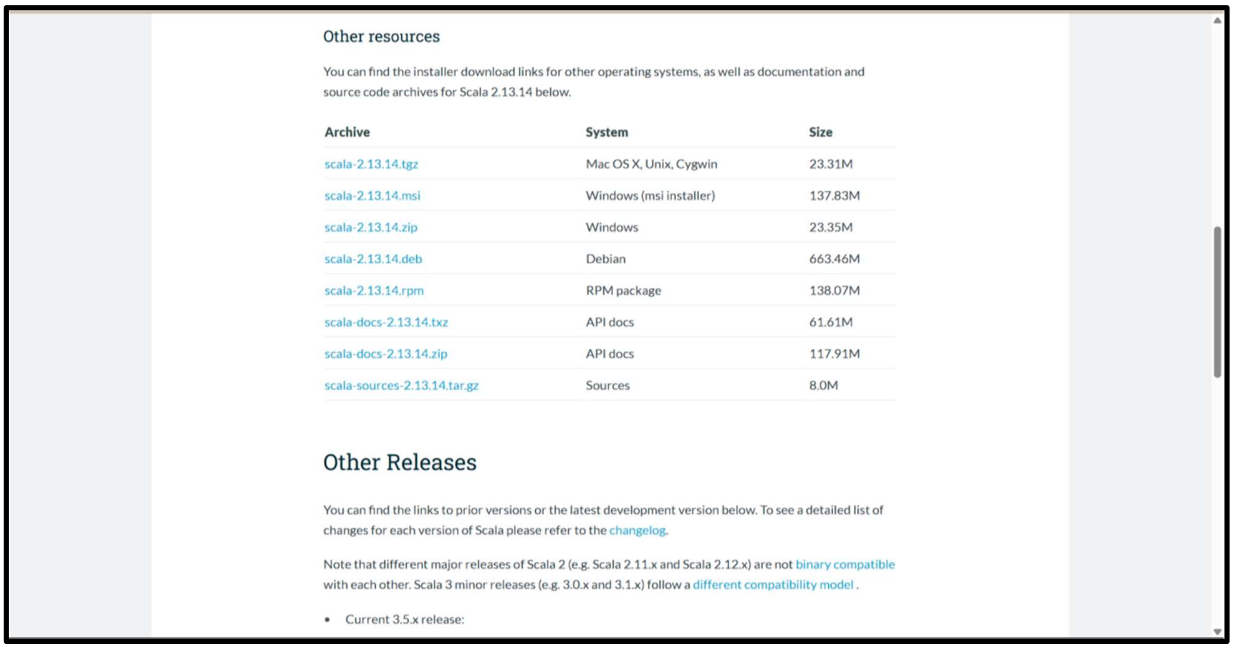
Practical 2: Installation of Scala and Apache Spark

1. Go to this link and download Scala msi installer.



Other resources

You can find the installer download links for other operating systems, as well as documentation and source code archives for Scala 2.13.14 below.

Archive	System	Size
scala-2.13.14.tgz	Mac OS X, Unix, Cygwin	23.31M
scala-2.13.14.msi	Windows (msi installer)	137.83M
scala-2.13.14.zip	Windows	23.35M
scala-2.13.14.deb	Debian	663.46M
scala-2.13.14.rpm	RPM package	138.07M
scala-docs-2.13.14.txz	API docs	61.61M
scala-docs-2.13.14.zip	API docs	117.91M
scala-sources-2.13.14.tar.gz	Sources	8.0M

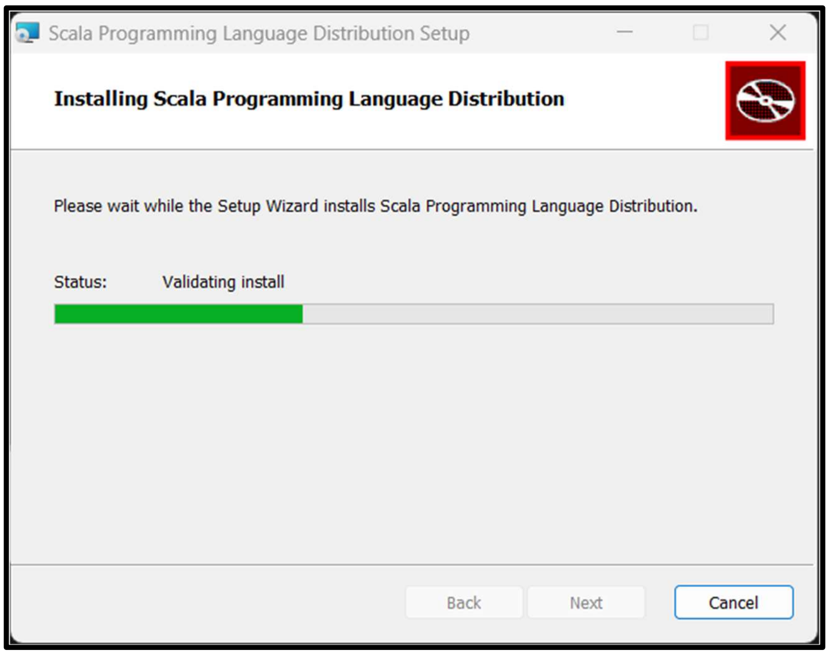
Other Releases

You can find the links to prior versions or the latest development version below. To see a detailed list of changes for each version of Scala please refer to the changelog.

Note that different major releases of Scala 2 (e.g. Scala 2.11.x and Scala 2.12.x) are not binary compatible with each other. Scala 3 minor releases (e.g. 3.0.x and 3.1.x) follow a different compatibility model.

- Current 3.5.x release:

2. Run the installer.



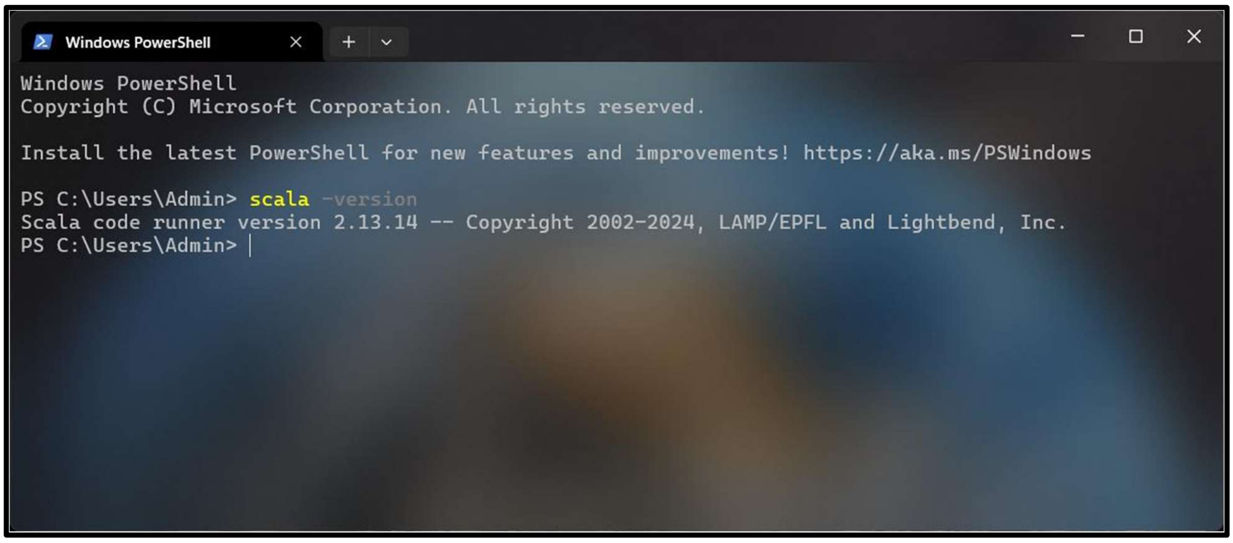
Scala Programming Language Distribution Setup

Installing Scala Programming Language Distribution

Please wait while the Setup Wizard installs Scala Programming Language Distribution.

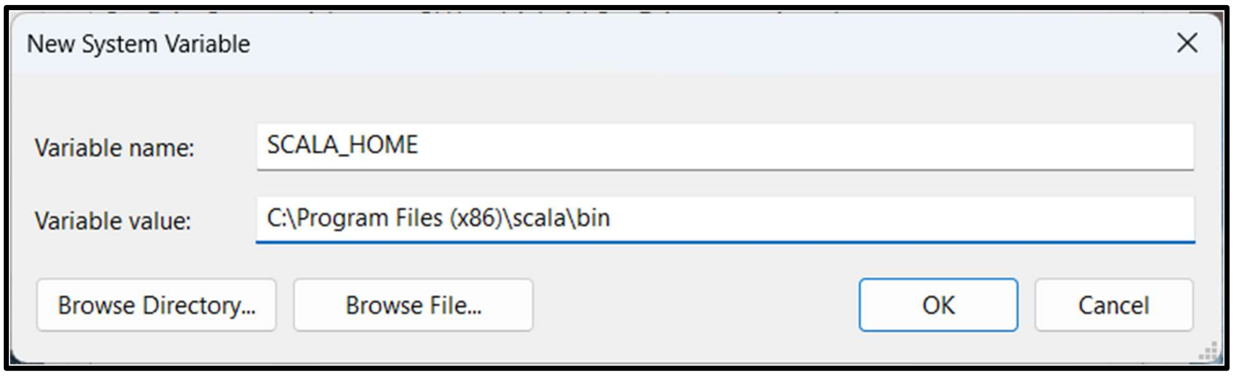
Status: Validating install

3. Using the command scala -version check if scala is installed.



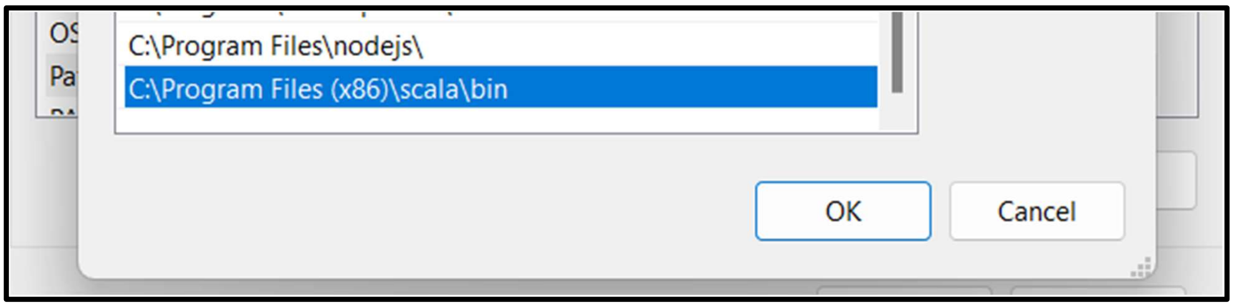
```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows

PS C:\Users\Admin> scala -version
Scala code runner version 2.13.14 -- Copyright 2002-2024, LAMP/EPFL and Lightbend, Inc.
PS C:\Users\Admin> |
```

