Name : Kajal Jitendra Pawar

PRN : 2019BTECS00010

ASSIGNMENT – 1

1) Weka is a GUI workbench that empowers data wranglers to assemble machine learning pipelines, train models, and run predictions without having to write code. Using Weka tool perform below tasks such as data preprocessing, data classification (use any appropriate ML algorithm) and data visualization efficiently on given dataset. Use the Iris dataset given https://drive.google.com/file/d/1A3Fxsfzm6BSfhFZGDrjI47RTe45bSgYP/view

   A) Installing weka from the link –
   https://sourceforge.net/projects/weka/

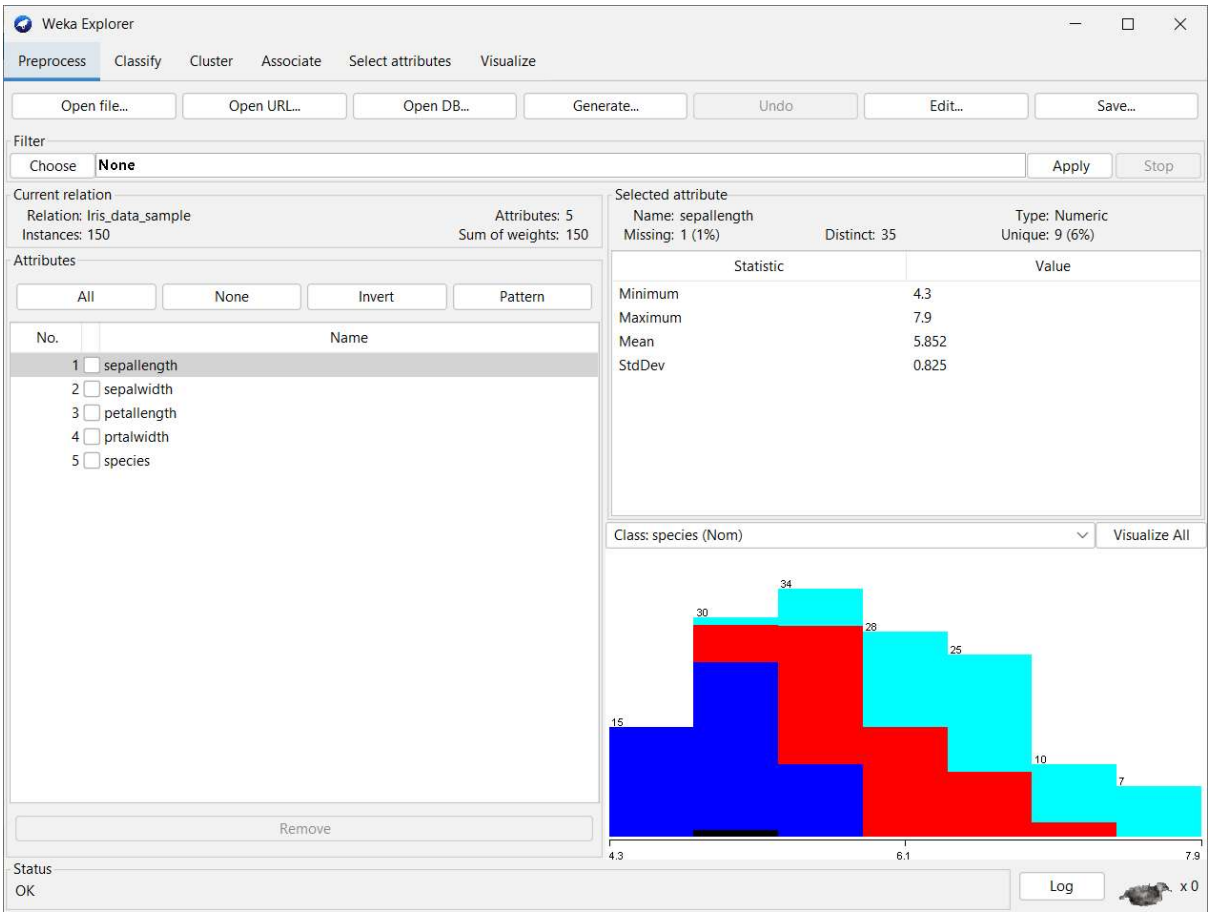
Data Preprocessing –

Firstly convert csv to arff and then open the file that is our dataset.

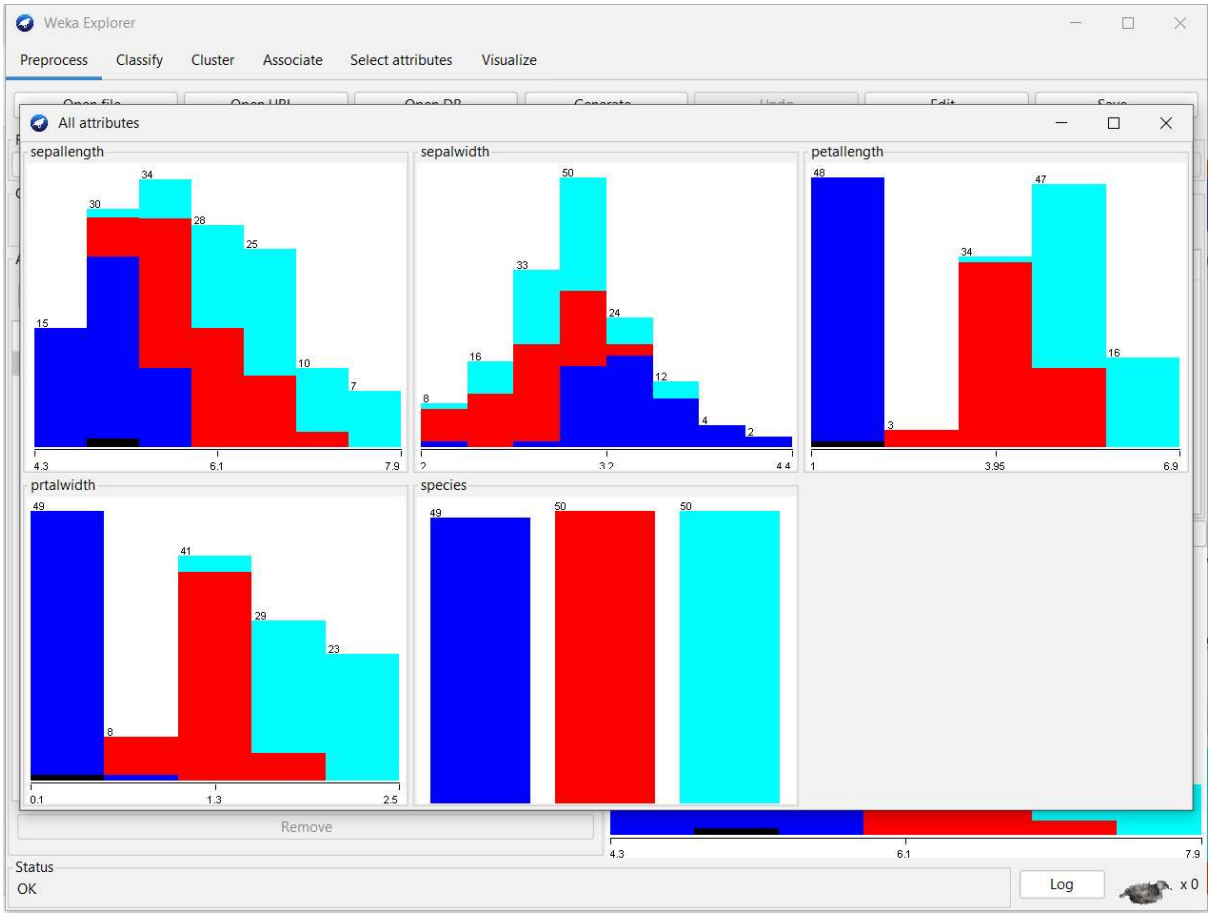
Click on Open file & import the Iris_dataset file.

Go to the Preprocessing section of the Explorer part & there we can see attribute names on the left side.

