1. Use the location coordinates to find the distance for each trip. Create a new column – *'distance'* and store its value there. Use the following formula:

$$distance = \sqrt{\left(lat_{drop} - lat_{pick}\right)^2 + \left(long_{drop} - long_{pick}\right)^2}$$

```
import csv
from math import sqrt
import numpy as np
import pandas as pd

df = pd.read_csv('Dataset_Day3.csv')

print(df)

df['distance'] = df.apply(lambda x:
sqrt((x['dropoff_latitude'] - x['pickup_latitude'])
** 2 + (
             x['dropoff_longitude'] -
x['pickup_longitude']) ** 2), axis=1)

print(df)
```

```
  tejas\PycharmProjects\pythonProject\START\Distance.py
2              key  fare_amount  ... dropoff_latitude  passenger_count
3 0       24238194          7.5  ...        40.723217                1
4 1       27835199          7.7  ...        40.750325                1
5 2       44984355         12.9  ...        40.772647                1
6 3       25894730          5.3  ...        40.803349                3
7 4       17610152         16.0  ...        40.761247                5
8 ...          ...          ...  ...              ...              ...
9 199995  42598914          3.0  ...        40.740297                1
10 199996  16382965          7.5  ...        40.739620                1
11 199997  27804658         30.9  ...        40.692588                2
12 199998  20259894         14.5  ...        40.695416                1
13 199999  11951496         14.1  ...        40.768793                1
14
15 [200000 rows x 8 columns]
16             key  fare_amount  ... passenger_count  distance
17 0       24238194          7.5  ...                1  0.015140
18 1       27835199          7.7  ...                1  0.022103
19 2       44984355         12.9  ...                1  0.053109
20 3       25894730          5.3  ...                3  0.016528
21 4       17610152         16.0  ...                5  0.051031
22 ...          ...          ...  ...              ...       ...
23 199995  42598914          3.0  ...                1  0.001064
24 199996  16382965          7.5  ...                1  0.022126
25 199997  27804658         30.9  ...                2  0.142223
26 199998  20259894         14.5  ...                1  0.033101
27 199999  11951496         14.1  ...                1  0.048729
28
29 [200000 rows x 9 columns]
30
31 Process finished with exit code 0
32
```

2.Find all the '*key*' values for which the attributes: *fare_amount* & *passenger_count* & *distance* are outliers. **Remove all rows with outliers**.

```
import csv
from math import sqrt
import numpy as np
import pandas as pd

df = pd.read_csv('Dataset_Day3.csv')
df['distance'] = df.apply(lambda x:
sqrt((x['dropoff_latitude'] -
x['pickup_latitude']) ** 2 + (
        x['dropoff_longitude'] -
x['pickup_longitude']) ** 2), axis=1)
print(df)
OutlierData = pd.DataFrame()
temp = df[["distance", "fare_amount",
"passenger_count"]]
for col in ["distance",
"fare_amount","passenger_count"]:
    Q1 = temp[col].quantile(0.25)  # Gives 25th
Percentile or Q1
    Q3 = temp[col].quantile(0.75)  # Gives 75th
Percentile or Q3

    IQR = Q3 - Q1

    UpperBound = Q3 + 1.5 * IQR
    LowerBound = Q1 - 1.5 * IQR

    OutlierData[col] = temp[col][(temp[col] <
LowerBound) | (temp[col] > UpperBound)]
    df_OutlierFree = df.drop(OutlierData.index,
axis=0)
    print(len(OutlierData))
```

```
C:\Users\tejas\PycharmProjects\pythonProject\venv\Scripts\python.exe C:\Users\
tejas\PycharmProjects\pythonProject\START\ol.py
             key  fare_amount  ... passenger_count  distance
0       24238194          7.5  ...               1  0.015140
1       27835199          7.7  ...               1  0.022103
2       44984355         12.9  ...               1  0.053109
3       25894730          5.3  ...               3  0.016528
4       17610152         16.0  ...               5  0.051031
...          ...          ...  ...             ...       ...
199995  42598914          3.0  ...               1  0.001064
199996  16382965          7.5  ...               1  0.022126
199997  27804658         30.9  ...               2  0.142223
199998  20259894         14.5  ...               1  0.033101
199999  11951496         14.1  ...               1  0.048729

[200000 rows x 9 columns]
17344
17344
17344

Process finished with exit code 0
```

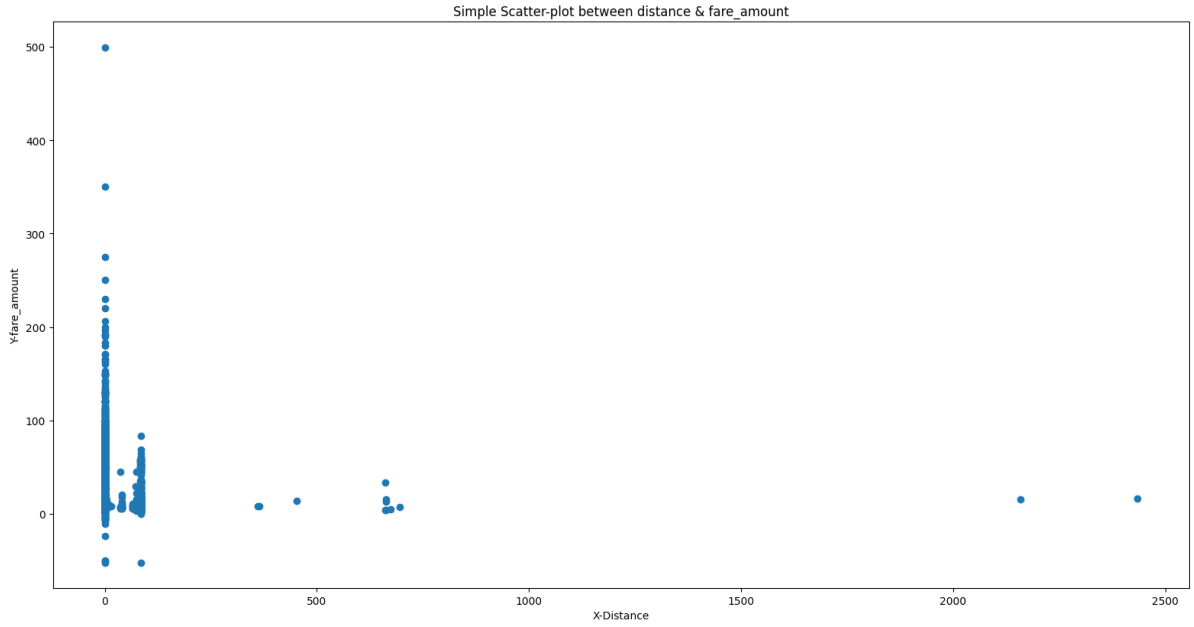
3.Show the scatterplot between *distance* & *fare_amount.* Is there any relationship that you can identify?(Relationship:Non-Linear)

```
import csv
from math import sqrt
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns

df = pd.read_csv('Dataset_Day3.csv')
df['distance'] = df.apply(lambda x:
sqrt((x['dropoff_latitude'] - x['pickup_latitude'])
** 2 + (
            x['dropoff_longitude'] -
x['pickup_longitude']) ** 2), axis=1)
print(df)

plt.scatter(df["distance"], df["fare_amount"])
plt.title('Simple Scatter-plot between distance &
fare_amount')
plt.xlabel('X-Distance')
plt.ylabel('Y-fare_amount')

plt.show()
```


Simple Scatter-plot between distance & fare_amount
Y-fare_amount
X-Distance
500
400
300
200
100
0
0
500
1000
1500
2000
2500