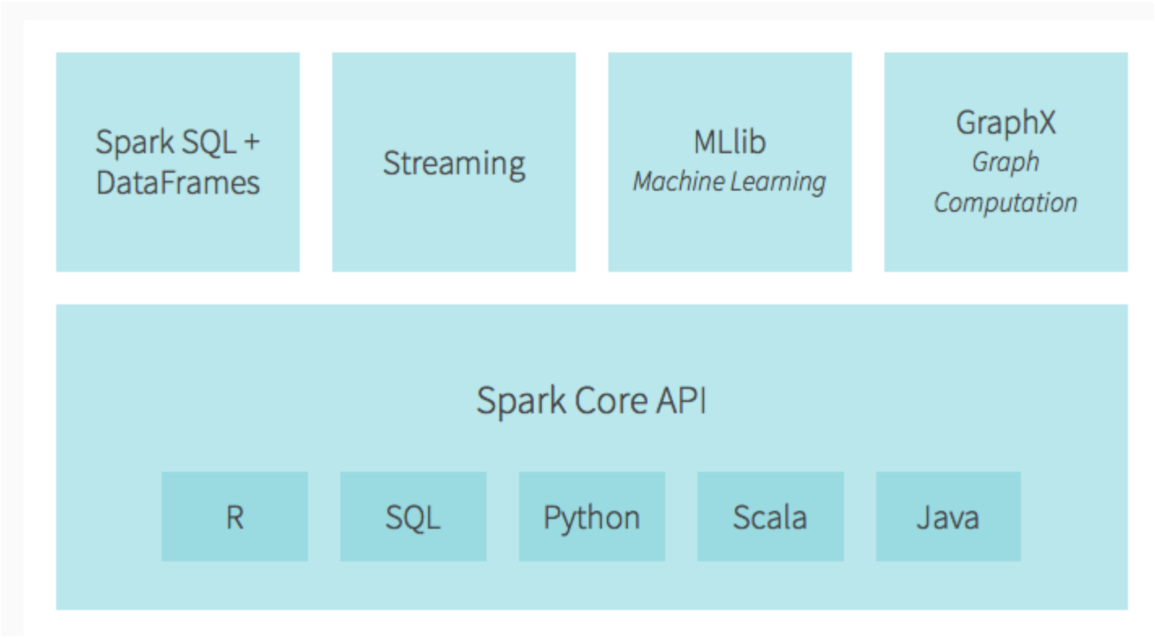
Apache Spark (http://spark.apache.org)

Spark Overview



Spark Core

- Basic functionality of Spark
  - Task scheduling, memory management, fault recovery, interacting with different kinds of storage systems, etc.
  - Provides the RDD (Resilient Distributed Datasets) abstraction
- RDD
  - Represents a collection of items distributed across many compute nodes that can be operated in parallel
  - APIs in various programming languages

Spark SQL

- Package for working with structured data
- Querying data via SQL/HQL (Hive Query Language)
- Supports various data sources (Parquet, JSON, etc.)

Spark DataFrame

- Distributed collection of data organized into named columns
- Provides various operations to filter, group, aggregate, etc.

Spark Streaming

- Component for processing live streams of data
- API for manipulating data streams closely matches the core RDD API

Spark MLib

- Library providing common machine learning functionality
- Classification, regression, clustering, etc.

Spark GraphX

- Library for manipulating graphs and graph-parallel computations
- API extends the Spark RDD API, enables a directed graph to be created with properties attached to each vertex and edge
- Provides common graph algorithms like PageRank, etc.

Installing Spark (MAC OSX)

Prereq: Java8 is required.

