Jesus
Maciel

```
#import Libraries
  import pandas as pd
  import numpy as np
  import tenserflow as tf

#import dataset as df
  df = pd.read_csv("data.csv")

#check data to see what I'm workin with
  print(df)

#The image on the board says the data has 8 columns
#columns: key, fare_amount, pickup_datetime, pickup_lonitude, pickup_latitude, dropoff_lonitude, dropoff_latitude
#passenger_count.
#Euclidean Distance = √[(x2 - x1)^2 + (y2 - y1)^2]
#where: (x1, y1) is the coordinate of the first point. (pickup location)
#       (x2, y2) is the coordinate of the second point. (dropoff location)
# make conditional statement to drop values from dataset that have zero for all coordinates.
  df = df.drop(0) ? something like this.
# drop null values from dataset
  df = df.dropna() ?

# reset index positions
  df = google it ?

# set euclidean distance code
def euc_distance(lat1, long1, lat2, long2):
      return(((lat1 - lat2)**2 + (long1 - long2)**2)**0.5)

# set euclidean distance column
  df['distance'] = euc_distance(df['pickup_latitude'], df['pickup_lonitude'], df['dropoff_latitude'],
                               df['dropoff_lonitude'])
```

Off the top of my head this is as much as I can think of. The next steps would be to see how I could use the data to make a model. I'd have to decide what my desired output is to look like and base my models architecture off it. This would als dictate what type of model I choose as well as its hyperparameters