**Project: Health care Data Analysis**

1. **Data Ingestion:**

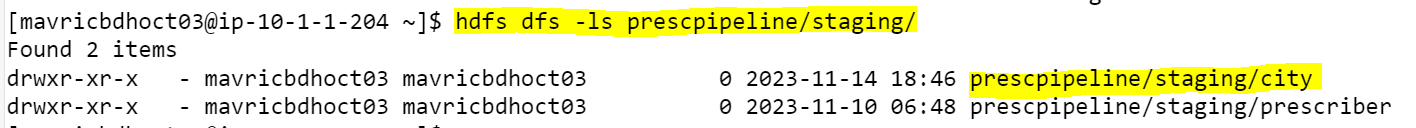
The 1st step in the project is Data Ingestion. Sample data files are provided and they have to be ingested in to HDFS in to the project’s location.

**City Report**

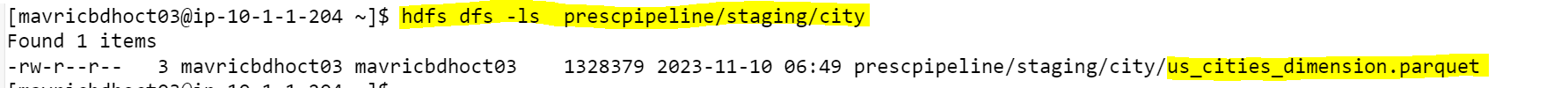
A close-up of a text

Description automatically generated





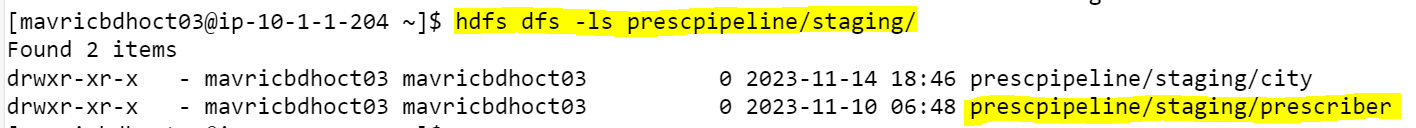




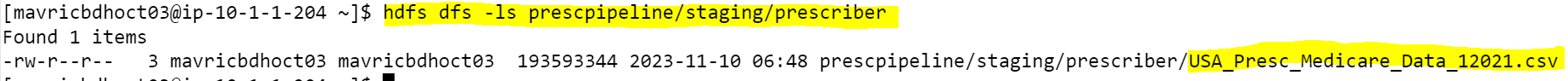
**Prescriber Report:**

A close-up of a text

Description automatically generated







1. **Pre-Processing:**

**The 2nd step is to cleanse the data and use only the data that is required for analysis/reports using pyspark.**

**Clean City Data:**

* + Select only required Columns in city data file like city, state\_id,state\_name,county\_name,population,zips

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

* + **Convert city, state and county fields to Upper Case**

A yellow line with black text

Description automatically generated with medium confidence

**Clean Prescriber Data:**

* + Select only required fields such as npi, nppes\_provider\_last\_org\_name, nppes\_provider\_first\_name, nppes\_provider\_city, nppes\_provider\_state, specialty\_description, drug\_name, total\_claim\_count, total\_day\_supply, total\_drug\_cost