4. Configure the System Environment Variable.



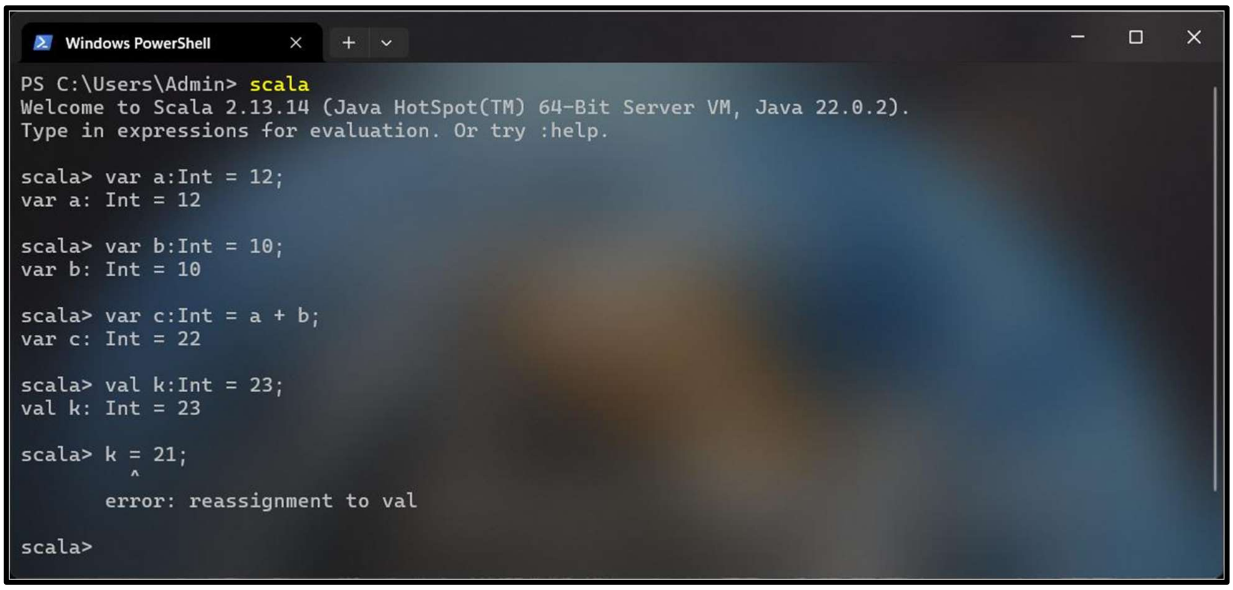
Variable name: SCALA_HOME

Variable value: C:\Program Files (x86)\scala\bin



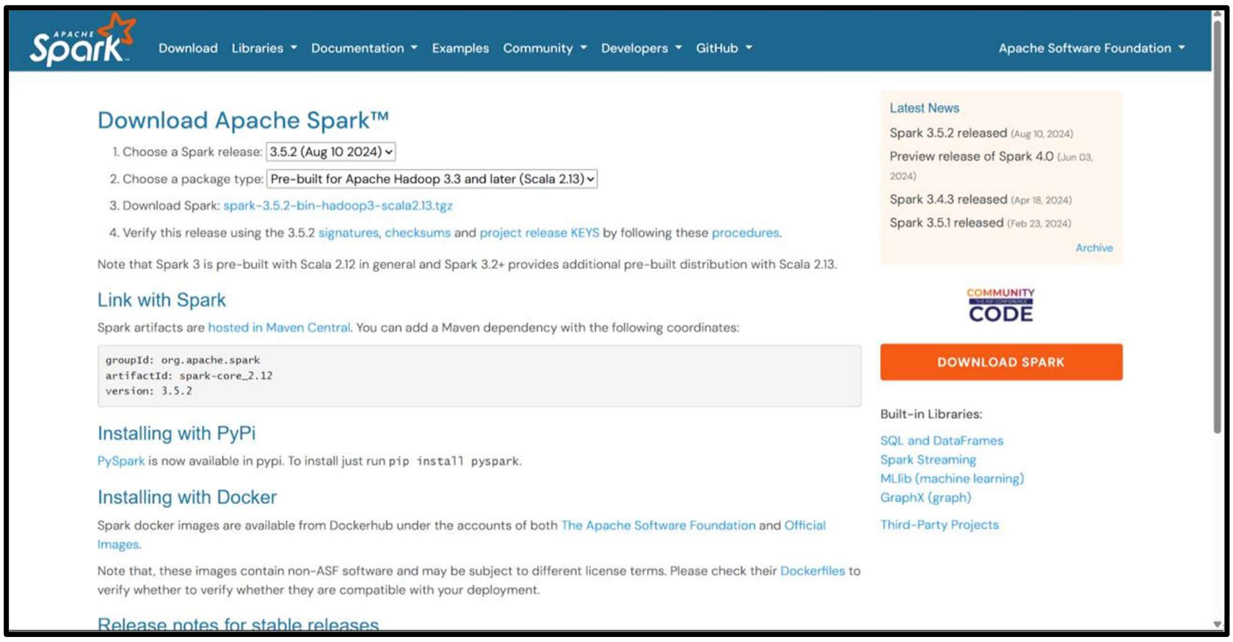
C:\Program Files\nodejs\
C:\Program Files (x86)\scala\bin

5. For mutable variables we use keyword 'var', whereas for immutable variables we use the keyword 'val'. Error will be thrown if values are reassigned to an immutable variable.



```
PS C:\Users\Admin> scala
Welcome to Scala 2.13.14 (Java HotSpot(TM) 64-Bit Server VM, Java 22.0.2).
Type in expressions for evaluation. Or try :help.

scala> var a:Int = 12;
var a: Int = 12

scala> var b:Int = 10;
var b: Int = 10

scala> var c:Int = a + b;
var c: Int = 22

scala> val k:Int = 23;
val k: Int = 23

scala> k = 21;
         ^
       error: reassignment to val

scala>
```

6. Download Apache Spark from here.



Download Apache Spark™

1. Choose a Spark release: 3.5.2 (Aug 10 2024)
2. Choose a package type: Pre-built for Apache Hadoop 3.3 and later (Scala 2.13)
3. Download Spark: spark-3.5.2-bin-hadoop3-scala2.13.tgz
4. Verify this release using the 3.5.2 signatures, checksums and project release KEYS by following these procedures.

Note that Spark 3 is pre-built with Scala 2.12 in general and Spark 3.2+ provides additional pre-built distribution with Scala 2.13.

Link with Spark

Spark artifacts are hosted in Maven Central. You can add a Maven dependency with the following coordinates:

```
groupId: org.apache.spark
artifactId: spark-core_2.12
version: 3.5.2
```

Installing with PyPi

PySpark is now available in pypi. To install just run `pip install pyspark`.

Installing with Docker

Spark docker images are available from Dockerhub under the accounts of both The Apache Software Foundation and Official Images.

Note that, these images contain non-ASF software and may be subject to different license terms. Please check their Dockerfiles to verify whether to verify whether they are compatible with your deployment.

Release notes for stable releases

Latest News

Spark 3.5.2 released (Aug 10, 2024)

Preview release of Spark 4.0 (Jun 03, 2024)

Spark 3.4.3 released (Apr 18, 2024)

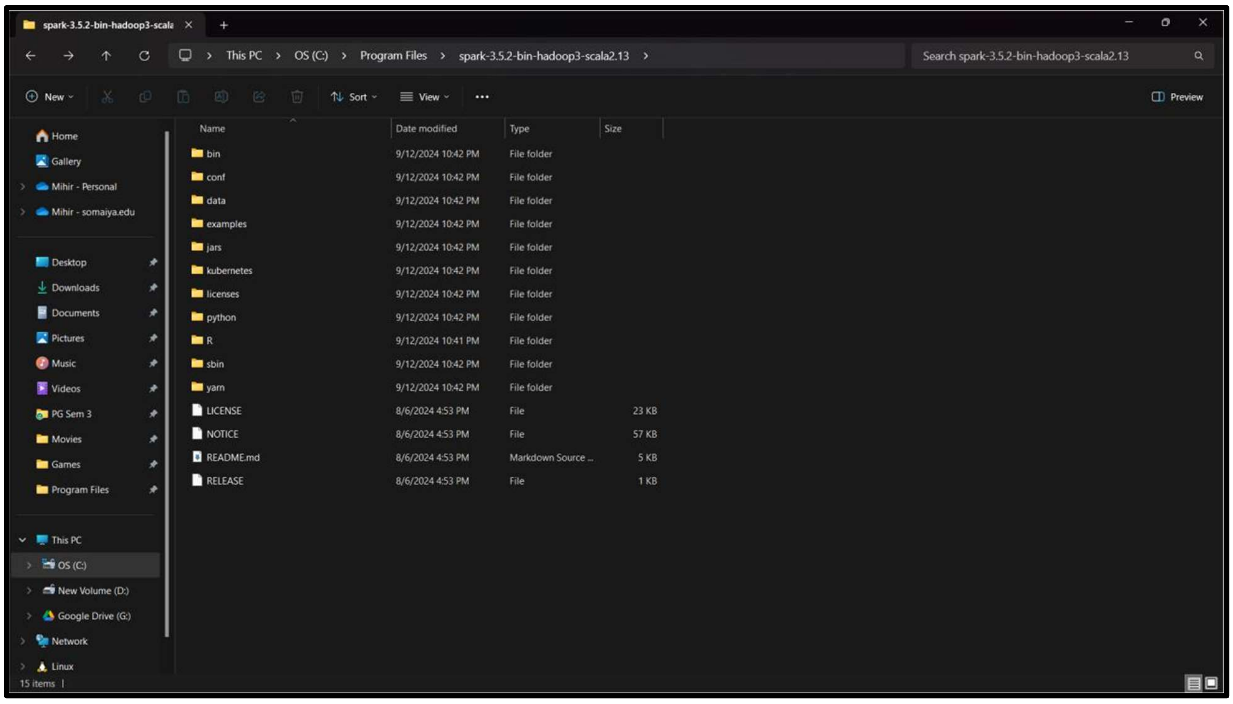
Spark 3.5.1 released (Feb 23, 2024)

