

Java-Success.com

Prepare to fast-track, choose & go places with 800+ Java & Big Data Q&As with lots of code & diagrams.

[Home](#) [Why? ▾](#) [300+ Java FAQs ▾](#) [300+ Big Data FAQs ▾](#) [Courses ▾](#)

[Membership ▾](#) [Your Career ▾](#)

[Home](#) › [bigdata-success.com](#) › [Tutorials - Big Data](#) › [TUT - Starting Spark & Scala](#) ›

05: Setting up & getting started with Spark local mode with Sbt & Scala

05: Setting up & getting started with Spark local mode with Sbt & Scala

 Posted on [November 7, 2018](#)

This extends [Setting up & getting started with sbt](#).

Spark 2.3.2 requires **Scala 2.11.x**. You need to use the right Scala version.

Step 1: On Eclipse set the Scala compiler to 2.11. Right mouse click on the project "sbt-tutorial" -> Properties -> Scala Compiler -> Tick "Use Project Settings" and select "latest **2.11** bundle (dynamic)".

300+ Java Interview FAQs

300+ Java FAQs



16+ Java Key Areas Q&As



150+ Java Architect FAQs



80+ Java Code Quality Q&As



150+ Java Coding Q&As



300+ Big Data Interview FAQs

300+ Big Data FAQs



Tutorials - Big Data



TUT -  Starting Big Data

TUT - Starting Spark & Scala

TOP

Step 2: Add the Spark dependency to “build.sbt” file.

```

1 organization := "com.sbt-tut"
2
3 version:= "0.1"
4
5 scalaVersion := "2.11.12"
6
7 libraryDependencies += "org.apache.spark" %% "spark-core" % "2.4.0"
8

```

Step 3: From the command-line run the following command, and it may take some time to download the dependencies.

```

1 ~/projects/sbt-tutorial]$ sbt eclipse
2

```

Step 4: Write a very simple Spark code in Scala as shown below within a package named “com.scalaproject” and the Scala file is named “SimpleSpark.scala”.

```

1 package com.scalaproject
2
3 import org.apache.spark.sql.SparkSession;
4 import org.apache.spark.sql.Dataset;
5
6 object SimpleSpark {
7
8     def main(args: Array[String]): Unit = {
9         val spark = SparkSession.builder.appName("SimpleSpark")
10            .config("spark.master", "local")
11            .getOrCreate()
12
13         val list = List(1,2,3,4,5)
14         val rdd = spark.sparkContext.parallelize(list)
15
16         val multipleBy2Rdd = rdd.map(_ * 2)
17         println(multipleBy2Rdd.collect().toList)
18     }
19 }
20
21

```

TUT - Starting with Python

TUT - Kafka

TUT - Pig

TUT - Apache Storm

TUT - Spark Scala on Zeppelin

TUT - Cloudera

TUT - Cloudera on Docker

TUT - File Formats

TUT - Spark on Docker

TUT - Flume

TUT - Hadoop (HDFS)

TUT - HBase (NoSQL)

TUT - Hive (SQL)

TUT - Hadoop & Spark

TUT - MapReduce

TUT - Spark and Scala

TUT - Spark & Java

TUT - PySpark on Databricks

TUT - Zookeeper

800+ Java Interview Q&As

300+ Core Java Q&As



300+ Enterprise Java Q&As



150+ Java Frameworks Q&As



120+ Companion Tech Q&As



Tutorials - Enterprise Java



Step 5: Right mouse click on the file

SimpleSpark.scala and “Run as” → “Scala Application” within eclipse.

