

▼ Day4_Exc_Key

Question 1

Write Pandas program to read the given “employees.csv” file, then accomplish the following functions:

- Replace all the values presented in the “Senior Management” such that, TRUE> T, FALSE> F, and the missing value>NA.
- By mapping, add a new column “Training” such that the employees having “C” or “PF” positions are eligible to attend the training.
- Use cut() method to generate three intervals [‘low’, ‘average’, and ‘high] according to the “Salary”. Show the details resulted from this binning.
- Figure out the results with using qcut() method.

#Start by uploading the file to Colab

```
from google.colab import files
uploaded = files.upload()
```

```
# Import the needed library
import pandas as pd
```

```
# Read the uploaded file to a new data frame
dataset = pd.read_csv("employees.csv")
```

```
#Replace all the values presented in the “Senior Management”
#such that, TRUE> T, FALSE> F, and the missing value>NA.
```

```
#first create the required mapping
mapping = {
    True: 'T',
    False: 'F',
    None: 'NA'
}
```

```
dataset['Senior Management'].replace(mapping,inplace=True) # use inplace=True to make the change
```

```
# By mapping, add a new column “Training”
# such that the employees having “C” or “PF” positions
# are eligible to attend the training.
```

```
# Import NumPy Library to include the needed functions
import numpy as np
```

```
# Create a new column 'Training' and
# assign the values according to the given condition
dataset['Training'] = np.where(dataset['Position'].isin({'C', 'PF'}), 'Eligible', 'Not Eligible')
```

```
# Use cut() method to generate three intervals [‘low’, ‘average’, and ‘high]
# according to the “Salary”.
```

```
bins=3;
bins_Lables = ['Low', 'Average', 'High']

categ = pd.cut(dataset['Salary'], bins, labels= bins_Lables)
categ.value_counts()
```

#Figure out the results with using qcut() method.

```
qcateg = pd.qcut(dataset['Salary'], bins, labels= bins_Lables)

print("\n Using cut() function \n",categ.value_counts())
print("\n Using qcut() function \n", qcateg.value_counts())
```

Question 2 Write Pandas program to read the salaries file given in the following link and accomplish the tasks listed below:

data URL = <http://vincentarelbundock.github.io/Rdatasets/csv/carData/Salaries.csv> (Note: You may read the link directly using pandas.read_csv()) The given dataset has the following columns (rank, discipline, yrs.since.phd, yrs.service, sex and salary).

- Group the data according to the “rank” and show the generated groups.
- Count the number of rows in each group.
- Check if there is any missing data in the datasets.
- Find out the number of unique values in each group.
- Calculate the mean, median, minimum and maximum salary, by groups, using the agg method. Rename all the generated columns so you capitalize each word.
- Do the grouping by considering three columns, “rank”, “discipline” and “sex” using groupby. Then show the size and count of each group.

```
# Import the needed libararies
import pandas as pd
```

```
# Read the given file using the URL
```

```
url_data = 'http://vincentarelbundock.github.io/Rdatasets/csv/carData/Salaries.csv'
datasetQ2 = pd.read_csv(url_data)
```

```
#Group the data according to the “rank” and show the generated groups.
```

```
rankgroups = datasetQ2.groupby("rank")
rankgroups.groups
```

```
# Count the number of rows in each group.
rankgroups.count()
```

```
# Check if there is any missing data in the datasets.
#from the execution of the above command, the results show equal number for the values present
```

```
#Find out the number of unique values in each group.
rankgroups.nunique()
```

```
# Calculate the mean, median, minimum and maximum salary, by groups,
```

```
# using the agg method.
#Rename all the generated columns so you capitalize each word.

rankgroups['salary'].agg([np.mean,np.median,np.min,np.max]).rename(
    columns={'mean':'Mean', 'median': 'Median', 'amin': 'Min', 'amax': 'max'})

# Do the grouping by considering three columns, "rank", "discipline" and "sex" using groupby.
# Then show the size and count of each group.

multilevels= datasetQ2.groupby(['rank', 'discipline', 'sex'])

print(multilevels.size())
print("\n\n\n",multilevels.count())
```