Archive

Built-in Libraries:

SQL and DataFrames
Spark Streaming
MLlib (machine learning)
GraphX (graph)
Third-Party Projects

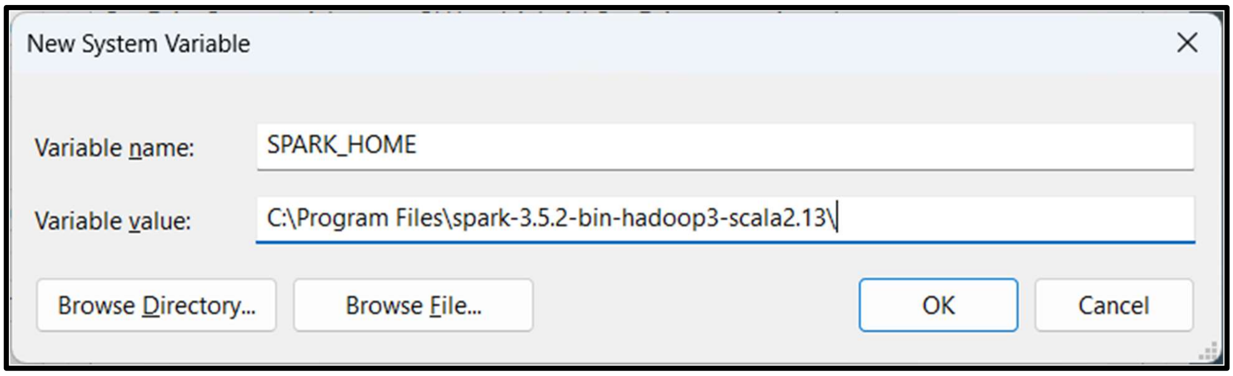
7. Extract the zip and paste it in Program Files.


This PC > OS (C:) > Program Files > spark-3.5.2-bin-hadoop3-scala2.13 >

Name	Date modified	Type	Size
bin	9/12/2024 10:42 PM	File folder	
conf	9/12/2024 10:42 PM	File folder	
data	9/12/2024 10:42 PM	File folder	
examples	9/12/2024 10:42 PM	File folder	
jars	9/12/2024 10:42 PM	File folder	
kubernetes	9/12/2024 10:42 PM	File folder	
licenses	9/12/2024 10:42 PM	File folder	
python	9/12/2024 10:42 PM	File folder	
R	9/12/2024 10:41 PM	File folder	
sbin	9/12/2024 10:42 PM	File folder	
yarn	9/12/2024 10:42 PM	File folder	
LICENSE	8/6/2024 4:53 PM	File	23 KB
NOTICE	8/6/2024 4:53 PM	File	57 KB
README.md	8/6/2024 4:53 PM	Markdown Source ...	5 KB
RELEASE	8/6/2024 4:53 PM	File	1 KB

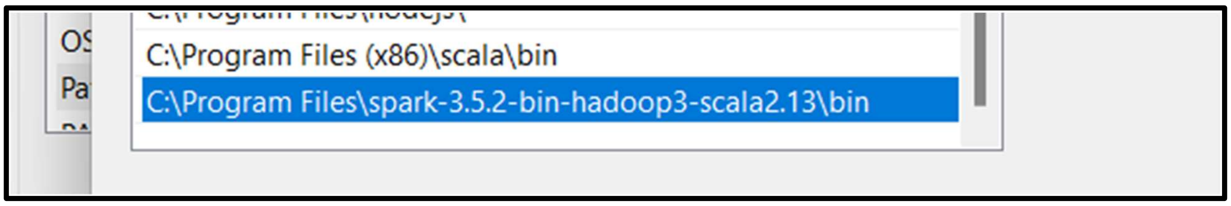
15 items

8. Configure the user and system environment variables.


New System Variable

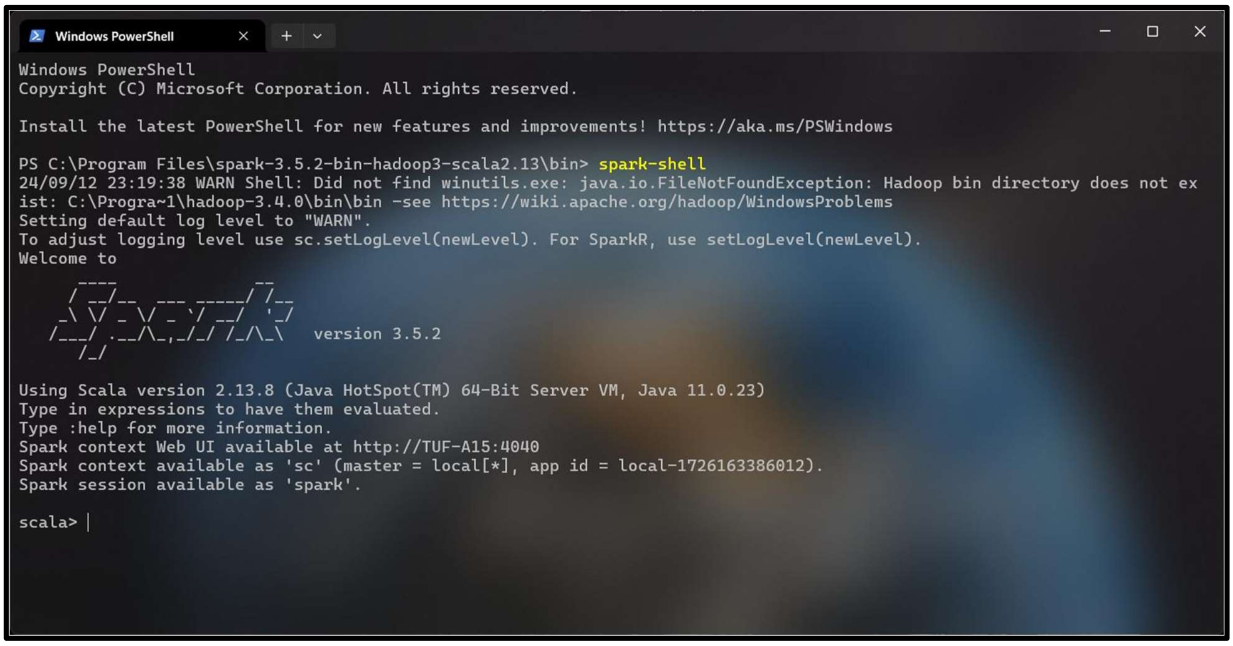
Variable name: SPARK_HOME

Variable value: C:\Program Files\spark-3.5.2-bin-hadoop3-scala2.13\


C:\Program Files (x86)\scala\bin

C:\Program Files\spark-3.5.2-bin-hadoop3-scala2.13\bin

9. Run spark-shell in the bin folder of spark.

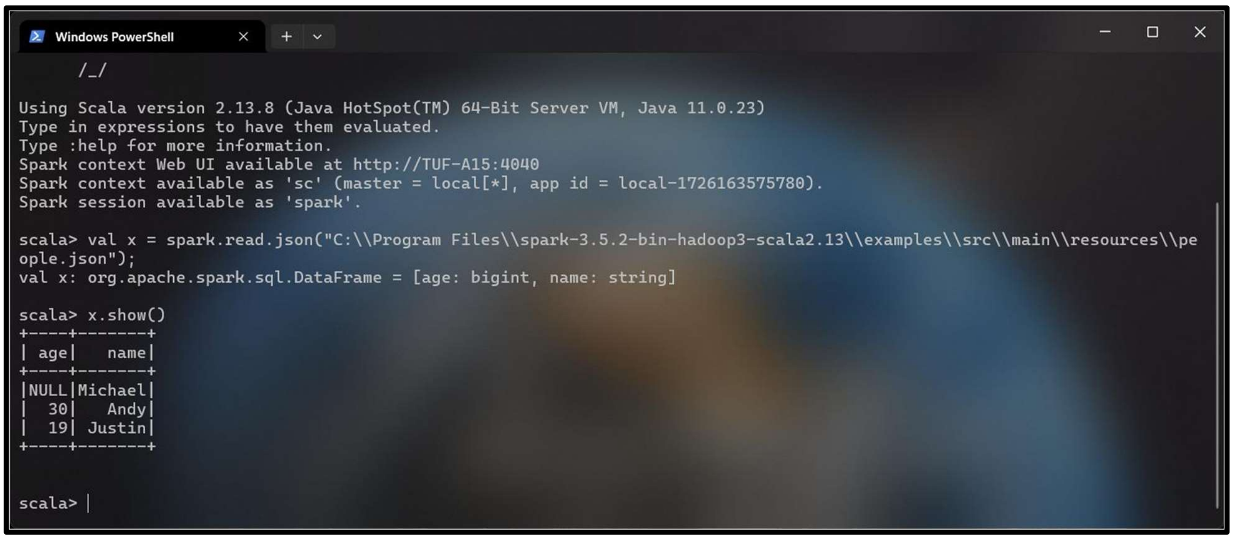


```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows

PS C:\Program Files\spark-3.5.2-bin-hadoop3-scala2.13\bin> spark-shell
24/09/12 23:19:38 WARN Shell: Did not find winutils.exe: java.io.FileNotFoundException: Hadoop bin directory does not ex
ist: C:\Progra~1\hadoop-3.4.0\bin\bin -see https://wiki.apache.org/hadoop/WindowsProblems
Setting default log level to "WARN".
To adjust logging level use sc.setLogLevel(newLevel). For SparkR, use setLogLevel(newLevel).
Welcome to
      ____              __
     / __/__  ___ _____/ /__
    _\ \/ _ \/ _ `/ __/  '_/
   /___/ .__/\_,_/_/ /_/\_\   version 3.5.2
      /_/

Using Scala version 2.13.8 (Java HotSpot(TM) 64-Bit Server VM, Java 11.0.23)
Type in expressions to have them evaluated.
Type :help for more information.
Spark context Web UI available at http://TUF-A15:4040
Spark context available as 'sc' (master = local[*], app id = local-1726163386012).
Spark session available as 'spark'.

scala> |
```

10. Run the following commands

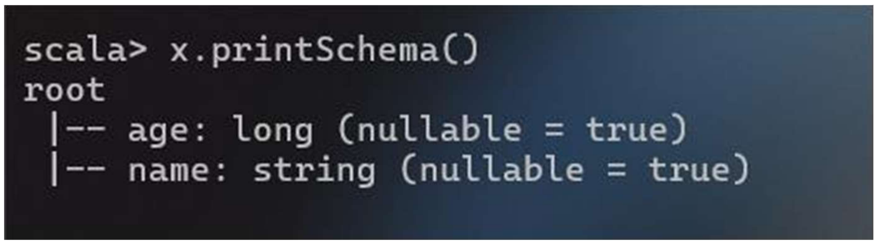
val x = spark.read.json("C:\\Program Files\\spark-3.5.2-bin-hadoop3-scala2.13\\examples\\src\\main\\resources\\people.json");

x.show()