A screenshot of a computer

Description automatically generated

A close up of a document

Description automatically generated

* + **Rename the above fields to shorter names.**

A yellow text on a white background

Description automatically generated

A close-up of a computer code

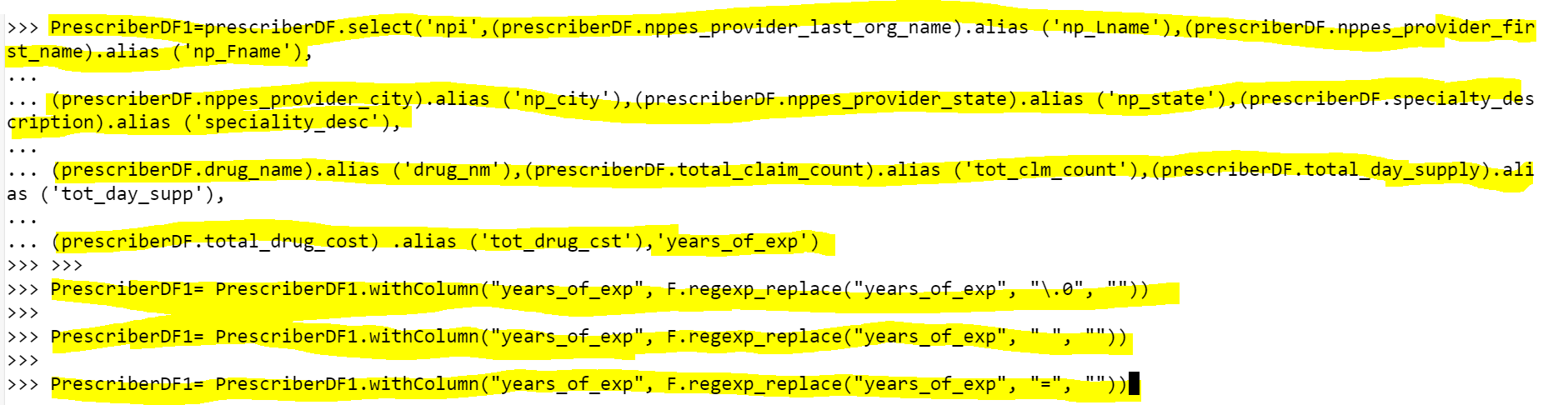
Description automatically generated

* + **Add a Country Field 'USA' to the above data.**

A close-up of a receipt

Description automatically generated

* + **Clean the “years\_of\_exp” to extract only the numbers.**



A close-up of a document

Description automatically generated

* + **Convert the years\_of\_exp field to integer.**

A screenshot of a computer code

Description automatically generated

* + **Combine First Name and Last Name into a single field and remove the individual columns.**

