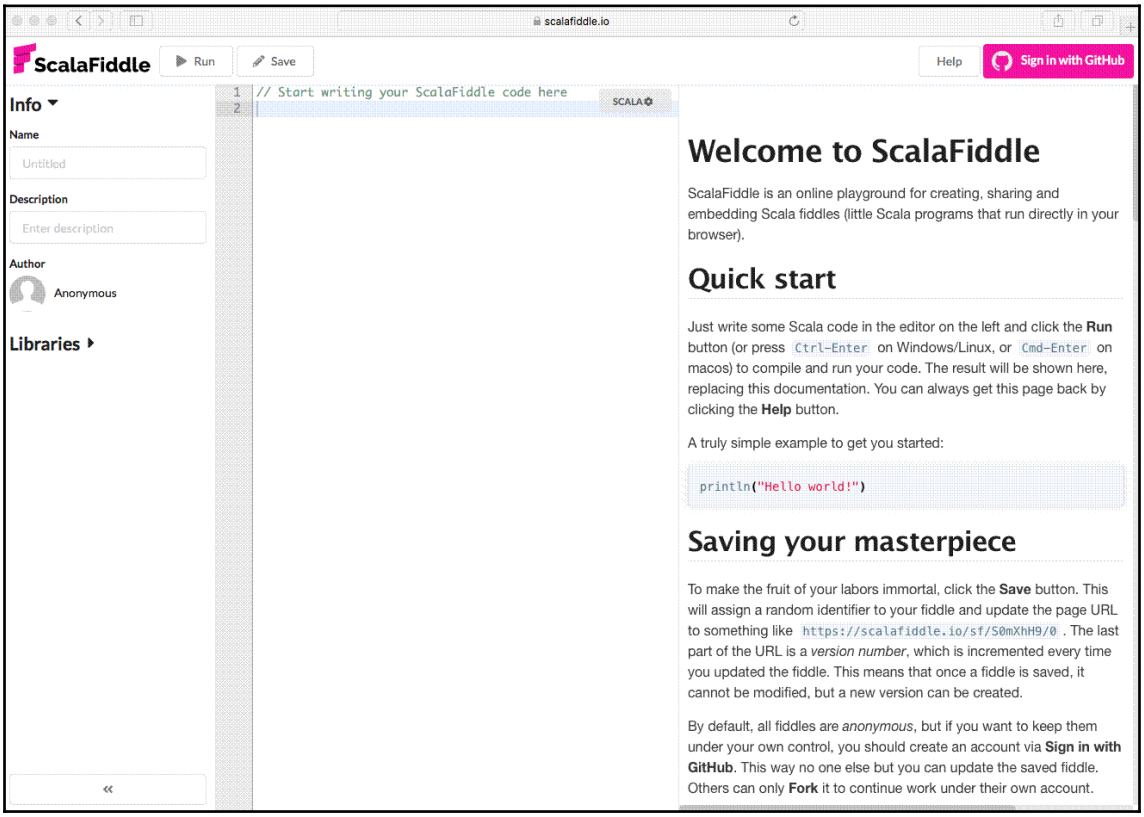
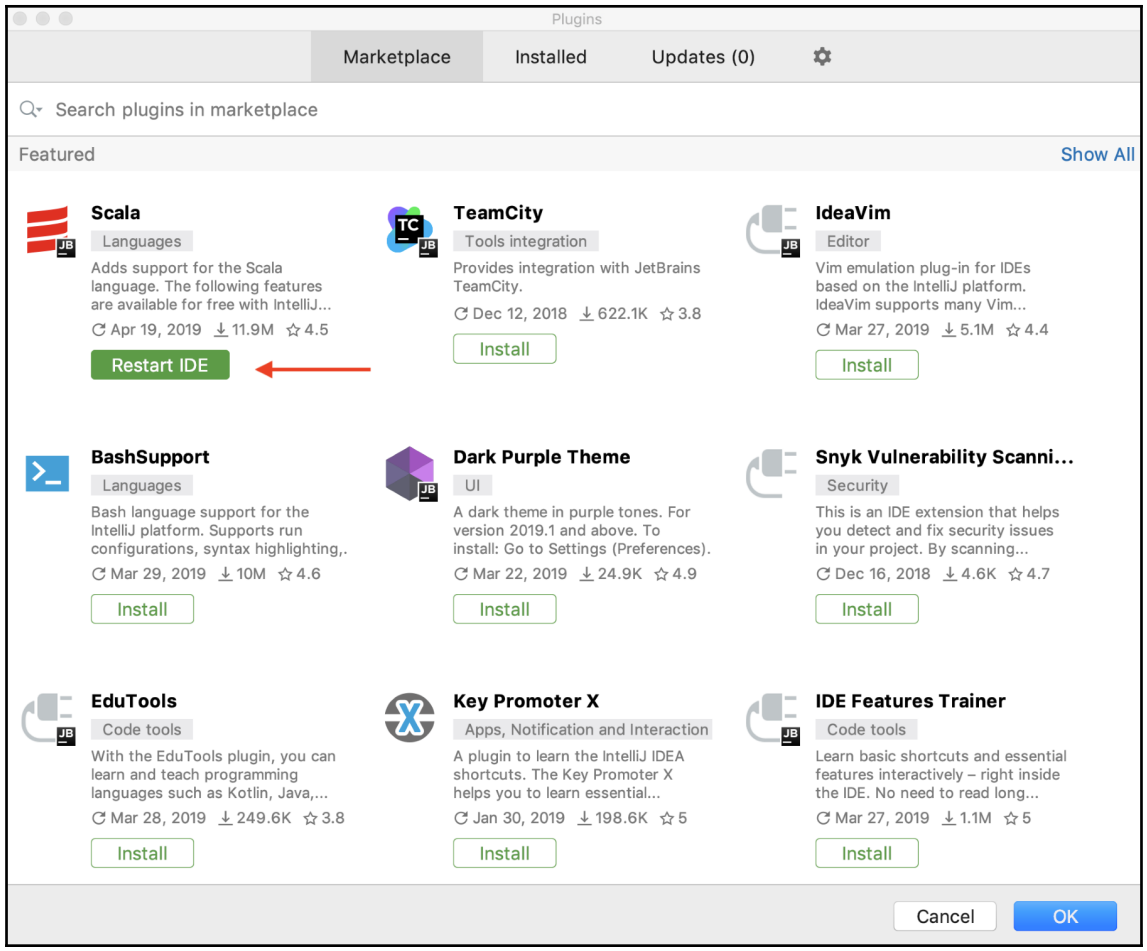
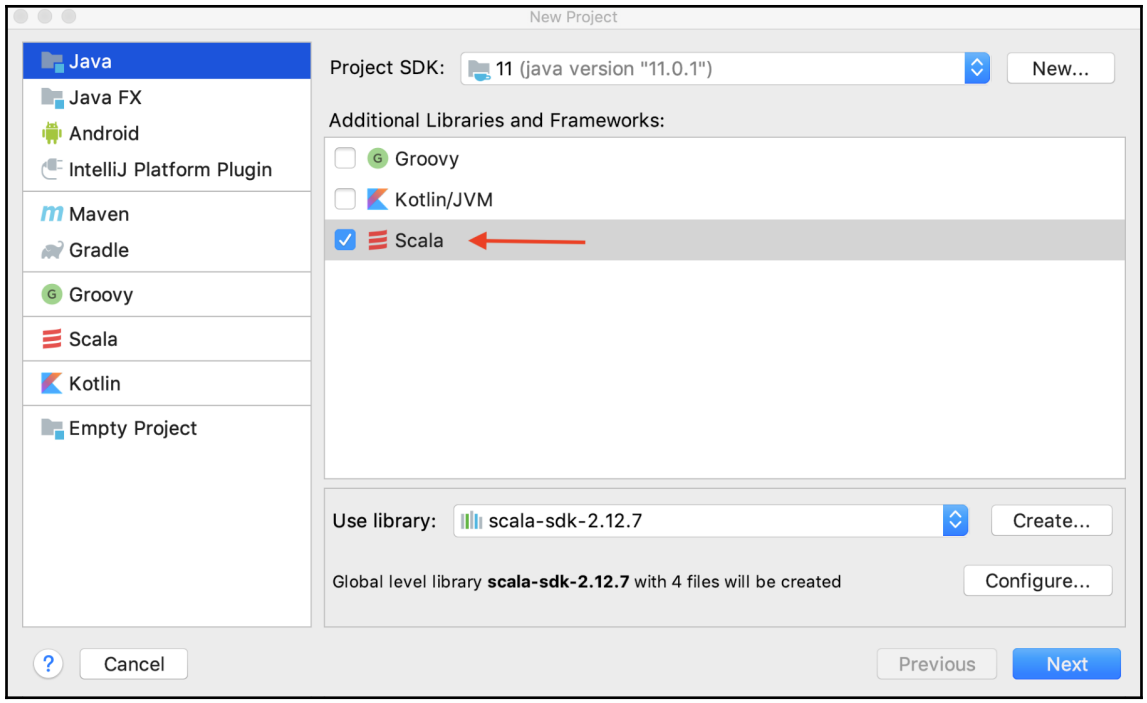
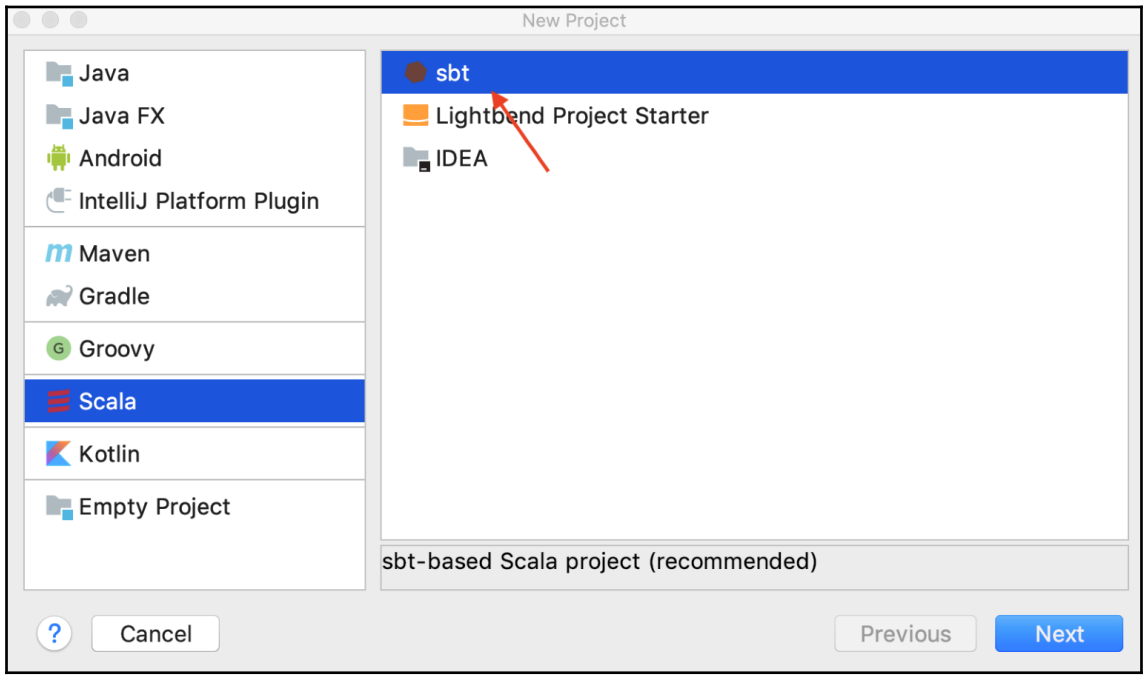
Chapter 1: Scala Overview

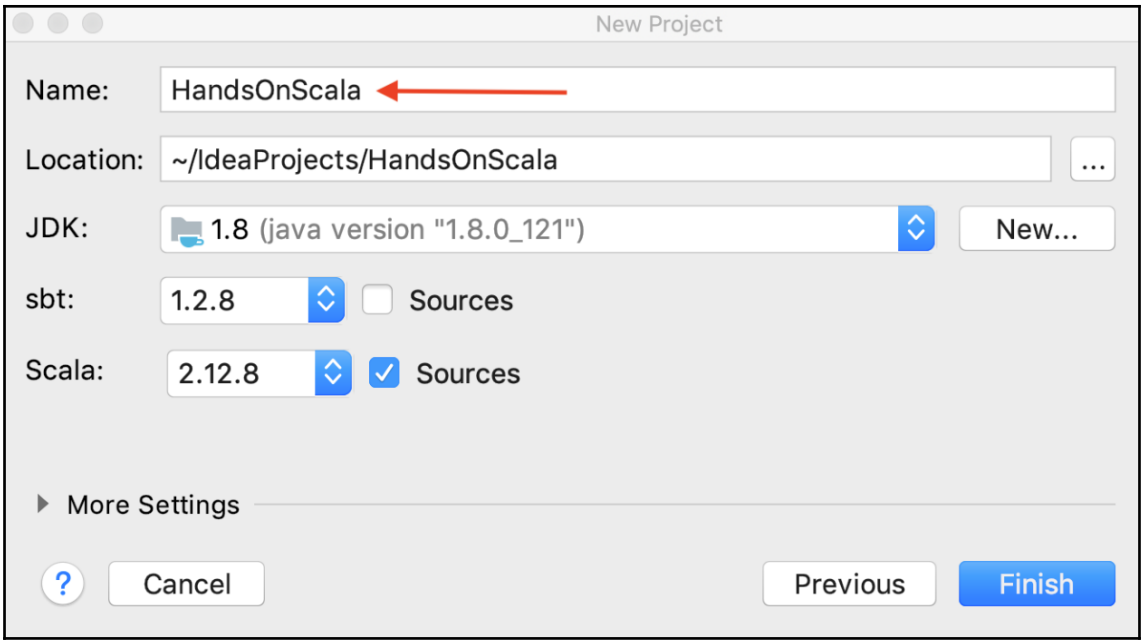


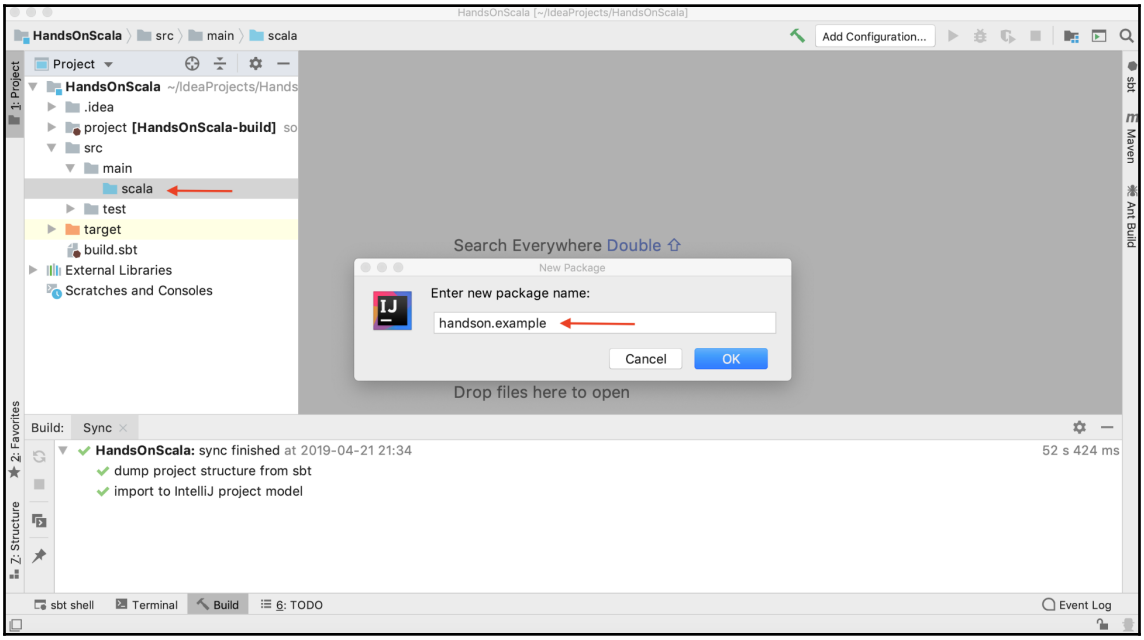

Plugins
Marketplace
Installed
Updates (0)
Search plugins in marketplace
Featured
Show All
Scala
Languages
Adds support for the Scala language. The following features are available for free with IntelliJ...
Apr 19, 2019 11.9M 4.5
Restart IDE
TeamCity
Tools integration
Provides integration with JetBrains TeamCity.
Dec 12, 2018 622.1K 3.8
Install
IdeaVim
Editor
Vim emulation plug-in for IDEs based on the IntelliJ platform. IdeaVim supports many Vim...
Mar 27, 2019 5.1M 4.4
Install
BashSupport
Languages
Bash language support for the IntelliJ platform. Supports run configurations, syntax highlighting,.
Mar 29, 2019 10M 4.6
Install
Dark Purple Theme
UI
A dark theme in purple tones. For version 2019.1 and above. To install: Go to Settings (Preferences).
Mar 22, 2019 24.9K 4.9
Install
Snyk Vulnerability Scanni...
Security
This is an IDE extension that helps you detect and fix security issues in your project. By scanning...
Dec 16, 2018 4.6K 4.7
Install
EduTools
Code tools
With the EduTools plugin, you can learn and teach programming languages such as Kotlin, Java,...
Mar 28, 2019 249.6K 3.8
Install
Key Promoter X
Apps, Notification and Interaction
A plugin to learn the IntelliJ IDEA shortcuts. The Key Promoter X helps you to learn essential...
Jan 30, 2019 198.6K 5
Install
IDE Features Trainer
Code tools
Learn basic shortcuts and essential features interactively – right inside the IDE. No need to read long...
Mar 27, 2019 1.1M 5
Install
Cancel
OK