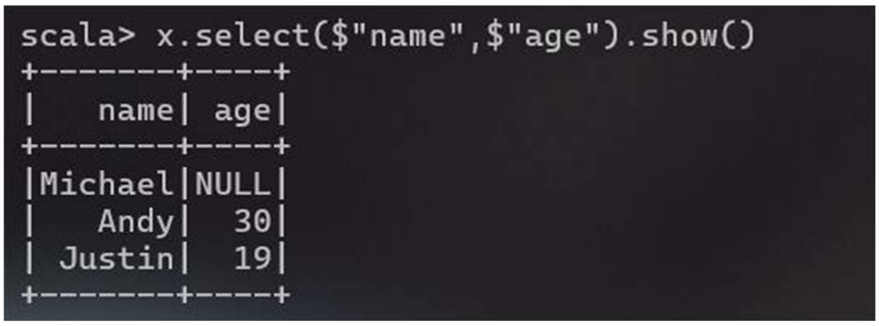
```
      /_/

Using Scala version 2.13.8 (Java HotSpot(TM) 64-Bit Server VM, Java 11.0.23)
Type in expressions to have them evaluated.
Type :help for more information.
Spark context Web UI available at http://TUF-A15:4040
Spark context available as 'sc' (master = local[*], app id = local-1726163575780).
Spark session available as 'spark'.

scala> val x = spark.read.json("C:\\Program Files\\spark-3.5.2-bin-hadoop3-scala2.13\\examples\\src\\main\\resources\\pe
ople.json");
val x: org.apache.spark.sql.DataFrame = [age: bigint, name: string]

scala> x.show()
+----+-------+
| age|   name|
+----+-------+
|NULL|Michael|
|  30|   Andy|
|  19| Justin|
+----+-------+


scala> |
```

x.printSchema()



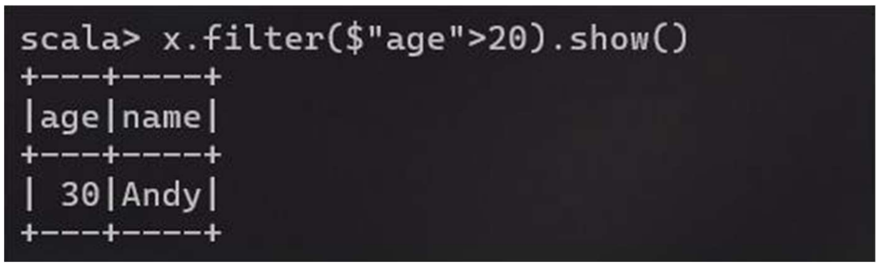
```
scala> x.printSchema()
root
 |-- age: long (nullable = true)
 |-- name: string (nullable = true)
```

x.select($"name",$"age").show()



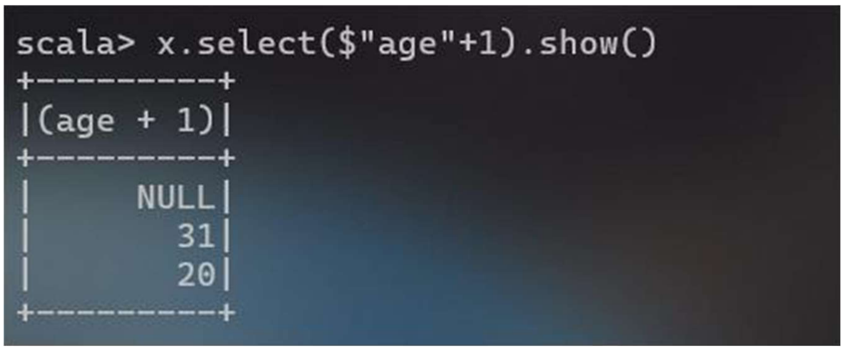
```
scala> x.select($"name",$"age").show()
+-------+----+
|   name| age|
+-------+----+
|Michael|NULL|
|   Andy|  30|
| Justin|  19|
+-------+----+
```

x.filter($"age">20).show()



```
scala> x.filter($"age">20).show()
+---+----+
|age|name|
+---+----+
| 30|Andy|
+---+----+
```

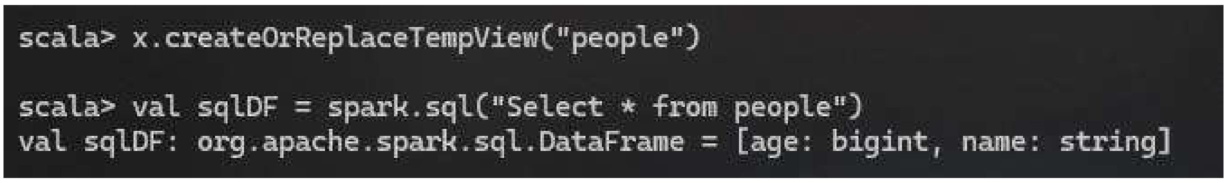
x.select($"age"+1).show()



```
scala> x.select($"age"+1).show()
+---------+
|(age + 1)|
+---------+
|     NULL|
|       31|
|       20|
+---------+
```

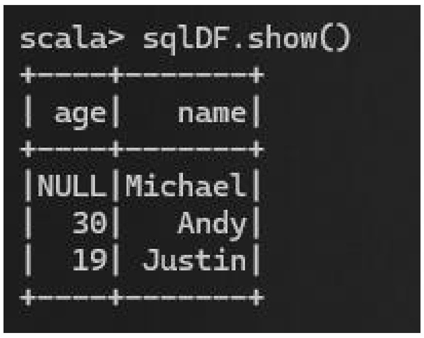
x.createOrReplaceTempView("people")

val sqlDF = spark.sql("Select * from people")



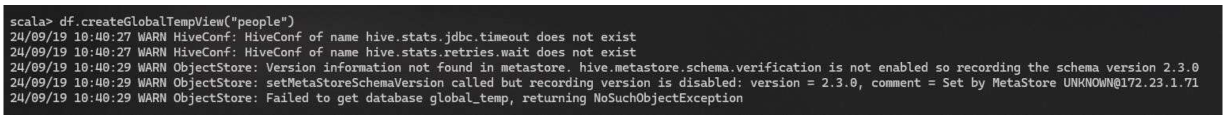
```
scala> x.createOrReplaceTempView("people")

scala> val sqlDF = spark.sql("Select * from people")
val sqlDF: org.apache.spark.sql.DataFrame = [age: bigint, name: string]
```

sqlDF.show()



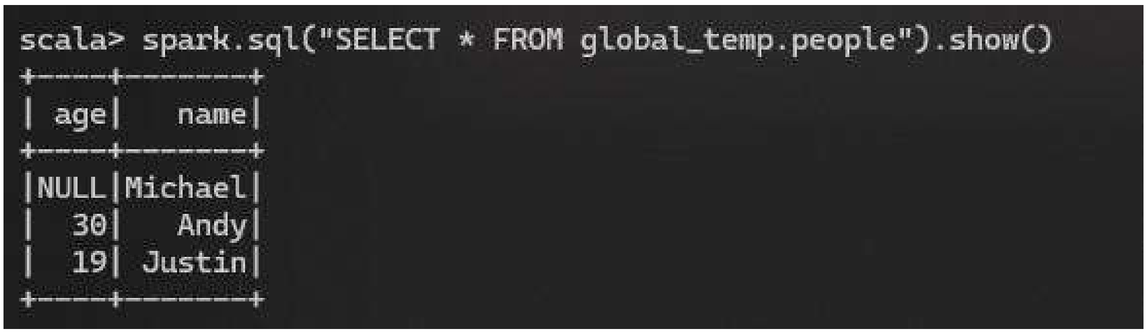
```
scala> sqlDF.show()
+----+-------+
| age|   name|
+----+-------+
|NULL|Michael|
|  30|   Andy|
|  19| Justin|
+----+-------+
```

df.createGlobalTempView("people")



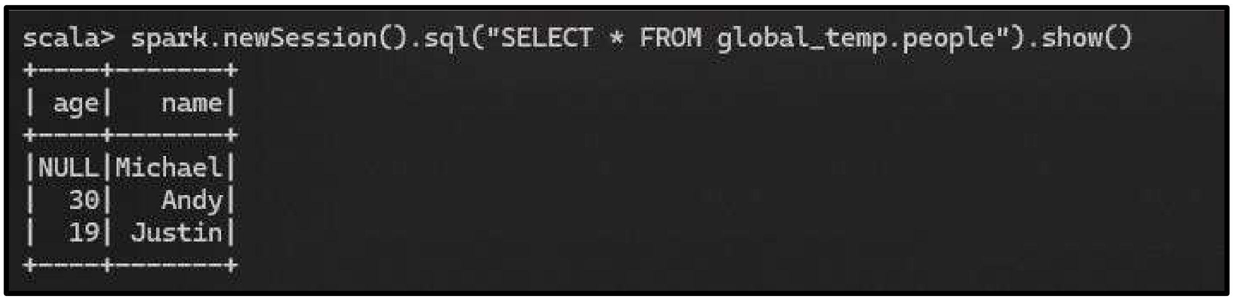
```
scala> df.createGlobalTempView("people")
24/09/19 10:40:27 WARN HiveConf: HiveConf of name hive.stats.jdbc.timeout does not exist
24/09/19 10:40:27 WARN HiveConf: HiveConf of name hive.stats.retries.wait does not exist
24/09/19 10:40:29 WARN ObjectStore: Version information not found in metastore. hive.metastore.schema.verification is not enabled so recording the schema version 2.3.0
24/09/19 10:40:29 WARN ObjectStore: setMetaStoreSchemaVersion called but recording version is disabled: version = 2.3.0, comment = Set by MetaStore UNKNOWN@172.23.1.71
24/09/19 10:40:29 WARN ObjectStore: Failed to get database global_temp, returning NoSuchObjectException
```

spark.sql("SELECT * FROM global_temp.people").show()



