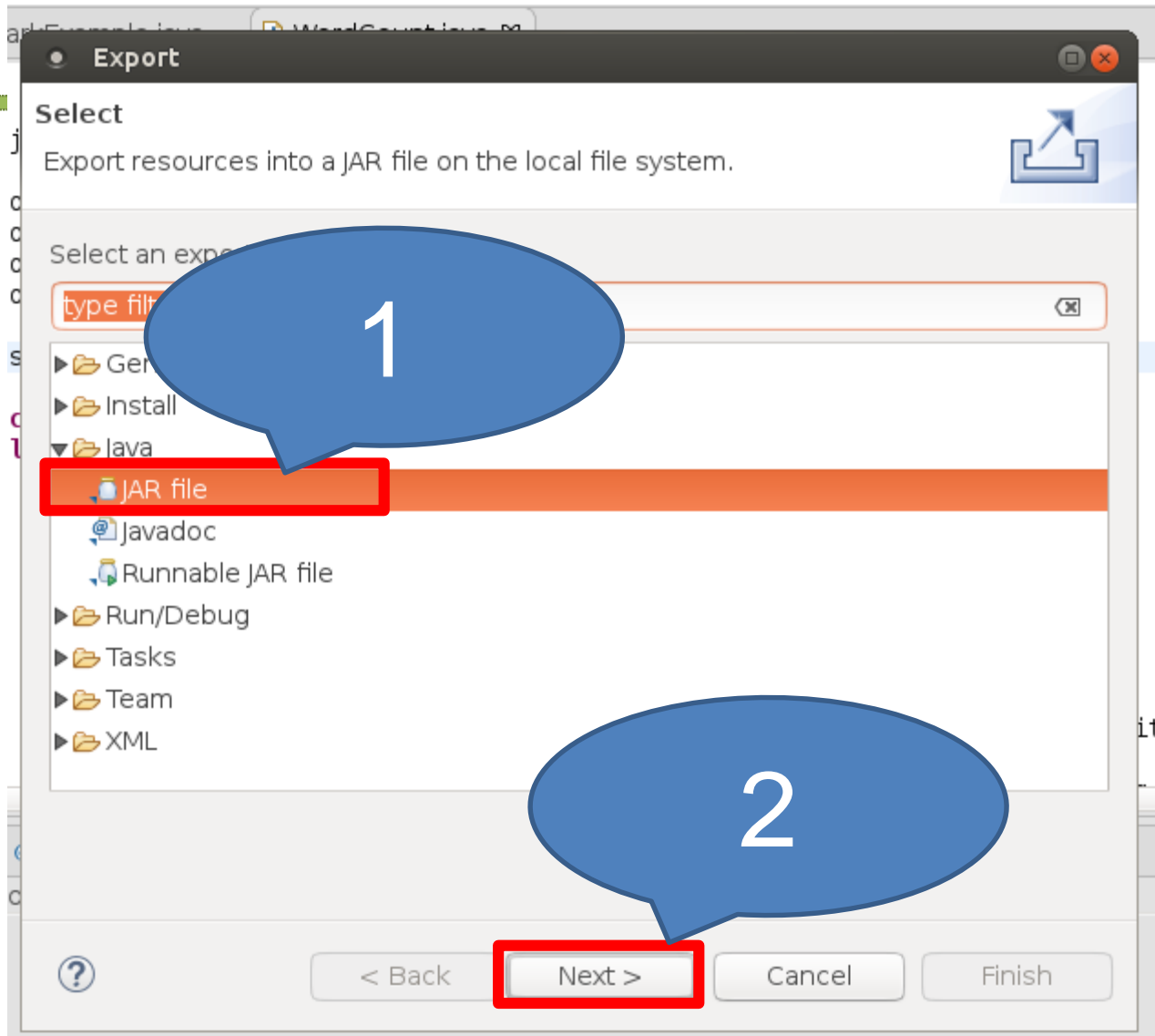
Export "jar"



The image shows a screenshot of an IDE's menu system. A dark-themed menu is open, listing various actions. The 'Export...' option, which includes a small folder icon, is highlighted with a red rectangular border. A blue speech bubble with the word 'Click' in white text points to this option. The background shows a code editor with Java code for a WordCount application, including imports for Arrays, SparkConf, JavaPairRDD, JavaRDD, and JavaSparkContext, and a class definition for WordCount with a main method.

Click

Export...







Run WordCount



\$SPARK_HOME/sbin/start-all.sh

```
~/Downloads$ $SPARK_HOME/sbin/start-all.sh
starting org.apache.spark.deploy.master.Master, logging to [REDACTED] spark/logs/spark-hado
op-org.apache.spark.deploy.master.Master-1-lemon.out
[REDACTED] starting org.apache.spark.deploy.worker.Worker, logging to [REDACTED] /spark/l
ogs/spark-hadoop-org.apache.spark.deploy.worker.Worker-1-lemon.out
[REDACTED] starting org.apache.spark.deploy.worker.Worker, logging to [REDACTED] /spark/l
ogs/spark-hadoop-org.apache.spark.deploy.worker.Worker-1-lime.out
[REDACTED] starting org.apache.spark.deploy.worker.Worker, logging to [REDACTED] /spark/l
ogs/spark-hadoop-org.apache.spark.deploy.worker.Worker-1-apple.out
```

Spark UI



Spark Master at spark:// localhost:7077

URL: spark://1

REST URL: spark: (cluster mode)

Alive Workers: 4

Cores in use: 32 Total, 0 Used

Memory in use: 58.4 GB Total, 0.0 B Used

Applications: 0 Running, 0 Completed

Drivers: 0 Running, 0 Completed

Status: ALIVE

Check where is spark running

Workers

Worker Id	Address	State	Cores	Memory
worker-20180906015228-		ALIVE	8 (0 Used)	22.5 GB (0.0 B Used)
worker-20180927161023-		ALIVE	8 (0 Used)	6.7 GB (0.0 B Used)
worker-20180927161156-		ALIVE	8 (0 Used)	6.7 GB (0.0 B Used)
worker-20180927161156-		ALIVE	8 (0 Used)	22.5 GB (0.0 B Used)

Running Applications

Application ID	Name	Cores	Memory per Executor	Submitted Time	User	State	Duration
----------------	------	-------	---------------------	----------------	------	-------	----------

Completed Applications

Application ID	Name	Cores	Memory per Executor	Submitted Time	User	State	Duration
----------------	------	-------	---------------------	----------------	------	-------	----------

Package name

For default package, it should be empty.
Ex) chapter.one.WordCount -> WordCount

Class name

`$SPARK_HOME/bin/spark-submit --
class chapter.one.WordCount --
master spark://localhost:7077
wordcount.jar
$SPARK_HOME/README.md`

File path

```
~/Downloads$ ~/spark/bin/spark-submit --class chapter.one.WordCount --master spa  
rk://:7077 wordcount.jar ~/spark/README.md
```

Result

```
find      1
sc.parallelize(range(1000)).count()      1
contains      1
you      4
project 1
Pi      1
protocols      1
that      2
a      8
or      3
high-level      1
name      1
Hadoop, 2
to      17
available      1
core      1
(You      1
instance:      1
more      1
see      3
of      5
tools      1
"local[N]"      1
programs      2
option 1
package.)      1
["Building      1
must      1
and      9
command,      2
system 1
Hadoop 3
```



If you have any question,
please feel free to email TA
(woongheelee@hanyang.ac.kr).

Make an appointment and visit
the lab.