# Table of Contents

Contents

[Table of Contents 1](#_Toc133921190)

[Working with restaurant dataset 2](#_Toc133921191)

[Creating Spark session 2](#_Toc133921192)

[Read restaurant\_csv dataset 2](#_Toc133921193)

[Applying Geocode UDF 3](#_Toc133921194)

[Applying Geohash 3](#_Toc133921195)

[Working with Weather dataset 4](#_Toc133921196)

[Reading Weather dataset and applying geohash UDF 4](#_Toc133921197)

[Deleting duplicated rows 5](#_Toc133921198)

[Saving weather cleaned dataset 5](#_Toc133921199)

[Joining Restaurant and Weather Datasets 6](#_Toc133921200)

[Cleaning joined dataset from Nulls and Drop Duplicates 6](#_Toc133921201)

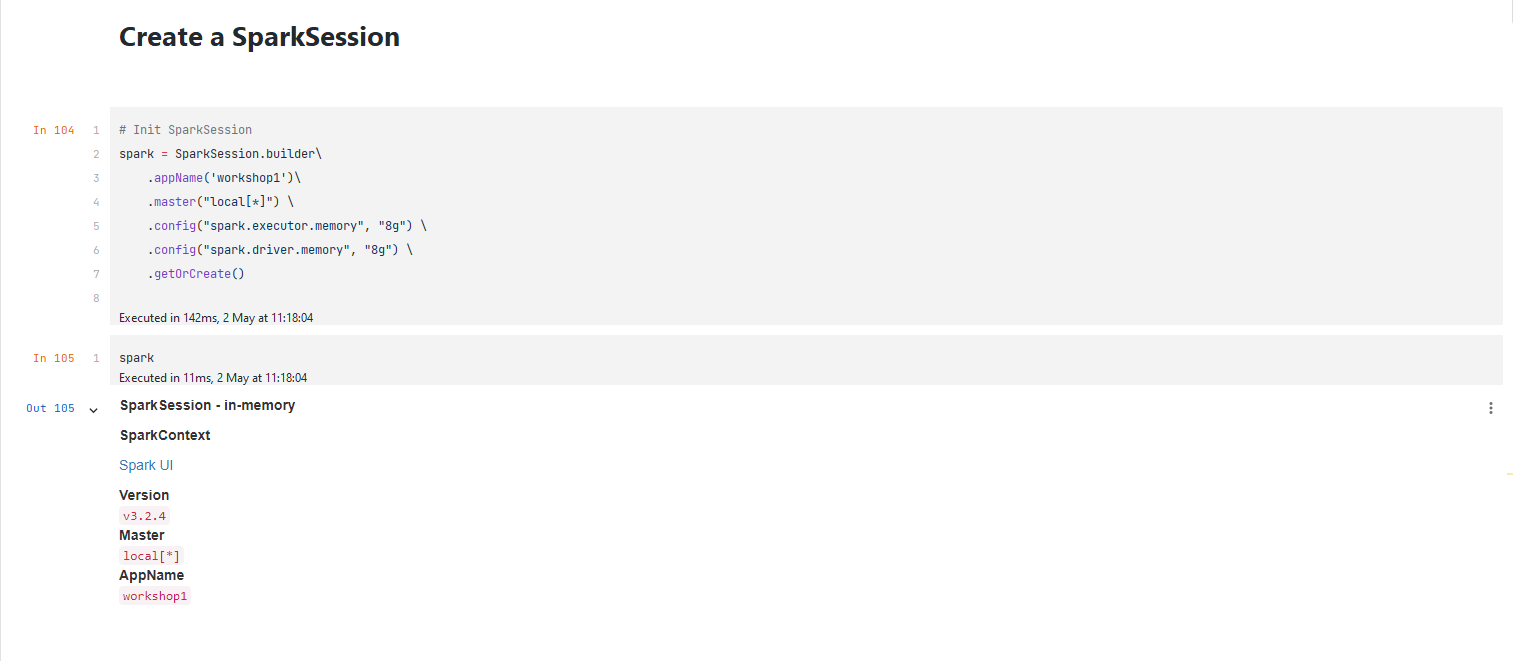
[Saving datasets. 7](#_Toc133921202)

[Testing 7](#_Toc133921203)