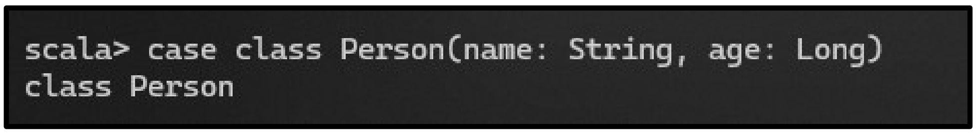
```
scala> spark.sql("SELECT * FROM global_temp.people").show()
+----+-------+
| age|   name|
+----+-------+
|NULL|Michael|
|  30|   Andy|
|  19| Justin|
+----+-------+
```

spark.newSession().sql("SELECT * FROM global_temp.people").show()



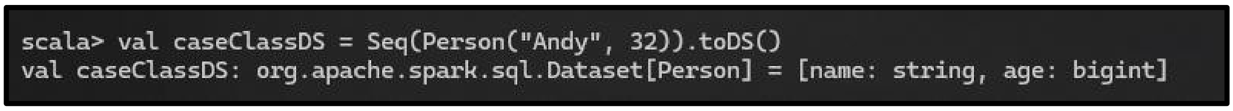
```
scala> spark.newSession().sql("SELECT * FROM global_temp.people").show()
+----+-------+
| age|   name|
+----+-------+
|NULL|Michael|
|  30|   Andy|
|  19| Justin|
+----+-------+
```

case class Person(name: String, age: Long)



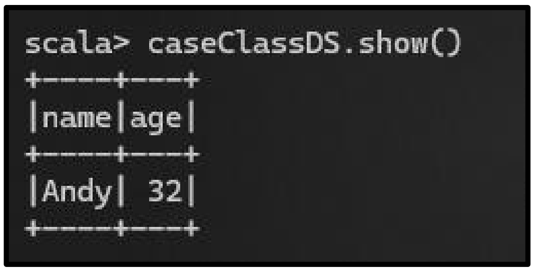
```
scala> case class Person(name: String, age: Long)
class Person
```

val caseClassDS = Seq(Person("Andy", 32)).toDS()



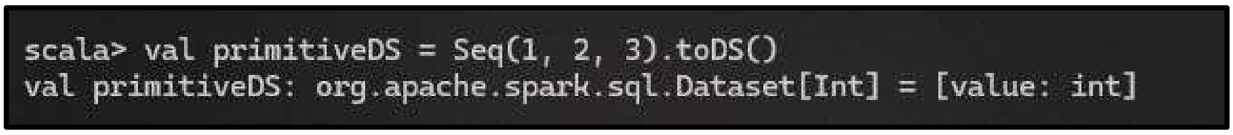
```
scala> val caseClassDS = Seq(Person("Andy", 32)).toDS()
val caseClassDS: org.apache.spark.sql.Dataset[Person] = [name: string, age: bigint]
```

caseClassDS.show()



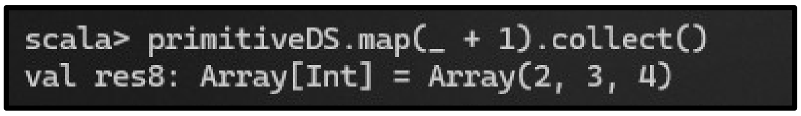
```
scala> caseClassDS.show()
+----+---+
|name|age|
+----+---+
|Andy| 32|
+----+---+
```

val primitiveDS = Seq(1, 2, 3).toDS()



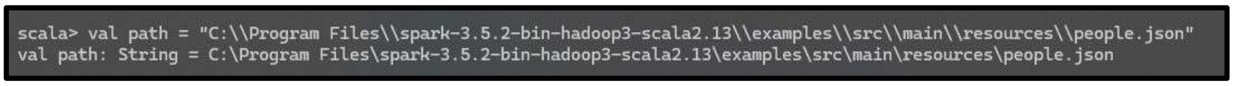
```
scala> val primitiveDS = Seq(1, 2, 3).toDS()
val primitiveDS: org.apache.spark.sql.Dataset[Int] = [value: int]
```

primitiveDS.map(_ + 1).collect()



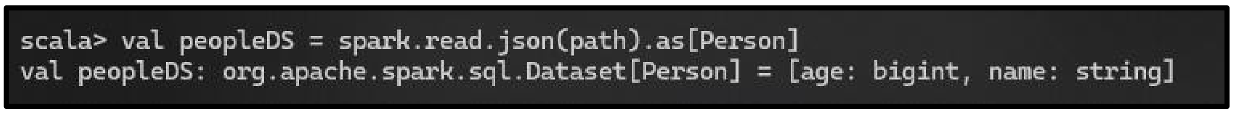
```
scala> primitiveDS.map(_ + 1).collect()
val res8: Array[Int] = Array(2, 3, 4)
```

val path = "C:\\Program Files\\spark-3.5.2-bin-hadoop3-scala2.13\\examples\\src\\main\\resources\\people.json"



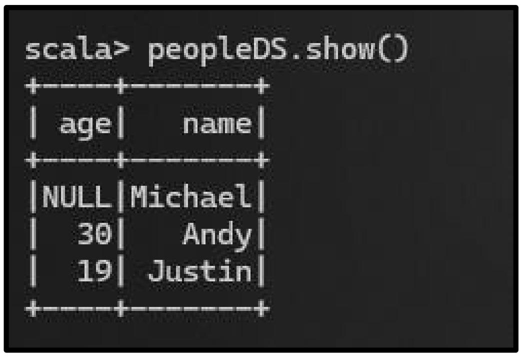
```
scala> val path = "C:\\Program Files\\spark-3.5.2-bin-hadoop3-scala2.13\\examples\\src\\main\\resources\\people.json"
val path: String = C:\Program Files\spark-3.5.2-bin-hadoop3-scala2.13\examples\src\main\resources\people.json
```

val peopleDS = spark.read.json(path).as[Person]

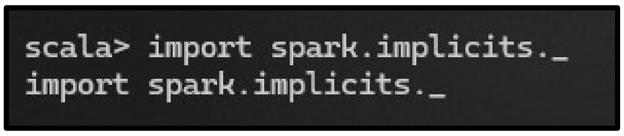


```
scala> val peopleDS = spark.read.json(path).as[Person]
val peopleDS: org.apache.spark.sql.Dataset[Person] = [age: bigint, name: string]
```

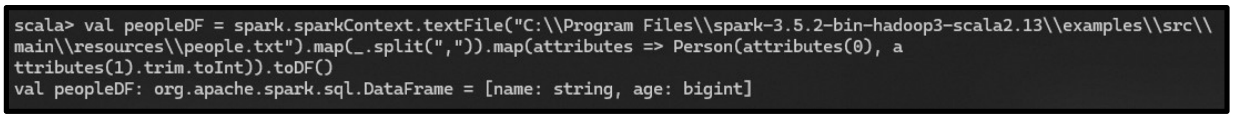
peopleDS.show()


```
scala> peopleDS.show()
+----+-------+
| age|   name|
+----+-------+
|NULL|Michael|
|  30|   Andy|
|  19| Justin|
+----+-------+
```

import spark.implicits._

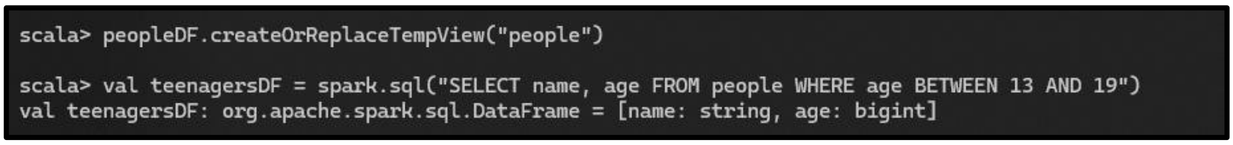

```
scala> import spark.implicits._
import spark.implicits._
```

val peopleDF = spark.sparkContext.textFile("C:\\Program Files\\spark-3.5.2-bin-hadoop3-scala2.13\\examples\\src\\main\\resources\\people.txt").map(_.split(",")).map(attributes => Person(attributes(0), attributes(1).trim.toInt)).toDF()

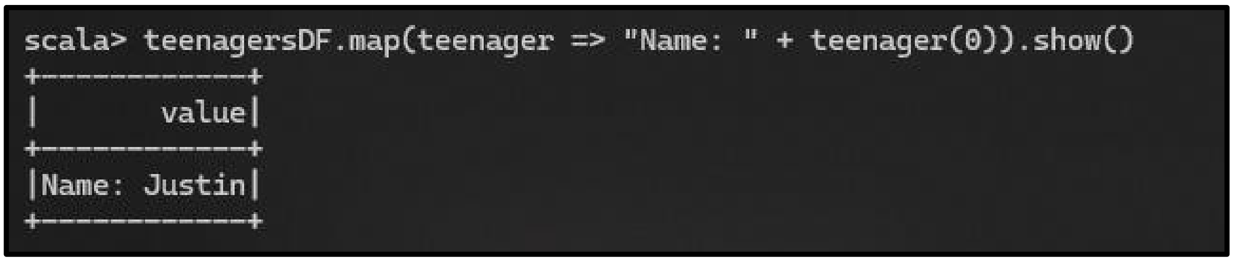

```
scala> val peopleDF = spark.sparkContext.textFile("C:\\Program Files\\spark-3.5.2-bin-hadoop3-scala2.13\\examples\\src\\
main\\resources\\people.txt").map(_.split(",")).map(attributes => Person(attributes(0), a
ttributes(1).trim.toInt)).toDF()
val peopleDF: org.apache.spark.sql.DataFrame = [name: string, age: bigint]
```

peopleDF.createOrReplaceTempView("people")

val teenagersDF = spark.sql("SELECT name, age FROM people WHERE age BETWEEN 13 AND 19")

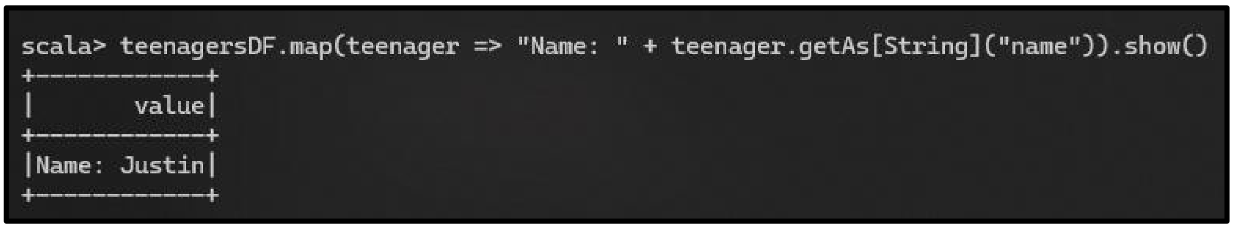

```
scala> peopleDF.createOrReplaceTempView("people")

scala> val teenagersDF = spark.sql("SELECT name, age FROM people WHERE age BETWEEN 13 AND 19")
val teenagersDF: org.apache.spark.sql.DataFrame = [name: string, age: bigint]
```

teenagersDF.map(teenager => "Name: " + teenager(0)).show()



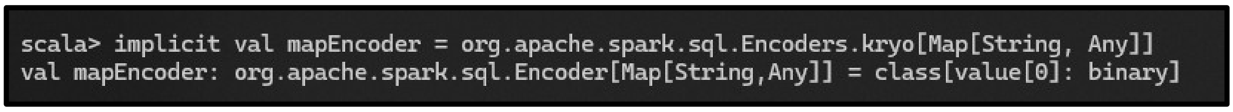
```
scala> teenagersDF.map(teenager => "Name: " + teenager(0)).show()
+------------+
|       value|
+------------+
|Name: Justin|
+------------+
```

teenagersDF.map(teenager => "Name: " + teenager.getAs[String]("name")).show()



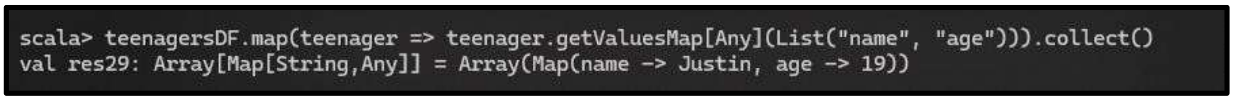
```
scala> teenagersDF.map(teenager => "Name: " + teenager.getAs[String]("name")).show()
+------------+
|       value|
+------------+
|Name: Justin|
+------------+
```

implicit val mapEncoder = org.apache.spark.sql.Encoders.kryo[Map[String, Any]]



```
scala> implicit val mapEncoder = org.apache.spark.sql.Encoders.kryo[Map[String, Any]]
val mapEncoder: org.apache.spark.sql.Encoder[Map[String,Any]] = class[value[0]: binary]
```

teenagersDF.map(teenager => teenager.getValuesMap[Any](List("name", "age"))).collect()

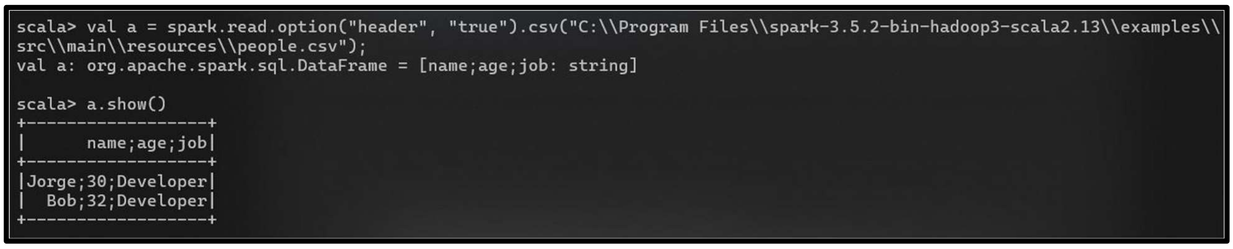


```
scala> teenagersDF.map(teenager => teenager.getValuesMap[Any](List("name", "age"))).collect()
val res29: Array[Map[String,Any]] = Array(Map(name -> Justin, age -> 19))
```

11. Perform the same operations on people.csv.

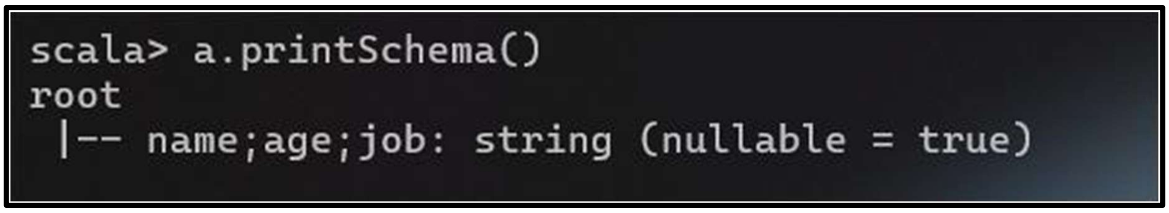
val a = spark.read.option("header", "true").csv("C:\\Program Files\\spark-3.5.2-bin-hadoop3-scala2.13\\examples\\src\\main\\resources\\people.csv");

a.show()



```
scala> val a = spark.read.option("header", "true").csv("C:\\Program Files\\spark-3.5.2-bin-hadoop3-scala2.13\\examples\\
src\\main\\resources\\people.csv");
val a: org.apache.spark.sql.DataFrame = [name;age;job: string]

scala> a.show()
+------------------+
|      name;age;job|
+------------------+
|Jorge;30;Developer|
|  Bob;32;Developer|
+------------------+
```

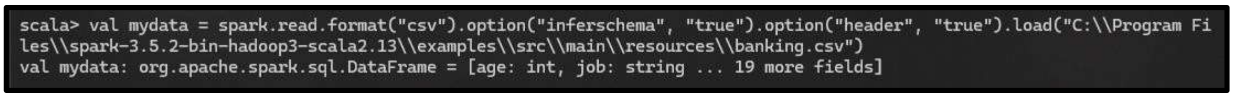
a.printSchema()



```
scala> a.printSchema()
root
 |-- name;age;job: string (nullable = true)
```

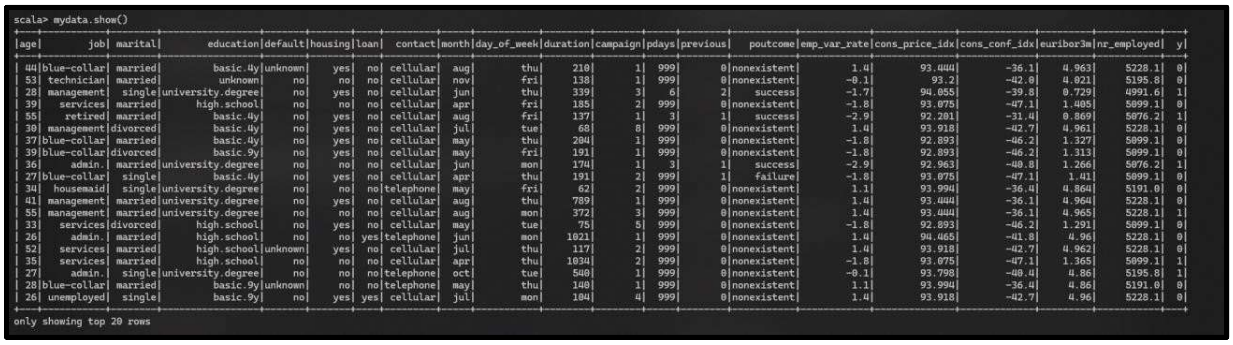
12. Performing operations on custom data.

val mydata = spark.read.format("csv").option("inferschema", "true").option("header", "true").load("C:\\Program Files\\spark-3.5.2-bin-hadoop3-scala2.13\\examples\\src\\main\\resources\\banking.csv")



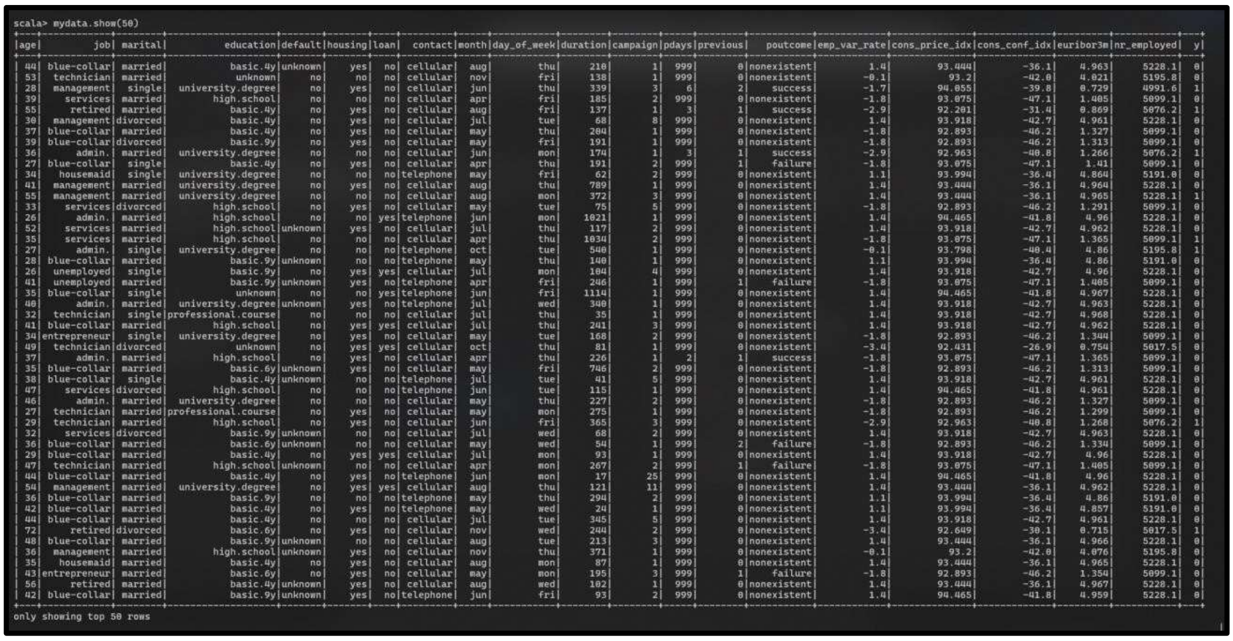
```
scala> val mydata = spark.read.format("csv").option("inferschema", "true").option("header", "true").load("C:\\Program Fi
les\\spark-3.5.2-bin-hadoop3-scala2.13\\examples\\src\\main\\resources\\banking.csv")
val mydata: org.apache.spark.sql.DataFrame = [age: int, job: string ... 19 more fields]
```

mydata.show()