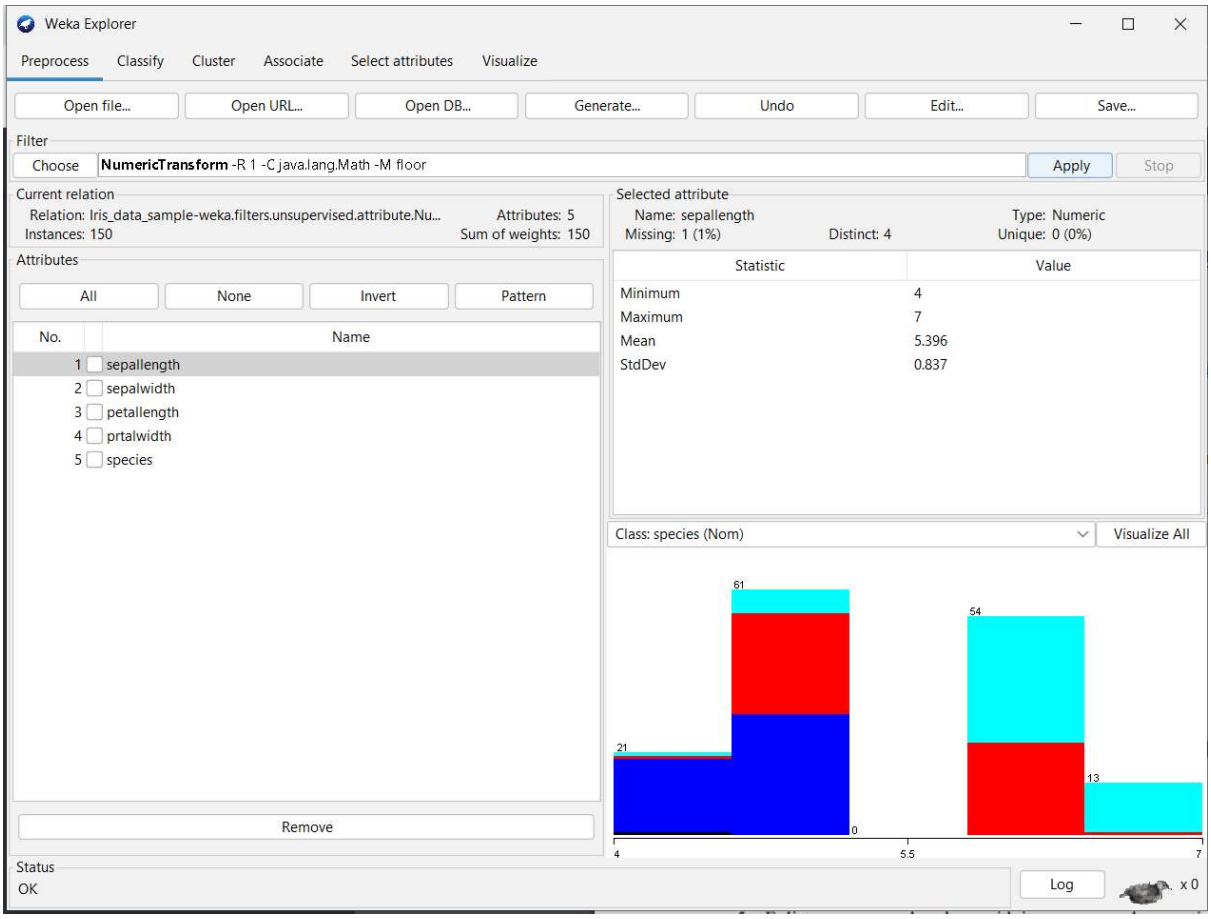
See visualization on bottom right



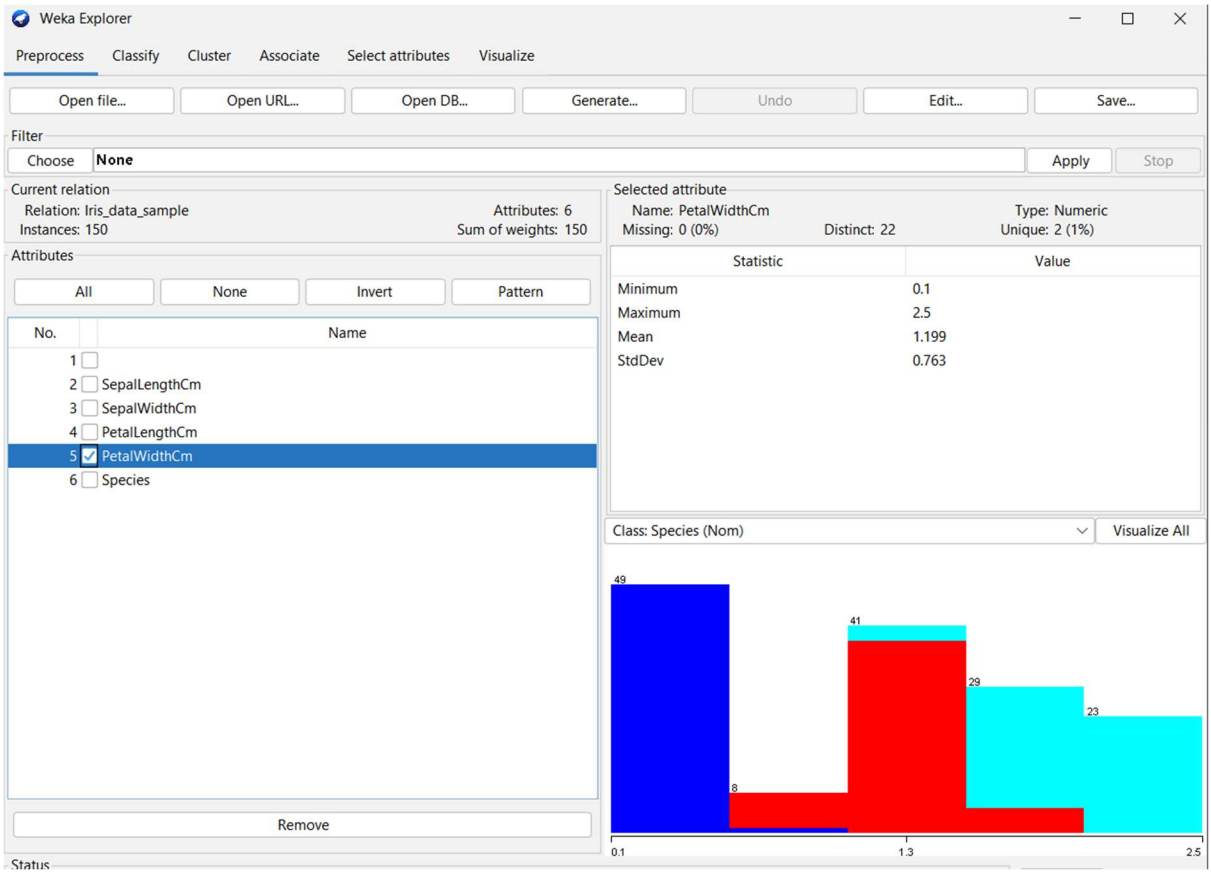
After clicking visualize all we can visualize all attributes



Transforming data into integer format



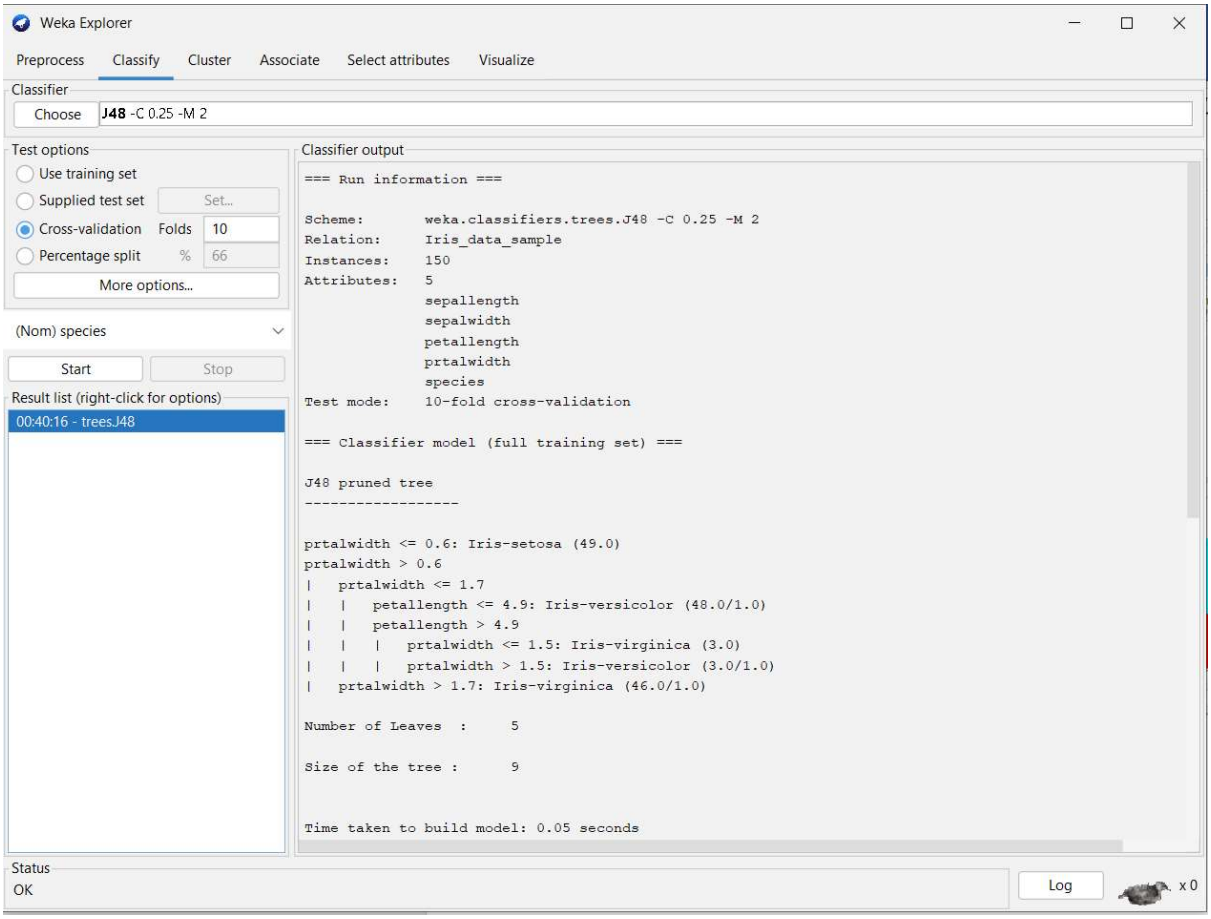
Visualize Petal width

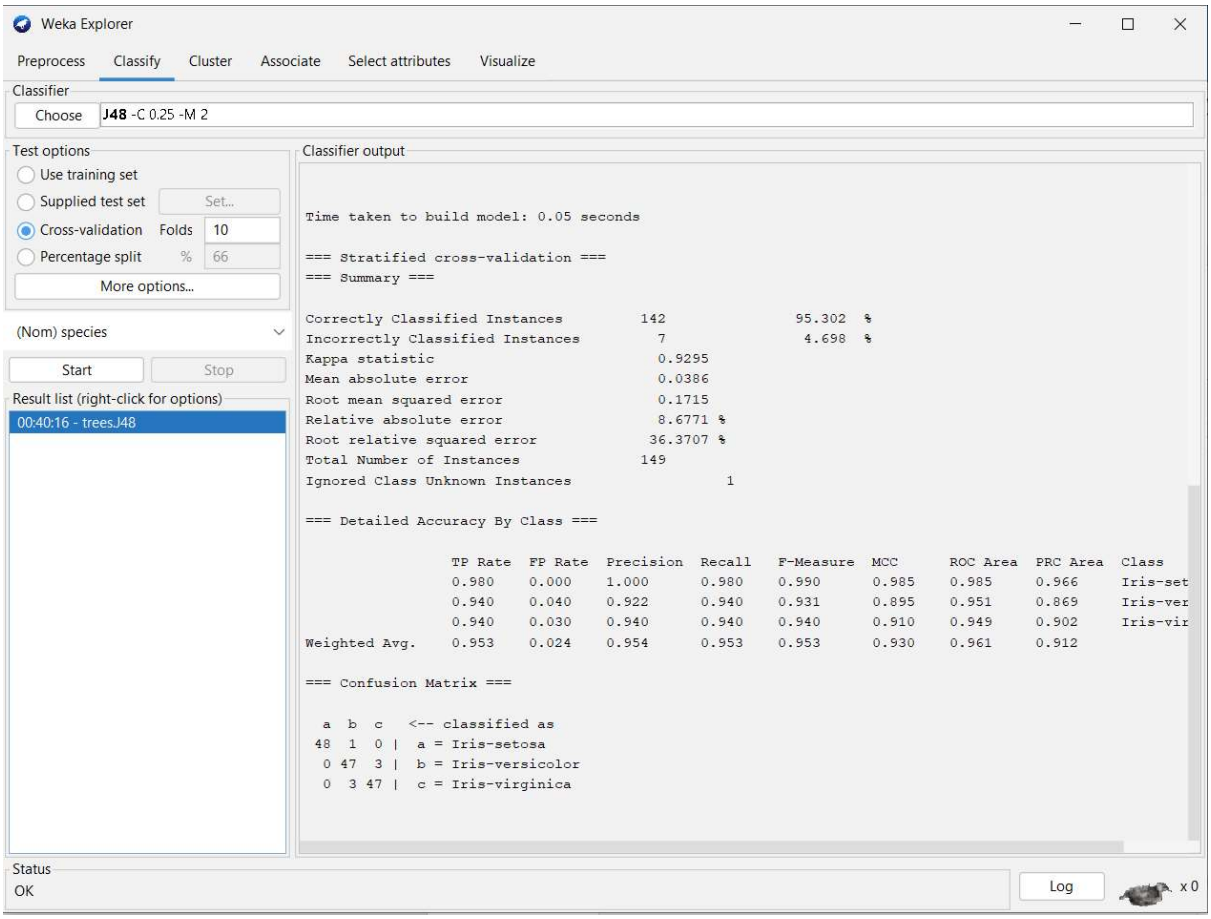


Data Classification –

Chose any ML algorithm from the options. Here I have chosen Naïve Bayes Multimonial Text under bays folder.

