```
import pandas as pd
import csv


# 1. Read Data from CSV File into Python List
csv_data = []
with open('orders.csv', mode='r') as csvfile:
    reader = csv.DictReader(csvfile)
    csv_data = [row for row in reader]

csv_data
```


```
[{'order_id': '1',
  'customer_id': '101',
  'order_date': '2023-01-15',
  'status': 'Completed',
  'amount': '250.00'},
 {'order_id': '2',
  'customer_id': '102',
  'order_date': '2023-02-20',
  'status': 'Pending',
  'amount': '150.00'},
 {'order_id': '3',
  'customer_id': '103',
  'order_date': '2023-02-22',
  'status': 'Completed',
  'amount': '200.00'},
 {'order_id': '4',
  'customer_id': '101',
  'order_date': '2023-03-10',
  'status': 'Canceled',
  'amount': '100.00'},
 {'order_id': '5',
  'customer_id': '104',
  'order_date': '2023-03-12',
  'status': 'Completed',
  'amount': '300.00'},
 {'order_id': '6',
  'customer_id': '102',
  'order_date': '2023-04-05',
  'status': 'Pending',
  'amount': '120.00'},
 {'order_id': '7',
  'customer_id': '105',
  'order_date': '2023-04-06',
  'status': 'Completed',
  'amount': '180.00'},
 {'order_id': '8',
  'customer_id': '103',
  'order_date': '2023-04-15',
  'status': 'Canceled',
  'amount': '220.00'}]
```


```
data = pd.read_csv('orders.csv')
data
```

	order_id	customer_id	order_date	status	amount
0	1	101	2023-01-15	Completed	250.0
1	2	102	2023-02-20	Pending	150.0
2	3	103	2023-02-22	Completed	200.0
3	4	101	2023-03-10	Canceled	100.0
4	5	104	2023-03-12	Completed	300.0
5	6	102	2023-04-05	Pending	120.0
6	7	105	2023-04-06	Completed	180.0
7	8	103	2023-04-15	Canceled	220.0

Next steps: Generate code with data | View recommended plots | New interactive sheet

```
import numpy as np
import matplotlib.pyplot as plt


s = pd.Series(csv_data)
s
```

	0
0	{'order_id': '1', 'customer_id': '101', 'order...
1	{'order_id': '2', 'customer_id': '102', 'order...
2	{'order_id': '3', 'customer_id': '103', 'order...
3	{'order_id': '4', 'customer_id': '101', 'order...
4	{'order_id': '5', 'customer_id': '104', 'order...
5	{'order_id': '6', 'customer_id': '102', 'order...
6	{'order_id': '7', 'customer_id': '105', 'order...
7	{'order_id': '8', 'customer_id': '103', 'order...

```
df = pd.DataFrame(csv_data)
df
```

	order_id	customer_id	order_date	status	amount
0	1	101	2023-01-15	Completed	250.00
1	2	102	2023-02-20	Pending	150.00
2	3	103	2023-02-22	Completed	200.00
3	4	101	2023-03-10	Canceled	100.00
4	5	104	2023-03-12	Completed	300.00
5	6	102	2023-04-05	Pending	120.00
6	7	105	2023-04-06	Completed	180.00
7	8	103	2023-04-15	Canceled	220.00

Next steps: Generate code with df View recommended plots New interactive sheet

```
df.dtypes
```

	0
order_id	object
customer_id	object
order_date	object
status	object
amount	object

```
# viewing top 5 data using head
df.head()
```

	order_id	customer_id	order_date	status	amount
0	1	101	2023-01-15	Completed	250.00
1	2	102	2023-02-20	Pending	150.00
2	3	103	2023-02-22	Completed	200.00
3	4	101	2023-03-10	Canceled	100.00
4	5	104	2023-03-12	Completed	300.00

Next steps: Generate code with df View recommended plots New interactive sheet

```
# viewing from last

df.tail(4)
```

	order_id	customer_id	order_date	status	amount
4	5	104	2023-03-12	Completed	300.00
5	6	102	2023-04-05	Pending	120.00
6	7	105	2023-04-06	Completed	180.00
7	8	103	2023-04-15	Canceled	220.00

```
df.index
df.columns
```

```
Index(['order_id', 'customer_id', 'order_date', 'status', 'amount'], dtype='object')
```

```
df.values
```

```
array([['1', '101', '2023-01-15', 'Completed', '250.00'],
       ['2', '102', '2023-02-20', 'Pending', '150.00'],
       ['3', '103', '2023-02-22', 'Completed', '200.00'],
       ['4', '101', '2023-03-10', 'Canceled', '100.00'],
       ['5', '104', '2023-03-12', 'Completed', '300.00'],
       ['6', '102', '2023-04-05', 'Pending', '120.00'],
       ['7', '105', '2023-04-06', 'Completed', '180.00'],
       ['8', '103', '2023-04-15', 'Canceled', '220.00']], dtype=object)
```

```
df.describe()
```

	order_id	customer_id	order_date	status	amount
count	8	8	8	8	8
unique	8	5	8	3	8
top	1	101	2023-01-15	Completed	250.00
freq	1	2	1	4	1

```
# Transposing
tr = df.T
tr
```

	0	1	2	3	4	5	6	7
order_id	1	2	3	4	5	6	7	8
customer_id	101	102	103	101	104	102	105	103
order_date	2023-01-15	2023-02-20	2023-02-22	2023-03-10	2023-03-12	2023-04-05	2023-04-06	2023-04-15
status	Completed	Pending	Completed	Canceled	Completed	Pending	Completed	Canceled
amount	250.00	150.00	200.00	100.00	300.00	120.00	180.00	220.00