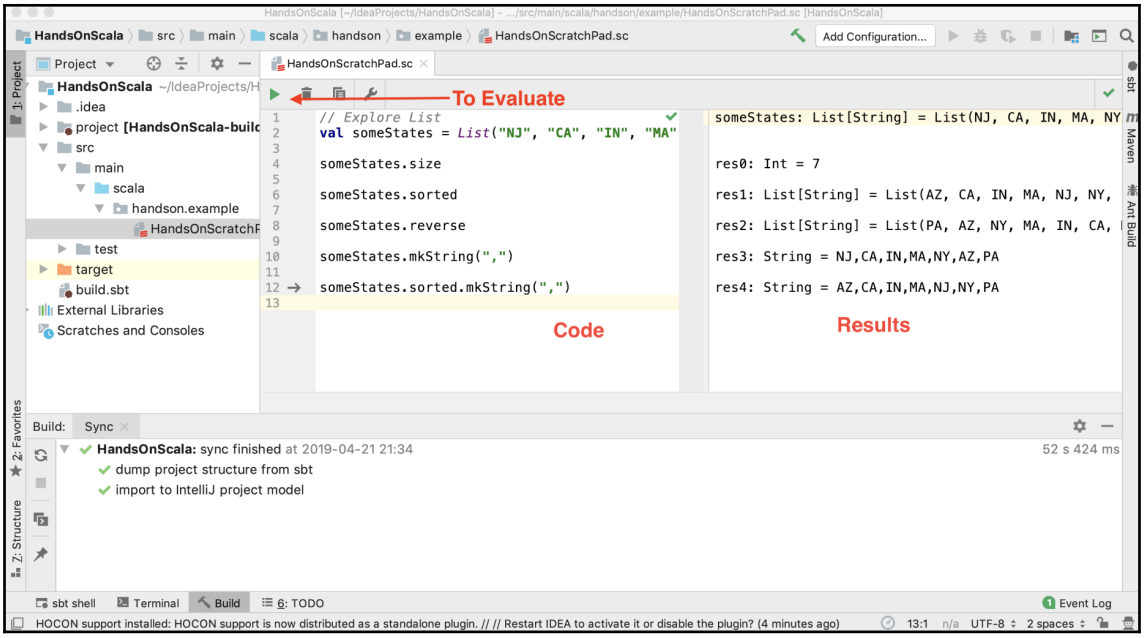

Welcome to IntelliJ IDEA
IJ
IntelliJ IDEA
Version 2019.1.1
Create New Project
Import Project
Open
Check out from Version Control
Configure
Get Help

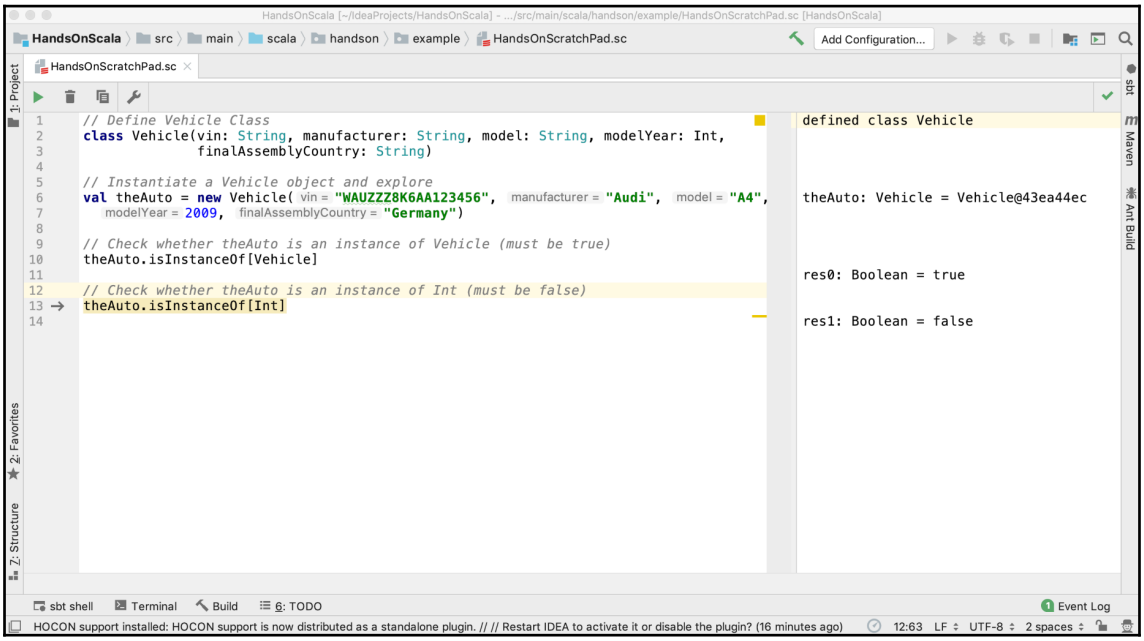

New Project
Java
Java FX
Android
IntelliJ Platform Plugin
Maven
Gradle
Groovy
Scala
Kotlin
Empty Project
Project SDK:
11 (java version "11.0.1")
New...
Additional Libraries and Frameworks:
Groovy
Kotlin/JVM
Scala
Use library:
scala-sdk-2.12.7
Create...
Global level library scala-sdk-2.12.7 with 4 files will be created
Configure...
Cancel
Previous
Next

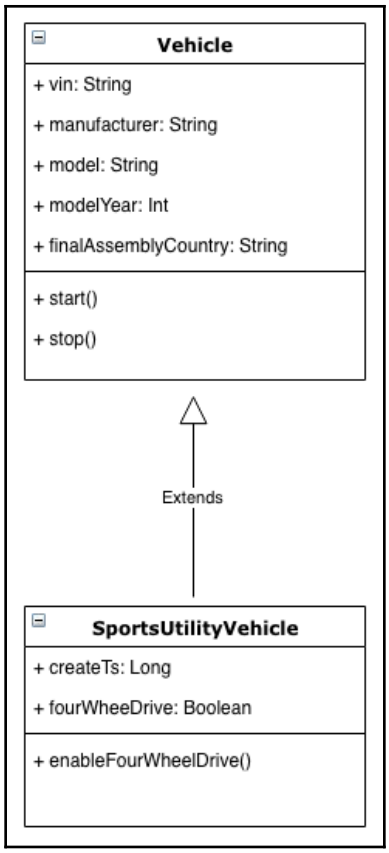

New Project
Java
Java FX
Android
IntelliJ Platform Plugin
Maven
Gradle
Groovy
Scala
Kotlin
Empty Project
sbt
Lightbend Project Starter
IDEA
sbt-based Scala project (recommended)
Cancel
Previous
Next

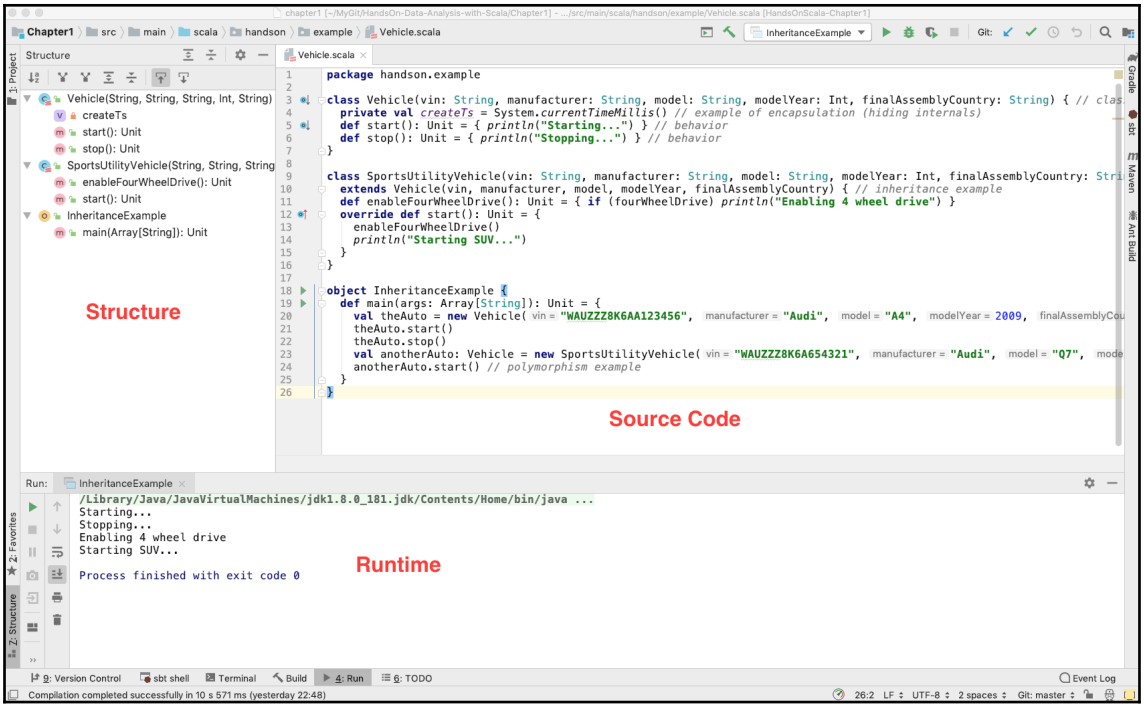

New Project
Name:
HandsOnScala
Location:
~/IdeaProjects/HandsOnScala
...
JDK:
1.8 (java version "1.8.0_121")
New...
sbt:
1.2.8
Sources
Scala:
2.12.8
Sources
More Settings
Cancel
Previous
Finish

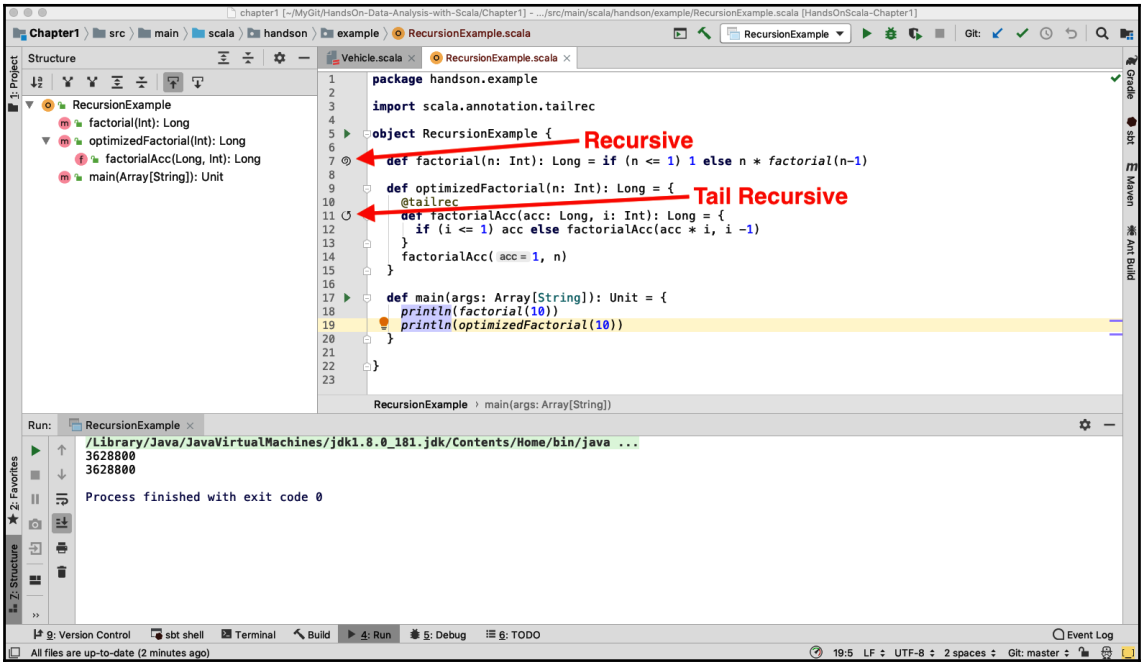

HandsOnScala [~/IdeaProjects/HandsOnScala]
HandsOnScala
src
main
scala
Add Configuration...
Project
HandsOnScala ~/IdeaProjects/Hands
.idea
project [HandsOnScala-build]
src
main
scala
test
target
build.sbt
External Libraries
Scratches and Consoles
Search Everywhere Double ⇧
New Package
Enter new package name:
handson.example
Cancel
OK
Drop files here to open
Build:
Sync
HandsOnScala: sync finished at 2019-04-21 21:34
52 s 424 ms
dump project structure from sbt
import to IntelliJ project model
sbt shell
Terminal
Build
6: TODO
Event Log
1: Project
2: Favorites
7: Structure
sbt
Maven
Ant Build

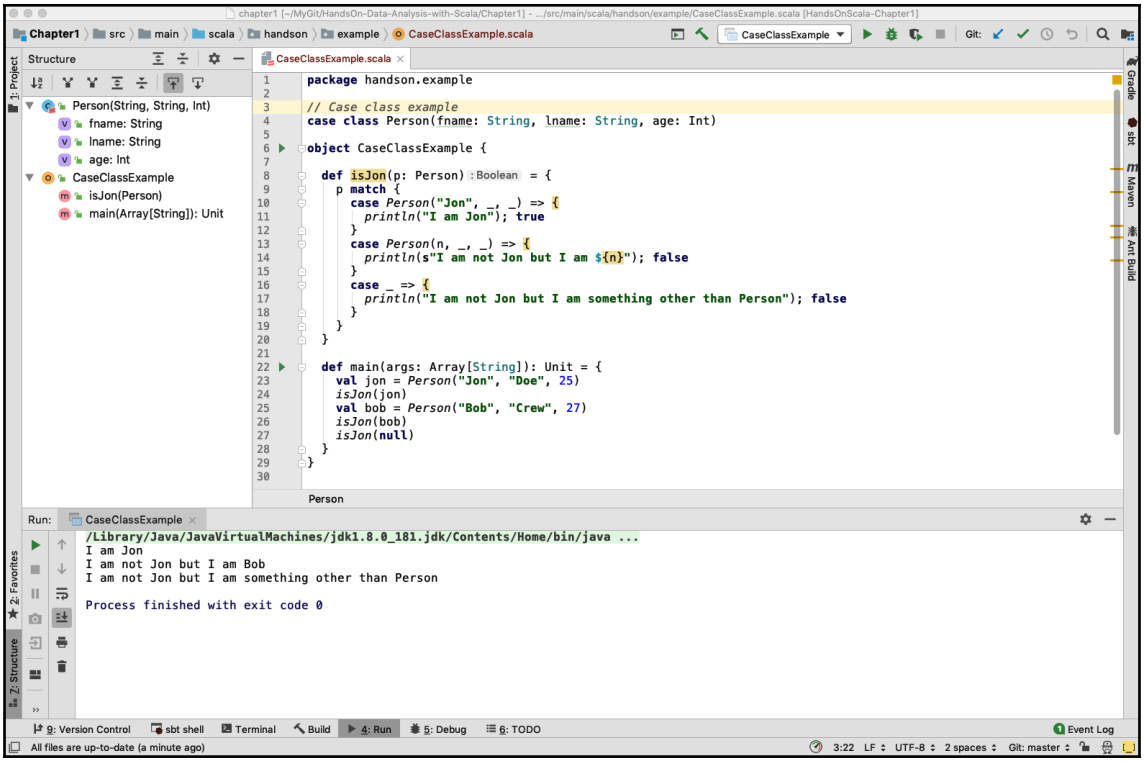

To Evaluate
// Explore List
val someStates = List("NJ", "CA", "IN", "MA"
someStates.size
someStates.sorted
someStates.reverse
someStates.mkString(",")
someStates.sorted.mkString(",")
someStates: List[String] = List(NJ, CA, IN, MA, NY
res0: Int = 7
res1: List[String] = List(AZ, CA, IN, MA, NJ, NY,
res2: List[String] = List(PA, AZ, NY, MA, IN, CA,
res3: String = NJ,CA,IN,MA,NY,AZ,PA
res4: String = AZ,CA,IN,MA,NJ,NY,PA
Code
Results

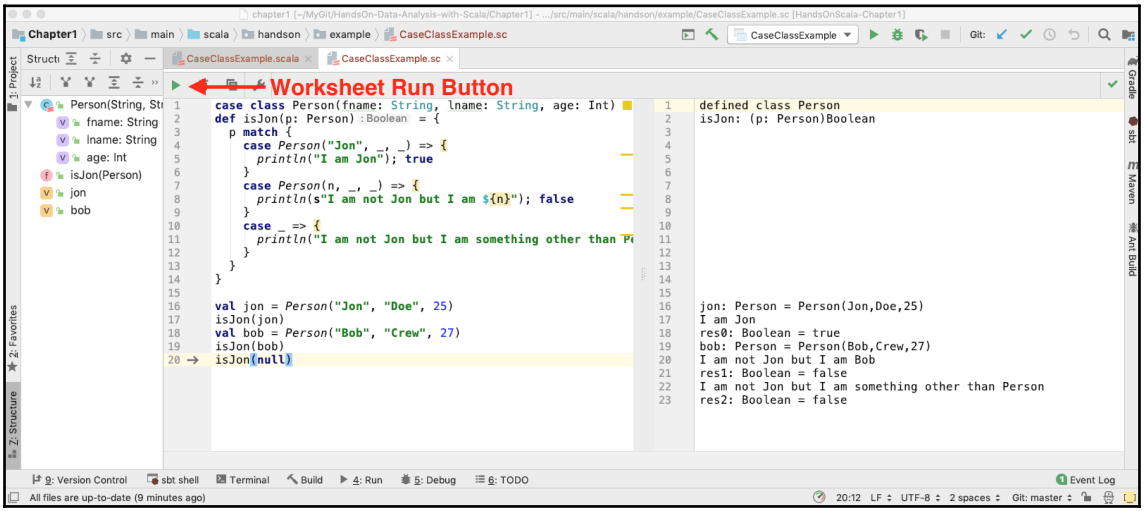

// Define Vehicle Class
class Vehicle(vin: String, manufacturer: String, model: String, modelYear: Int,
finalAssemblyCountry: String)
// Instantiate a Vehicle object and explore
val theAuto = new Vehicle(vin = "WAUZZZ8K6AA123456", manufacturer = "Audi", model = "A4",
modelYear = 2009, finalAssemblyCountry = "Germany")
// Check whether theAuto is an instance of Vehicle (must be true)
theAuto.isInstanceOf[Vehicle]
// Check whether theAuto is an instance of Int (must be false)
theAuto.isInstanceOf[Int]
defined class Vehicle
theAuto: Vehicle = Vehicle@43ea44ec
res0: Boolean = true
res1: Boolean = false