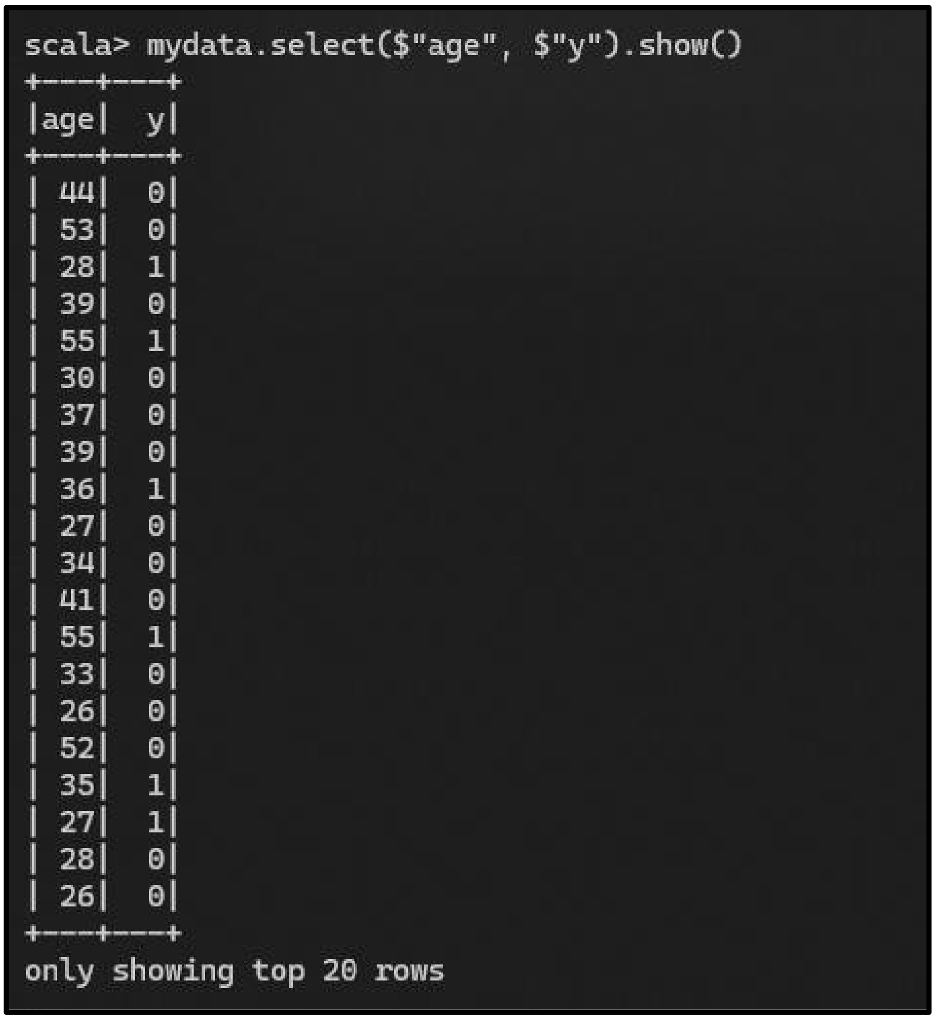
```
scala> mydata.show()
+---+-----------+--------+-----------------+-------+-------+----+---------+-----+-----------+--------+--------+-----+--------+-----------+------------+--------------+-------------+---------+-----------+---+
|age|        job| marital|        education|default|housing|loan|  contact|month|day_of_week|duration|campaign|pdays|previous|   poutcome|emp_var_rate|cons_price_idx|cons_conf_idx|euribor3m|nr_employed|  y|
+---+-----------+--------+-----------------+-------+-------+----+---------+-----+-----------+--------+--------+-----+--------+-----------+------------+--------------+-------------+---------+-----------+---+
| 44|blue-collar| married|         basic.4y|unknown|    yes|  no| cellular|  aug|        thu|     210|       1|  999|       0|nonexistent|         1.4|        93.444|        -36.1|    4.963|     5228.1|  0|
| 53| technician| married|          unknown|     no|     no|  no| cellular|  nov|        fri|     138|       1|  999|       0|nonexistent|        -0.1|          93.2|        -42.0|    4.021|     5195.8|  0|
| 28| management|  single|university.degree|     no|    yes|  no| cellular|  jun|        thu|     339|       3|    6|       2|    success|        -1.7|        94.055|        -39.8|    0.729|     4991.6|  1|
| 39|   services| married|      high.school|     no|     no|  no| cellular|  apr|        fri|     185|       2|  999|       0|nonexistent|        -1.8|        93.075|        -47.1|    1.405|     5099.1|  0|
| 55|    retired| married|         basic.4y|     no|    yes|  no| cellular|  aug|        fri|     137|       1|    3|       1|    success|        -2.9|        92.201|        -31.4|    0.869|     5076.2|  1|
| 30| management|divorced|         basic.4y|     no|    yes|  no| cellular|  jul|        tue|      68|       8|  999|       0|nonexistent|         1.4|        93.918|        -42.7|    4.961|     5228.1|  0|
| 37|blue-collar| married|         basic.4y|     no|    yes|  no| cellular|  may|        thu|     204|       1|  999|       0|nonexistent|        -1.8|        92.893|        -46.2|    1.327|     5099.1|  0|
| 39|blue-collar|divorced|         basic.9y|     no|    yes|  no| cellular|  may|        fri|     191|       1|  999|       0|nonexistent|        -1.8|        92.893|        -46.2|    1.313|     5099.1|  0|
| 36|     admin.| married|university.degree|     no|     no|  no| cellular|  jun|        mon|     174|       1|    3|       1|    success|        -2.9|        92.963|        -40.8|    1.266|     5076.2|  1|
| 27|blue-collar|  single|         basic.4y|     no|    yes|  no| cellular|  apr|        thu|     191|       2|  999|       1|    failure|        -1.8|        93.075|        -47.1|     1.41|     5099.1|  0|
| 34|  housemaid|  single|university.degree|     no|     no|  no|telephone|  may|        fri|      62|       2|  999|       0|nonexistent|         1.1|        93.994|        -36.4|    4.864|     5191.0|  0|
| 41| management| married|university.degree|     no|    yes|  no| cellular|  aug|        thu|     789|       1|  999|       0|nonexistent|         1.4|        93.444|        -36.1|    4.964|     5228.1|  0|
| 55| management| married|university.degree|     no|     no|  no| cellular|  aug|        mon|     372|       3|  999|       0|nonexistent|         1.4|        93.444|        -36.1|    4.965|     5228.1|  1|
| 33|   services|divorced|      high.school|     no|    yes|  no| cellular|  may|        tue|      75|       5|  999|       0|nonexistent|        -1.8|        92.893|        -46.2|    1.291|     5099.1|  0|
| 26|     admin.| married|      high.school|     no|     no| yes|telephone|  jun|        mon|    1021|       1|  999|       0|nonexistent|         1.4|        94.465|        -41.8|     4.96|     5228.1|  0|
| 52|   services| married|      high.school|unknown|    yes|  no| cellular|  jul|        thu|     117|       2|  999|       0|nonexistent|         1.4|        93.918|        -42.7|    4.962|     5228.1|  0|
| 35|   services| married|      high.school|     no|     no|  no| cellular|  apr|        thu|    1034|       2|  999|       0|nonexistent|        -1.8|        93.075|        -47.1|    1.365|     5099.1|  1|
| 27|     admin.|  single|university.degree|     no|     no|  no|telephone|  oct|        tue|     540|       1|  999|       0|nonexistent|        -0.1|        93.798|        -40.4|     4.86|     5195.8|  1|
| 28|blue-collar| married|         basic.9y|unknown|     no|  no|telephone|  may|        thu|     140|       1|  999|       0|nonexistent|         1.1|        93.994|        -36.4|     4.86|     5191.0|  0|
| 26| unemployed|  single|         basic.9y|     no|    yes| yes| cellular|  jul|        mon|     104|       4|  999|       0|nonexistent|         1.4|        93.918|        -42.7|     4.96|     5228.1|  0|
+---+-----------+--------+-----------------+-------+-------+----+---------+-----+-----------+--------+--------+-----+--------+-----------+------------+--------------+-------------+---------+-----------+---+
only showing top 20 rows
```

mydata.show(50)