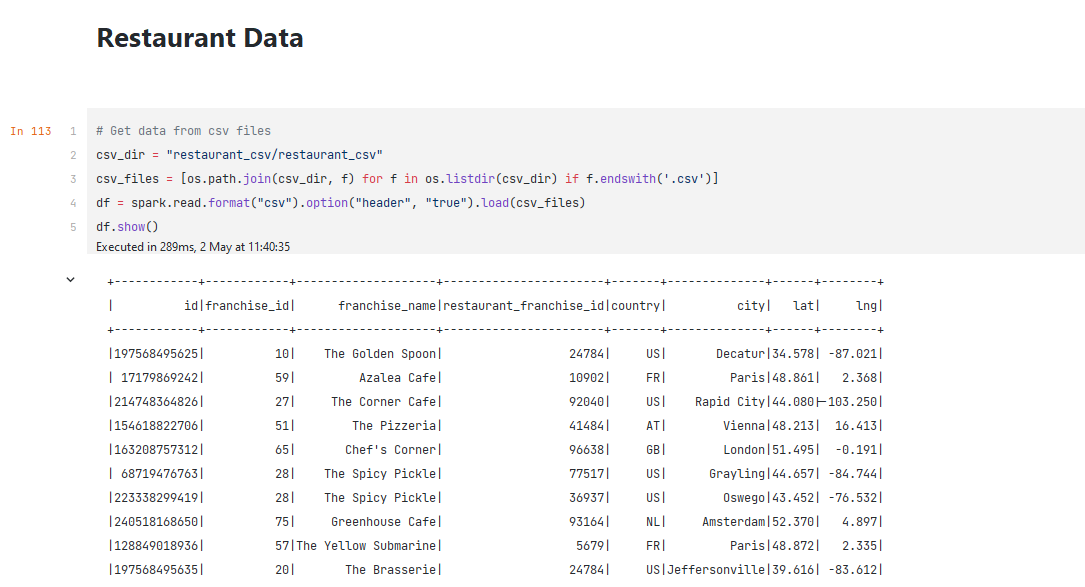
# Working with restaurant dataset

Install Spark locally using one of the methods described [here](https://spark.apache.org/downloads.html) or in Docker.

## Creating Spark session



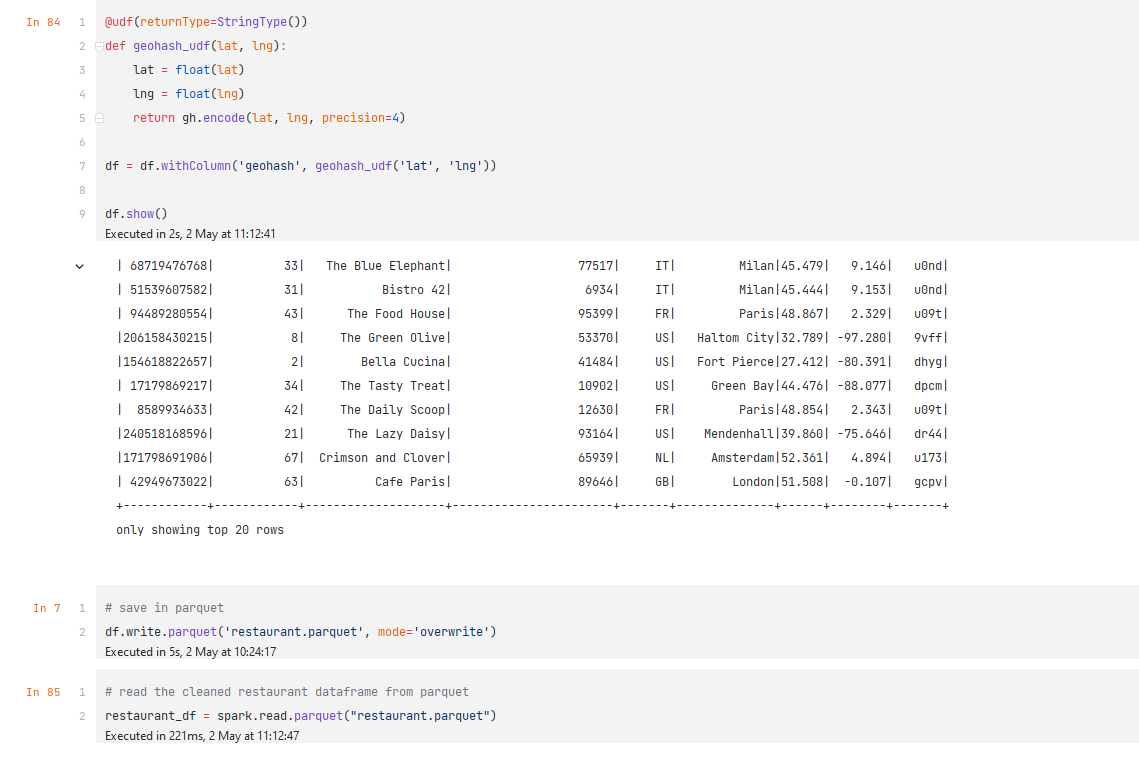
## Read restaurant\_csv dataset



## Applying Geocode UDF



## Applying Geohash

I also created geohash UDF in order to hash my geocode. Saved it as parquest file as “restaurant”