Vehicle
+ vin: String
+ manufacturer: String
+ model: String
+ modelYear: Int
+ finalAssemblyCountry: String
+ start()
+ stop()
Extends
SportsUtilityVehicle
+ createTs: Long
+ fourWheeDrive: Boolean
+ enableFourWheelDrive()

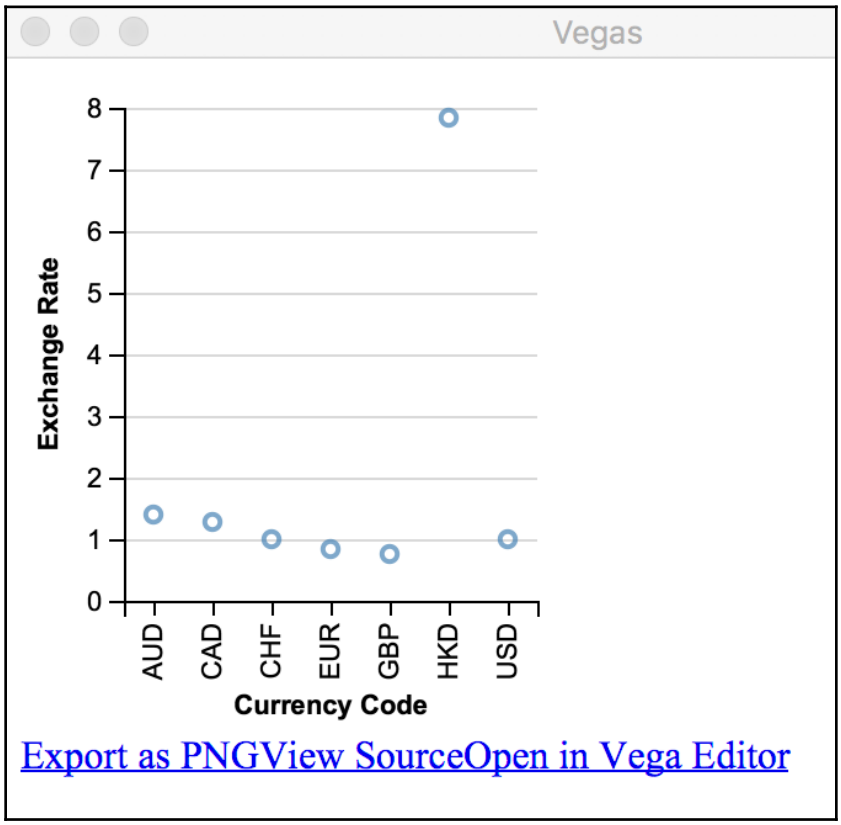

Structure
Source Code
Runtime
/Library/Java/JavaVirtualMachines/jdk1.8.0_181.jdk/Contents/Home/bin/java ...
Starting...
Stopping...
Enabling 4 wheel drive
Starting SUV...
Process finished with exit code 0

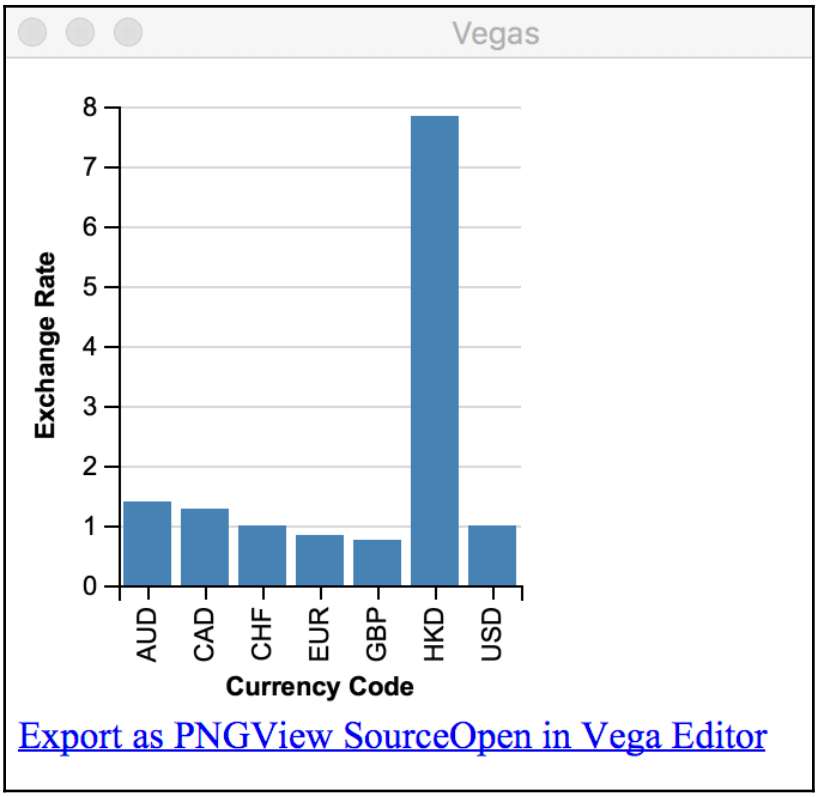

Recursive
Tail Recursive
/Library/Java/JavaVirtualMachines/jdk1.8.0_181.jdk/Contents/Home/bin/java ...
3628800
3628800
Process finished with exit code 0

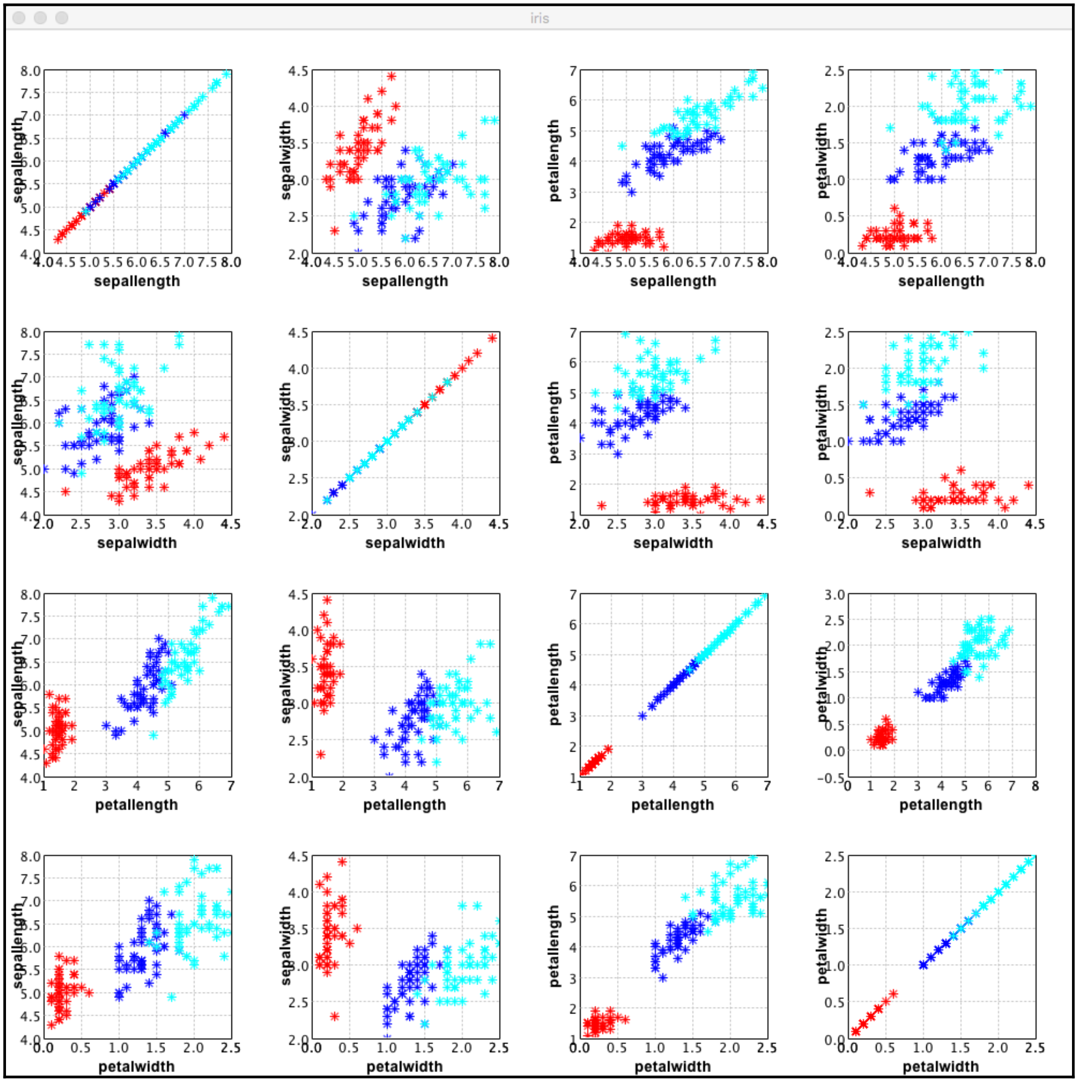

chapter1 [~/MyGit/HandsOn-Data-Analysis-with-Scala/Chapter1] - .../src/main/scala/handson/example/CaseClassExample.scala [HandsOnScala-Chapter1]
Chapter1 src main scala handson example CaseClassExample.scala
CaseClassExample
Git:
Structure
CaseClassExample.scala
Person(String, String, Int)
fname: String
lname: String
age: Int
CaseClassExample
isJon(Person)
main(Array[String]): Unit
1 package handson.example
2
3 // Case class example
4 case class Person(fname: String, lname: String, age: Int)
5
6 object CaseClassExample {
7
8 def isJon(p: Person) :Boolean = {
9 p match {
10 case Person("Jon", _, _) => {
11 println("I am Jon"); true
12 }
13 case Person(n, _, _) => {
14 println(s"I am not Jon but I am ${n}"); false
15 }
16 case _ => {
17 println("I am not Jon but I am something other than Person"); false
18 }
19 }
20 }
21
22 def main(args: Array[String]): Unit = {
23 val jon = Person("Jon", "Doe", 25)
24 isJon(jon)
25 val bob = Person("Bob", "Crew", 27)
26 isJon(bob)
27 isJon(null)
28 }
29 }
30
Person
Run: CaseClassExample
/Library/Java/JavaVirtualMachines/jdk1.8.0_181.jdk/Contents/Home/bin/java ...
I am Jon
I am not Jon but I am Bob
I am not Jon but I am something other than Person
Process finished with exit code 0
9: Version Control
sbt shell
Terminal
Build
4: Run
5: Debug
6: TODO
Event Log
All files are up-to-date (a minute ago)
3:22 LF UTF-8 2 spaces Git: master
1: Project
2: Favorites
Z: Structure
Gradle
sbt
Maven
Ant Build


chapter1 [~/MyGit/HandsOn-Data-Analysis-with-Scala/Chapter1] - .../src/main/scala/handson/example/CaseClassExample.sc [HandsOnScala-Chapter1]
Chapter1 src main scala handson example CaseClassExample.sc
CaseClassExample
Git:
CaseClassExample.scala
CaseClassExample.sc
Worksheet Run Button
Person(String, Str
fname: String
lname: String
age: Int
isJon(Person)
jon
bob
1 case class Person(fname: String, lname: String, age: Int)
2 def isJon(p: Person) :Boolean = {
3 p match {
4 case Person("Jon", _, _) => {
5 println("I am Jon"); true
6 }
7 case Person(n, _, _) => {
8 println(s"I am not Jon but I am ${n}"); false
9 }
10 case _ => {
11 println("I am not Jon but I am something other than Pe
12 }
13 }
14 }
15
16 val jon = Person("Jon", "Doe", 25)
17 isJon(jon)
18 val bob = Person("Bob", "Crew", 27)
19 isJon(bob)
20 isJon(null)
1 defined class Person
2 isJon: (p: Person)Boolean
16 jon: Person = Person(Jon,Doe,25)
17 I am Jon
18 res0: Boolean = true
19 bob: Person = Person(Bob,Crew,27)
20 I am not Jon but I am Bob
21 res1: Boolean = false
22 I am not Jon but I am something other than Person
23 res2: Boolean = false
9: Version Control
sbt shell
Terminal
Build
4: Run
5: Debug
6: TODO
Event Log
All files are up-to-date (9 minutes ago)
20:12 LF UTF-8 2 spaces Git: master
1: Project
2: Favorites
Z: Structure
Gradle
sbt
Maven
Ant Build

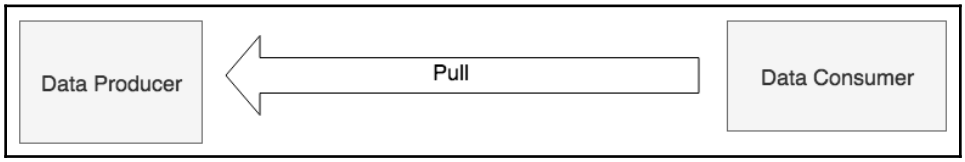
Chapter 2: Data Analysis Life Cycle

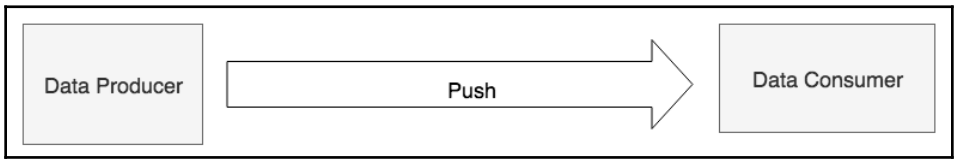


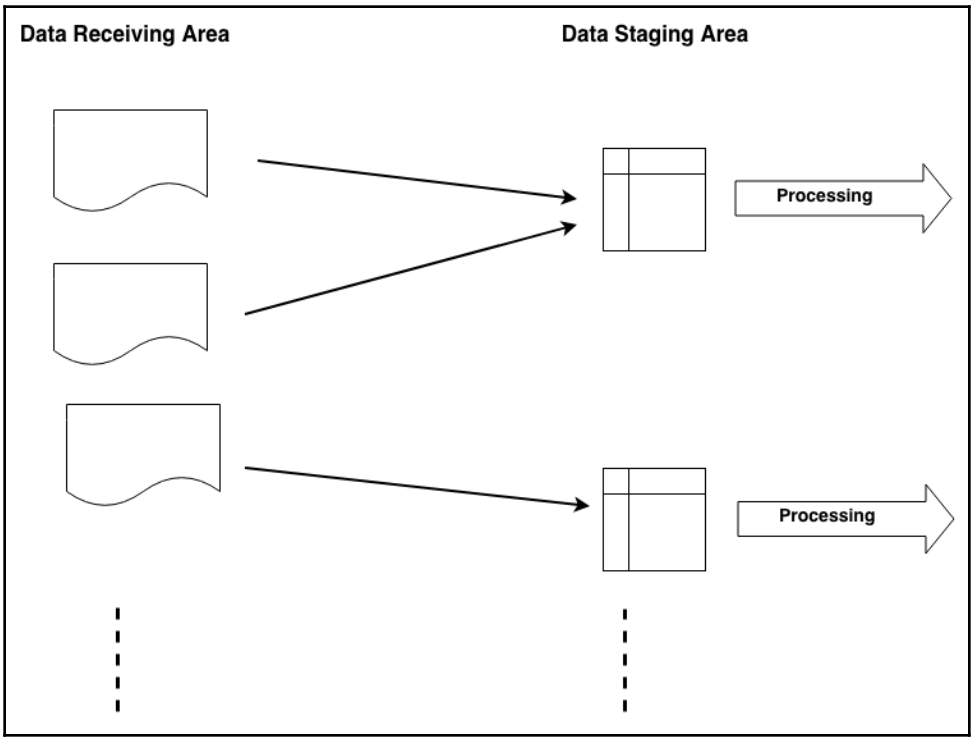

Vegas
Exchange Rate
8
7
6
5
4
3
2
1
0
AUD
CAD
CHF
EUR
GBP
HKD
USD
Currency Code
Export as PNGView SourceOpen in Vega Editor

