

🔍 Search the docs ...

[Installation](#)

[Quickstart: DataFrame](#)

[Quickstart: Spark Connect](#)

[Quickstart: Pandas API on](#)

[Spark](#)

[Testing PySpark](#)

## Quickstart: Spark Connect

Spark Connect introduced a decoupled client-server architecture for Spark that allows remote connectivity to Spark clusters using the [DataFrame API](#).

This notebook walks through a simple step-by-step example of how to use Spark Connect to build any type of application that needs to leverage the power of Spark when working with data.

Spark Connect includes both client and server components and we will show you how to set up and use both.

☰ On this page

[Launch Spark server with Spark Connect](#)

[Connect to Spark Connect server](#)

[Create DataFrame](#)

## Launch Spark server with Spark Connect 🚀

To launch Spark with support for Spark Connect sessions, run the `start-connect-server.sh` script.

```
[1]: !${HOME}/sbin/start-connect-server.sh --packages org.apache.spark:spark-connect_2.12:${SPARK_VERSION}
```

## Connect to Spark Connect server

Now that the Spark server is running, we can connect to it remotely using Spark Connect. We do this by creating a remote Spark session on the client where our application runs. Before we can do that, we need to make sure to stop the existing regular Spark session because it cannot coexist with the remote Spark Connect session we are about to create.

```
[2]: from pyspark.sql import SparkSession

SparkSession.builder.master("local[*]").getOrCreate().stop()
```

The command we used above to launch the server configured Spark to run as `localhost:15002`. So now we can create a remote Spark session on the client using the following command.

```
[3]: spark = SparkSession.builder.remote("sc://localhost:15002").getOrCreate()
```

## Create DataFrame

Once the remote Spark session is created successfully, it can be used the same way as a regular Spark session. Therefore, you can create a DataFrame with the following command.

```
[4]: from datetime import datetime, date
from pyspark.sql import Row

df = spark.createDataFrame([
    Row(a=1, b=2., c='string1', d=date(2000, 1, 1), e=datetime(2000, 1, 1, 12, 0)),
    Row(a=2, b=3., c='string2', d=date(2000, 2, 1), e=datetime(2000, 1, 2, 12, 0)),
    Row(a=4, b=5., c='string3', d=date(2000, 3, 1), e=datetime(2000, 1, 3, 12, 0))
])
df.show()
```

See 'Live Notebook: DataFrame' at [the quickstart page](#) for more detail usage of DataFrame API.

◀ Previous  
**Quickstart: DataFrame**

Next ▶  
**Quickstart: Pandas API on Spark**

---

© Copyright .

Created using [Sphinx](#) 3.0.4.