

PYTHON LAB - 28

PANDAS DATA ANALYSIS

NAME: Keerthana K R

ID: AF0363623

QUESTIONS

1. Write a Pandas program to split the following data frame into groups based on Class and count the number of students in that particular class. Also generate a bar chart based on the result and explain the conclusion. Input:
2. Write a Pandas program to split the following dataframe by school code and get mean, min, and max value of age for each school. Also generate a horizontal bar chart based on the result and explain the conclusion.
3. Write a Pandas program to split a dataset, group by one column and get mean, min, and max values by group. Using the following dataset find the mean, min, and max values of purchase amount (purch_amt) group by customer id (customer_id). Also generate a line chart based on the result and explain the conclusion. Input:
4. Write a Pandas program to split the following data frame into groups and calculate monthly purchase amount. Also generate a bar chart based on the result and explain the conclusion.

1. Write a Pandas program to split the following data frame into groups based on Class and count the number of students in that particular class. Also generate a bar chart based on the result and explain the conclusion. Input:
student_data = pd.DataFrame({ 'school_code': ['s001', 's002', 's003', 's001', 's002','s004'], 'class': ['V', 'VI', 'VI', 'VI', 'V', 'VI'], 'name': ['Alberto Franco','Gino Mcneill','Ryan Parkes', 'Eesha Hinton', 'Gino Mcneill', 'David Parkes'], 'age': [12, 12, 13, 13, 14, 12], 'height': [173, 192, 186, 167, 151, 159], 'weight': [35, 32, 33, 30, 31, 32], 'address': ['street1', 'street2', 'street3', 'street1', 'street2', 'street4']})

Code:

```
# Import necessary packages
import pandas as pd
import matplotlib.pyplot as plt

# Define the DataFrame
student_data = pd.DataFrame({
    'school_code': ['s001', 's002', 's003', 's001',
's002', 's004'],
    'class': ['V', 'VI', 'VI', 'VI', 'V', 'VI'],
    'name': ['Alberto Franco', 'Gino Mcneill', 'Ryan
Parkes', 'Eesha Hinton', 'Gino Mcneill', 'David Parkes'],
    'age': [12, 12, 13, 13, 14, 12],
    'height': [173, 192, 186, 167, 151, 159],
    'weight': [35, 32, 33, 30, 31, 32],
    'address': ['street1', 'street2', 'street3',
'street1', 'street2', 'street4']
})

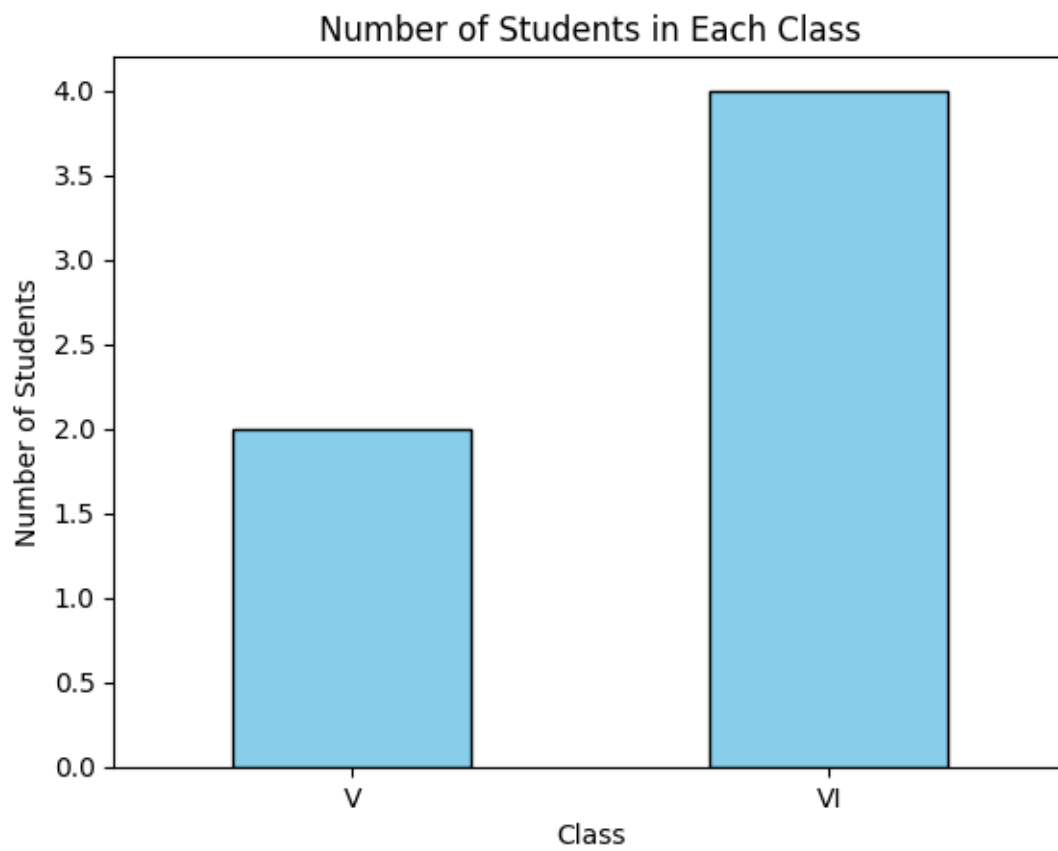
# Grouping the DataFrame by 'class' and counting the
number of students in each class
class_counts = student_data.groupby('class').size()

# Plotting the bar chart
class_counts.plot(kind='bar', color='skyblue',
edgecolor='black')

# Adding labels and title
plt.xlabel('Class')
plt.ylabel('Number of Students')
plt.title('Number of Students in Each Class')

# Display the plot
plt.xticks(rotation=0)
plt.show()
```