# Working with Weather dataset

## Reading Weather dataset and applying geohash UDF



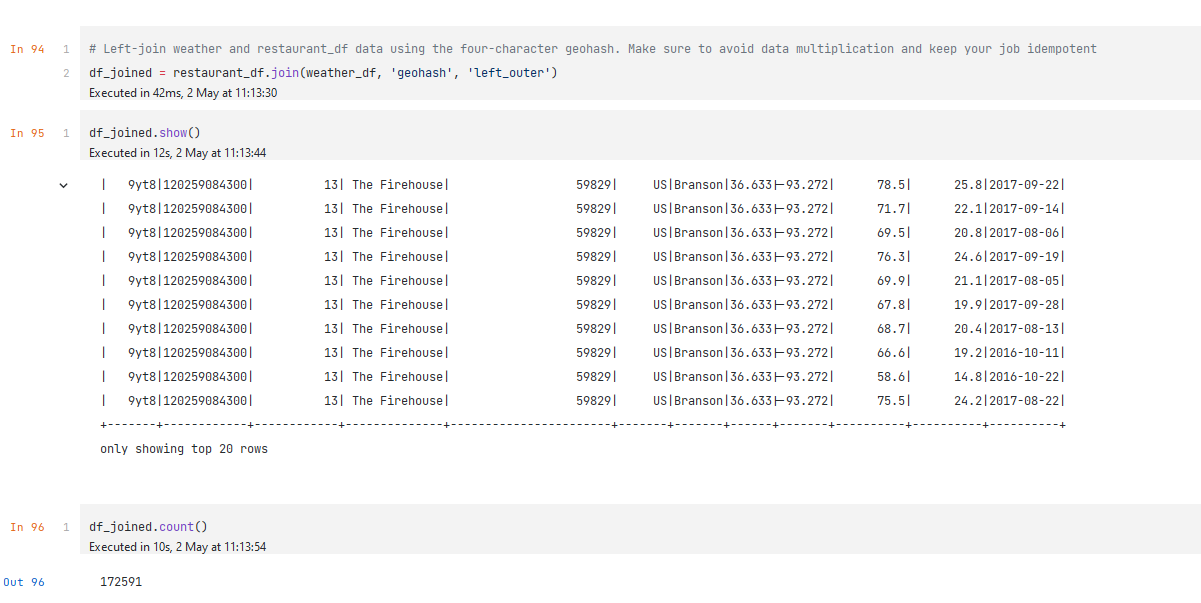
## Deleting duplicated rows



## Saving weather cleaned dataset

Save dataset then read it again to decrease computing time

# Joining Restaurant and Weather Datasets



## Cleaning joined dataset from Nulls and Drop Duplicates



## Saving datasets.

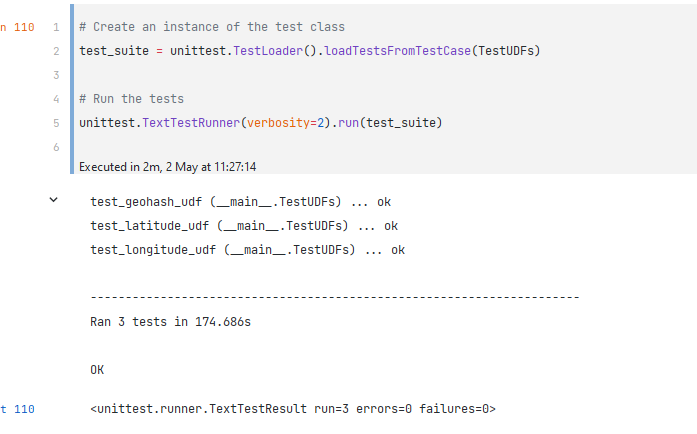
Saved dataset using partitioning by “city” column



# Testing

I unit test in order to check following UDFs: latitude\_udf, longitude\_udf, geohash\_udf.

Unit test results with positive result:



Unit test results with some negative result:

