Abon, Benedict Aldous A.

CPE311-CPE22S3

```
import pandas as pd
```

1. Create DataFrame; Examine the first 5 rows

```
data = pd.read_csv('/2019_Yellow_Taxi_Trip_Data.csv')
data.head(5)
```


```
{"summary":"{\n  \"name\": \"data\",\n  \"rows\": 10000,\n
\"fields\": [\n    {\n      \"column\": \"vendorid\",\n
\"properties\": {\n        \"dtype\": \"number\",\n        \"std\":
0,\n        \"min\": 1,\n        \"max\": 2,\n
\"num_unique_values\": 2,\n        \"samples\": [\n          1,\n
2\n        ],\n        \"semantic_type\": \"\",\n
\"description\": \"\"\n      }\n    },\n    {\n      \"column\":
\"tpep_pickup_datetime\",\n      \"properties\": {\n        \"dtype\":
\"object\",\n        \"num_unique_values\": 4281,\n
\"samples\": [\n          \"2019-10-23T16:44:33.000\",\n
\"2019-10-23T16:26:46.000\"\n        ],\n        \"semantic_type\":
\"\",\n        \"description\": \"\"\n      }\n    },\n    {\n
\"column\": \"tpep_dropoff_datetime\",\n      \"properties\": {\n
\"dtype\": \"object\",\n        \"num_unique_values\": 5079,\n
\"samples\": [\n          \"2019-10-23T17:45:43.000\",\n
\"2019-10-23T17:49:28.000\"\n        ],\n        \"semantic_type\":
\"\",\n        \"description\": \"\"\n      }\n    },\n    {\n
\"column\": \"passenger_count\",\n      \"properties\": {\n
\"dtype\": \"number\",\n        \"std\": 1,\n        \"min\": 0,\n
\"max\": 6,\n        \"num_unique_values\": 7,\n        \"samples\":
[\n          1,\n          2\n        ],\n        \"semantic_type\":
\"\",\n        \"description\": \"\"\n      }\n    },\n    {\n
\"column\": \"trip_distance\",\n      \"properties\": {\n
\"dtype\": \"number\",\n        \"std\": 4.148063386748824,\n
\"min\": 0.0,\n        \"max\": 38.11,\n        \"num_unique_values\":
1243,\n        \"samples\": [\n          5.41,\n          10.29\n
],\n        \"semantic_type\": \"\",\n        \"description\": \"\"\n
}\n    },\n    {\n      \"column\": \"ratecodeid\",\n
\"properties\": {\n        \"dtype\": \"number\",\n        \"std\":
0,\n        \"min\": 1,\n        \"max\": 5,\n
\"num_unique_values\": 5,\n        \"samples\": [\n          2,\n
4\n        ],\n        \"semantic_type\": \"\",\n
\"description\": \"\"\n      }\n    },\n    {\n      \"column\":
\"store_and_fwd_flag\",\n      \"properties\": {\n        \"dtype\":
\"category\",\n        \"num_unique_values\": 2,\n        \"samples\":
[\n          \"Y\",\n          \"N\"\n        ],\n
\"semantic_type\": \"\",\n        \"description\": \"\"\n      }\
n    },\n    {\n      \"column\": \"pulocationid\",\n
```

```
\"properties\": {\n        \"dtype\": \"number\",\n        \"std\":
63,\n        \"min\": 1,\n        \"max\": 265,\n
\"num_unique_values\": 129,\n        \"samples\": [\n          24,\n
164\n        ],\n        \"semantic_type\": \"\",\n
\"description\": \"\"\n      }\n    },\n    {\n      \"column\":
\"dolocationid\",\n      \"properties\": {\n        \"dtype\":
\"number\",\n        \"std\": 68,\n        \"min\": 1,\n
\"max\": 265,\n        \"num_unique_values\": 183,\n
\"samples\": [\n          193,\n          249\n        ],\n
\"semantic_type\": \"\",\n        \"description\": \"\"\n      }\
n    },\n    {\n      \"column\": \"payment_type\",\n
\"properties\": {\n        \"dtype\": \"number\",\n        \"std\":
0,\n        \"min\": 1,\n        \"max\": 4,\n
\"num_unique_values\": 4,\n        \"samples\": [\n          2,\n
4\n        ],\n        \"semantic_type\": \"\",\n
\"description\": \"\"\n      }\n    },\n    {\n      \"column\":
\"fare_amount\",\n      \"properties\": {\n        \"dtype\":
\"number\",\n        \"std\": 13.954761757679899,\n        \"min\": -
52.0,\n        \"max\": 176.0,\n        \"num_unique_values\": 199,\n
\"samples\": [\n          42.5,\n          5.0\n        ],\n
\"semantic_type\": \"\",\n        \"description\": \"\"\n      }\
n    },\n    {\n      \"column\": \"extra\",\n      \"properties\": {\
n        \"dtype\": \"number\",\n        \"std\": 1.392940064388004,\n
\"min\": -4.5,\n        \"max\": 7.0,\n        \"num_unique_values\":
12,\n        \"samples\": [\n          2.05,\n          3.0\
n        ],\n        \"semantic_type\": \"\",\n
\"description\": \"\"\n      }\n    },\n    {\n      \"column\":
\"mta_tax\",\n      \"properties\": {\n        \"dtype\": \"number\",\
n        \"std\": 0.07154375829473356,\n        \"min\": -0.5,\n
\"max\": 0.5,\n        \"num_unique_values\": 3,\n        \"samples\":
[\n          0.5,\n          -0.5\n        ],\n
\"semantic_type\": \"\",\n        \"description\": \"\"\n      }\
n    },\n    {\n      \"column\": \"tip_amount\",\n
\"properties\": {\n        \"dtype\": \"number\",\n        \"std\":
3.409800031906561,\n        \"min\": 0.0,\n        \"max\": 43.0,\n
\"num_unique_values\": 657,\n        \"samples\": [\n          16.16,\
n          5.56\n        ],\n        \"semantic_type\": \"\",\n
\"description\": \"\"\n      }\n    },\n    {\n      \"column\":
\"tolls_amount\",\n      \"properties\": {\n        \"dtype\":
\"number\",\n        \"std\": 6.437507127609063,\n        \"min\": -
6.12,\n        \"max\": 612.0,\n        \"num_unique_values\": 37,\n
\"samples\": [\n          10.0,\n          14.65\n        ],\n
\"semantic_type\": \"\",\n        \"description\": \"\"\n      }\
n    },\n    {\n      \"column\": \"improvement_surcharge\",\n
\"properties\": {\n        \"dtype\": \"number\",\n        \"std\":
0.03253684472450732,\n        \"min\": -0.3,\n        \"max\": 0.3,\n
\"num_unique_values\": 3,\n        \"samples\": [\n          0.3,\n
-0.3\n        ],\n        \"semantic_type\": \"\",\n
\"description\": \"\"\n      }\n    },\n    {\n      \"column\":
```

```
\"total_amount\",\n      \"properties\": {\n        \"dtype\":
\"number\",\n        \"std\": 19.209255488783793,\n        \"min\": -
65.92,\n        \"max\": 671.8,\n        \"num_unique_values\": 1097,\
n        \"samples\": [\n          14.16,\n          85.7\n        ],\
n        \"semantic_type\": \"\",\n        \"description\": \"\"\n
}\n    },\n    {\n      \"column\": \"congestion_surcharge\",\n
\"properties\": {\n        \"dtype\": \"number\",\n        \"std\":
0.7209463477704644,\n        \"min\": -2.5,\n        \"max\": 2.75,\n
\"num_unique_values\": 4,\n        \"samples\": [\n          0.0,\n
2.75\n        ],\n        \"semantic_type\": \"\",\n
\"description\": \"\"\n      }\n    }\n  ]\
n}","type":"dataframe","variable_name":"data"}
```


1. Find the dimensions (number of rows and number of columns) in the data.

```
data.shape
```

```
(10000, 18)
```

1. Calculate summary statistics for the fare_amount, tip_amount, tolls_amount, and total_amount columns.

```
# fare amount stats
fa_mean = data['fare_amount'].mean()
fa_median = data['fare_amount'].median()
fa_mode = data['fare_amount'].mode()
fa_std = data['fare_amount'].std()
# print stats
print(f'Fare amount mean: {fa_mean}')
print(f'Fare amount median: {fa_median}')
print(f'Fare amount mode: {fa_mode[0]}')
print(f'Fare amount standard deviation: {fa_std}')
```

```
Fare amount mean: 15.106313
Fare amount median: 10.0
Fare amount mode: 52.0
Fare amount standard deviation: 13.954761757679899
```

```
# tip_amount stats
ta_mean = data['tip_amount'].mean()
ta_median = data['tip_amount'].median()
ta_mode = data['tip_amount'].mode()
ta_std = data['tip_amount'].std()
# print stats
print(f'Tip amount mean:{ta_mean}')
print(f'Tip amount median: {ta_median}')
print(f'Tip amount mode: {ta_mode[0]}')
print(f'Tip amount standard deviation: {ta_std}')
```

```
Tip amount:2.6344939999999997
Tip amount median: 2.0
```

```
Tip amount mode: 0.0
Tip amount standard deviation: 3.409800031906561

# tolls_amount stats
taa_mean = data['tolls_amount'].mean()
taa_median = data['tolls_amount'].median()
taa_mode = data['tolls_amount'].mode()
taa_std = data['tolls_amount'].std()
# print stats
print(f'Tolls amount mean: {taa_mean}')
print(f'Tolls amount median: {taa_median}')
print(f'Tolls amount mode: {taa_mode[0]}')
print(f'Tolls amount standard deviation: {taa_std}')

Tolls amount: 0.623447
Tolls amount median: 0.0
Tolls amount: 0.0
Tolls amount standard deviation: 6.437507127609063

# total_amount stats
taat_mean = data['total_amount'].mean()
taat_median = data['total_amount'].median()
taat_mode = data['total_amount'].mode()
taat_std = data['total_amount'].std()
# print stats
print(f'Total amount mean: {taat_mean}')
print(f'Total amount median: {taat_median}')
print(f'Total amount mode: {taat_mode[0]}')
print(f'Total amount standard deviation: {taat_std}')

Total amount mean: 22.564659
Total amount median: 16.3
Total amount mode: 11.8
Total amount standard deviation: 19.209255488783793
```

1. Isolate the fare_amount, tip_amount, tolls_amount, and total_amount for the longest trip by distance (trip_distance).

```
data.sort_values(by='trip_distance', ascending=False).head(1)

{"repr_error":"0","type":"dataframe"}
```