Output:



2. Write a Pandas program to split the following dataframe by school code and get mean, min, and max value of age for each school. Also generate a horizontal bar chart based on the result and explain the conclusion. Input:
- ```
student_data = pd.DataFrame({ 'school_code': ['s001', 's002', 's003', 's001', 's002', 's004'], 'class': ['V', 'V', 'VI', 'VI', 'V', 'VI'], 'name': ['Alberto Franco', 'Gino Mcneill', 'Ryan Parkes', 'Eesha Hinton', 'Gino Mcneill', 'David Parkes'], 'age': [12, 12, 13, 13, 14, 12], 'height': [173, 192, 186, 167, 151, 159], 'weight': [35, 32, 33, 30, 31, 32], 'address': ['street1', 'street2', 'street3', 'street1', 'street2', 'street4']})
```

### Code:

```
import pandas as pd
import matplotlib.pyplot as plt

Define the DataFrame
student_data = pd.DataFrame({
 'school_code': ['s001', 's002', 's003', 's001', 's002', 's004'],
 'class': ['V', 'V', 'VI', 'VI', 'V', 'VI'],
 'name': ['Alberto Franco', 'Gino Mcneill', 'Ryan Parkes', 'Eesha Hinton', 'Gino Mcneill', 'David Parkes'],
 'age': [12, 12, 13, 13, 14, 12],
 'height': [173, 192, 186, 167, 151, 159],
 'weight': [35, 32, 33, 30, 31, 32],
 'address': ['street1', 'street2', 'street3', 'street1', 'street2', 'street4']
})

Grouping the DataFrame by 'school_code' and calculating mean, min, and max age for each school
school_age = student_data.groupby('school_code')['age'].agg(['mean', 'min', 'max'])

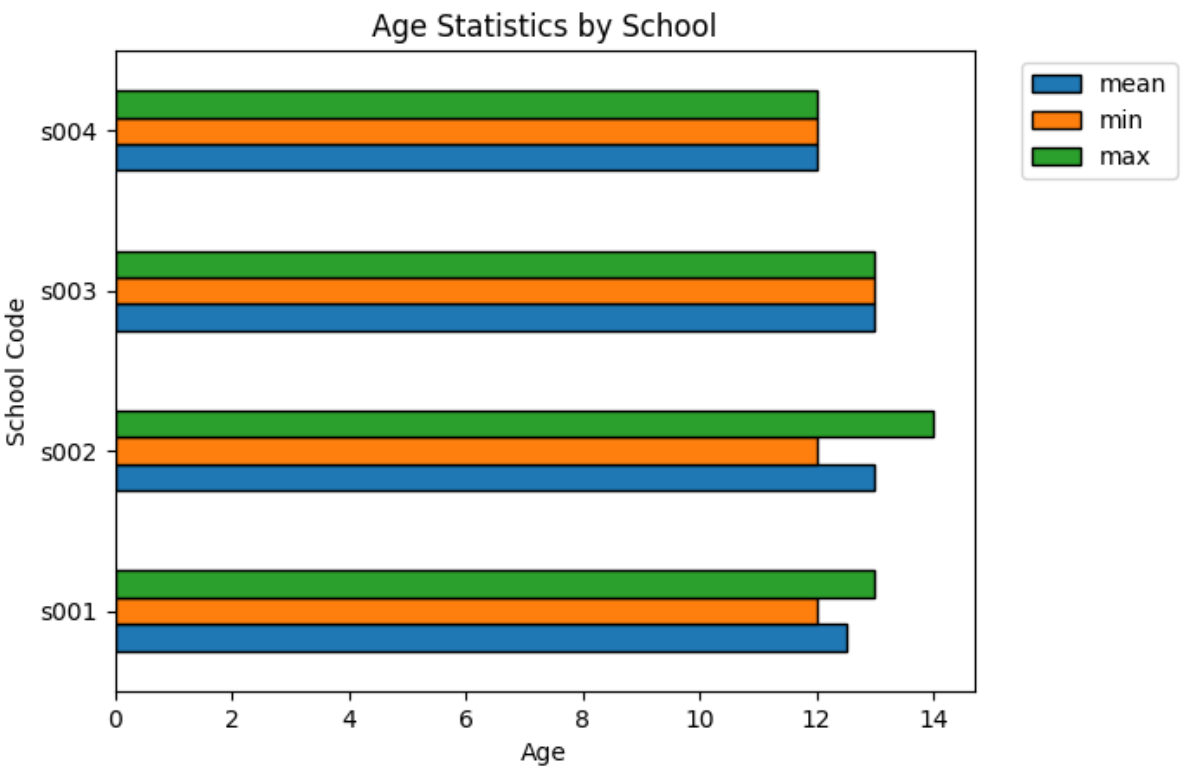
Plotting the horizontal bar chart
school_age.plot(kind='barh', edgecolor='black')

Adding labels and title
plt.xlabel('Age')
plt.ylabel('School Code')
plt.title('Age Statistics by School')

Adding legend
plt.legend(loc='upper right', bbox_to_anchor=(1.25, 1))

Display the plot
plt.show()
```