ML algorithm used : J48 Tree based algorithm

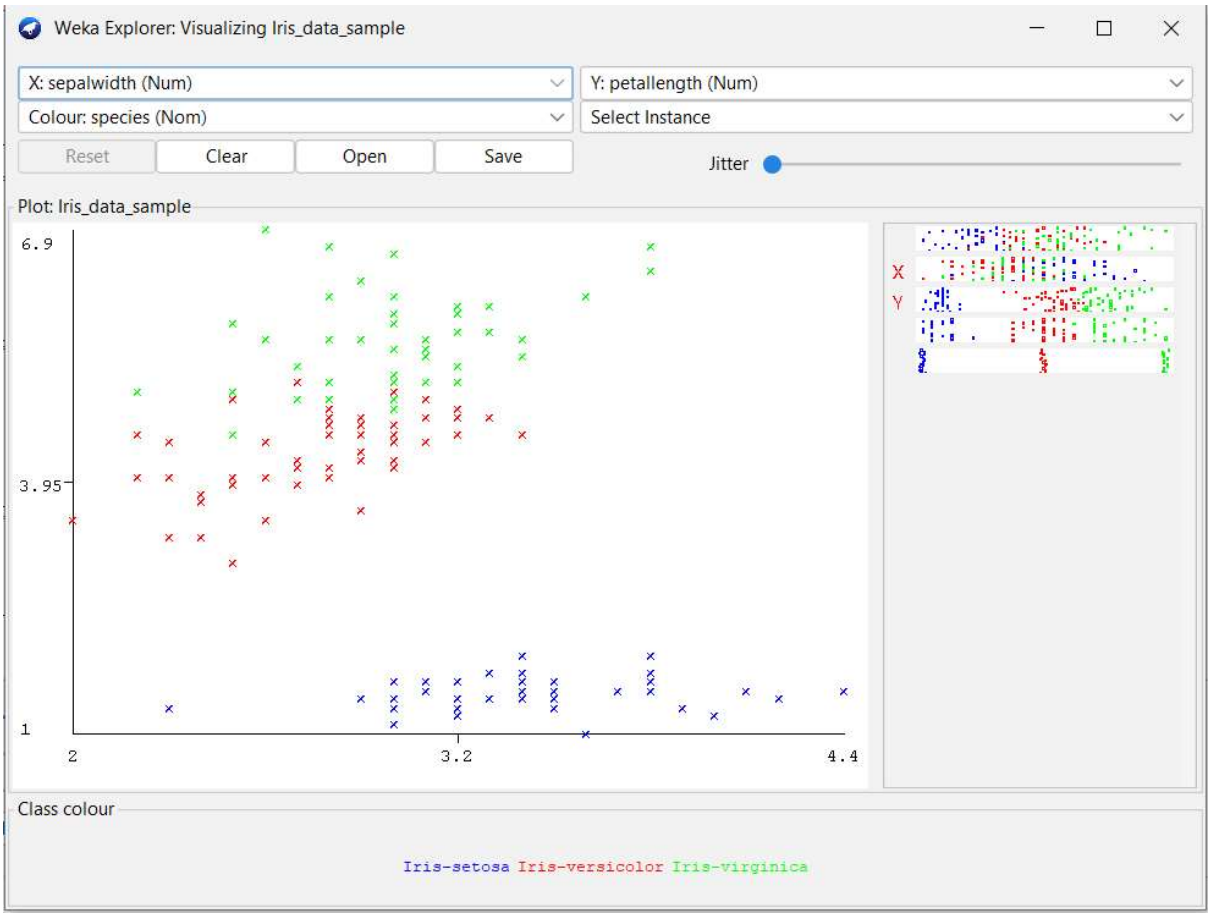


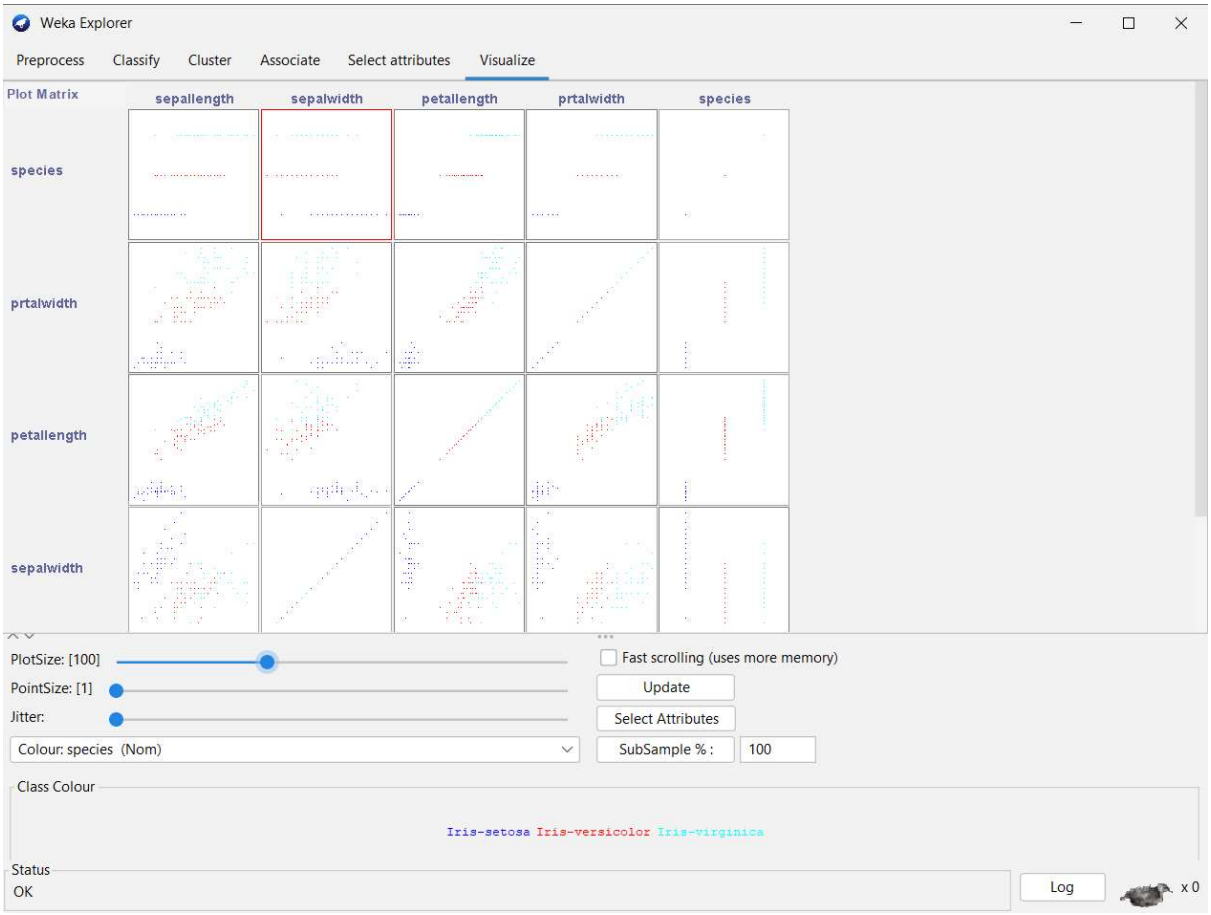


Data Visualization –

In the Visulaize section , A plot matrix is prepared & below that there are following options –

Change plot size , point size , jitter etc & the attributes can be selected according to our requirement.

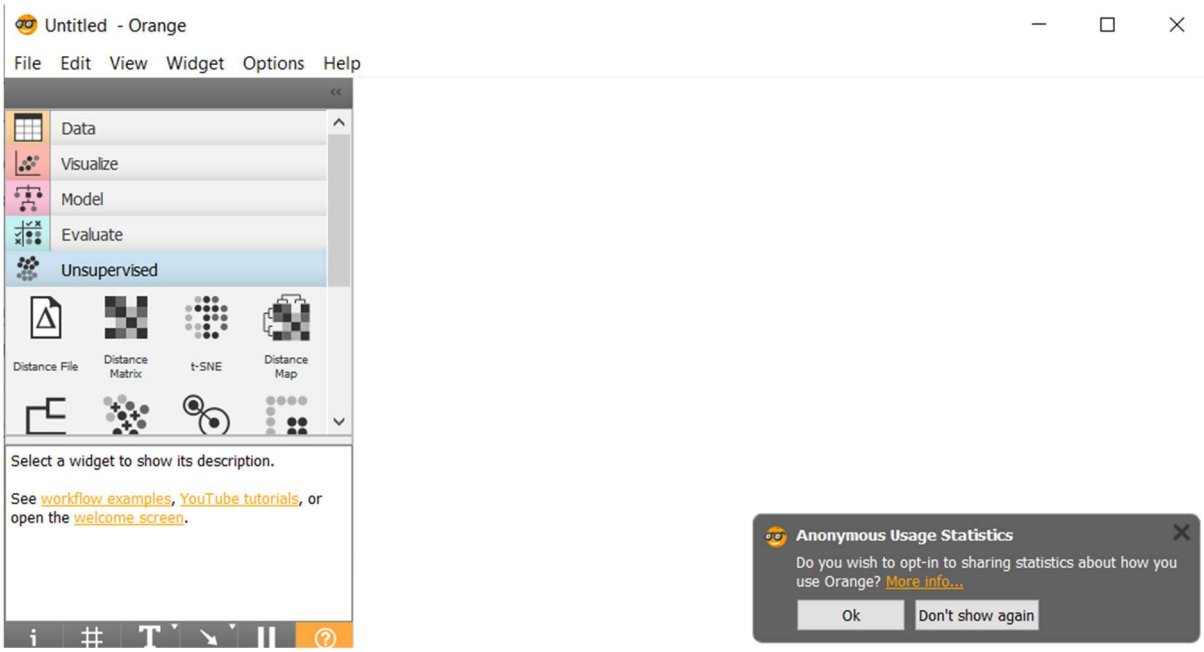

Weka Explorer: Visualizing Iris_data_sample
X: sepalwidth (Num)
Y: petallength (Num)
Colour: species (Nom)
Select Instance
Reset
Clear
Open
Save
Jitter
Plot: Iris_data_sample
6.9
3.95
1
2
3.2
4.4
X
Y
Class colour
Iris-setosa Iris-versicolor Iris-virginica


Weka Explorer
Preprocess
Classify
Cluster
Associate
Select attributes
Visualize
Plot Matrix
sepallength
sepalwidth
petallength
prtalwidth
species
PlotSize: [100]
PointSize: [1]
Jitter:
Fast scrolling (uses more memory)
Update
Select Attributes
Colour: species (Nom)
SubSample % :
100
Class Colour
Iris-setosa Iris-versicolor Iris-virginica
Status
OK
Log
x 0

Q 2) Orange is an easy to use data visualization tool with a large toolkit. In spite of being a GUI-based beginner-friendly tool, you mustn't mistake it for a light-weight one. It can do statistical distributions and box plots as well as decision trees, hierarchical clustering and linear projections. a. Install orange b. Show data distribution c. Show linear projection d. Show FreeViz