Website: http://spark.apache.org/downloads.html

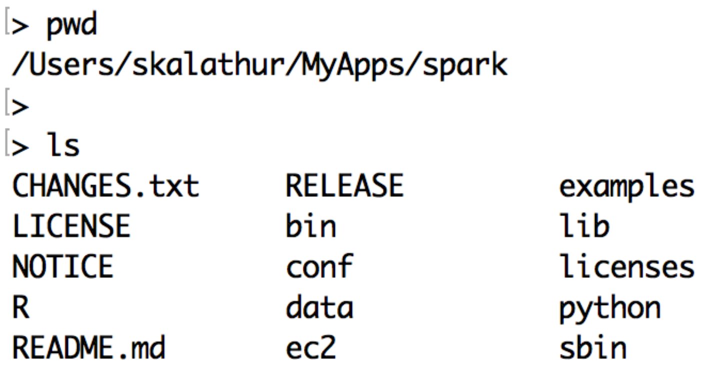
Please select Spark release **Version 2.2.0.**

Package type: Pre-built for Apache Hadoop 2.7 and later

Download and extract the contents (Use a folder with no spaces)
Rename the `spark-2.2.0-bin-hadoop2.7.tar` folder as `spark`

For example, my installation folder now will be

```
/Users/skalathur/MyApps/spark
```



```
> pwd
/Users/skalathur/MyApps/spark
>
> ls
CHANGES.txt     RELEASE         examples
LICENSE         bin             lib
NOTICE          conf            licenses
R               data            python
README.md       ec2             sbin
```

Environment Variables (MAC OSX)

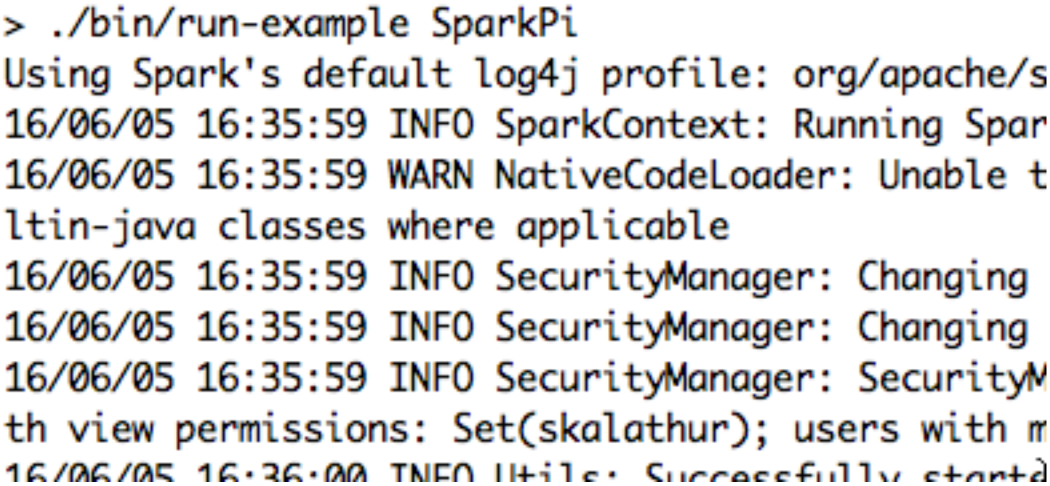
The following environment variables are to be set for MAC.

```
export SPARK_HOME=/Users/skalathur/MyApps/spark
export SPARK_LOCAL_IP=localhost
```

To test the installation, run the following command from the SPARK HOME directory.

```
./bin/run-example SparkPi
```

The sample output is like as shown below.



```
> ./bin/run-example SparkPi
Using Spark's default log4j profile: org/apache/s
16/06/05 16:35:59 INFO SparkContext: Running Spar
16/06/05 16:35:59 WARN NativeCodeLoader: Unable t
ltin-java classes where applicable
16/06/05 16:35:59 INFO SecurityManager: Changing
16/06/05 16:35:59 INFO SecurityManager: Changing
16/06/05 16:35:59 INFO SecurityManager: SecurityM
th view permissions: Set(skalathur); users with m
16/06/05 16:36:00 INFO Utils: Successfully startd
```

In order to reduce the INFO and WARN log messages to the output, do the following:

Navigate to the ***conf*** folder under SPARK HOME.
Copy the log4j.properties.template file to log4j.properties file.

```
cp log4j.properties.template log4j.properties
```

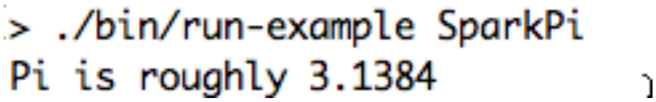
Change the following line in the log4j.properties file

```
log4j.rootCategory=INFO, console
```

to the following:

```
log4j.rootCategory=ERROR, console
```

Run the previous example to see the following output.