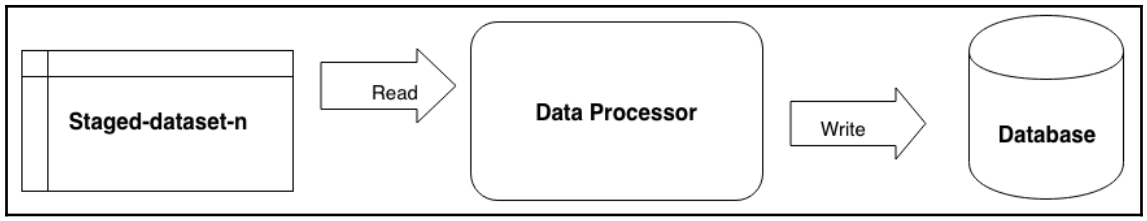

iris
sepallength
sepalwidth
petallength
petalwidth

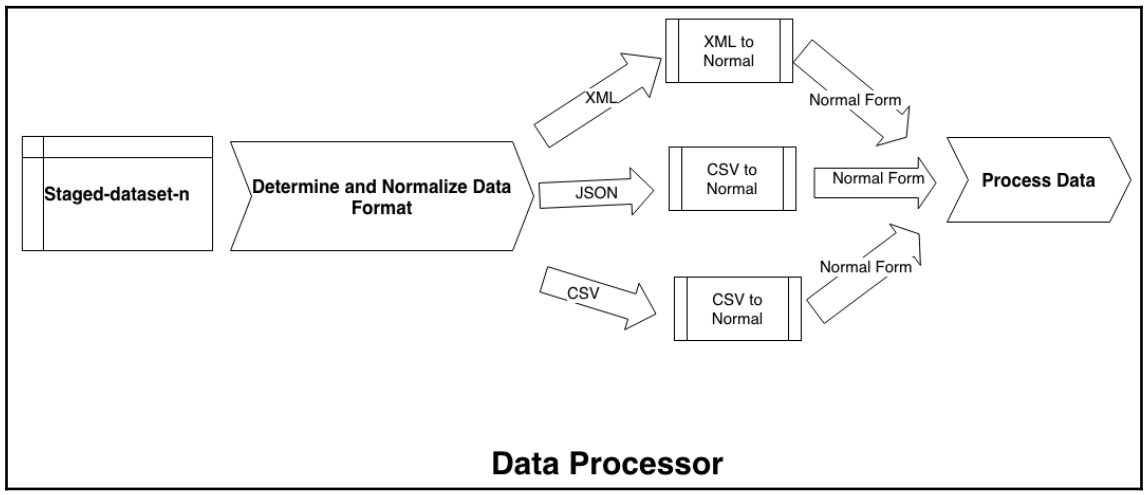
Chapter 3: Data Ingestion

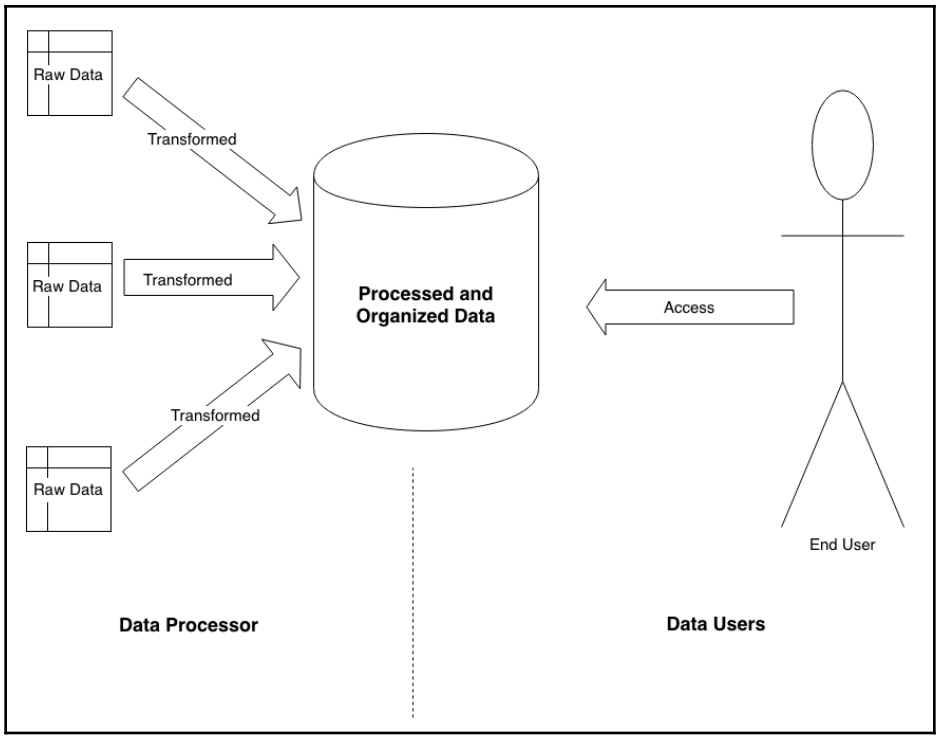




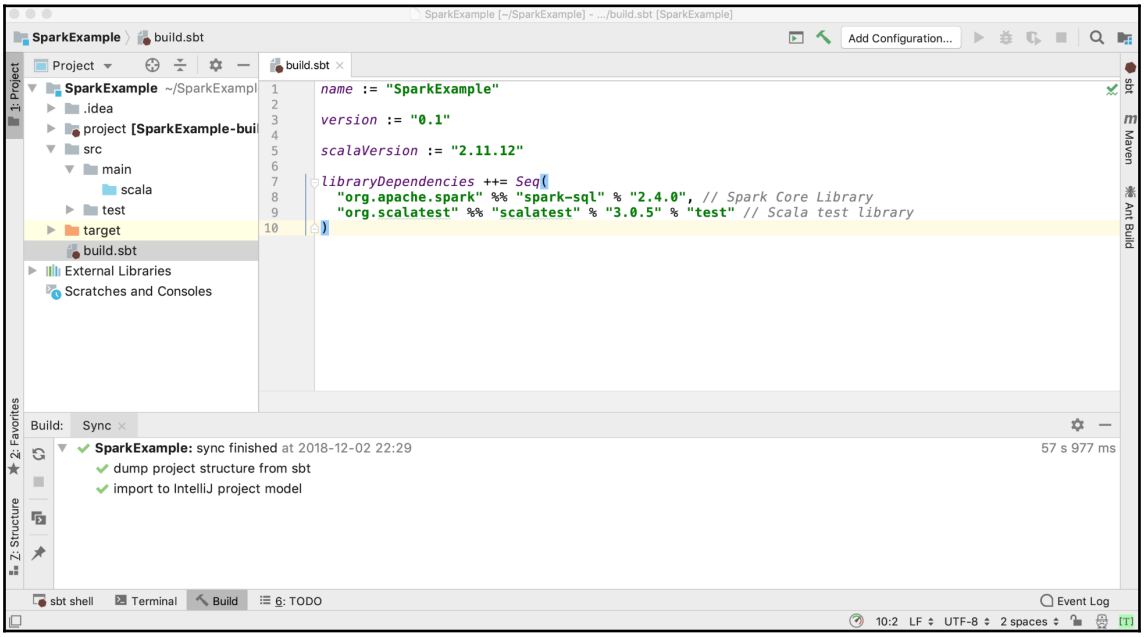


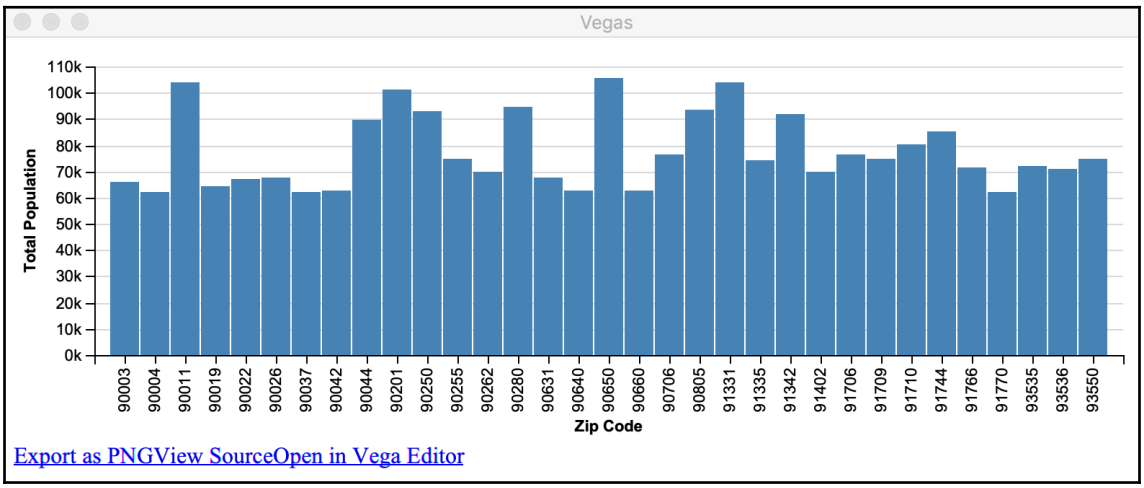

Staged-dataset-n
Read
Data Processor
Write
Database

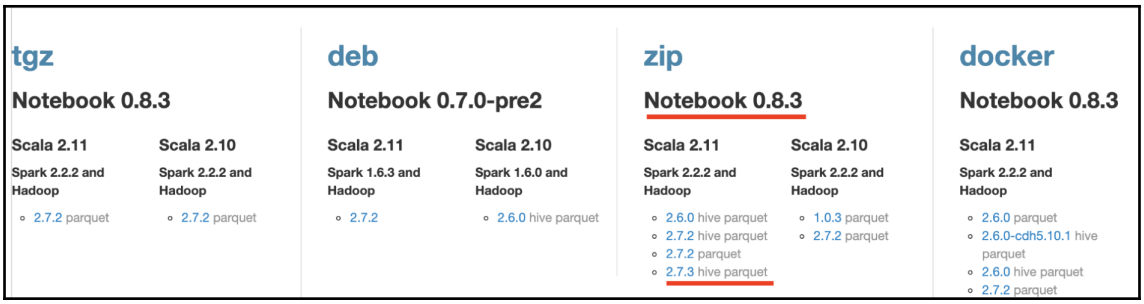

Staged-dataset-n
Determine and Normalize Data Format
XML
JSON
CSV
XML to Normal
CSV to Normal
CSV to Normal
Normal Form
Normal Form
Normal Form
Process Data
Data Processor

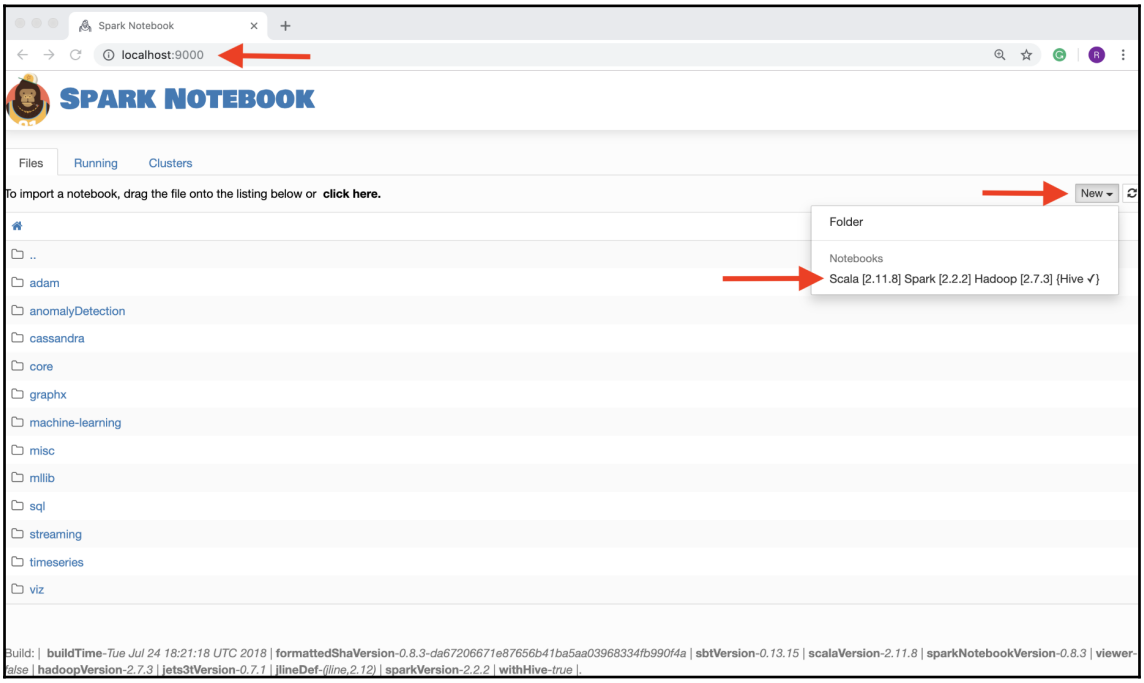

Raw Data
Transformed
Raw Data
Transformed
Processed and
Organized Data
Access
Transformed
Raw Data
End User
Data Processor
Data Users

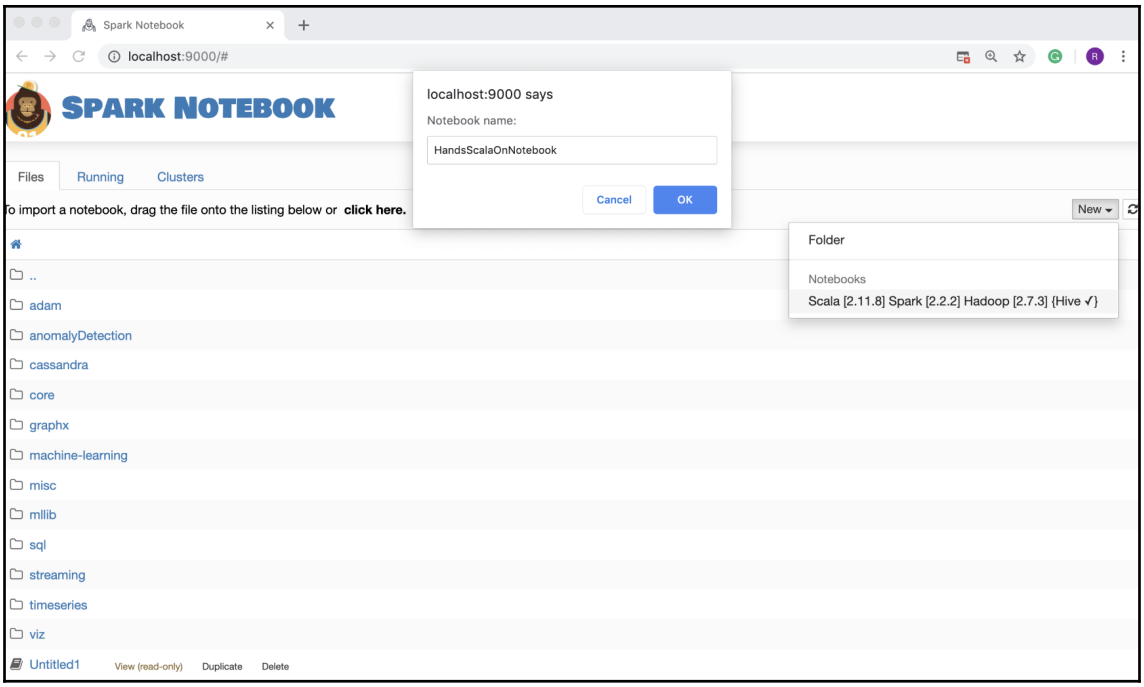
Chapter 4: Data Exploration and Visualization

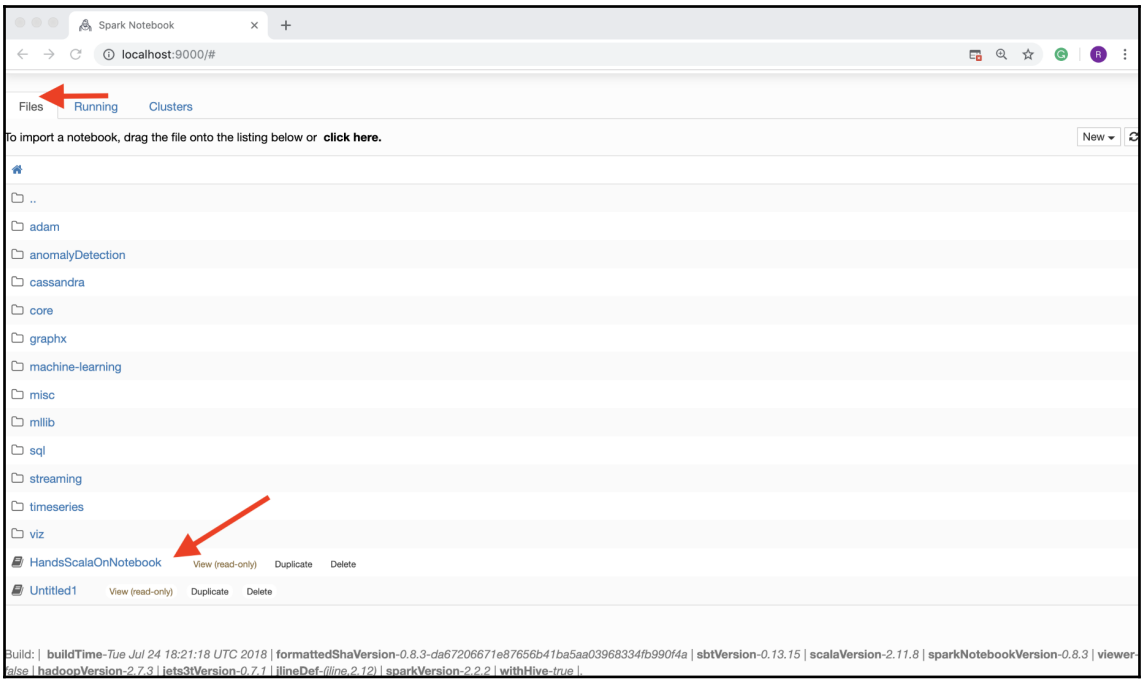


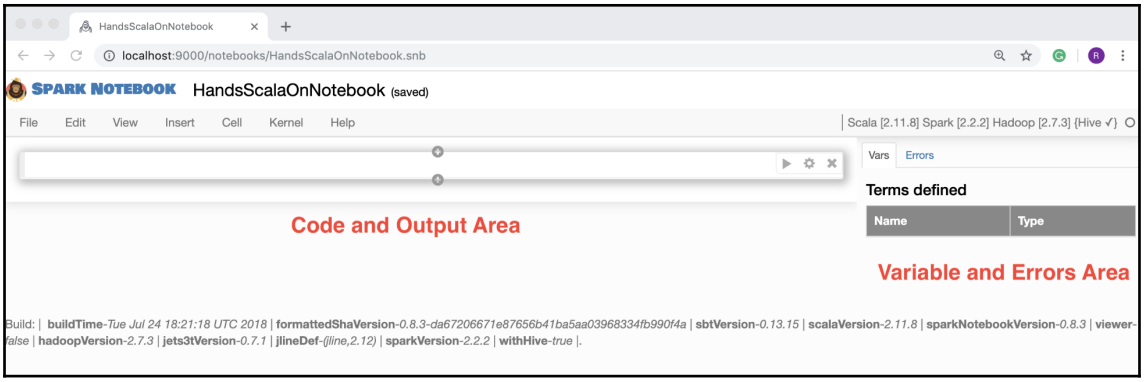


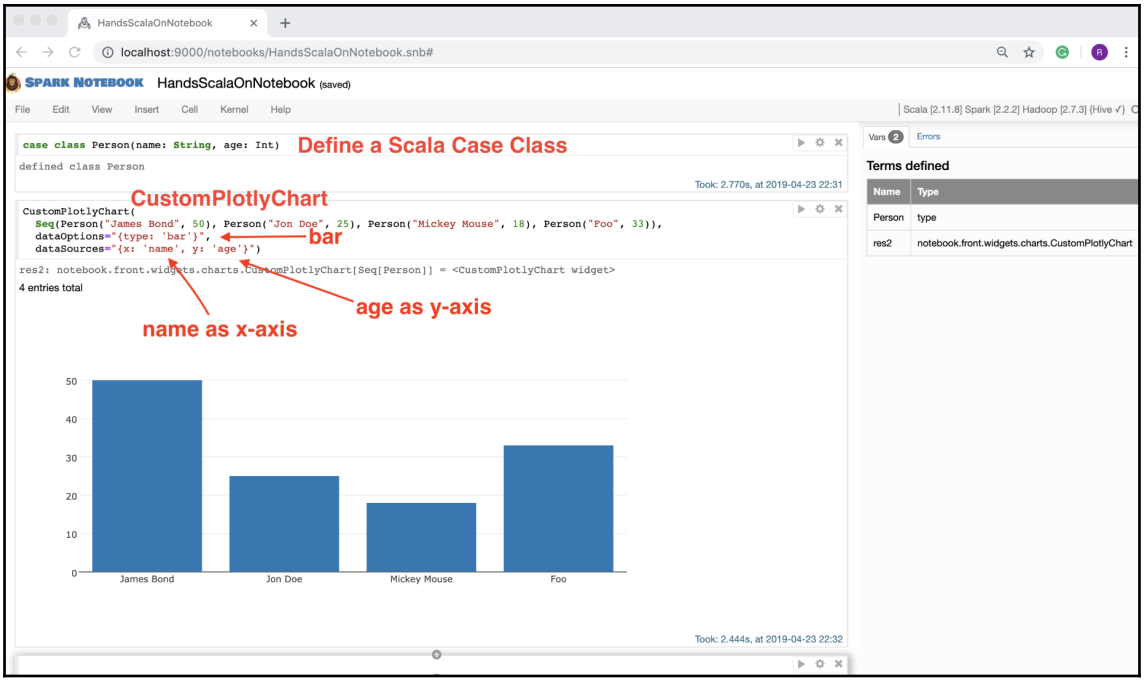

tgz
Notebook 0.8.3
Scala 2.11
Spark 2.2.2 and Hadoop
2.7.2 parquet
Scala 2.10
Spark 2.2.2 and Hadoop
2.7.2 parquet
deb
Notebook 0.7.0-pre2
Scala 2.11
Spark 1.6.3 and Hadoop
2.7.2
Scala 2.10
Spark 1.6.0 and Hadoop
2.6.0 hive parquet
zip
Notebook 0.8.3
Scala 2.11
Spark 2.2.2 and Hadoop
2.6.0 hive parquet
2.7.2 hive parquet
2.7.2 parquet
2.7.3 hive parquet
Scala 2.10
Spark 2.2.2 and Hadoop
1.0.3 parquet
2.7.2 parquet
docker
Notebook 0.8.3
Scala 2.11
Spark 2.2.2 and Hadoop
2.6.0 parquet
2.6.0-cdh5.10.1 hive parquet
2.6.0 hive parquet
2.7.2 parquet

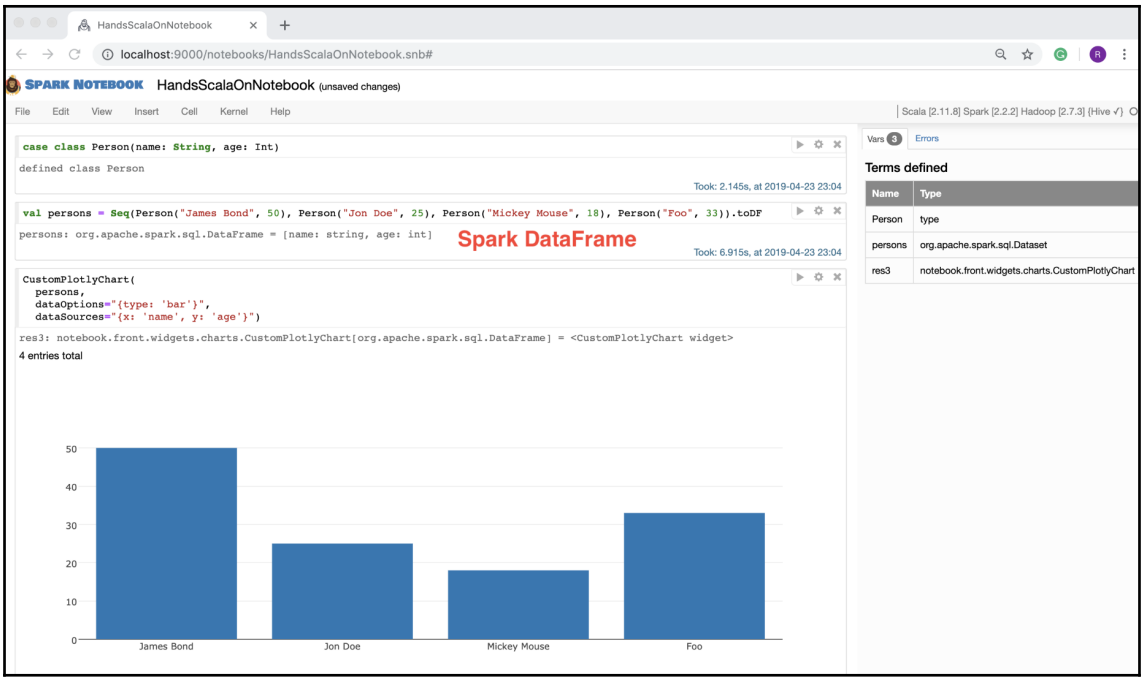

Spark Notebook
localhost:9000
SPARK NOTEBOOK
Files
Running
Clusters
To import a notebook, drag the file onto the listing below or click here.
New
Folder
Notebooks
Scala [2.11.8] Spark [2.2.2] Hadoop [2.7.3] {Hive ✓}
..
adam
anomalyDetection
cassandra
core
graphx
machine-learning
misc
mllib
sql
streaming
timeseries
viz
Build: | buildTime-Tue Jul 24 18:21:18 UTC 2018 | formattedShaVersion-0.8.3-da67206671e87656b41ba5aa03968334fb990f4a | sbtVersion-0.13.15 | scalaVersion-2.11.8 | sparkNotebookVersion-0.8.3 | viewer-false | hadoopVersion-2.7.3 | jets3tVersion-0.7.1 | jlineDef-(jline,2.12) | sparkVersion-2.2.2 | withHive-true |.

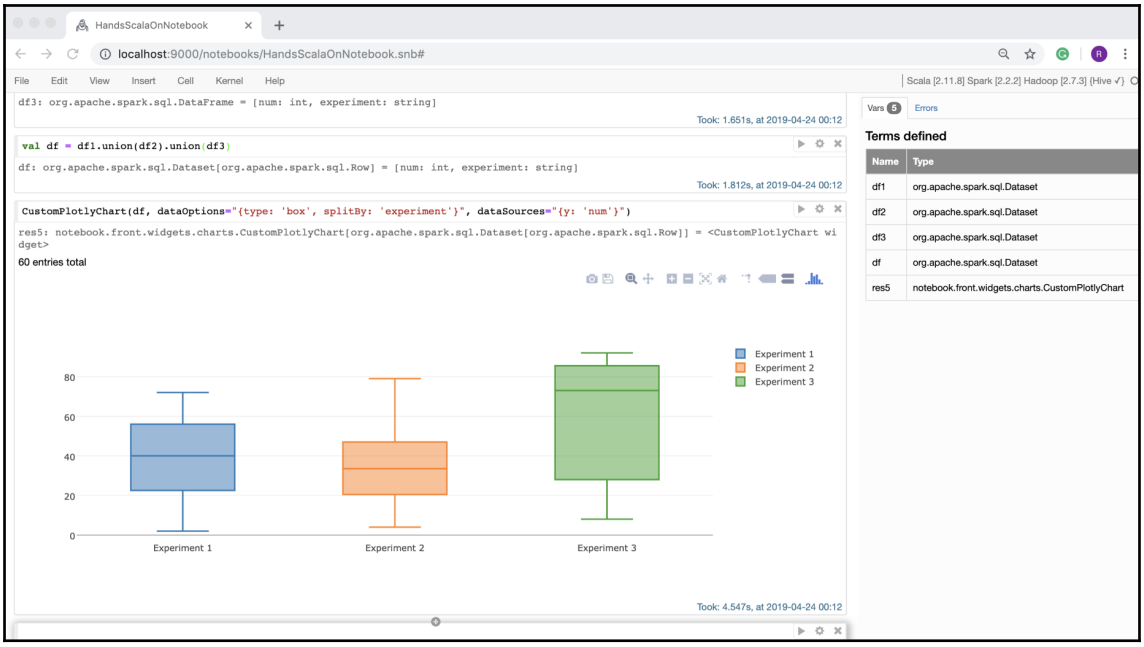

Spark Notebook
localhost:9000/#
SPARK NOTEBOOK
localhost:9000 says
Notebook name:
HandsScalaOnNotebook
Cancel
OK
Files
Running
Clusters
To import a notebook, drag the file onto the listing below or click here.
New
Folder
Notebooks
Scala [2.11.8] Spark [2.2.2] Hadoop [2.7.3] {Hive ✓}
..
adam
anomalyDetection
cassandra
core
graphx
machine-learning
misc
mllib
sql
streaming
timeseries
viz
Untitled1
View (read-only)
Duplicate
Delete

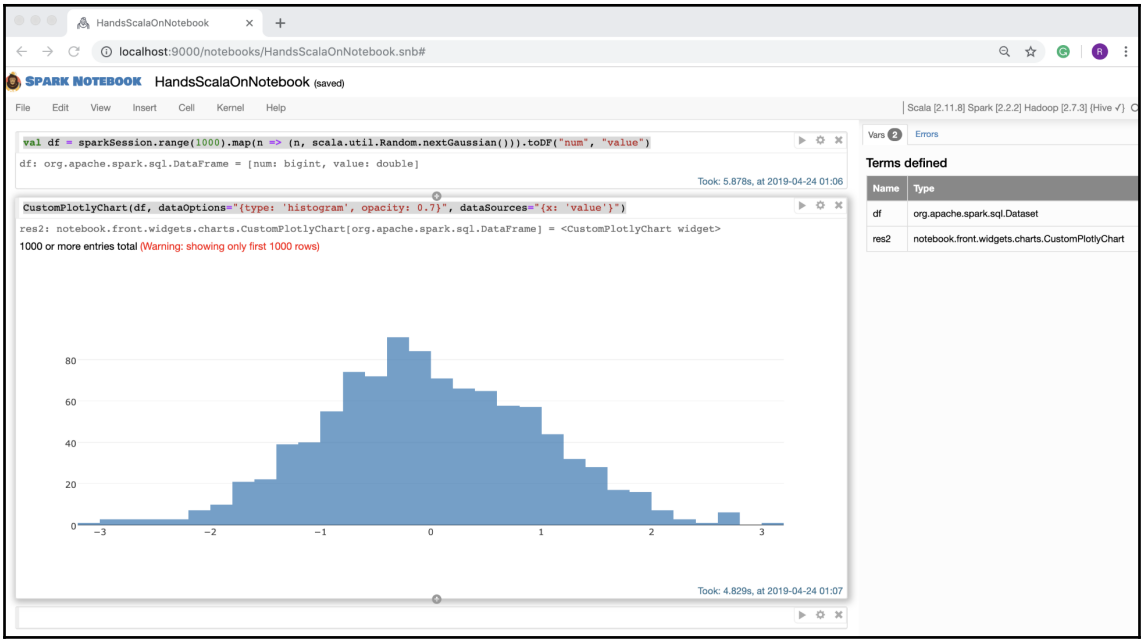

Spark Notebook
localhost:9000/#
Files
Running
Clusters
To import a notebook, drag the file onto the listing below or click here.
New
..
adam
anomalyDetection
cassandra
core
graphx
machine-learning
misc
mllib
sql
streaming
timeseries
viz
HandsScalaOnNotebook
View (read-only)
Duplicate
Delete
Untitled1
View (read-only)
Duplicate
Delete
Build: | buildTime-Tue Jul 24 18:21:18 UTC 2018 | formattedShaVersion-0.8.3-da67206671e87656b41ba5aa03968334fb990f4a | sbtVersion-0.13.15 | scalaVersion-2.11.8 | sparkNotebookVersion-0.8.3 | viewer-
false | hadoopVersion-2.7.3 | jets3tVersion-0.7.1 | jlineDef-(jline,2.12) | sparkVersion-2.2.2 | withHive-true |

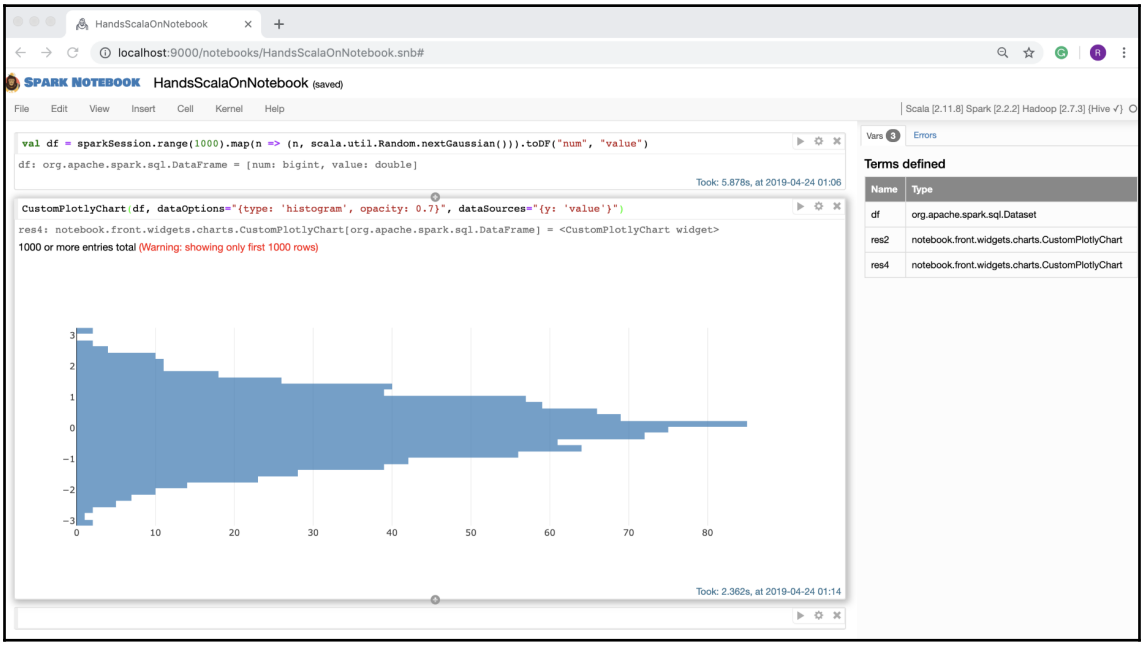

HandsScalaOnNotebook
localhost:9000/notebooks/HandsScalaOnNotebook.snb
SPARK NOTEBOOK HandsScalaOnNotebook (saved)
File Edit View Insert Cell Kernel Help
Scala [2.11.8] Spark [2.2.2] Hadoop [2.7.3] {Hive ✓}
Vars Errors
Terms defined
Name Type
Code and Output Area
Variable and Errors Area
Build: | buildTime-Tue Jul 24 18:21:18 UTC 2018 | formattedShaVersion-0.8.3-da67206671e87656b41ba5aa03968334fb990f4a | sbtVersion-0.13.15 | scalaVersion-2.11.8 | sparkNotebookVersion-0.8.3 | viewer-false | hadoopVersion-2.7.3 | jets3tVersion-0.7.1 | jlineDef-(jline,2.12) | sparkVersion-2.2.2 | withHive-true |.