Output:



3. Write a Pandas program to split a dataset, group by one column and get mean, min, and max values by group. Using the following dataset find the mean, min, and max values of purchase amount (purch\_amt) group by customer id (customer\_id). Also generate a line chart based on the result and explain the conclusion. Input:

```
orders_data = pd.DataFrame({ 'ord_no': [70001, 70009, 70002, 70004, 70007, 70005, 70008, 70010, 70003, 70012, 70011, 70013], 'purch_amt': [150.5, 270.65, 65.26, 110.5, 948.5, 2400.6, 5760, 1983.43, 2480.4, 250.45, 75.29, 3045.6], 'ord_date': ['2012-10-05', '2012-09-10', '2012-10-05', '2012-08-17', '2012-09-10', '2012-07-27', '2012-09-10', '2012-10-10', '2012-10-10', '2012-06-27', '2012-08-17', '2012-04-25'], 'customer_id': [3005, 3001, 3002, 3009, 3005, 3007, 3002, 3004, 3009, 3008, 3003, 3002], 'salesman_id': [5002, 5005, 5001, 5003, 5002, 5001, 5001, 5006, 5003, 5002, 5007, 5001]})
```

### Code:

```
import pandas as pd
import matplotlib.pyplot as plt

Define the DataFrame
orders_data = pd.DataFrame({
 'ord_no': [70001, 70009, 70002, 70004, 70007, 70005, 70008, 70010, 70003, 70012, 70011, 70013],
 'purch_amt': [150.5, 270.65, 65.26, 110.5, 948.5, 2400.6, 5760, 1983.43, 2480.4, 250.45, 75.29, 3045.6],
 'ord_date': ['2012-10-05', '2012-09-10', '2012-10-05', '2012-08-17', '2012-09-10', '2012-07-27', '2012-09-10', '2012-10-10', '2012-10-10', '2012-06-27', '2012-08-17', '2012-04-25'],
 'customer_id': [3005, 3001, 3002, 3009, 3005, 3007, 3002, 3004, 3009, 3008, 3003, 3002],
 'salesman_id': [5002, 5005, 5001, 5003, 5002, 5001, 5001, 5006, 5003, 5002, 5007, 5001]})

Grouping by 'customer id' and calculating mean, min, and max of purchase amount
customer_amt = orders_data.groupby('customer_id')['purch_amt'].agg(['mean', 'min', 'max'])

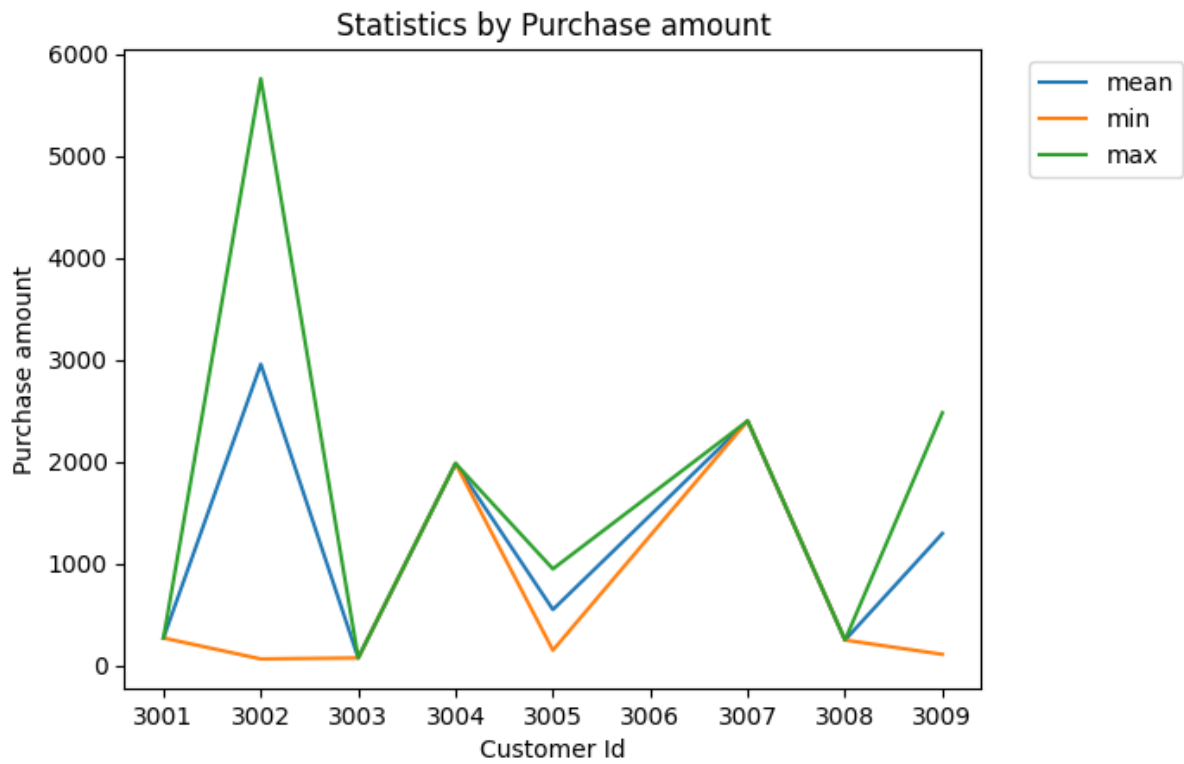
Plotting the horizontal bar chart
customer_amt.plot()

Adding labels and title
plt.xlabel('Customer Id')
plt.ylabel('Purchase amount')
plt.title('Statistics by Purchase amount')

Adding legend
plt.legend(loc='upper right', bbox_to_anchor=(1.25, 1))
```

```
Display the plot
plt.show()
```