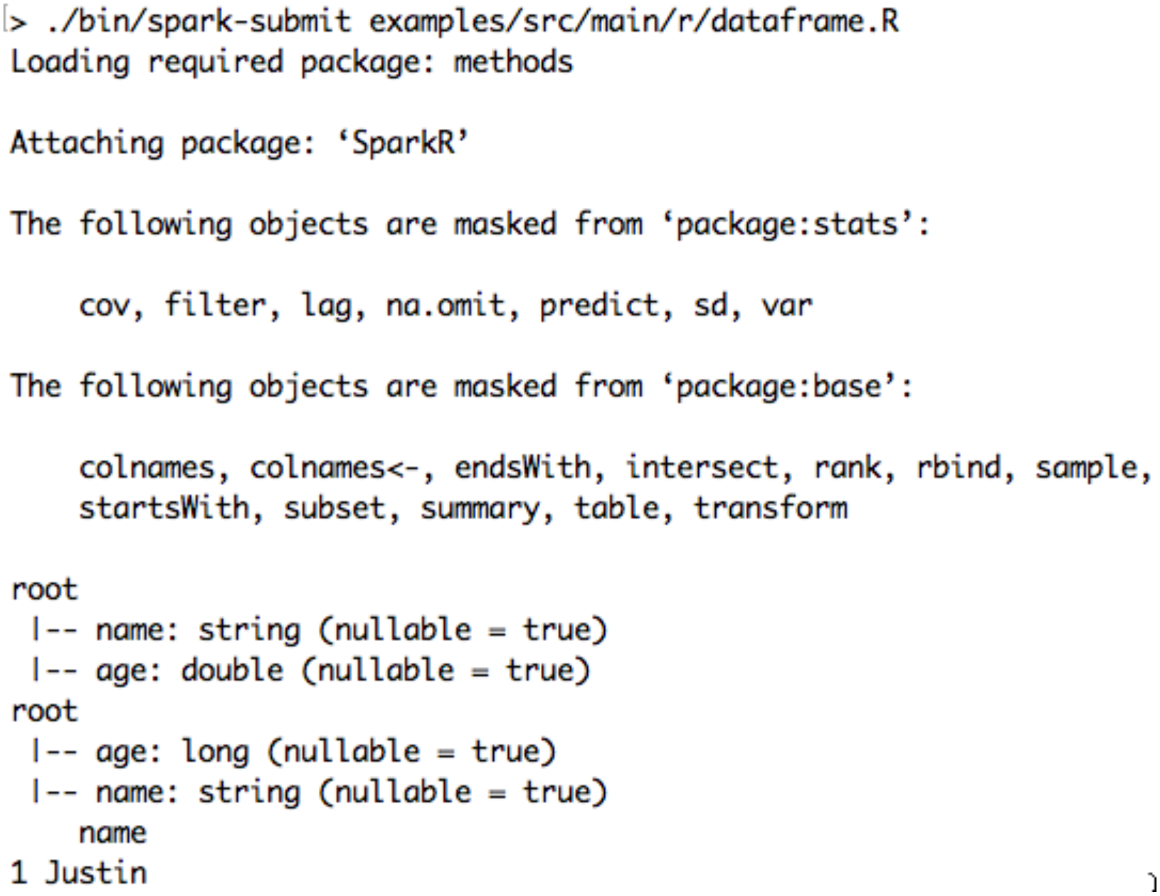


```
> ./bin/run-example SparkPi
Pi is roughly 3.1384
```

Test the Spark and R configuration using the following.

```
./bin/spark-submit examples/src/main/r/dataframe.R
```