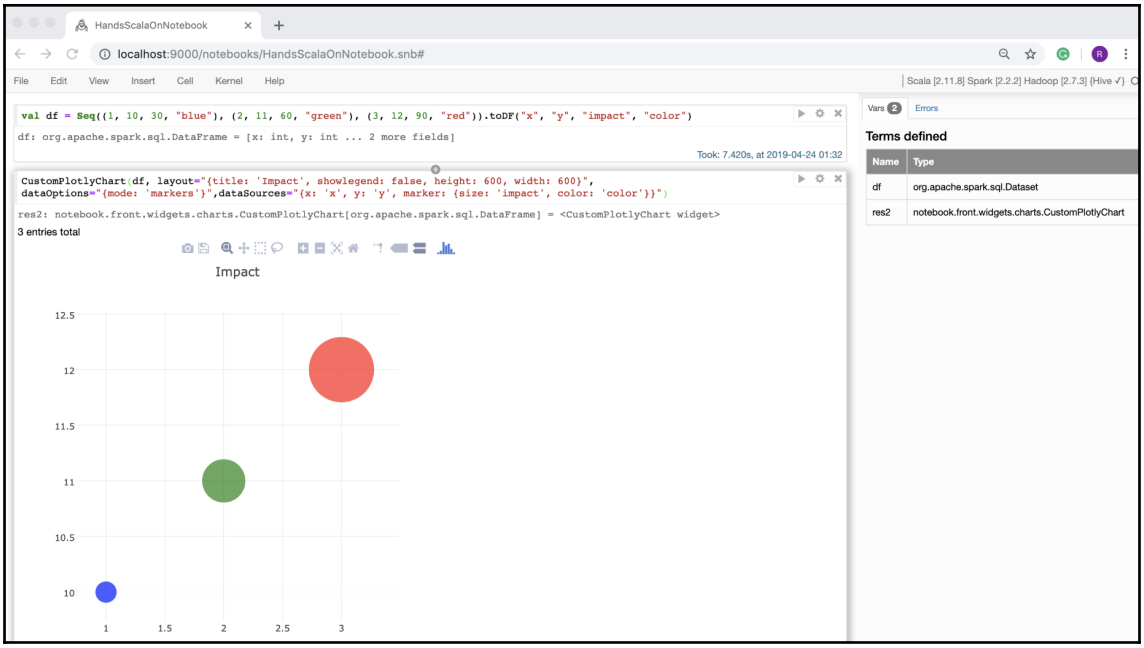

HandsScalaOnNotebook
localhost:9000/notebooks/HandsScalaOnNotebook.snb#
SPARK NOTEBOOK HandsScalaOnNotebook (saved)
File Edit View Insert Cell Kernel Help
Scala [2.11.8] Spark [2.2.2] Hadoop [2.7.3] {Hive ✓}
case class Person(name: String, age: Int)
Define a Scala Case Class
defined class Person
Took: 2.770s, at 2019-04-23 22:31
CustomPlotlyChart
CustomPlotlyChart(
Seq(Person("James Bond", 50), Person("Jon Doe", 25), Person("Mickey Mouse", 18), Person("Foo", 33)),
dataOptions="{type: 'bar'}",
dataSources="{x: 'name', y: 'age'}")
bar
res2: notebook.front.widgets.charts.CustomPlotlyChart[Seq[Person]] = <CustomPlotlyChart widget>
4 entries total
age as y-axis
name as x-axis
50
40
30
20
10
0
James Bond
Jon Doe
Mickey Mouse
Foo
Took: 2.444s, at 2019-04-23 22:32
Vars 2
Errors
Terms defined
Name Type
Person type
res2 notebook.front.widgets.charts.CustomPlotlyChart


HandsScalaOnNotebook
localhost:9000/notebooks/HandsScalaOnNotebook.snb#
SPARK NOTEBOOK HandsScalaOnNotebook (unsaved changes)
File Edit View Insert Cell Kernel Help
Scala [2.11.8] Spark [2.2.2] Hadoop [2.7.3] (Hive ✓)
case class Person(name: String, age: Int)
defined class Person
Took: 2.145s, at 2019-04-23 23:04
val persons = Seq(Person("James Bond", 50), Person("Jon Doe", 25), Person("Mickey Mouse", 18), Person("Foo", 33)).toDF
persons: org.apache.spark.sql.DataFrame = [name: string, age: int]
Spark DataFrame
Took: 6.915s, at 2019-04-23 23:04
CustomPlotlyChart(
persons,
dataOptions="{type: 'bar'}",
dataSources="{x: 'name', y: 'age'}")
res3: notebook.front.widgets.charts.CustomPlotlyChart[org.apache.spark.sql.DataFrame] = <CustomPlotlyChart widget>
4 entries total
50
40
30
20
10
0
James Bond
Jon Doe
Mickey Mouse
Foo
Vars 3
Errors
Terms defined
Name
Type
Person
type
persons
org.apache.spark.sql.Dataset
res3
notebook.front.widgets.charts.CustomPlotlyChart


HandsScalaOnNotebook
localhost:9000/notebooks/HandsScalaOnNotebook.snb#
File Edit View Insert Cell Kernel Help
Scala [2.11.8] Spark [2.2.2] Hadoop [2.7.3] (Hive ✓)
df3: org.apache.spark.sql.DataFrame = [num: int, experiment: string]
Took: 1.651s, at 2019-04-24 00:12
val df = df1.union(df2).union(df3)
df: org.apache.spark.sql.Dataset[org.apache.spark.sql.Row] = [num: int, experiment: string]
Took: 1.812s, at 2019-04-24 00:12
CustomPlotlyChart(df, dataOptions="{type: 'box', splitBy: 'experiment'}", dataSources="{y: 'num'}")
res5: notebook.front.widgets.charts.CustomPlotlyChart[org.apache.spark.sql.Dataset[org.apache.spark.sql.Row]] = <CustomPlotlyChart widget>
60 entries total
Experiment 1
Experiment 2
Experiment 3
80
60
40
20
0
Experiment 1
Experiment 2
Experiment 3
Took: 4.547s, at 2019-04-24 00:12
Vars 5
Errors
Terms defined
Name
Type
df1
org.apache.spark.sql.Dataset
df2
org.apache.spark.sql.Dataset
df3
org.apache.spark.sql.Dataset
df
org.apache.spark.sql.Dataset
res5
notebook.front.widgets.charts.CustomPlotlyChart


HandsScalaOnNotebook
localhost:9000/notebooks/HandsScalaOnNotebook.snb#
SPARK NOTEBOOK HandsScalaOnNotebook (saved)
File Edit View Insert Cell Kernel Help
Scala [2.11.8] Spark [2.2.2] Hadoop [2.7.3] {Hive ✓}
val df = sparkSession.range(1000).map(n => (n, scala.util.Random.nextGaussian())).toDF("num", "value")
df: org.apache.spark.sql.DataFrame = [num: bigint, value: double]
Took: 5.878s, at 2019-04-24 01:06
CustomPlotlyChart(df, dataOptions="{type: 'histogram', opacity: 0.7}", dataSources="{x: 'value'}")
res2: notebook.front.widgets.charts.CustomPlotlyChart[org.apache.spark.sql.DataFrame] = <CustomPlotlyChart widget>
1000 or more entries total (Warning: showing only first 1000 rows)
80
60
40
20
0
−3
−2
−1
0
1
2
3
Took: 4.829s, at 2019-04-24 01:07
Vars 2
Errors
Terms defined
Name
Type
df
org.apache.spark.sql.Dataset
res2
notebook.front.widgets.charts.CustomPlotlyChart

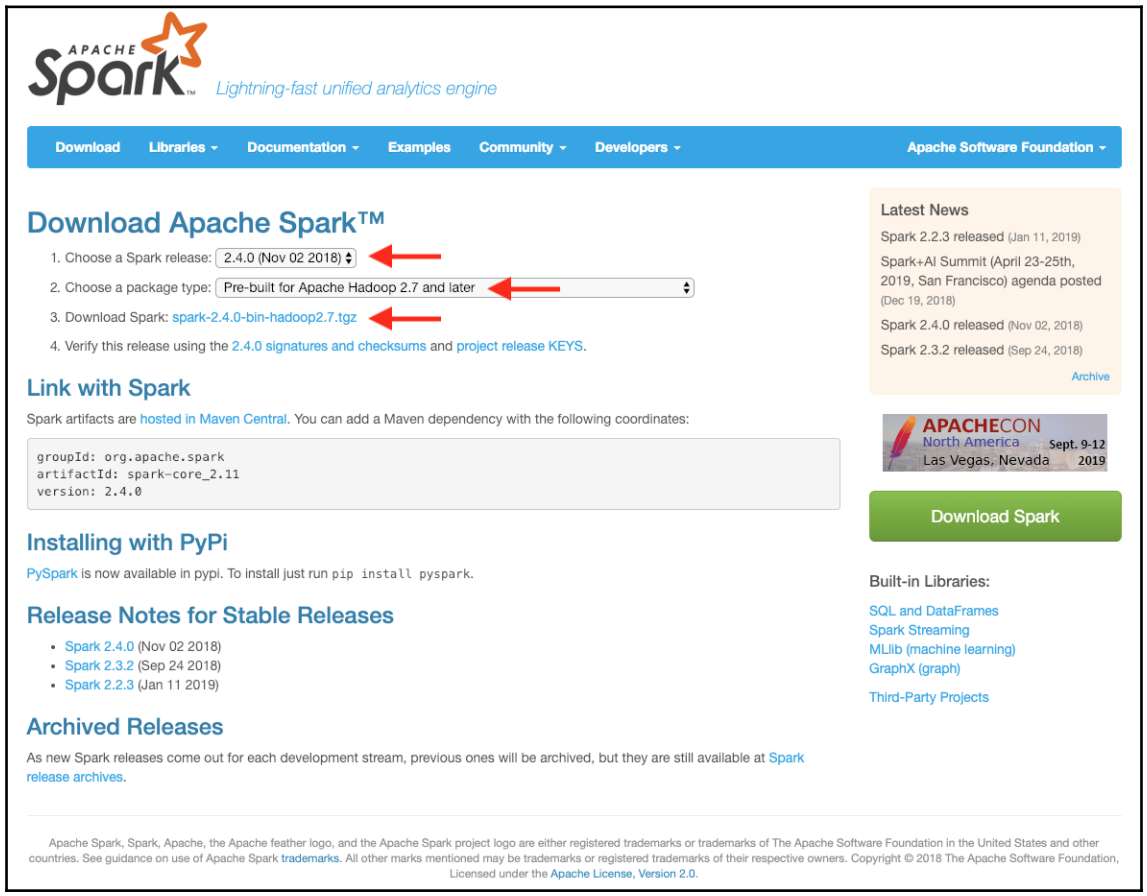

HandsScalaOnNotebook
localhost:9000/notebooks/HandsScalaOnNotebook.snb#
SPARK NOTEBOOK HandsScalaOnNotebook (saved)
File Edit View Insert Cell Kernel Help
Scala [2.11.8] Spark [2.2.2] Hadoop [2.7.3] {Hive ✓}
val df = sparkSession.range(1000).map(n => (n, scala.util.Random.nextGaussian())).toDF("num", "value")
df: org.apache.spark.sql.DataFrame = [num: bigint, value: double]
Took: 5.878s, at 2019-04-24 01:06
CustomPlotlyChart(df, dataOptions="{type: 'histogram', opacity: 0.7}", dataSources="{y: 'value'}")
res4: notebook.front.widgets.charts.CustomPlotlyChart[org.apache.spark.sql.DataFrame] = <CustomPlotlyChart widget>
1000 or more entries total (Warning: showing only first 1000 rows)
3
2
1
0
−1
−2
−3
0
10
20
30
40
50
60
70
80
Took: 2.362s, at 2019-04-24 01:14
Vars 3
Errors
Terms defined
Name
Type
df
org.apache.spark.sql.Dataset
res2
notebook.front.widgets.charts.CustomPlotlyChart
res4
notebook.front.widgets.charts.CustomPlotlyChart


HandsScalaOnNotebook
localhost:9000/notebooks/HandsScalaOnNotebook.snb#
File Edit View Insert Cell Kernel Help
Scala [2.11.8] Spark [2.2.2] Hadoop [2.7.3] {Hive ✓}
val df = Seq((1, 10, 30, "blue"), (2, 11, 60, "green"), (3, 12, 90, "red")).toDF("x", "y", "impact", "color")
df: org.apache.spark.sql.DataFrame = [x: int, y: int ... 2 more fields]
Took: 7.420s, at 2019-04-24 01:32
CustomPlotlyChart(df, layout="{title: 'Impact', showlegend: false, height: 600, width: 600}",
dataOptions="{mode: 'markers'}",dataSources="{x: 'x', y: 'y', marker: {size: 'impact', color: 'color'}}")
res2: notebook.front.widgets.charts.CustomPlotlyChart[org.apache.spark.sql.DataFrame] = <CustomPlotlyChart widget>
3 entries total
Impact
12.5
12
11.5
11
10.5
10
1
1.5
2
2.5
3
Vars 2
Errors
Terms defined
Name
Type
df
org.apache.spark.sql.Dataset
res2
notebook.front.widgets.charts.CustomPlotlyChart

Chapter 5: Applying Statistics and Hypothesis Testing

```
numsSorted: scala.collection.immutable.IndexedSeq[Int] = Vector(1, 2, 3, 4, 5, 6, 7, 8, 9, 10,
11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34,
35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49, 50, 51, 52, 53, 54, 55, 56, 57, 58,
59, 60, 61, 62, 63, 64, 65, 66, 67, 68, 69, 70, 71, 72, 73, 74, 75, 76, 77, 78, 79, 80, 81, 82,
83, 84, 85, 86, 87, 88, 89, 90, 91, 92, 93, 94, 95, 96, 97, 98, 99, 100)
```

Chapter 6: Introduction to Spark for Distributed Data Analysis



THE APACHE® SOFTWARE FOUNDATION

SUPPORT APACHE

Google Custom

The Apache Way

Contribute

ASF Sponsors

We suggest the following mirror site for your download:

http://mirror.cogentco.com/pub/apache/spark/spark-2.4.0/spark-2.4.0-bin-hadoop2.7.tgz

Other mirror sites are suggested below.

It is essential that you verify the integrity of the downloaded file using the PGP signature (`.asc` file) or a hash (`.md5` or `.sha*` file).

Please only use the backup mirrors to download KEYS, PGP and MD5 sigs/hashes or if no other mirrors are working.

HTTP

http://apache.claz.org/spark/spark-2.4.0/spark-2.4.0-bin-hadoop2.7.tgz

http://apache.cs.utah.edu/spark/spark-2.4.0/spark-2.4.0-bin-hadoop2.7.tgz

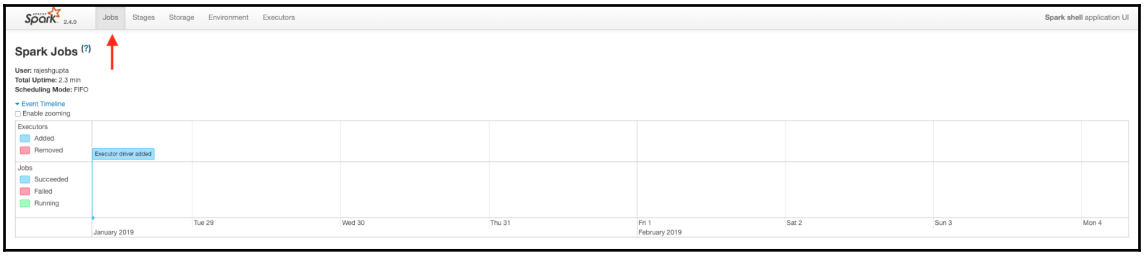
http://apache.mirrors.hoobly.com/spark/spark-2.4.0/spark-2.4.0-bin-hadoop2.7.tgz

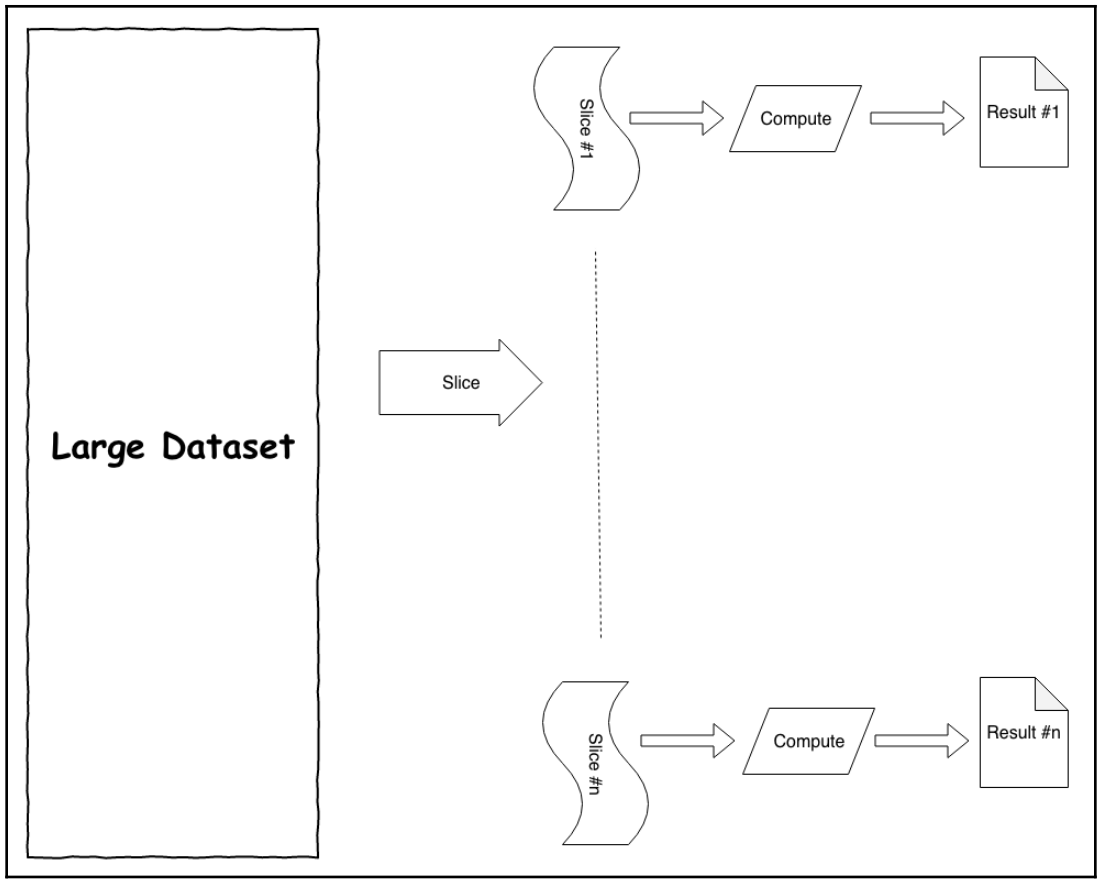
http://apache.mirrors.ionfish.org/spark/spark-2.4.0/spark-2.4.0-bin-hadoop2.7.tgz

