What is the difference between RDD, Dataframe and Dataset in Spark?

RDD, Dataframe and Dataset are all Spark APIs introduced in Spark at different points in time. The goal of these API is to help us work with large datasets in a distributed fashion in Spark with performance in mind.

RDD:

RDD is now considered to be a low level API

RDD is still the core of Spark

Whether you use Dataframe or Dataset, all your operations eventually get transformed to RDD .

RDD API provides many transformation functions like map(), filter() and reduce() etc. for performing computations on Data.

RDD distribute a collection of JVM objects

Can not catch syntax and analysis errors at compile time

RDD Example

val rdd = sc.textFile("student.txt")

val rdd\_filtered = rdd.filter(\_.age > 13) // transformation

rdd\_filtered.saveAsTextFile("students\_over13.txt") // action

In this above example, first we read a file and we get a RDD.

Next we apply a filter function on RDD which again results in a RDD and we are calling it rdd\_filtered.

filter() is a transformation function. All transformation functions result in a RDD.

Finally we call saveAsTextFile on rdd\_filtered which is an action function to save the contents of RDD.

DataFrame:

All operations with Data Frames go through Spark’s catalyst optimizer which converts our Dataframe code to an optimized set of code in RDD.

Let’s you perform “SQL like” operations on data

DataFrame distribute a collection of Row objects

Catch syntax errors at compile time and analysis errors only at run time.

DataFrame Example

val df = spark.read.csv("student.txt")

val df\_filtered = df.filter("age > 13")

Dataset:

Considered high level API Just like Data frames, all operations with Datasets go through Spark’s catalyst optimizer which converts our Dataframe code to an optimized set of code in RDD.

Just like Data frames, let’s you perform “SQL like” operations on data

Dataset lets you work with data represented as an user defined object.

Datasets are considered type safe

Catch syntax errors at compile time and due to type safety catch analysis errors also at compile time

val ds = spark.read.csv("student.txt").as[Student]

val ds\_filtered =ds.filter("age > 13")

Data Representation

RDD : Distributed collection of elements.

DataFrame : Distributed collection of data organized into columns.

DataSet : Combination of RDD and DataFrame.

Schema Projection

RDD : Schemas need to be defined manually.

DataFrame : Auto-discovery of file schemas.

DataSet : Auto-discovery of file schemas.

Programming Language Support:

RDD :Java Scala Python R

DF :Java Scala Python R

DS :Java Scala

Ref link

[What is the difference between RDD, Dataframe and Dataset in Spark? – Hadoop In Real World](https://www.hadoopinrealworld.com/what-is-the-difference-between-rdd-dataframe-and-dataset-in-spark/)