Next steps: Generate code with tr | View recommended plots | New interactive sheet

```
sortasc = df.sort_index(axis=1, ascending=False)
sortasc
```

	status	order_id	order_date	customer_id	amount
0	Completed	1	2023-01-15	101	250.00
1	Pending	2	2023-02-20	102	150.00
2	Completed	3	2023-02-22	103	200.00
3	Canceled	4	2023-03-10	101	100.00
4	Completed	5	2023-03-12	104	300.00
5	Pending	6	2023-04-05	102	120.00
6	Completed	7	2023-04-06	105	180.00
7	Canceled	8	2023-04-15	103	220.00

Next steps: Generate code with sortasc | View recommended plots | New interactive sheet

```
sortdes = df.sort_index(axis=1, ascending=True)
sortdes
```

	amount	customer_id	order_date	order_id	status
0	250.00	101	2023-01-15	1	Completed
1	150.00	102	2023-02-20	2	Pending
2	200.00	103	2023-02-22	3	Completed
3	100.00	101	2023-03-10	4	Canceled
4	300.00	104	2023-03-12	5	Completed
5	120.00	102	2023-04-05	6	Pending
6	180.00	105	2023-04-06	7	Completed
7	220.00	103	2023-04-15	8	Canceled

Next steps: Generate code with sortdes | View recommended plots | New interactive sheet

```
sorted = df.sort_values(by='amount')
sorted
```

	order_id	customer_id	order_date	status	amount
3	4	101	2023-03-10	Canceled	100.00
5	6	102	2023-04-05	Pending	120.00
1	2	102	2023-02-20	Pending	150.00
6	7	105	2023-04-06	Completed	180.00
2	3	103	2023-02-22	Completed	200.00
7	8	103	2023-04-15	Canceled	220.00
0	1	101	2023-01-15	Completed	250.00
4	5	104	2023-03-12	Completed	300.00

Next steps: Generate code with sorted | View recommended plots | New interactive sheet

```
df['amount']
```

	amount
0	250.00
1	150.00
2	200.00
3	100.00
4	300.00
5	120.00
6	180.00
7	220.00

```
df[2:4]
```

	order_id	customer_id	order_date	status	amount
2	3	103	2023-02-22	Completed	200.00
3	4	101	2023-03-10	Canceled	100.00

```
df.loc[[2,3]]
```

	order_id	customer_id	order_date	status	amount
2	3	103	2023-02-22	Completed	200.00
3	4	101	2023-03-10	Canceled	100.00

```
df.iloc[3]
```

	3
order_id	4
customer_id	101
order_date	2023-03-10
status	Canceled
amount	100.00

```
df['amount'] = pd.to_numeric(df['amount'], errors='coerce')
filtered_df = df[df['amount'] > 150]
print(filtered_df)
```

```
   order_id customer_id  order_date     status  amount
0         1         101  2023-01-15  Completed   250.0
2         3         103  2023-02-22  Completed   200.0
4         5         104  2023-03-12  Completed   300.0
6         7         105  2023-04-06  Completed   180.0
7         8         103  2023-04-15   Canceled   220.0
```

```
pd.isnull(df)
```

	order_id	customer_id	order_date	status	amount
0	False	False	False	False	False
1	False	False	False	False	False
2	False	False	False	False	False
3	False	False	False	False	False
4	False	False	False	False	False
5	False	False	False	False	False
6	False	False	False	False	False
7	False	False	False	False	False

```
df['amount'].mean()
```

```
190.0
```

```
import matplotlib.pyplot as plt

# Sample data
data = {
    'order_id': [1, 3, 5, 7, 8, 9, 10],
    'customer_id': [101, 103, 104, 105, 103, 106, 107],
    'order_date': pd.to_datetime(['2023-01-15', '2023-02-22', '2023-03-12', '2023-04-06', '2023-04-15', '2023-04-18', '2023-05-10']),
    'status': ['Completed', 'Completed', 'Completed', 'Completed', 'Canceled', 'Completed', 'Pending'],
    'amount': [250.0, 200.0, 300.0, 180.0, 220.0, 275.0, 150.0]
}
df = pd.DataFrame(data)

# 1. Total Amount per Order Status
plt.figure(figsize=(8, 5))
df.groupby('status')['amount'].sum().plot(kind='bar', color=['blue', 'red', 'green'])
plt.title('Total Amount per Order Status')
plt.xlabel('Order Status')
plt.ylabel('Total Amount')
plt.show()

# 2. Orders over Time
plt.figure(figsize=(10, 6))
df.groupby('order_date').size().plot(kind='line', marker='o', color='purple')
plt.title('Number of Orders Over Time')
plt.xlabel('Order Date')
plt.ylabel('Number of Orders')
plt.grid()
plt.show()

# 3. Amount per Order by Date
plt.figure(figsize=(10, 6))
plt.plot(df['order_date'], df['amount'], marker='o', color='teal', linestyle='-', linewidth=2)
plt.title('Amount per Order by Date')
plt.xlabel('Order Date')
plt.ylabel('Order Amount')
```

```
plt.ylabel('Order Amount')
plt.grid()
plt.show()

# 4. Distribution of Order Amounts
plt.figure(figsize=(8, 5))
plt.hist(df['amount'], bins=5, color='orange', edgecolor='black')
plt.title('Distribution of Order Amounts')
plt.xlabel('Order Amount')
plt.ylabel('Frequency')
plt.show()
```