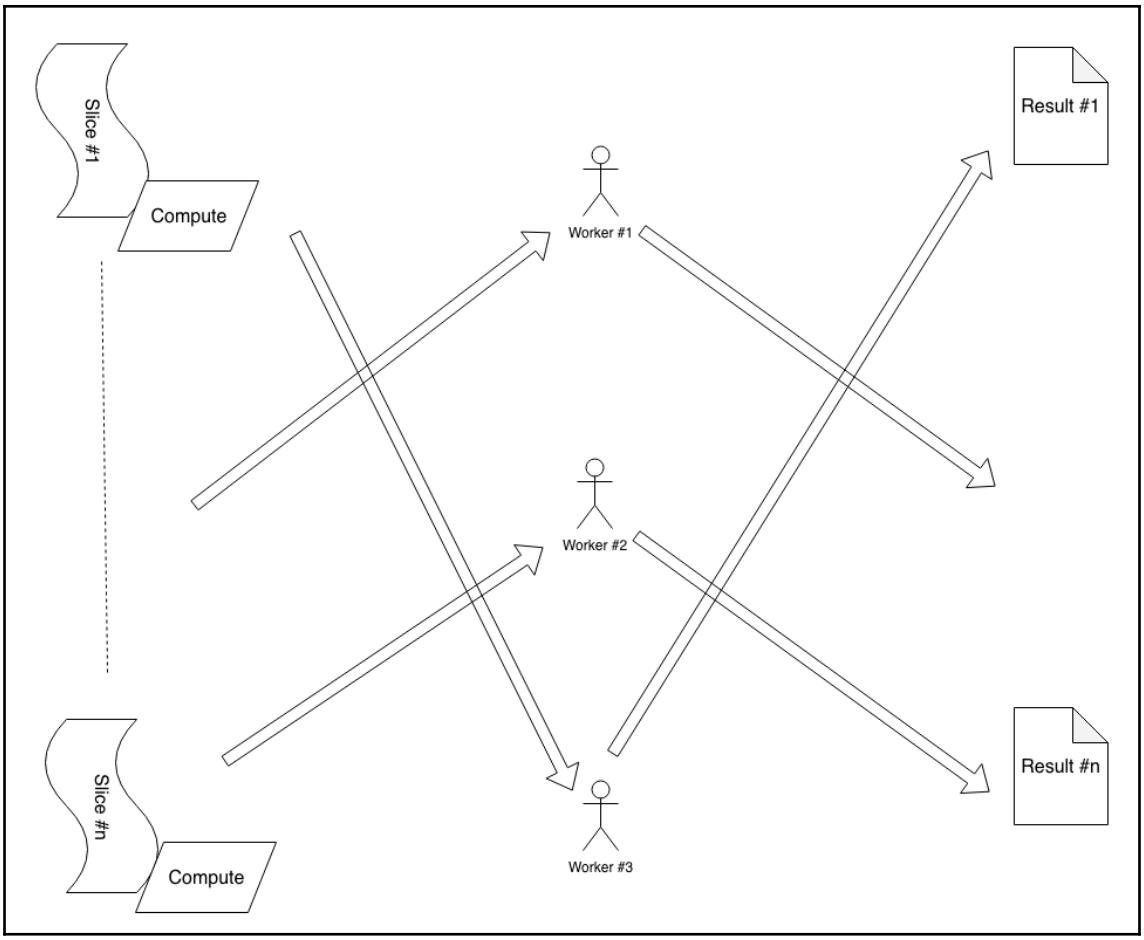
http://apache.mirrors.lucidnetworks.net/spark/spark-2.4.0/spark-2.4.0-bin-hadoop2.7.tgz

http://apache.mirrors.pair.com/spark/spark-2.4.0/spark-2.4.0-bin-hadoop2.7.tgz

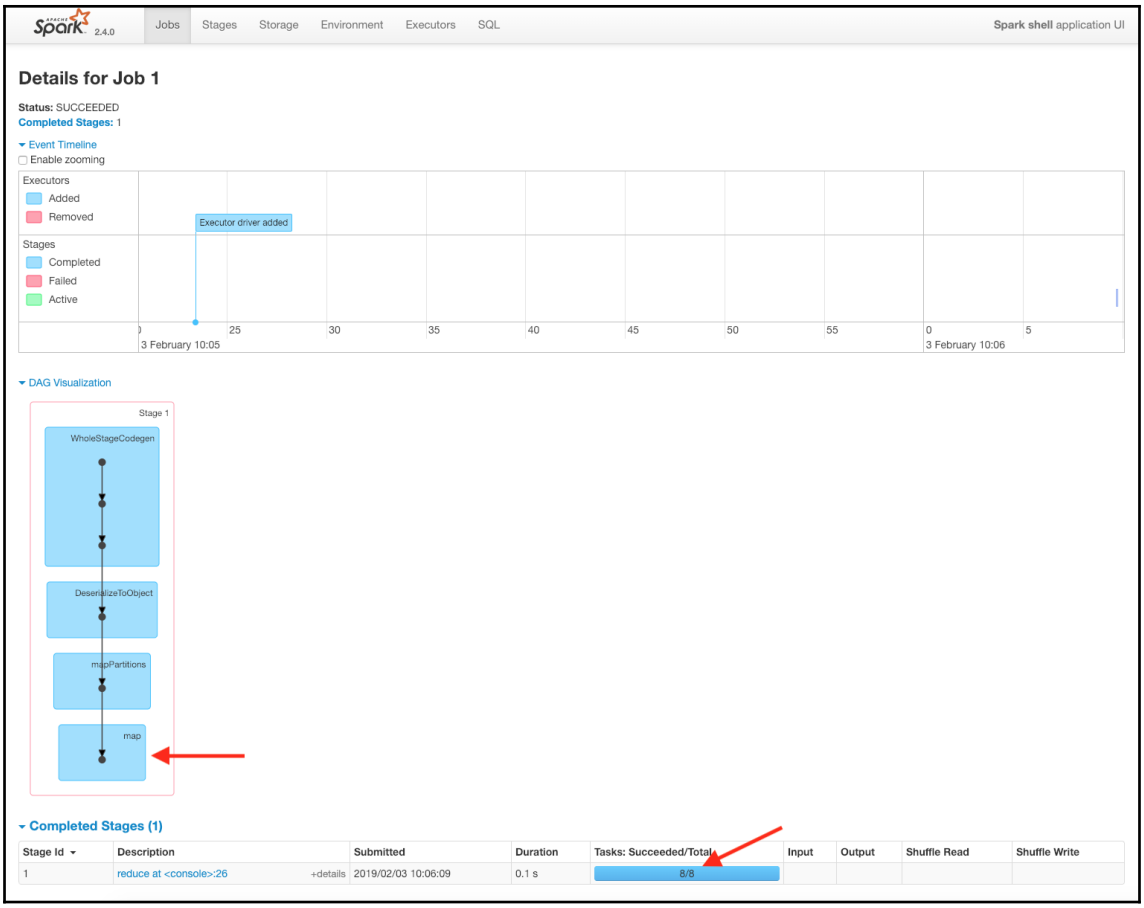
http://apache.mirrors.tds.net/spark/spark-2.4.0/spark-2.4.0-bin-hadoop2.7.tgz

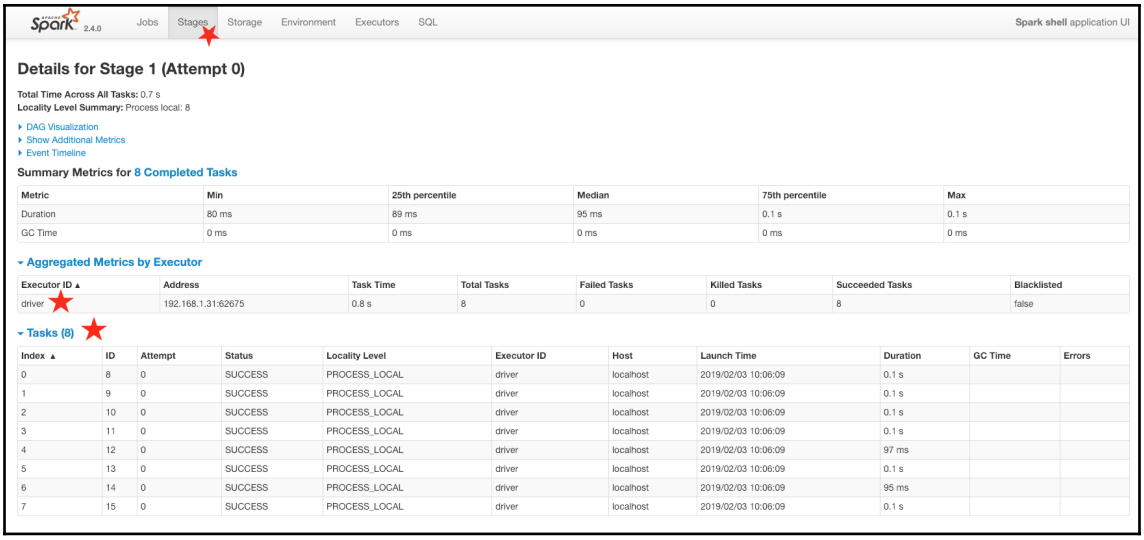



Large Dataset
Slice
Slice #1
Compute
Result #1
Slice #n
Compute
Result #n

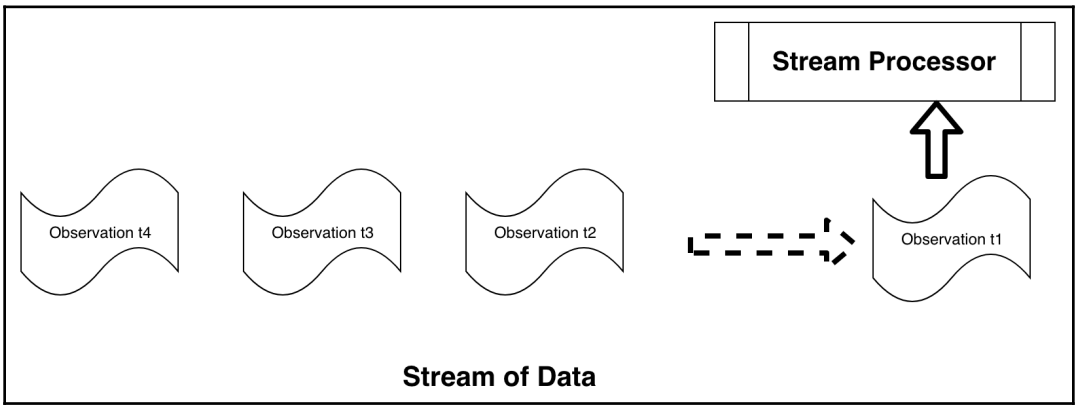

Slice #1
Compute
Worker #1
Result #1
Worker #2
Slice #n
Compute
Worker #3
Result #n

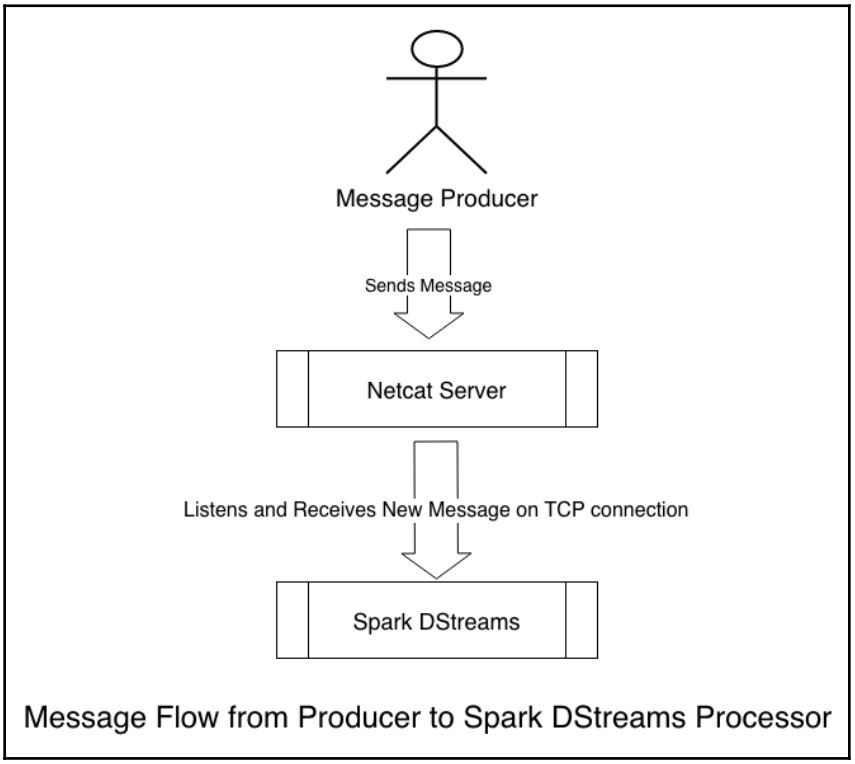

Spark 2.4.0
Jobs
Stages
Storage
Environment
Executors
SQL
Spark shell application UI
Spark Jobs (?)
User: rajeshgupta
Total Uptime: 2.2 min
Scheduling Mode: FIFO
Completed Jobs: 2
Event Timeline
Completed Jobs (2)
numRDD.reduce((a, b) => if (a >= b) a else b)
Job Id
Description
Submitted
Duration
Stages: Succeeded/Total
Tasks (for all stages): Succeeded/Total
1
reduce at <console>:26
reduce at <console>:26
2019/02/03 10:06:09
0.1 s
1/1
8/8
0
foreachPartition at <console>:26
foreachPartition at <console>:26
2019/02/03 10:05:59
0.4 s
1/1
8/8
numRDD.foreachPartition(p => println(p.size))

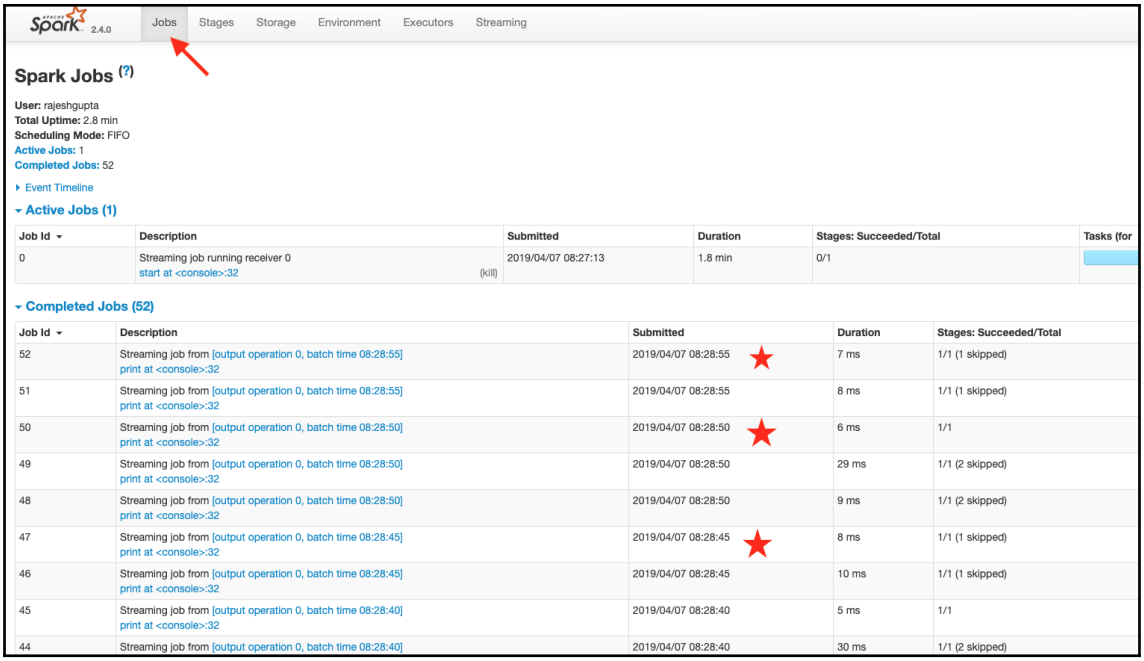

Spark 2.4.0
Jobs
Stages
Storage
Environment
Executors
SQL
Spark shell application UI
Details for Job 1
Status: SUCCEEDED
Completed Stages: 1
Event Timeline
Enable zooming
Executors
Added
Removed
Executor driver added
Stages
Completed
Failed
Active
0
25
30
35
40
45
50
55
3 February 10:05
0
5
3 February 10:06
DAG Visualization
Stage 1
WholeStageCodegen
DeserializeToObject
mapPartitions
map
Completed Stages (1)
Stage Id
Description
Submitted
Duration
Tasks: Succeeded/Total
Input
Output
Shuffle Read
Shuffle Write
1
reduce at <console>:26
+details
2019/02/03 10:06:09
0.1 s
8/8