Install Orange from the website –

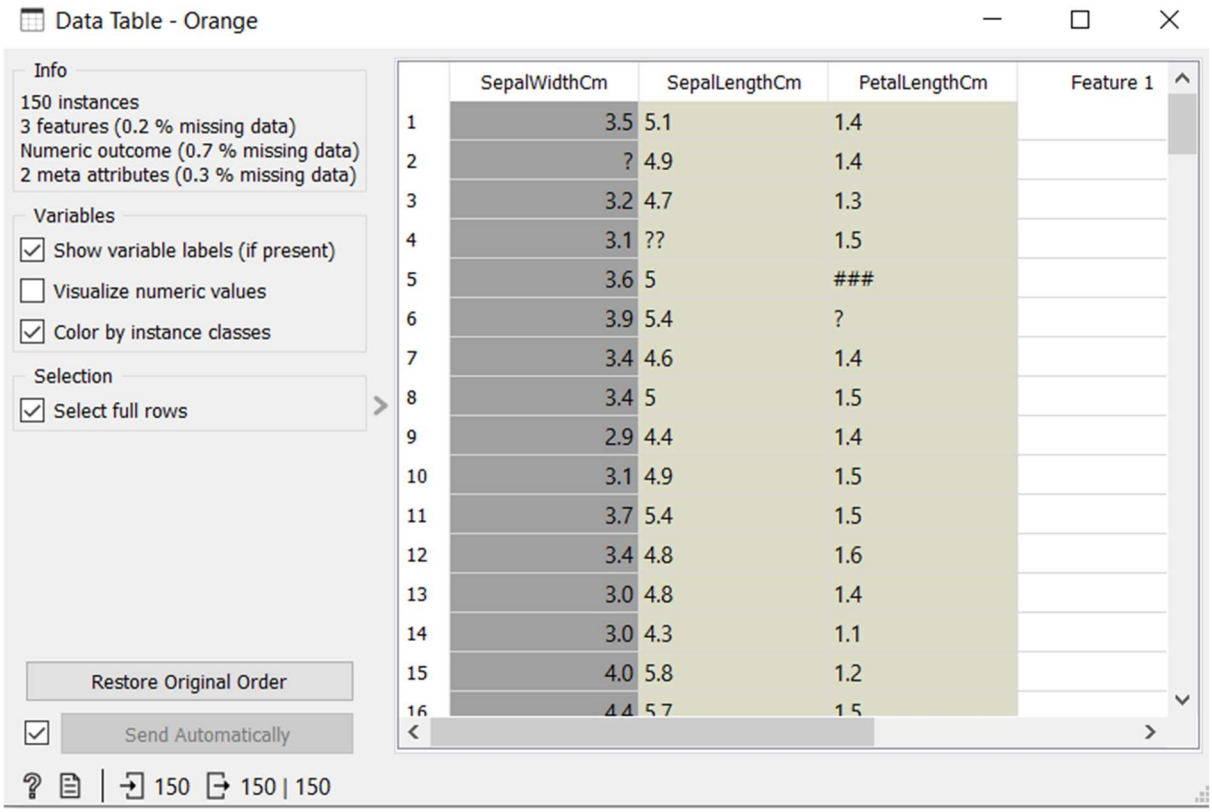
https://orangedatamining.com/



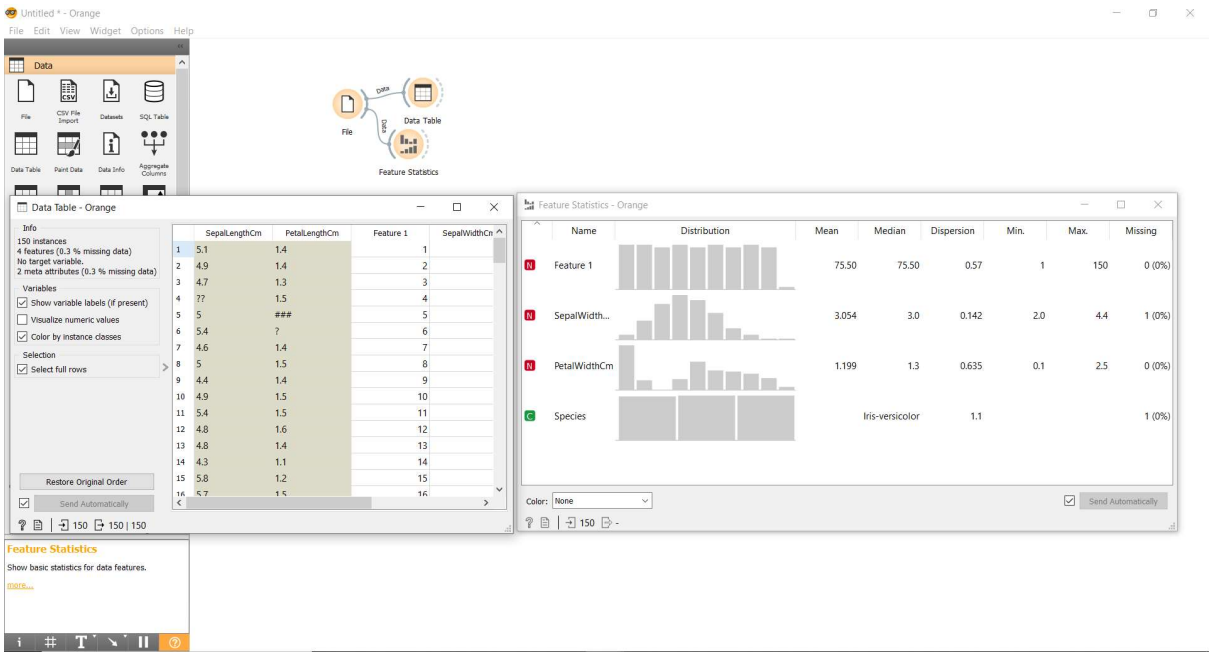
Data Table

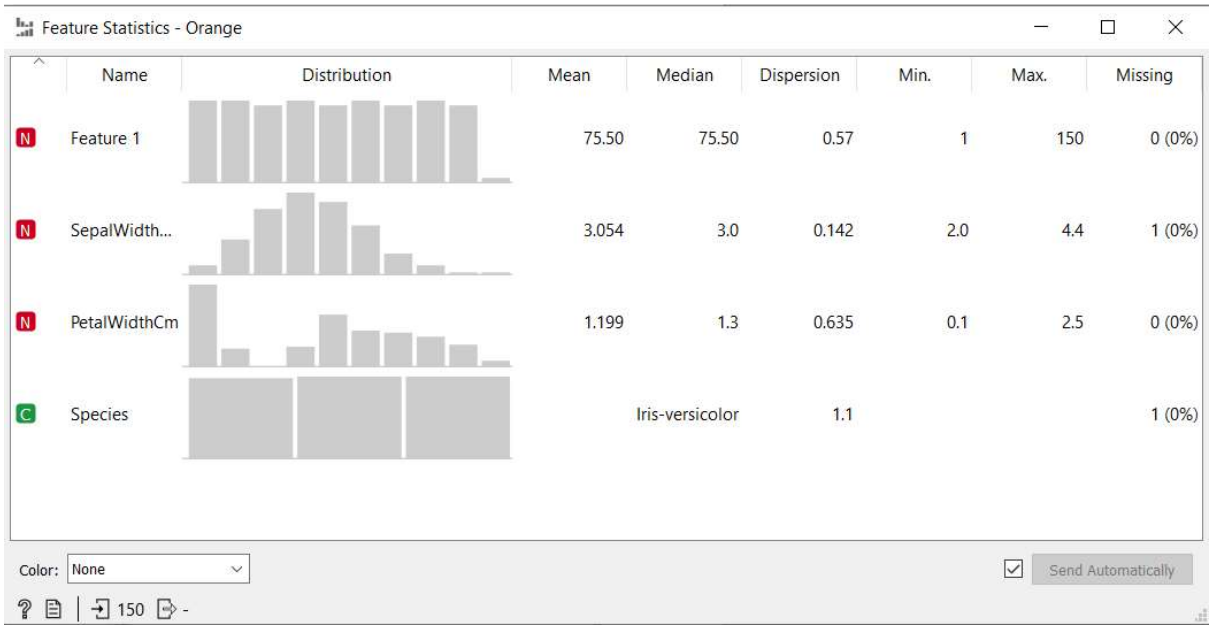
Import the dataset file i.e. , iris_dataset.xlsx.

Choose the options according to requirements . For example – Data distribution , Freewiz , linear regression etc