A screenshot of a computer

Description automatically generated

A white paper with black text

Description automatically generated

* + **Count the number of null values for each column.**

A close up of a computer screen

Description automatically generated

* + **Clean all the Null/Nan Values.**

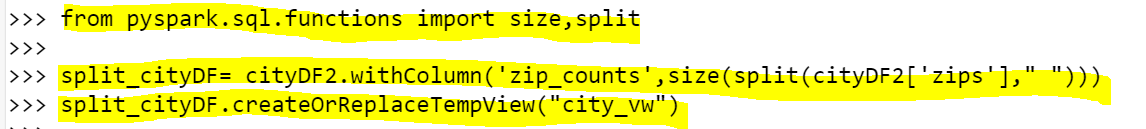
A yellow text on a white background

Description automatically generated

**The 3rd step is to transform the cleansed data in to the required reports using pyspark.**

**Transform Logic: City Report**

* + **Calculate the Number of zips in each city.**



A screenshot of a computer

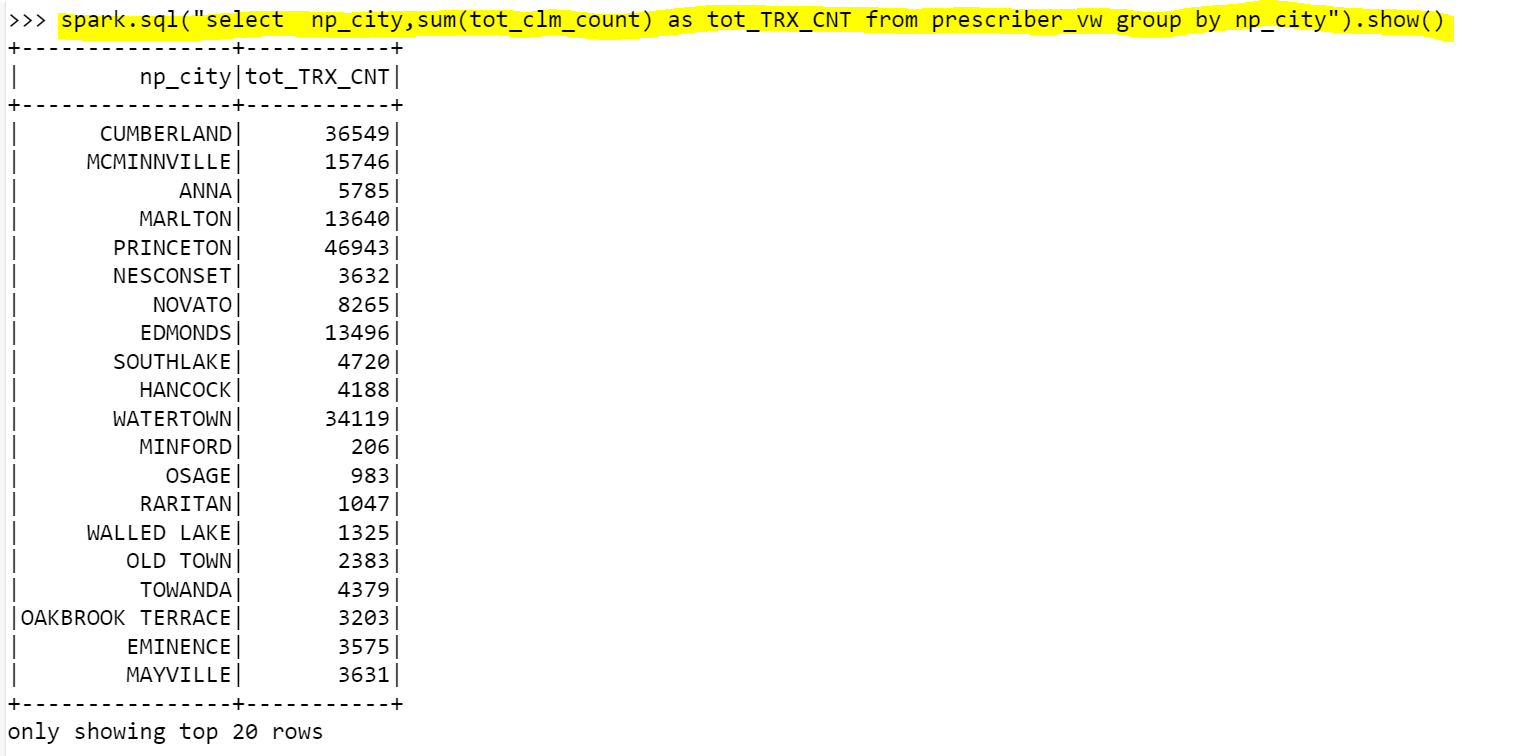
Description automatically generated

* + **Calculate the number of distinct Prescribers assigned for each City.**

A screenshot of a computer

Description automatically generated

* + **Calculate total TRX\_CNT prescribed for each city.**



* + **Do not report a city in the final report if no prescriber is assigned to it.**

A yellow and black text

Description automatically generated

**Transform logic: Prescriber Report:**

* + **Top 5 Prescribers with highest total\_claim\_count per each state.**

A screenshot of a computer

Description automatically generated

* + **Consider the prescribers only from 20 to 50 years of experience.**

A screenshot of a computer screen

Description automatically generated

**Output City report Layout:**

A yellow and black text

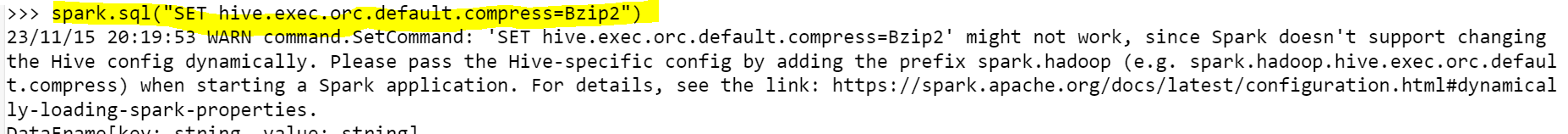
Description automatically generated

**Hive storage:**

**Pyspark:**

A yellow text on a white background

Description automatically generated



A screenshot of a computer

Description automatically generated

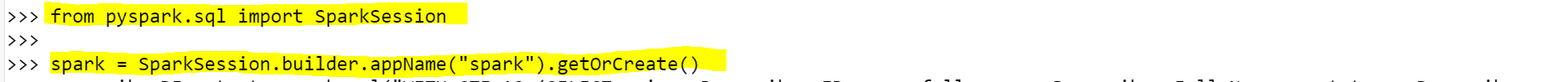
**Output Prescriber report layout**

A yellow and black text

Description automatically generated

**Hive storage:**

**Pyspark:**



A yellow text on a white background

Description automatically generated

A screenshot of a computer screen

Description automatically generated