The output appears as shown below.



```
> ./bin/spark-submit examples/src/main/r/dataframe.R
Loading required package: methods

Attaching package: ‘SparkR’

The following objects are masked from ‘package:stats’:

    cov, filter, lag, na.omit, predict, sd, var

The following objects are masked from ‘package:base’:

    colnames, colnames<-, endsWith, intersect, rank, rbind, sample,
    startsWith, subset, summary, table, transform

root
 |-- name: string (nullable = true)
 |-- age: double (nullable = true)
root
 |-- age: long (nullable = true)
 |-- name: string (nullable = true)
    name
1 Justin
```

Installing Spark (Windows)

Java installation is required (Java8). The environment variable **JAVA_HOME** should point to the Java installation location.

HADOOP

Download a pre-built version of Hadoop from the following link: https://drive.google.com/open?id=0B6ZM44-N3wySc2hZOGhZY3JnV2M

Extract the contents of the zip file and set the environment variable **HADOOP_HOME** to this location.

SPARK (Version 2.2.0)

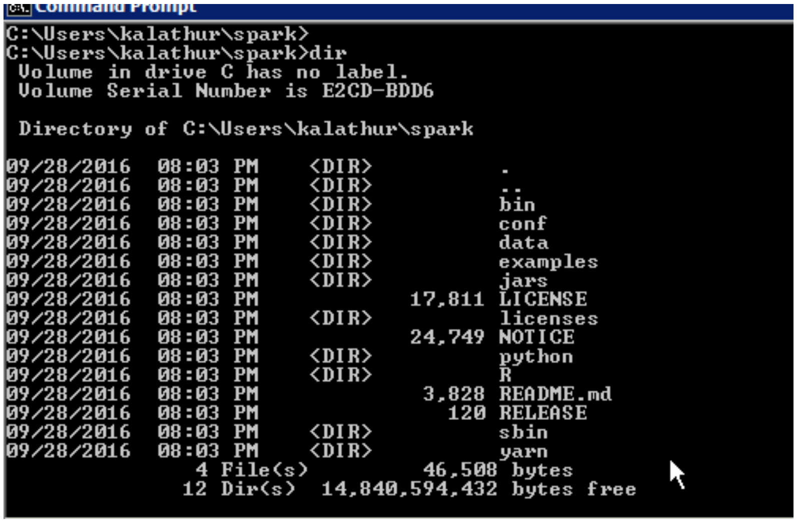
Download Spark from the Website: http://spark.apache.org/downloads.html

Package type: Pre-built for Hadoop 2.7 and later

Download the spark-2.2.0-bin-hadoop2.7.tgz file and extract the contents (You can use 7-zip for this purpose, if you don't have any other tool.)
The extraction above creates `spark-2.2.0-bin-hadoop2.7.tar` file. Extract this tar file one more time (7-zip Extract here… option). This creates the `spark-2.2.0-bin-hadoop2.7 folder.`

Rename the `spark-2.2.0-bin-hadoop2.7` folder as `spark`

For example, my installation folder now will be
`C:\Users\kalathur\spark`



```
C:\Users\kalathur\spark>
C:\Users\kalathur\spark>dir
 Volume in drive C has no label.
 Volume Serial Number is E2CD-BDD6

 Directory of C:\Users\kalathur\spark

09/28/2016  08:03 PM    <DIR>          .
09/28/2016  08:03 PM    <DIR>          ..
09/28/2016  08:03 PM    <DIR>          bin
09/28/2016  08:03 PM    <DIR>          conf
09/28/2016  08:03 PM    <DIR>          data
09/28/2016  08:03 PM    <DIR>          examples
09/28/2016  08:03 PM    <DIR>          jars
09/28/2016  08:03 PM            17,811 LICENSE
09/28/2016  08:03 PM    <DIR>          licenses
09/28/2016  08:03 PM            24,749 NOTICE
09/28/2016  08:03 PM    <DIR>          python
09/28/2016  08:03 PM    <DIR>          R
09/28/2016  08:03 PM             3,828 README.md
09/28/2016  08:03 PM               120 RELEASE
09/28/2016  08:03 PM    <DIR>          sbin
09/28/2016  08:03 PM    <DIR>          yarn
               4 File(s)         46,508 bytes
              12 Dir(s)  14,840,594,432 bytes free
```