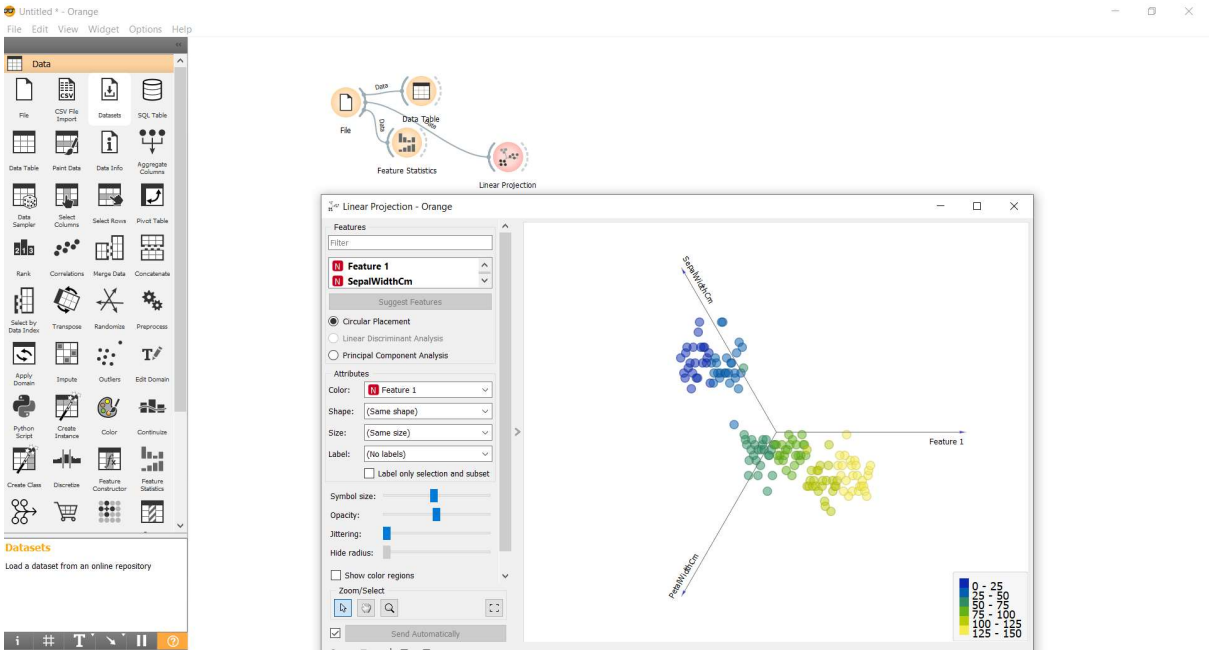


Data Distribution

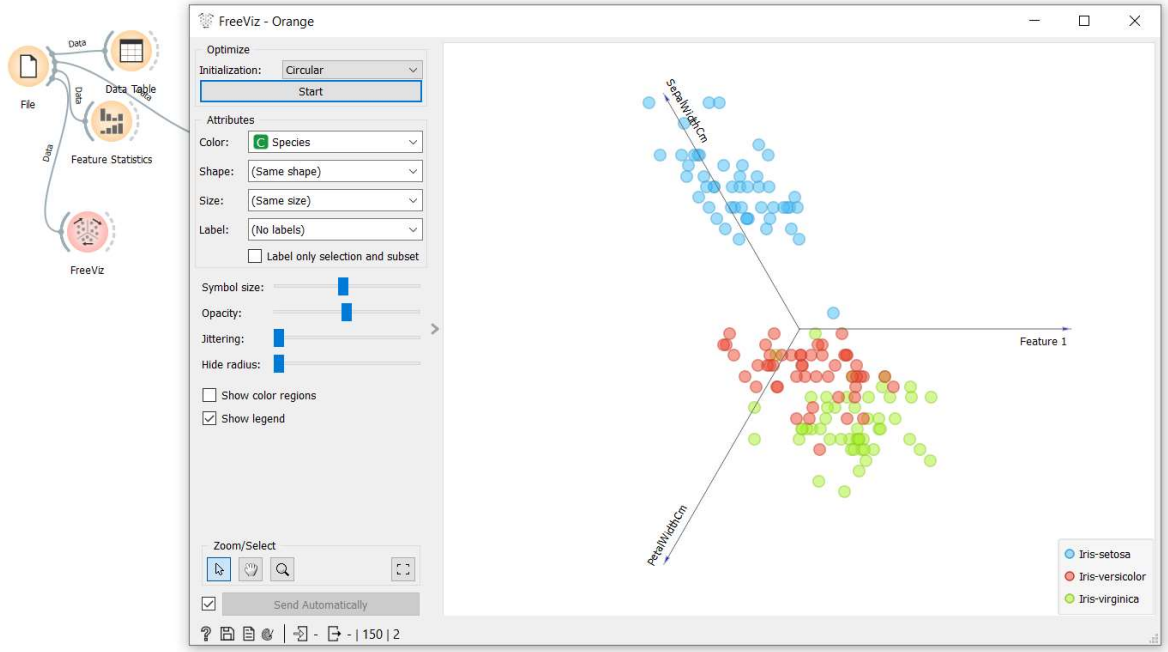




Linear Projection



Free Viz



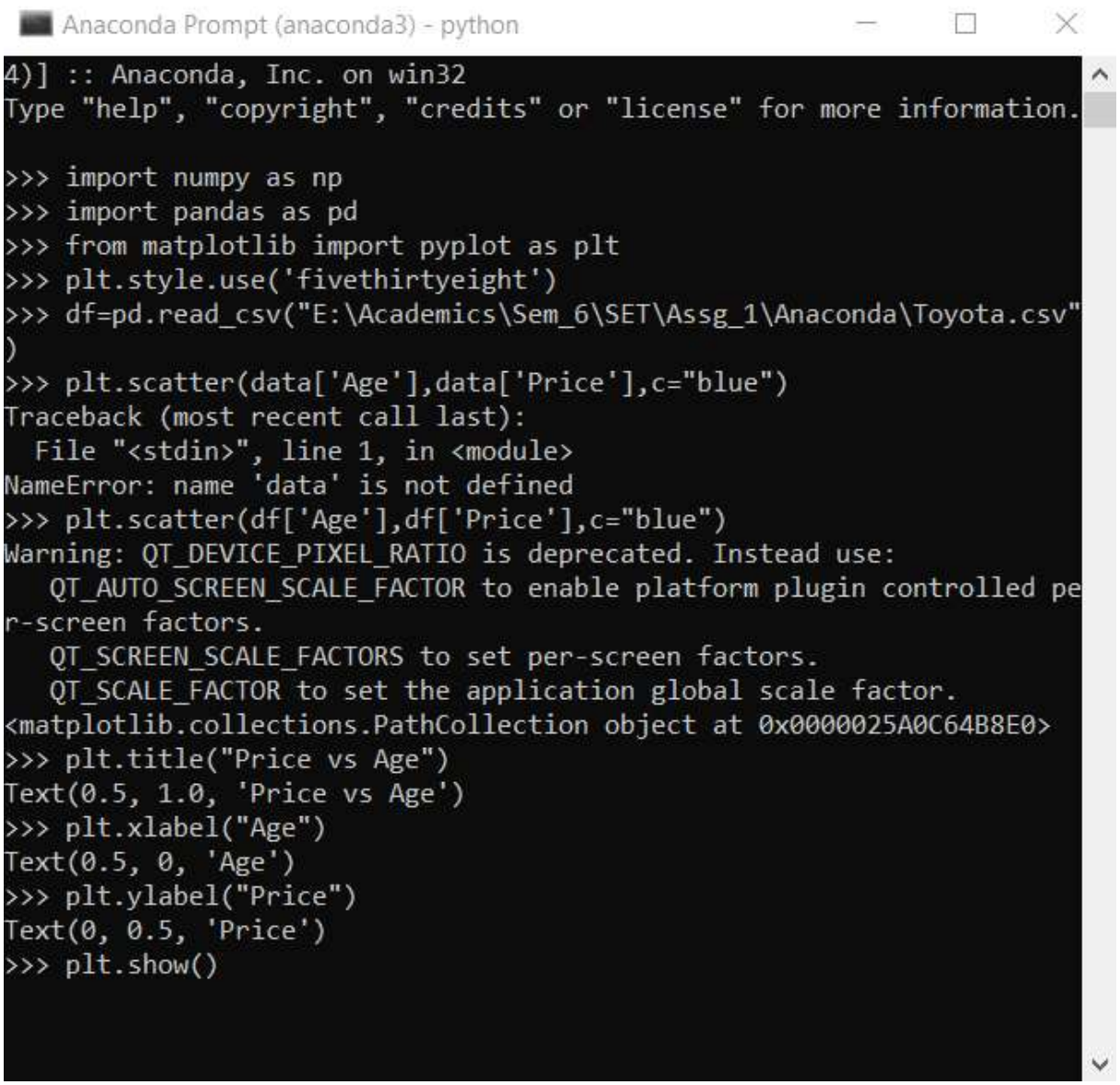
Q.3) Differentiate in between free software, Open source software and proprietary software with respect to its properties.

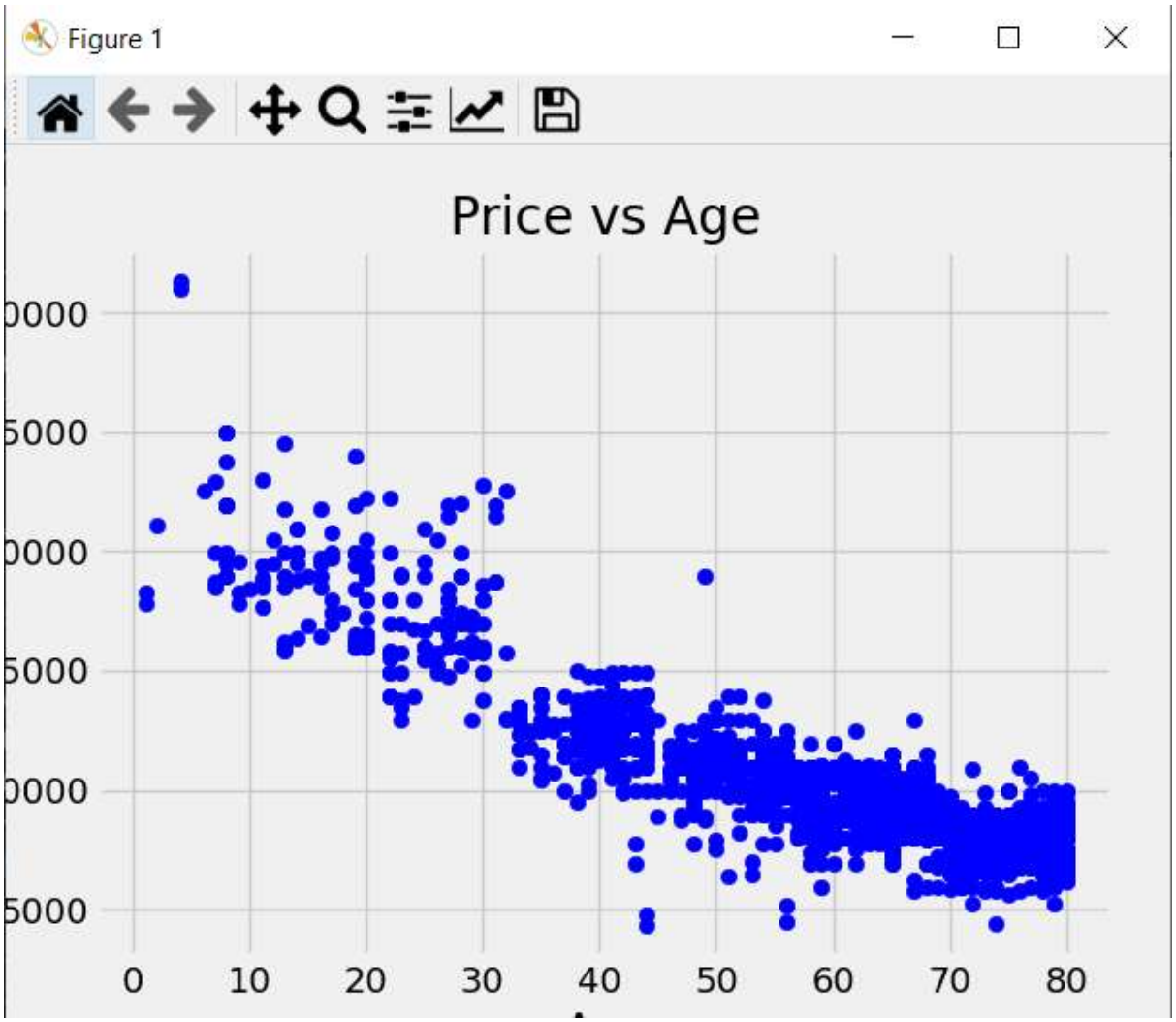
- Open Source software is flexible, that is, anyone can use, modify and distribute it. But Proprietary software has limited flexibility. Its creators own the right to its source code and only those who have paid and bought it can further use it.
- Open Source software has an open access. But Proprietary software has a limited and restricted access.
- Open Source software is the software that is open to use for all. But Proprietary software is the software that is copyrighted.
- While Open Source software is developed for open collaboration, Proprietary software is not meant for collaboration, but only for the creators and its users who paid for it.
- Open Source software is not meant for those who do not have any basic knowledge about programming. Whereas Proprietary software can be used by anyone irrespective of their skill level.
- While the examples of Open Source software are FreeBSD, Android, LibreOffice, Ubuntu and Firefox, the examples of Proprietary software are Windows, Microsoft, Adobe Flash Player and Photoshop an iTunes.