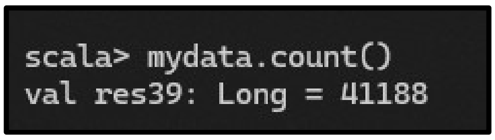
```
scala> mydata.show(50)
+---+------------+--------+-------------------+-------+-------+----+---------+-----+-----------+--------+--------+-----+--------+-----------+------------+--------------+-------------+---------+-----------+---+
|age|         job| marital|          education|default|housing|loan|  contact|month|day_of_week|duration|campaign|pdays|previous|   poutcome|emp_var_rate|cons_price_idx|cons_conf_idx|euribor3m|nr_employed|  y|
+---+------------+--------+-------------------+-------+-------+----+---------+-----+-----------+--------+--------+-----+--------+-----------+------------+--------------+-------------+---------+-----------+---+
| 44| blue-collar| married|           basic.4y|unknown|    yes|  no| cellular|  aug|        thu|     210|       1|  999|       0|nonexistent|         1.4|        93.444|        -36.1|    4.963|     5228.1|  0|
| 53|  technician| married|            unknown|     no|     no|  no| cellular|  nov|        fri|     138|       1|  999|       0|nonexistent|        -0.1|          93.2|        -42.0|    4.021|     5195.8|  0|
| 28|  management|  single|  university.degree|     no|    yes|  no| cellular|  jun|        thu|     339|       3|    6|       2|    success|        -1.7|        94.055|        -39.8|    0.729|     4991.6|  1|
| 39|    services| married|        high.school|     no|     no|  no| cellular|  apr|        fri|     185|       2|  999|       0|nonexistent|        -1.8|        93.075|        -47.1|    1.405|     5099.1|  0|
| 55|     retired| married|           basic.4y|     no|    yes|  no| cellular|  aug|        fri|     137|       1|    3|       1|    success|        -2.9|        92.201|        -31.4|    0.869|     5076.2|  1|
| 30|  management|divorced|           basic.4y|     no|    yes|  no| cellular|  jul|        tue|      68|       8|  999|       0|nonexistent|         1.4|        93.918|        -42.7|    4.961|     5228.1|  0|
| 37| blue-collar| married|           basic.4y|     no|    yes|  no| cellular|  may|        thu|     204|       1|  999|       0|nonexistent|        -1.8|        92.893|        -46.2|    1.327|     5099.1|  0|
| 39| blue-collar|divorced|           basic.9y|     no|    yes|  no| cellular|  may|        fri|     191|       1|  999|       0|nonexistent|        -1.8|        92.893|        -46.2|    1.313|     5099.1|  0|
| 36|      admin.| married|  university.degree|     no|     no|  no| cellular|  jun|        mon|     174|       1|    3|       1|    success|        -2.9|        92.963|        -40.8|    1.266|     5076.2|  1|
| 27| blue-collar|  single|           basic.4y|     no|    yes|  no| cellular|  apr|        thu|     191|       2|  999|       1|    failure|        -1.8|        93.075|        -47.1|     1.41|     5099.1|  0|
| 34|   housemaid|  single|  university.degree|     no|     no|  no|telephone|  may|        fri|      62|       2|  999|       0|nonexistent|         1.1|        93.994|        -36.4|    4.864|     5191.0|  0|
| 41|  management| married|  university.degree|     no|    yes|  no| cellular|  aug|        thu|     789|       1|  999|       0|nonexistent|         1.4|        93.444|        -36.1|    4.964|     5228.1|  0|
| 55|  management| married|  university.degree|     no|     no|  no| cellular|  aug|        mon|     372|       3|  999|       0|nonexistent|         1.4|        93.444|        -36.1|    4.965|     5228.1|  1|
| 33|    services|divorced|        high.school|     no|    yes|  no| cellular|  may|        tue|      75|       5|  999|       0|nonexistent|        -1.8|        92.893|        -46.2|    1.291|     5099.1|  0|
| 26|      admin.| married|        high.school|     no|     no| yes|telephone|  jun|        mon|    1021|       1|  999|       0|nonexistent|         1.4|        94.465|        -41.8|     4.96|     5228.1|  0|
| 52|    services| married|        high.school|unknown|    yes|  no| cellular|  jul|        thu|     117|       2|  999|       0|nonexistent|         1.4|        93.918|        -42.7|    4.962|     5228.1|  0|
| 35|    services| married|        high.school|     no|     no|  no| cellular|  apr|        thu|    1034|       2|  999|       0|nonexistent|        -1.8|        93.075|        -47.1|    1.365|     5099.1|  1|
| 27|      admin.|  single|  university.degree|     no|     no|  no|telephone|  oct|        tue|     540|       1|  999|       0|nonexistent|        -0.1|        93.798|        -40.4|     4.86|     5195.8|  1|
| 28| blue-collar| married|           basic.9y|unknown|     no|  no|telephone|  may|        thu|     140|       1|  999|       0|nonexistent|         1.1|        93.994|        -36.4|     4.86|     5191.0|  0|
| 26|  unemployed|  single|           basic.9y|     no|    yes| yes| cellular|  jul|        mon|     104|       4|  999|       0|nonexistent|         1.4|        93.918|        -42.7|     4.96|     5228.1|  0|
| 41|  unemployed| married|           basic.9y|unknown|    yes|  no|telephone|  apr|        fri|     246|       1|  999|       1|    failure|        -1.8|        93.075|        -47.1|    1.405|     5099.1|  0|
| 35| blue-collar|  single|            unknown|     no|     no| yes|telephone|  jun|        fri|    1114|       1|  999|       0|nonexistent|         1.4|        94.465|        -41.8|    4.967|     5228.1|  0|
| 40|      admin.| married|  university.degree|unknown|    yes|  no|telephone|  jul|        wed|     340|       1|  999|       0|nonexistent|         1.4|        93.918|        -42.7|    4.963|     5228.1|  0|
| 32|  technician|  single|professional.course|     no|     no|  no| cellular|  jul|        thu|      35|       1|  999|       0|nonexistent|         1.4|        93.918|        -42.7|    4.968|     5228.1|  0|
| 41| blue-collar| married|        high.school|     no|    yes| yes| cellular|  jul|        thu|     241|       3|  999|       0|nonexistent|         1.4|        93.918|        -42.7|    4.962|     5228.1|  0|
| 34|entrepreneur|  single|  university.degree|     no|    yes|  no| cellular|  may|        tue|     168|       2|  999|       0|nonexistent|        -1.8|        92.893|        -46.2|    1.344|     5099.1|  0|
| 49|  technician|divorced|            unknown|     no|    yes| yes| cellular|  oct|        thu|      81|       1|  999|       0|nonexistent|        -3.4|        92.431|        -26.9|    0.754|     5017.5|  0|
| 37|      admin.| married|        high.school|     no|    yes|  no| cellular|  apr|        thu|     226|       1|    2|       1|    success|        -1.8|        93.075|        -47.1|    1.365|     5099.1|  0|
| 35| blue-collar| married|           basic.6y|unknown|    yes|  no| cellular|  may|        fri|     746|       2|  999|       0|nonexistent|        -1.8|        92.893|        -46.2|    1.313|     5099.1|  0|
| 38| blue-collar|  single|           basic.4y|unknown|     no|  no|telephone|  jul|        tue|      41|       5|  999|       0|nonexistent|         1.4|        93.918|        -42.7|    4.961|     5228.1|  0|
| 47|    services|divorced|        high.school|     no|     no|  no|telephone|  jun|        tue|     115|       1|  999|       0|nonexistent|         1.4|        94.465|        -41.8|    4.961|     5228.1|  0|
| 46|      admin.| married|  university.degree|     no|     no|  no| cellular|  may|        thu|     227|       2|  999|       0|nonexistent|        -1.8|        92.893|        -46.2|    1.327|     5099.1|  0|
| 27|  technician| married|professional.course|     no|    yes|  no| cellular|  may|        mon|     275|       1|  999|       0|nonexistent|        -1.8|        92.893|        -46.2|    1.299|     5099.1|  0|
| 29|  technician| married|        high.school|     no|    yes|  no| cellular|  jun|        fri|     365|       3|  999|       0|nonexistent|        -2.9|        92.963|        -40.8|    1.268|     5076.2|  1|
| 32|    services|divorced|           basic.9y|unknown|     no|  no| cellular|  jul|        wed|      68|       2|  999|       0|nonexistent|         1.4|        93.918|        -42.7|    4.963|     5228.1|  0|
| 36| blue-collar| married|           basic.6y|unknown|     no|  no| cellular|  may|        wed|      54|       1|  999|       2|    failure|        -1.8|        92.893|        -46.2|    1.334|     5099.1|  0|
| 29| blue-collar| married|           basic.4y|     no|    yes| yes| cellular|  jul|        mon|      93|       1|  999|       0|nonexistent|         1.4|        93.918|        -42.7|     4.96|     5228.1|  0|
| 47|  technician| married|        high.school|unknown|     no|  no| cellular|  apr|        mon|     267|       2|  999|       1|    failure|        -1.8|        93.075|        -47.1|    1.405|     5099.1|  0|
| 44| blue-collar| married|           basic.4y|     no|    yes|  no|telephone|  jun|        mon|      17|      25|  999|       0|nonexistent|         1.4|        94.465|        -41.8|     4.96|     5228.1|  0|
| 54|  management| married|  university.degree|     no|    yes| yes| cellular|  aug|        thu|     121|      11|  999|       0|nonexistent|         1.4|        93.444|        -36.1|    4.962|     5228.1|  0|
| 36| blue-collar| married|           basic.9y|     no|     no|  no|telephone|  may|        thu|     294|       2|  999|       0|nonexistent|         1.1|        93.994|        -36.4|     4.86|     5191.0|  0|
| 42| blue-collar| married|           basic.4y|     no|    yes|  no|telephone|  may|        wed|      24|       1|  999|       0|nonexistent|         1.1|        93.994|        -36.4|    4.857|     5191.0|  0|
| 44| blue-collar| married|           basic.4y|     no|     no|  no| cellular|  jul|        tue|     345|       5|  999|       0|nonexistent|         1.4|        93.918|        -42.7|    4.961|     5228.1|  0|
| 72|     retired|divorced|           basic.6y|     no|    yes|  no| cellular|  nov|        wed|     244|       2|  999|       0|nonexistent|        -3.4|        92.649|        -30.1|    0.715|     5017.5|  1|
| 48| blue-collar| married|           basic.9y|unknown|     no|  no| cellular|  aug|        tue|     213|       3|  999|       0|nonexistent|         1.4|        93.444|        -36.1|    4.966|     5228.1|  0|
| 36|  management| married|        high.school|unknown|    yes|  no| cellular|  nov|        thu|     371|       1|  999|       0|nonexistent|        -0.1|          93.2|        -42.0|    4.076|     5195.8|  0|
| 35|   housemaid| married|           basic.4y|     no|    yes|  no| cellular|  aug|        mon|      87|       1|  999|       0|nonexistent|         1.4|        93.444|        -36.1|    4.965|     5228.1|  0|
| 43|entrepreneur| married|           basic.6y|     no|    yes|  no| cellular|  may|        mon|     195|       3|  999|       1|    failure|        -1.8|        92.893|        -46.2|    1.354|     5099.1|  0|
| 56|     retired| married|           basic.4y|unknown|    yes|  no| cellular|  aug|        wed|     102|       1|  999|       0|nonexistent|         1.4|        93.444|        -36.1|    4.967|     5228.1|  0|
| 42| blue-collar| married|           basic.9y|unknown|    yes|  no|telephone|  jun|        fri|      93|       2|  999|       0|nonexistent|         1.4|        94.465|        -41.8|    4.959|     5228.1|  0|
+---+------------+--------+-------------------+-------+-------+----+---------+-----+-----------+--------+--------+-----+--------+-----------+------------+--------------+-------------+---------+-----------+---+
only showing top 50 rows
```

mydata.select($"age", $"y").show()



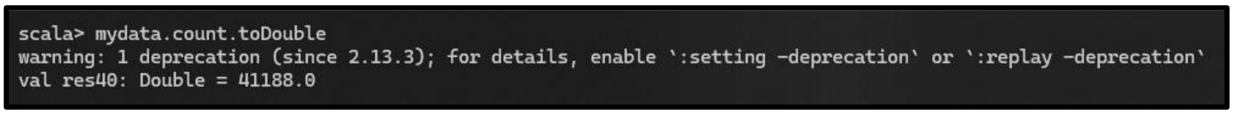
```
scala> mydata.select($"age", $"y").show()
+---+---+
|age|  y|
+---+---+
| 44|  0|
| 53|  0|
| 28|  1|
| 39|  0|
| 55|  1|
| 30|  0|
| 37|  0|
| 39|  0|
| 36|  1|
| 27|  0|
| 34|  0|
| 41|  0|
| 55|  1|
| 33|  0|
| 26|  0|
| 52|  0|
| 35|  1|
| 27|  1|
| 28|  0|
| 26|  0|
+---+---+
only showing top 20 rows
```

mydata.count()



```
scala> mydata.count()
val res39: Long = 41188
```

mydata.count.toDouble



```
scala> mydata.count.toDouble
warning: 1 deprecation (since 2.13.3); for details, enable `:setting -deprecation` or `:replay -deprecation`
val res40: Double = 41188.0
```

Practical 3: Spark GraphX