Environment Variables (WINDOWS)

The following environment variables are to be set for WINDOWS.

```
SPARK_HOME=C:\Users\kalathur\spark
SPARK_LOCAL_IP=localhost
```

To test the installation, run the following command from the SPARK HOME directory.

```
bin\run-example SparkPi
```

In order to reduce the INFO and WARN log messages to the output, do the following:

Navigate to the ***conf*** folder under SPARK HOME.
Copy the log4j.properties.template file to log4j.properties file.

```
copy log4j.properties.template log4j.properties
```

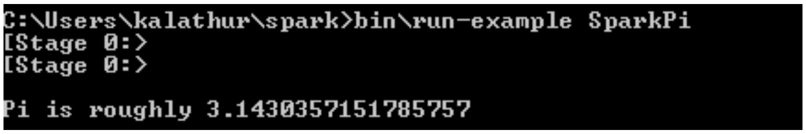
Change the following line in the log4j.properties file (Use Wordpad)

```
log4j.rootCategory=INFO, console
```

to the following:

```
log4j.rootCategory=ERROR, console
```

Run the previous example to see the following output. You may see some error messages after the following output. You can ignore.



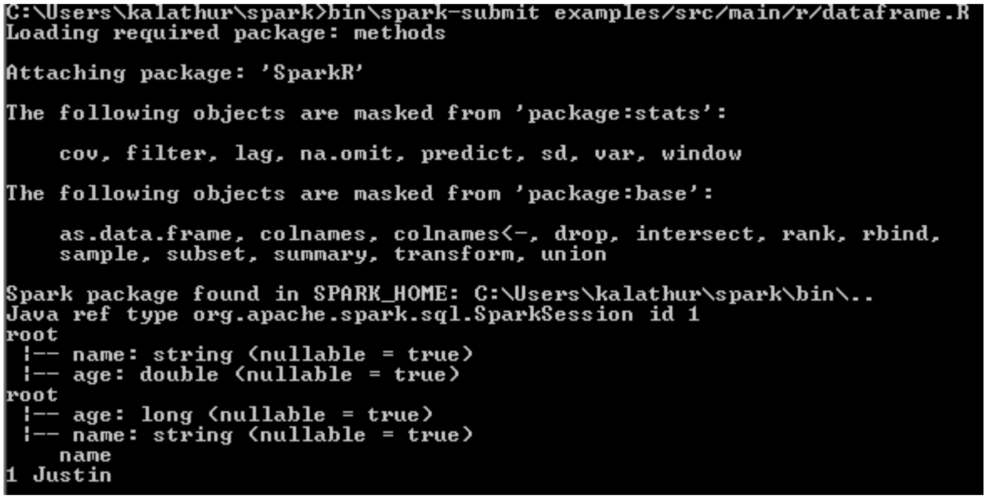
```
C:\Users\kalathur\spark>bin\run-example SparkPi
[Stage 0:>
[Stage 0:>

Pi is roughly 3.1430357151785757
```

Test the Spark and R configuration using the following.

```
bin\spark-submit examples/src/main/r/dataframe.R
```

The output appears as shown below.



```
C:\Users\kalathur\spark>bin\spark-submit examples/src/main/r/dataframe.R
Loading required package: methods

Attaching package: 'SparkR'

The following objects are masked from 'package:stats':

    cov, filter, lag, na.omit, predict, sd, var, window

The following objects are masked from 'package:base':

    as.data.frame, colnames, colnames<-, drop, intersect, rank, rbind,
    sample, subset, summary, transform, union

Spark package found in SPARK_HOME: C:\Users\kalathur\spark\bin\..
Java ref type org.apache.spark.sql.SparkSession id 1
root
 |-- name: string (nullable = true)
 |-- age: double (nullable = true)
root
 |-- age: long (nullable = true)
 |-- name: string (nullable = true)
    name
1 Justin
```