Q4) Using Anaconda Python create Histogram, Scatter plot and Bar plot for the dataset given below. Dataset-

https://drive.google.com/file/d/1i11BZFe8Xj9kNq7eeE9KOa_Iz1KhEdXJ/view

a. Scatter plot- Scatter plot of Price Vs Age

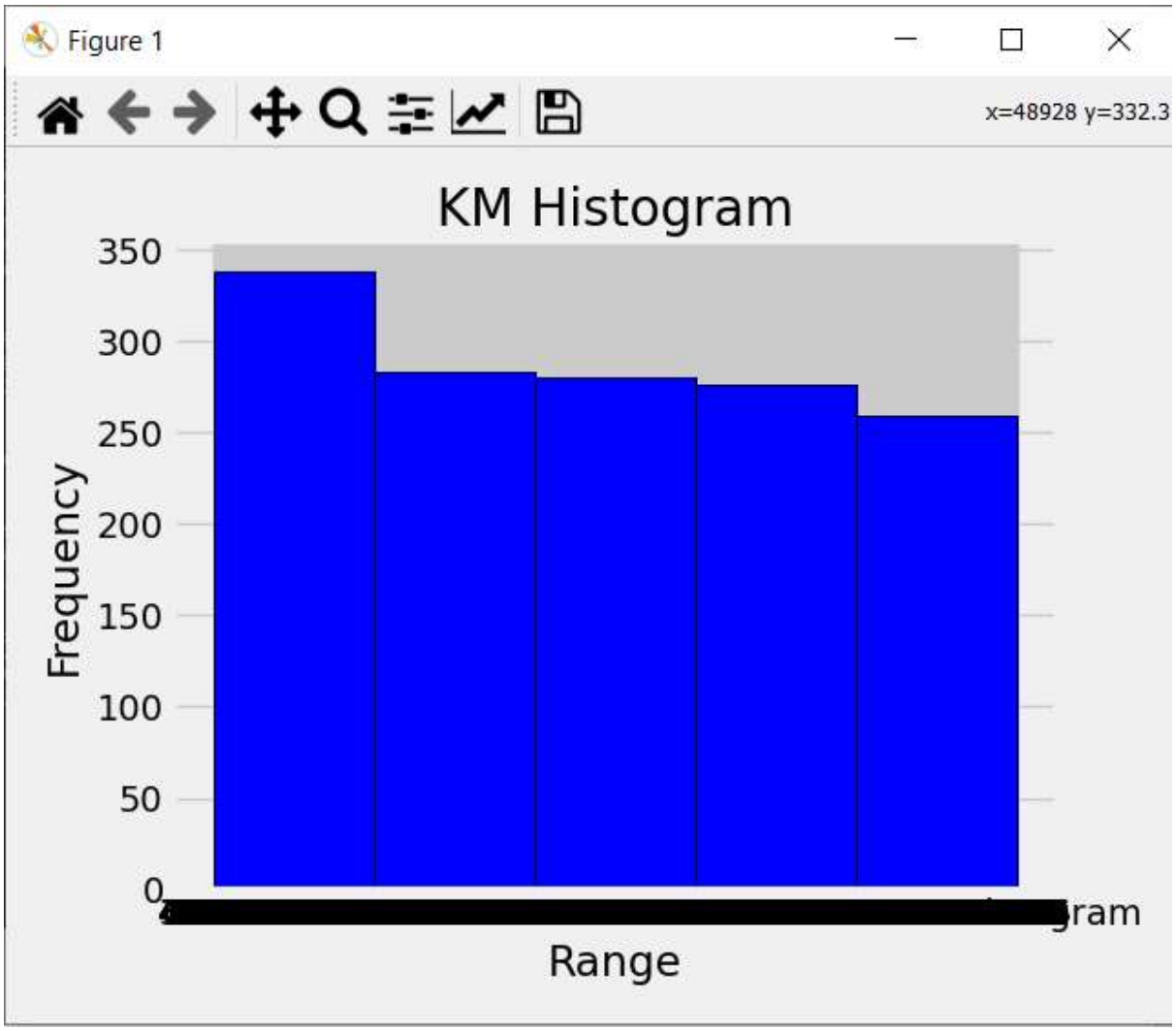




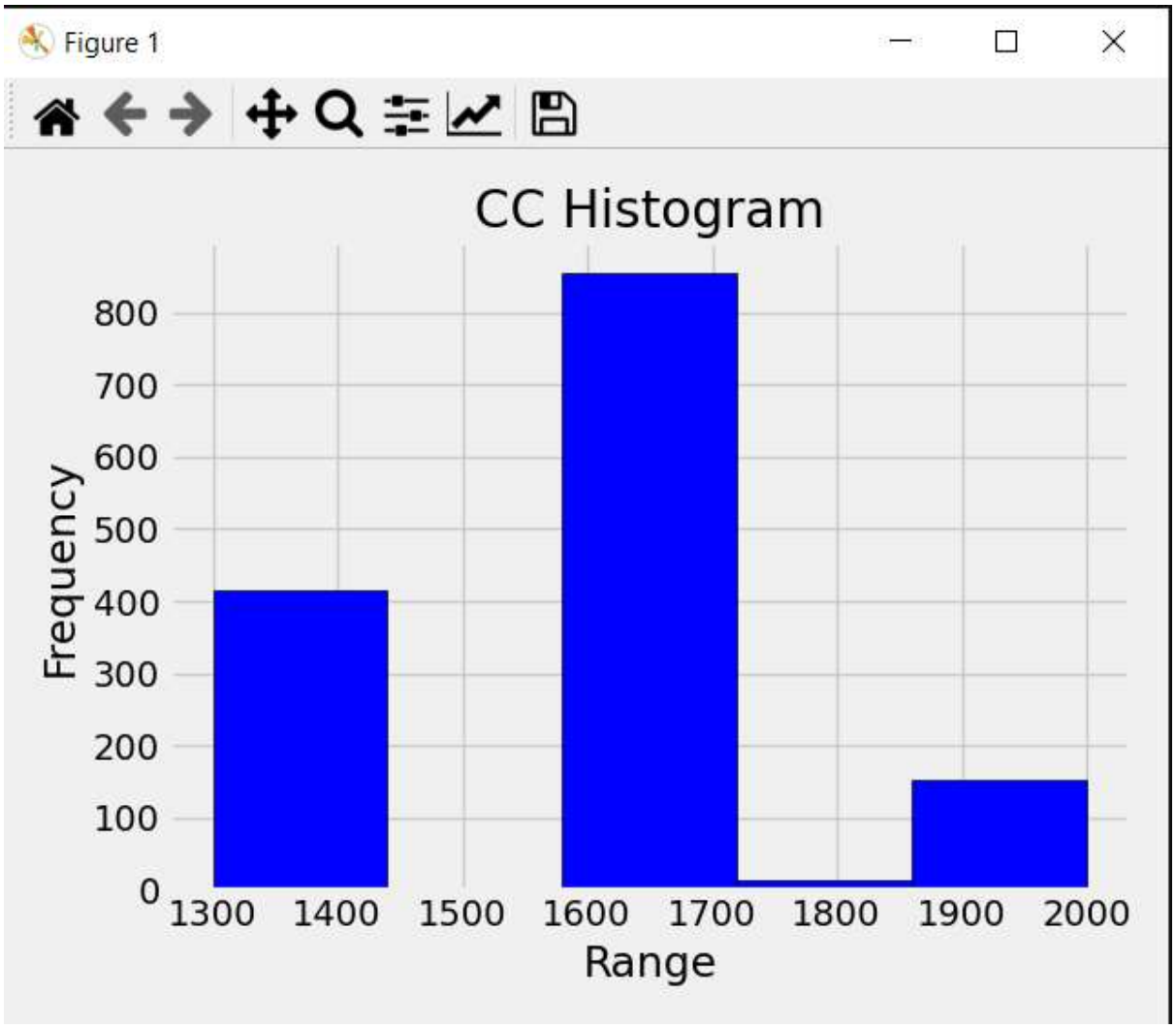
b. Histogram- for Kilometer and CC

```
>>> import numpy as np
>>> import pandas as pd
>>> from matplotlib import pyplot as plt
>>> plt.style.use("fivethirtyeight")
>>> df-pd.read_csv("E:\Academics\Sem_6\SET\Assg_1\Anaconda\Toyota.csv")
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
NameError: name 'df' is not defined
>>> df=pd.read_csv("E:\Academics\Sem_6\SET\Assg_1\Anaconda\Toyota.csv">>> kmdata=df['KM']
>>> df.head(3)
   Unnamed: 0  Price   Age     KM  ... Automatic    CC  Doors  Weight
0           0  13500  23.0  46986  ...         0  2000  three    1165
1           1  13750  23.0  72937  ...         0  2000      3    1165
2           2  13950  24.0  41711  ...         0  2000      3    1165

[3 rows x 11 columns]
>>> plt.hist(kmdata,bins=5,edgecolor="black",color="blue")
Warning: QT_DEVICE_PIXEL_RATIO is deprecated. Instead use:
   QT_AUTO_SCREEN_SCALE_FACTOR to enable platform plugin controlled per-screen factors.
   QT_SCREEN_SCALE_FACTORS to set per-screen factors.
   QT_SCALE_FACTOR to set the application global scale factor.
(array([338., 283., 280., 276., 259.]), array([   0.,  251.,  502.,  753., 1004., 1255.]), <BarContainer object of 5 artists>)
>>> plt.hist("KM Histogram")
(array([0., 0., 0., 0., 0., 1., 0., 0., 0., 0.]), array([1255.5, 1255.6, 1255.7, 1255.8, 1255.9, 1256. , 1256.1, 1256.2,
       1256.3, 1256.4, 1256.5]), <BarContainer object of 10 artists>)
>>> plt.hist(kmdata,bins=5,edgecolor="black",color="blue")
(array([338., 283., 280., 276., 259.]), array([   0.,  251.,  502.,  753., 1004., 1255.]), <BarContainer object of 5 artists>)
>>> plt.title("KM Histogram")
Text(0.5, 1.0, 'KM Histogram')
>>> plt.xlabel("Range")
Text(0.5, 0, 'Range')
>>> plt.ylabel("Frequency")
Text(0, 0.5, 'Frequency')
>>> plt.tight_layout()
>>> plt.show()
```

