Assignment 04

Sabrina Minaya Vasquez

October 8, 2025

|  |
| --- |
| Important |
| ConnectionRefusedError Traceback (most recent call last) Cell In[72], line 12 9 pio.renderers.default = “notebook+notebook\_connected+vscode” 11 # Initialize Spark Session —> 12 spark = SparkSession.builder.appName(“LightcastData”).getOrCreate() 14 # Load Data 15 df = spark.read.option(“header”, “true”).option(“inferSchema”, “true”).option(“multiLine”,“true”).option(“escape”, “"”)options=self.\_options 501 else: 502 getattr( –> 503 getattr(session.\_jvm, “SparkSession$"), "MODULE$” 504 ).applyModifiableSettings(session.\_jsparkSession, self.\_options) 505 return session |

|  |
| --- |
| Tip |
| That “ConnectionRefusedError” when calling SparkSession.builder.getOrCreate() almost always means the Python side couldn’t talk to the Java gateway that Spark starts. In practice it’s one of these:  Java isn’t installed or JAVA\_HOME isn’t set in the environment your notebook/venv sees  You’re launching without a local master (rare, but notebooks sometimes need it)  A stale/failed gateway from a previous run is blocking the new one  Here’s a quick, do-it-now fix checklist. Do the terminal steps in your Ubuntu shell, then restart your kernel and run the Python snippet.  Terminal (Ubuntu) — verify and set Java 1. 1) Install Java 11 if needed sudo apt-get update sudo apt-get install -y openjdk-11-jdk 2. 2) Find JAVA\_HOME path readlink -f “$(which java)” # e.g., /usr/lib/jvm/java-11-openjdk-amd64/bin/java 3. 3) Set JAVA\_HOME for your shell sessions echo ‘export JAVA\_HOME=/usr/lib/jvm/java-11-openjdk-amd64’ >> ~/.bashrc echo ‘export PATH=PATH’ >> ~/.bashrc source ~/.bashrc 4. 4) Clean any stale Spark temp dirs (optional but helpful) rm -rf /tmp/spark-\* /tmp/hsperfdata\_${USER} 2>/dev/null || true 5. 5) (Optional) kill stray Java/Spark processes from previous runs pgrep -af ‘org.apache.spark|py4j’ && kill -9 $(pgrep -f ‘org.apache.spark|py4j’) 2>/dev/null || true  n your notebook (Python) — make the env explicit and force local mode  Run this at the very top of your notebook before creating the session:  import os, sys 6. Make sure the notebook/venv sees Java and uses the current Python os.environ.setdefault(“JAVA\_HOME”, “/usr/lib/jvm/java-11-openjdk-amd64”) os.environ[“PYSPARK\_PYTHON”] = sys.executable os.environ[“PYSPARK\_DRIVER\_PYTHON”] = sys.executable  from pyspark.sql import SparkSession  spark = ( SparkSession.builder .master(“local[\*]”) # ensure local master .appName(“LightcastData”) .config(“spark.ui.enabled”, “false”) # avoid trying to bind a UI port .getOrCreate() ) |