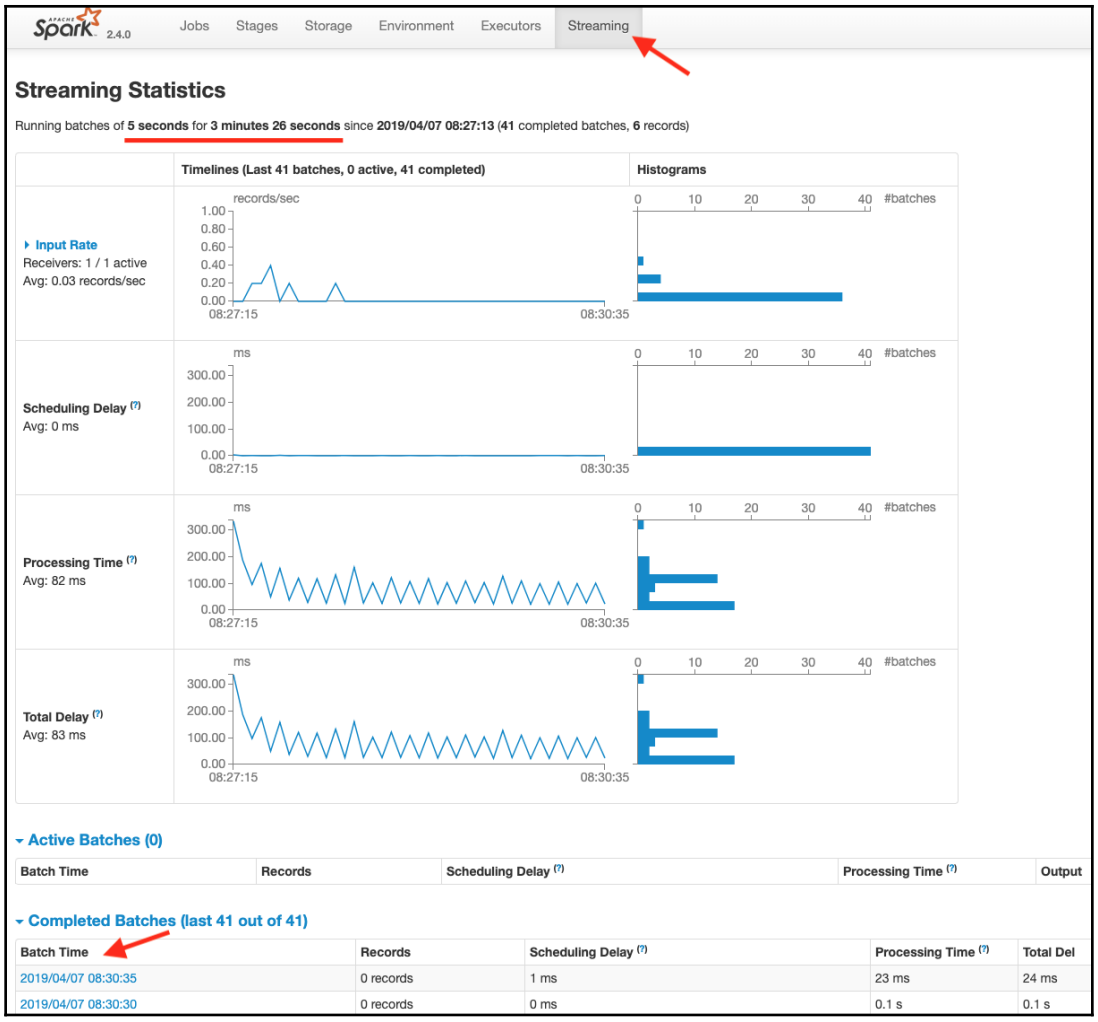

Spark 2.4.0
Jobs
Stages
Storage
Environment
Executors
SQL
Spark shell application UI
Details for Stage 1 (Attempt 0)
Total Time Across All Tasks: 0.7 s
Locality Level Summary: Process local: 8
DAG Visualization
Show Additional Metrics
Event Timeline
Summary Metrics for 8 Completed Tasks
Metric | Min | 25th percentile | Median | 75th percentile | Max
Duration | 80 ms | 89 ms | 95 ms | 0.1 s | 0.1 s
GC Time | 0 ms | 0 ms | 0 ms | 0 ms | 0 ms
Aggregated Metrics by Executor
Executor ID ▲ | Address | Task Time | Total Tasks | Failed Tasks | Killed Tasks | Succeeded Tasks | Blacklisted
driver | 192.168.1.31:62675 | 0.8 s | 8 | 0 | 0 | 8 | false
Tasks (8)
Index ▲ | ID | Attempt | Status | Locality Level | Executor ID | Host | Launch Time | Duration | GC Time | Errors
0 | 8 | 0 | SUCCESS | PROCESS_LOCAL | driver | localhost | 2019/02/03 10:06:09 | 0.1 s
1 | 9 | 0 | SUCCESS | PROCESS_LOCAL | driver | localhost | 2019/02/03 10:06:09 | 0.1 s
2 | 10 | 0 | SUCCESS | PROCESS_LOCAL | driver | localhost | 2019/02/03 10:06:09 | 0.1 s
3 | 11 | 0 | SUCCESS | PROCESS_LOCAL | driver | localhost | 2019/02/03 10:06:09 | 0.1 s
4 | 12 | 0 | SUCCESS | PROCESS_LOCAL | driver | localhost | 2019/02/03 10:06:09 | 97 ms
5 | 13 | 0 | SUCCESS | PROCESS_LOCAL | driver | localhost | 2019/02/03 10:06:09 | 0.1 s
6 | 14 | 0 | SUCCESS | PROCESS_LOCAL | driver | localhost | 2019/02/03 10:06:09 | 95 ms
7 | 15 | 0 | SUCCESS | PROCESS_LOCAL | driver | localhost | 2019/02/03 10:06:09 | 0.1 s

Chapter 8: Near Real-Time Data Analysis Using Streaming

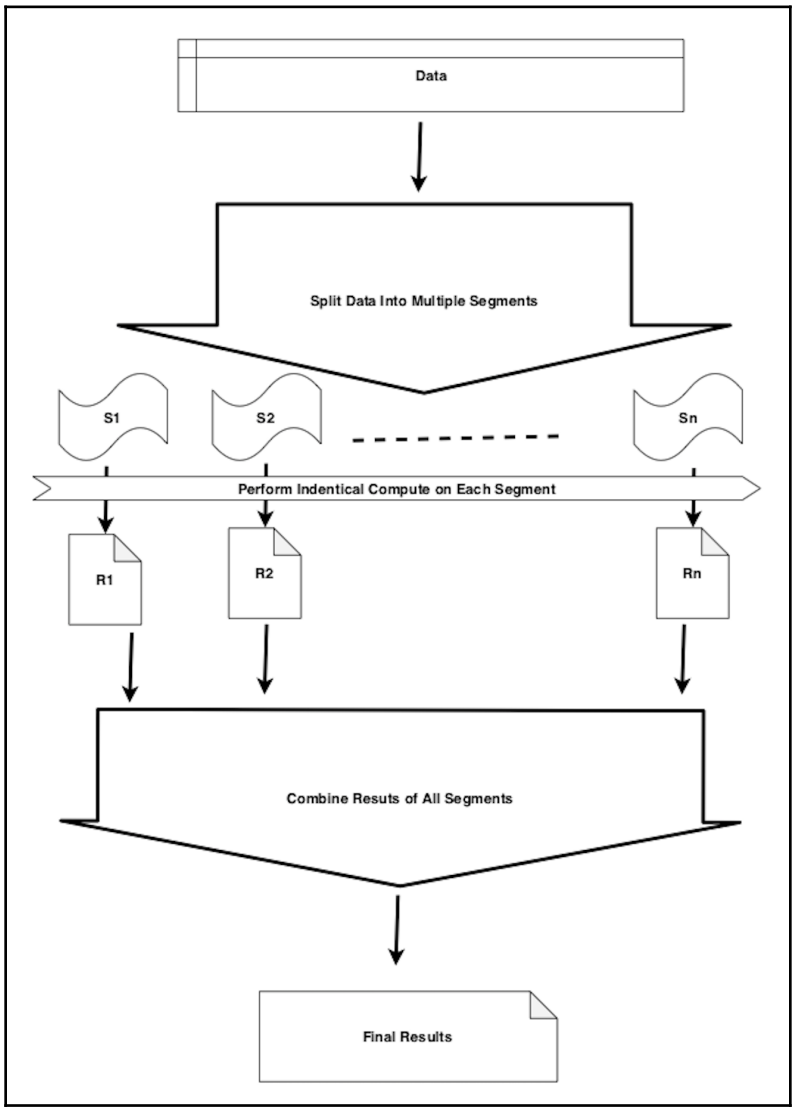


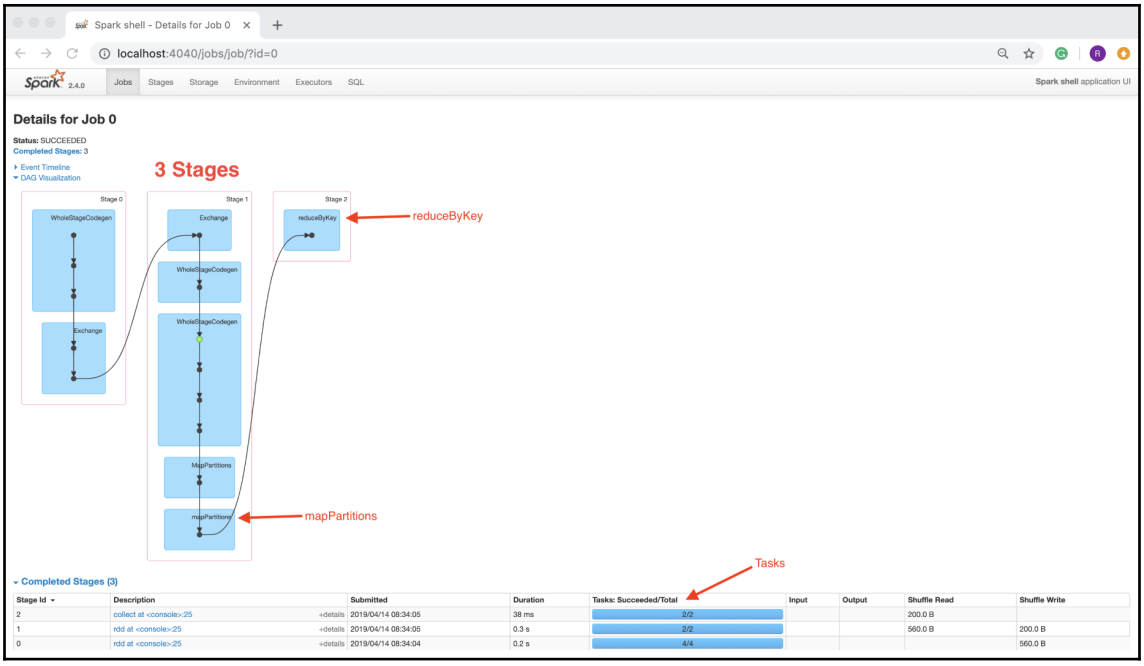


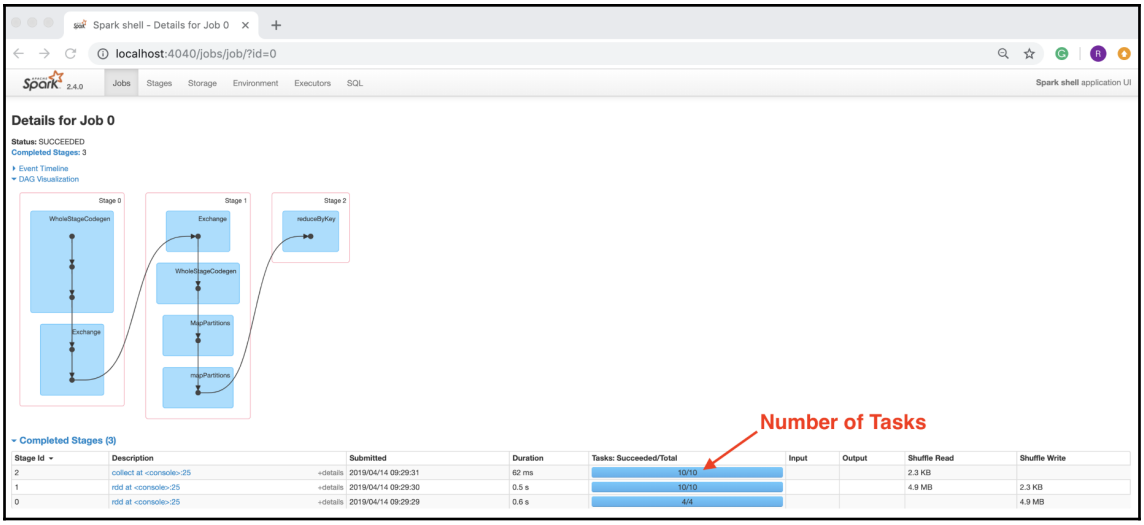

Spark 2.4.0
Jobs
Stages
Storage
Environment
Executors
Streaming
Spark Jobs (?)
User: rajeshgupta
Total Uptime: 2.8 min
Scheduling Mode: FIFO
Active Jobs: 1
Completed Jobs: 52
Event Timeline
Active Jobs (1)
Job Id
Description
Submitted
Duration
Stages: Succeeded/Total
Tasks (for
0
Streaming job running receiver 0
start at <console>:32
(kill)
2019/04/07 08:27:13
1.8 min
0/1
Completed Jobs (52)
Job Id
Description
Submitted
Duration
Stages: Succeeded/Total
52
Streaming job from [output operation 0, batch time 08:28:55]
print at <console>:32
2019/04/07 08:28:55
7 ms
1/1 (1 skipped)
51
Streaming job from [output operation 0, batch time 08:28:55]
print at <console>:32
2019/04/07 08:28:55
8 ms
1/1 (1 skipped)
50
Streaming job from [output operation 0, batch time 08:28:50]
print at <console>:32
2019/04/07 08:28:50
6 ms
1/1
49
Streaming job from [output operation 0, batch time 08:28:50]
print at <console>:32
2019/04/07 08:28:50
29 ms
1/1 (2 skipped)
48
Streaming job from [output operation 0, batch time 08:28:50]
print at <console>:32
2019/04/07 08:28:50
9 ms
1/1 (2 skipped)
47
Streaming job from [output operation 0, batch time 08:28:45]
print at <console>:32
2019/04/07 08:28:45
8 ms
1/1 (1 skipped)
46
Streaming job from [output operation 0, batch time 08:28:45]
print at <console>:32
2019/04/07 08:28:45
10 ms
1/1 (1 skipped)
45
Streaming job from [output operation 0, batch time 08:28:40]
print at <console>:32
2019/04/07 08:28:40
5 ms
1/1
44
Streaming job from [output operation 0, batch time 08:28:40]
2019/04/07 08:28:40
30 ms
1/1 (2 skipped)

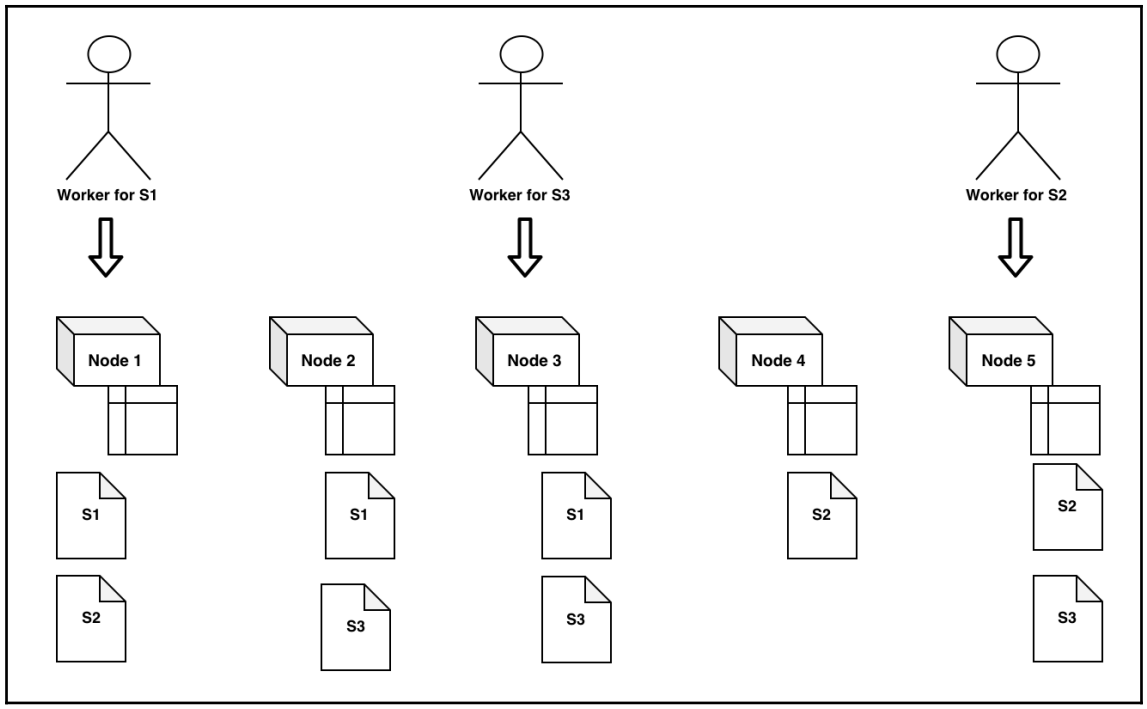

Spark 2.4.0
Jobs
Stages
Storage
Environment
Executors
Streaming
Streaming Statistics
Running batches of 5 seconds for 3 minutes 26 seconds since 2019/04/07 08:27:13 (41 completed batches, 6 records)
Timelines (Last 41 batches, 0 active, 41 completed)
Histograms
Input Rate
Receivers: 1 / 1 active
Avg: 0.03 records/sec
records/sec
#batches
Scheduling Delay (?)
Avg: 0 ms
Processing Time (?)
Avg: 82 ms
Total Delay (?)
Avg: 83 ms
ms
08:27:15
08:30:35
Active Batches (0)
Batch Time
Records
Scheduling Delay (?)
Processing Time (?)
Output
Completed Batches (last 41 out of 41)
Batch Time
Records
Scheduling Delay (?)
Processing Time (?)
Total Del
2019/04/07 08:30:35
0 records
1 ms
23 ms
24 ms
2019/04/07 08:30:30
0 records
0 ms
0.1 s
0.1 s

Chapter 9: Working with Data at Scale




Spark shell - Details for Job 0
localhost:4040/jobs/job/?id=0
Spark 2.4.0
Jobs
Stages
Storage
Environment
Executors
SQL
Spark shell application UI
Details for Job 0
Status: SUCCEEDED
Completed Stages: 3
Event Timeline
DAG Visualization
3 Stages
Stage 0
Stage 1
Stage 2
WholeStageCodegen
Exchange
reduceByKey
MapPartitions
mapPartitions
Tasks
Completed Stages (3)
Stage Id
Description
Submitted
Duration
Tasks: Succeeded/Total
Input
Output
Shuffle Read
Shuffle Write
2 collect at <console>:25 +details 2019/04/14 08:34:05 38 ms 2/2 200.0 B
1 rdd at <console>:25 +details 2019/04/14 08:34:05 0.3 s 2/2 560.0 B 200.0 B
0 rdd at <console>:25 +details 2019/04/14 08:34:04 0.2 s 4/4 560.0 B


Spark shell - Details for Job 0
localhost:4040/jobs/job/?id=0
Spark 2.4.0
Jobs
Stages
Storage
Environment
Executors
SQL
Spark shell application UI
Details for Job 0
Status: SUCCEEDED
Completed Stages: 3
Event Timeline
DAG Visualization
Stage 0
Stage 1
Stage 2
WholeStageCodegen
Exchange
reduceByKey
MapPartitions
mapPartitions
Number of Tasks
Completed Stages (3)
Stage Id
Description
Submitted
Duration
Tasks: Succeeded/Total
Input
Output
Shuffle Read
Shuffle Write
2 collect at <console>:25 +details 2019/04/14 09:29:31 62 ms 10/10 2.3 KB
1 rdd at <console>:25 +details 2019/04/14 09:29:30 0.5 s 10/10 4.9 MB 2.3 KB
0 rdd at <console>:25 +details 2019/04/14 09:29:29 0.6 s 4/4 4.9 MB


Worker for S1
Worker for S3
Worker for S2
Node 1
Node 2
Node 3
Node 4
Node 5
S1
S1
S1
S2
S2
S2
S3
S3
S3