Create a Dataframe with following data

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| slno | F-name | Type | Department | YOE | Salary |
| 0 | Tejas K S | H R | CS | 9 | 40000 |
| 1 | Roshana | IT | ME | 5 | 50000 |
| 2 | Sangeetha | Manager | CE | 7 | 80000 |
| 3 | Nikhil Gowda | H R | EC | 9 | 70000 |
| 4 | Umesh H N | Developer | CS | 5 | 90000 |
| 5 | Jeevan | IT | ME | 9 | 38000 |

1. Make pivot table which source average salary of each type of employee for each Department
2. Make a pivot table which source sum and mean() the salary of each type of employee and the number of employee of each type
3. Make pivot Table which source a standard deviation for salary column

#import pandas module to make DataFrame

import pandas as pd

data={'Employee':['Tejas K S','Roshana','Sangeetha','Nikhil Gowda','Umesh H N','Jeevan'],

'Department':['H R','IT','Manager','H R','Developer','IT'],

'Type':['CS','ME','CE','EC','CS','ME'],

'Salary':[40000,50000,80000,70000,90000,38000,]}

df=pd.DataFrame(data)

print(df)

Output

Employee Department Type Salary

0 Tejas K S H R CS 40000

1 Roshana IT ME 50000

2 Sangeetha Manager CE 80000

3 Nikhil Gowda H R EC 70000

4 Umesh H N Developer CS 90000

5 Jeevan IT ME 38000

1. Make pivot table which source average salary of each type of employee for each Department

avg\_sal\_pivote=df.pivot\_table(values='Salary', index='Department', columns='Type', aggfunc='mean')

print(avg\_sal\_pivote)

Output

Type CE CS EC ME

Department

Developer NaN 90000.0 NaN NaN

H R NaN 40000.0 70000.0 NaN

IT NaN NaN NaN 44000.0

Manager 80000.0 NaN NaN NaN

1. Make a pivot table which shows sum() and mean() the salary of each type of employee and the number of employee of each type

summary\_pivot=df.pivot\_table(values="Salary",index="Type",aggfunc=["sum","mean","count"])

print(summary\_pivot)

Output

sum mean count

Salary Salary Salary

Type

CE 80000 80000 1

CS 130000 65000 2

EC 70000 70000 1

ME 88000 44000 2

1. Make pivot Table which source a standard deviation for salary column

standard\_pivot=df.pivot\_table(values="Salary",index="Type",aggfunc='std')

print(standard\_pivot)

Output

Salary

Type

CS 35355.339059

*ME 8485.281374*

Create two Series as shown using pandas Series() function

Series\_A=[10,20,30,40,50,60]

Series\_B=[40,50,60,70,80,90]

a) Get the item not common in Series\_B

b) Identify the smallest and largest elements in Series\_A

c) Find the sum of Series\_B

D) Calculate average in a Series\_B

# Get the item not common in Series\_B

import pandas as pd

Series\_A=[10,20,30,40,50,60]

Series\_B=[40,50,60,70,80,90]

Series\_AA=set(Series\_A)

Series\_BB=set(Series\_B)

not\_common=Series\_AA.difference(Series\_BB)

print(not\_common)

Output

{10, 20, 30}

# Identify the smallest and largest elements in Series\_A

smallest=min(Series\_A)

largest=max(Series\_A)

print("Min value:",smallest)

print("maximum Value:",largest)

Output

Min value: 10

maximum Value: 60

# Find the sum of Series\_B

sum\_b=sum(Series\_B)

print("Sum of Series\_b:",sum\_b)

Output

Sum of Series\_b: 390

# Calculate average in a Series\_B

import statistics as st

avg=st.mean(Series\_B)

print("Average of Series\_B:",avg)

Output

Average of Series\_B: 65

Perform the following operations on car manufacturing company dataset auto-mpg.csv given below using pandas

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| mpg | cylinders | displacement | HP | Weight | acceleration | Model year | car name |
| 10 | 8 | 307 | 120 | 2800 | 12 | 2000 | Santro |
| 15 | 8 | 398 | 180 | 4200 | 12.9 | 2004 | Maruthi |
| 15 | 4 | 360 | 140 | 3800 | 11.4 | 1998 | Inovva |
| 18 | 8 | 348 | 152 | 4000 | 8.7 | 1995 | Punch |
| 17 | 4 | 321 | 185 | 4000 | 7.6 | 2010 | Nexon |
| 16 | 8 | 374 | 400 | 7000 | 18.5 | 2022 | Fortuner |
| 15 | 4 | 329 | 350 | 6500 | 17.2 | 2018 | Fiat |

1. Read data From existing File
2. Get statistical details of dataset
3. Get all car with 8 cylinders
4. Get the number of car manufactured in each year

# Read data From existing File

import pandas as pd

df=pd.read\_csv("Desktop/auto-mpg.csv")

print(df)

Output

mpg cylinders displacement HP Weight acceleration Model year \

0 10 8 307 120 2800 12.0 2000

1 15 8 398 180 4200 12.9 2004

2 15 4 360 140 3800 11.4 1998

3 18 8 348 152 4000 8.7 1995

4 17 4 321 185 4000 7.6 2010

5 16 8 374 400 7000 18.5 2022

6 15 4 329 350 6500 17.2 2018

car name

0 Santro

1 Maruthi

2 Inovva

3 Punch

4 Nexon

5 Fortuner

6 Fiat

# Get statistical details of dataset

stats=df.describe()

print(stats)

Output

mpg cylinders displacement HP Weight \

count 7.000000 7.000000 7.000000 7.000000 7.000000

mean 15.142857 6.285714 348.142857 218.142857 4614.285714

std 2.544836 2.138090 31.861307 110.396903 1534.523348

min 10.000000 4.000000 307.000000 120.000000 2800.000000

25% 15.000000 4.000000 325.000000 146.000000 3900.000000

50% 15.000000 8.000000 348.000000 180.000000 4000.000000

75% 16.500000 8.000000 367.000000 267.500000 5350.000000

max 18.000000 8.000000 398.000000 400.000000 7000.000000

acceleration Model year

count 7.000000 7.000000

mean 12.614286 2006.714286

std 4.042866 10.307187

min 7.600000 1995.000000

25% 10.050000 1999.000000

50% 12.000000 2004.000000

75% 15.050000 2014.000000

max 18.500000 2022.000000

# Get all car with 8 cylinders

eight\_cylinder=df[df["cylinders"]==8]

print(eight\_cylinder)

Output

mpg cylinders displacement HP Weight acceleration Model year \

0 10 8 307 120 2800 12.0 2000

1 15 8 398 180 4200 12.9 2004

3 18 8 348 152 4000 8.7 1995

5 16 8 374 400 7000 18.5 2022

car name

0 Santro

1 Maruthi

3 Punch

5 Fortuner

# Get the number of car manufactured in each year

car\_by\_year=df.groupby("Model\_year")["car\_name"].count()

print(car\_by\_year)

Output

Model\_year

1995 1

1998 1

2000 1

2004 1

2010 1

2018 1

2022 1

Name: car\_name, dtype: int64