**Output:**





4. Write a Pandas program to split the following data frame into groups and calculate monthly purchase amount. Also generate a bar chart based on the result and explain the conclusion. Input:

```
df = pd.DataFrame({ 'ord_no': [70001, 70009, 70002, 70004, 70007, 70005, 70008, 70010, 70003, 70012, 70011, 70013], 'purch_amt': [150.5, 270.65, 65.26, 110.5, 948.5, 2400.6, 5760, 1983.43, 2480.4, 250.45, 75.29, 3045.6], 'ord_date': ['05-10-2012', '09-10-2012', '05-10-2012', '08-17-2012', '10-09-2012', '07-27-2012', '10-09-2012', '10-10-2012', '10-10-2012', '06-17-2012', '07-08-2012', '04-25-2012'], 'customer_id': [3001, 3001, 3005, 3001, 3005, 3001, 3005, 3001, 3005, 3001, 3005, 3001, 3005, 3001, 3005, 3001, 3005, 3005]})
```

### Code:

```
import pandas as pd
import matplotlib.pyplot as plt

Define the DataFrame
df = pd.DataFrame({
 'ord_no': [70001, 70009, 70002, 70004, 70007, 70005, 70008, 70010, 70003, 70012, 70011, 70013],
 'purch_amt': [150.5, 270.65, 65.26, 110.5, 948.5, 2400.6, 5760, 1983.43, 2480.4, 250.45, 75.29, 3045.6],
 'ord_date': ['05-10-2012', '09-10-2012', '05-10-2012', '08-17-2012', '10-09-2012', '07-27-2012', '10-09-2012', '10-10-2012', '10-10-2012', '06-17-2012', '07-08-2012', '04-25-2012'],
 'customer_id': [3001, 3001, 3005, 3001, 3005, 3001, 3005, 3001, 3005, 3001, 3005, 3001, 3005, 3001, 3005, 3001, 3005, 3005]})

Convert 'ord_date' to datetime format
df['ord_date'] = pd.to_datetime(df['ord_date'], format='%m-%d-%Y')

Extract month from 'ord_date'
df['month'] = df['ord_date'].dt.month

Group by month and calculate monthly purchase amount
monthly_purchase = df.groupby('month')['purch_amt'].sum()

Plot the bar chart
monthly_purchase.plot(kind='bar', edgecolor='black')

Adding labels and title
plt.xlabel('Month')
plt.ylabel('Total Purchase Amount')
plt.title('Monthly Purchase Amount')

Display the plot
plt.xticks(rotation=0)
plt.show()
```

**Output:**