Output:

```

1 Using Spark's default log4j profile: org/apache/spark/log4j-defaults.properties
2 18/11/07 23:01:44 INFO SparkContext: Running Spark version 2.4.0
3 18/11/07 23:01:44 WARN NativeCodeLoader: Unable to load native code loader: scala-native
4 18/11/07 23:01:44 INFO SparkContext: Submitted application: SimpleSpark
5 18/11/07 23:01:45 INFO SecurityManager: Changing view acls to: {}
6 18/11/07 23:01:45 INFO SecurityManager: Changing view map acls to: {}
7 18/11/07 23:01:45 INFO SecurityManager: Changing view acl to: {}
8 18/11/07 23:01:45 INFO SecurityManager: Changing view acl to: {}
9 18/11/07 23:01:45 INFO SecurityManager: SecurityManager: SecurityManager: SecurityManager
10 18/11/07 23:01:45 INFO Utils: Successfully started service 'org.apache.spark.network.netty.NettyBlockTransferService' on port 4444
11 18/11/07 23:01:45 INFO SparkEnv: Registering MapOutputTracker
12 18/11/07 23:01:45 INFO SparkEnv: Registering BlockManagerMasterEndpoint
13 18/11/07 23:01:45 INFO BlockManagerMasterEndpoint: Using org.apache.spark.storage.DefaultTopologyMapper for getting topology
14 18/11/07 23:01:45 INFO BlockManagerMasterEndpoint: BlockManagerMasterEndpoint: BlockManagerMasterEndpoint
15 18/11/07 23:01:45 INFO DiskBlockManager: Created disk block manager
16 18/11/07 23:01:45 INFO MemoryStore: MemoryStore started with capacity 1.0 GiB
17 18/11/07 23:01:45 INFO SparkEnv: Registering OutputCommitCoordinator
18 18/11/07 23:01:45 INFO Utils: Successfully started service 'org.apache.spark.network.netty.NettyBlockTransferService' on port 4444
19 18/11/07 23:01:45 INFO SparkUI: Bound SparkUI to 0.0.0.0, port 4040
20 18/11/07 23:01:45 INFO Executor: Starting executor with driver 0.0.0.0, port 4040
21 18/11/07 23:01:45 INFO Utils: Successfully started service 'org.apache.spark.network.netty.NettyBlockTransferService' on port 4444
22 18/11/07 23:01:45 INFO NettyBlockTransferService: NettyBlockTransferService: NettyBlockTransferService
23 18/11/07 23:01:45 INFO BlockManager: Using org.apache.spark.storage.DefaultTopologyMapper for getting topology
24 18/11/07 23:01:45 INFO BlockManagerMaster: Registering block manager
25 18/11/07 23:01:45 INFO BlockManagerMasterEndpoint: Registering block manager
26 18/11/07 23:01:45 INFO BlockManagerMaster: Registering block manager
27 18/11/07 23:01:45 INFO BlockManager: Initialized block manager
28 18/11/07 23:01:46 INFO SparkContext: Starting job 0 (coarse grained)
29 18/11/07 23:01:46 INFO DAGScheduler: Got job 0 (coarse grained)
30 18/11/07 23:01:46 INFO DAGScheduler: Final stage:
31 18/11/07 23:01:46 INFO DAGScheduler: Parents of final stage:
32 18/11/07 23:01:46 INFO DAGScheduler: Missing parents:
33 18/11/07 23:01:46 INFO DAGScheduler: Submitting 1 tasks
34 18/11/07 23:01:46 INFO MemoryStore: Block broadcast_0 stored as value in memory (estimated size 1.0 GiB)
35 18/11/07 23:01:46 INFO MemoryStore: Block broadcast_0 stored as value in memory (estimated size 1.0 GiB)
36 18/11/07 23:01:46 INFO BlockManagerInfo: Added broadcast_0 to storage
37 18/11/07 23:01:46 INFO SparkContext: Created broadcast_0
38 18/11/07 23:01:46 INFO DAGScheduler: Submitting 2 tasks
39 18/11/07 23:01:46 INFO TaskSchedulerImpl: Adding task
40 18/11/07 23:01:46 INFO TaskSetManager: Starting task
41 18/11/07 23:01:46 INFO Executor: Running task 0.0
42 18/11/07 23:01:46 INFO Executor: Finished task 0.0
43 18/11/07 23:01:46 INFO TaskSetManager: Starting task
44 18/11/07 23:01:46 INFO Executor: Running task 1.0
45 18/11/07 23:01:46 INFO TaskSetManager: Finished task

```



```
46 18/11/07 23:01:46 INFO Executor: Finished task 1.0
47 18/11/07 23:01:46 INFO TaskSetManager: Finished task
48 18/11/07 23:01:46 INFO TaskSchedulerImpl: Removed
49 18/11/07 23:01:46 INFO DAGScheduler: ResultStage 0
50 18/11/07 23:01:46 INFO DAGScheduler: Job 0 finished
51 List(2, 4, 6, 8, 10)
52 18/11/07 23:01:46 INFO SparkContext: Invoking stop
53 18/11/07 23:01:46 INFO SparkUI: Stopped Spark web
54 18/11/07 23:01:46 INFO ContextCleaner: Cleaned acc
55 18/11/07 23:01:46 INFO MapOutputTrackerMasterEndpo
56 18/11/07 23:01:46 INFO MemoryStore: MemoryStore c
57 18/11/07 23:01:46 INFO BlockManager: BlockManager
58 18/11/07 23:01:46 INFO BlockManagerMaster: BlockMa
59 18/11/07 23:01:46 INFO OutputCommitCoordinator$Out
60 18/11/07 23:01:46 INFO SparkContext: Successfully
61 18/11/07 23:01:46 INFO ShutdownHookManager: Shutd
62 18/11/07 23:01:46 INFO ShutdownHookManager: Delet
63
```

Note: In this tutorial we run it in **local mode**, which is good for development & debugging. When you run in cluster mode, you need to package your code as a jar file, and run it with the **spark-submit command against a Spark cluster**.

Step 6: You can package it as a jar file in the command-line.

```
1 ~/projects/sbt-tutorial]$ sbt package
2
```

projects/sbt-tutorial/target/scala-2.11/sbt-tutorial_2.11-0.1.jar will be built.

◀ 21: Q121 – Q124 Scala Extractor Interview Q&As

06: Setting up Spark-shell on Mac & getting started ▶

Disclaimer



The contents in this Java-Success are copyrighted and from EmpoweringTech pty ltd. The EmpoweringTech pty ltd has the right to correct or enhance the current content without any prior notice. These are general advice only, and one needs to take his/her own circumstances into consideration. The EmpoweringTech pty ltd will not be held liable for any damages caused or alleged to be caused either directly or indirectly by these materials and resources. Any trademarked names or labels used in this blog remain the property of their respective trademark owners. Links to external sites do not imply endorsement of the linked-to sites. [Privacy Policy](#).

© 2022 [java-success.com](https://www.java-success.com)

Top