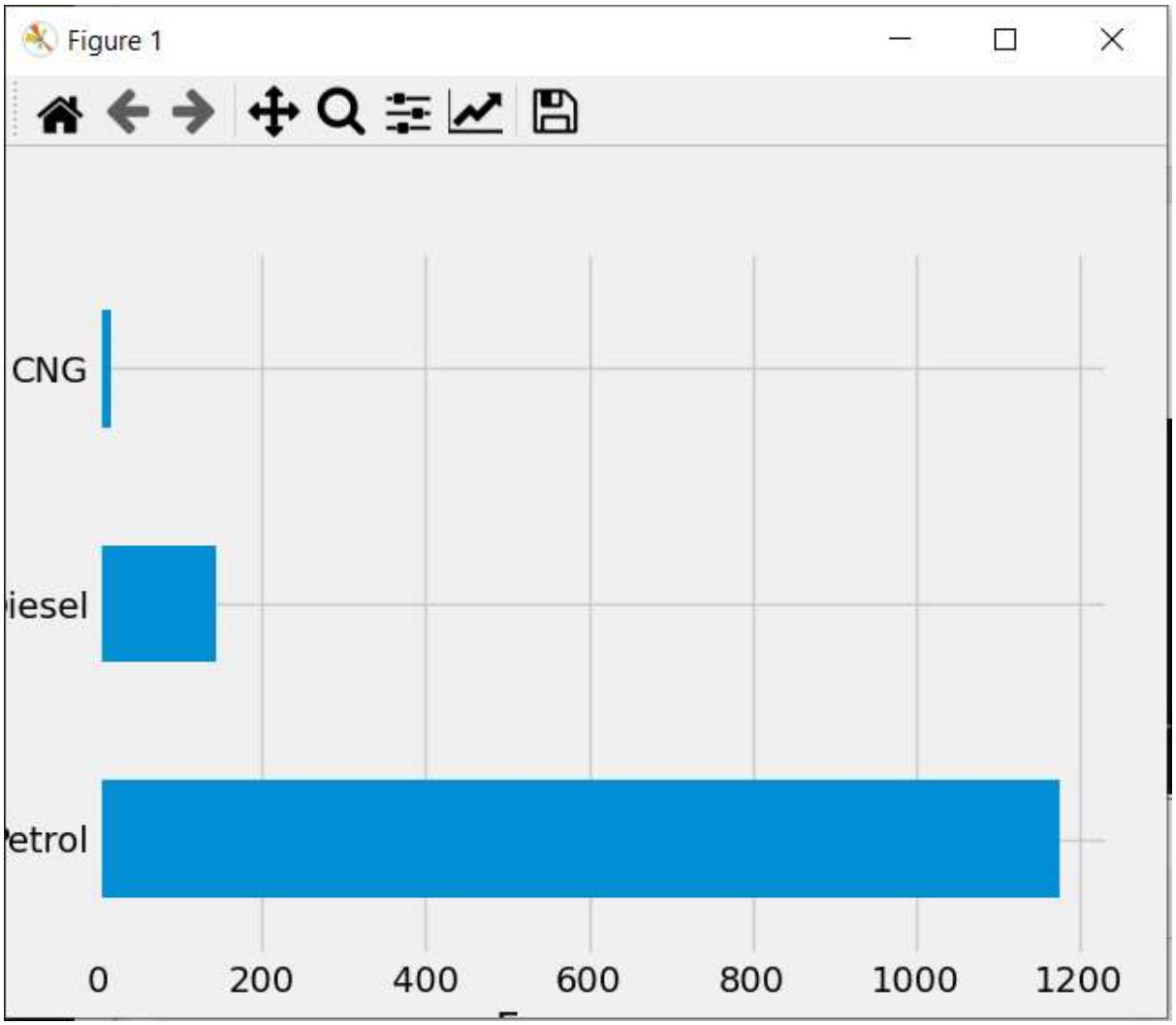



```
>>> ccdata=df['CC']
>>> plt.hist(ccdata,bins=5,edgecolor="black",color="blue")
(array([416.,   0., 854.,  14., 152.]), array([1300., 1440., 1580., 1720., 1860., 2000.]), <BarContainer object of 5 artists>)
>>> plt.title("CC Histogram")
Text(0.5, 1.0, 'CC Histogram')
>>>  plt.xlabel("Range")
  File "<stdin>", line 1
    plt.xlabel("Range")
    ^
IndentationError: unexpected indent
>>> plt.xlabel("Range")
Text(0.5, 0, 'Range')
>>> plt.ylabel("Frequency")
Text(0, 0.5, 'Frequency')
>>> plt.tight_layout()
>>> plt.show()
```



c. Bar plot- Bar plot for different fuel types

```
>>> fuels=pd.value_counts(df['FuelType'].values,sort=True)
>>> plt.xlabel("Frequency")
Text(0.5, 0, 'Frequency')
>>> plt.ylabel("Fuel Type")
Text(0, 0.5, 'Fuel Type')
>>> fuels.plot.barh()
<AxesSubplot:xlabel='Frequency', ylabel='Fuel Type'>
>>> plt.show()
```



Q 5) Enlist some examples along with its purpose and properties (at least 10) of FOSS and proprietary software with respect to database.

FOSS (Free-open-source-software):

Free and open-source software (FOSS) is a software that can be classified as both free software and open-source software.

Examples: Ubuntu, VLC Player, Android system, etc.

Advantages of FOSS:

1. Expansive licensing: Proprietary software licenses are usually quite restrictive in terms of use, number of users, type of machine and other. There is fee to own license of a proprietary software. Open-source software are free to own there is no restriction on how we use the software, we can install it on unlimited machines.

2. Transparency: Open-source development is carried out openly. As software code is openly available anyone who finds some bug can fix it for others. As development process is carried out publicly its development process is transparent. Users can easily communicate with product developers to understand their product decisions and offer opinions for betterment of software.

3. Source Code inspection: As source code is openly available anyone can view code of the software for better understanding of how the application works.

4. Source Code modification: We can also modify code after inspecting the code. One can modify code and for himself and also, he can add those changes to main version of software so that changes will be available for everyone for use.

5. Community: Foundation of open source projects is community, it includes developers of software and also users. Users in community can easily share there feedback with developers so that developers can improve the software.

Proprietary Software:

This type of software requires licenses for their use. Company or organization that owns the software provides rights to use the software to customer. Users can only install software only on limited number of machines and cannot redistribute it.

1) **Increased Functionality and Convenience**

Proprietary systems are easier to use and learn, leading to faster work processes. Skype, for example, is used by organizations worldwide. It takes minutes to sign up for an account and make international phone calls or conduct video interviews online. On top of that, your customers, suppliers and employees may already have a Skype account, so they know how to use it

2) **Superior Customer Support**

Open-source software can be difficult to install and set up. Customizing it isn't easier either. Plus, your staff may not be familiar with the program and may need additional training.

The average employee lacks the expertise to use open-source programs. Therefore, your team members may need help with most tasks. They will spend hours trying to figure things out instead of focusing on the tasks at hand.
Proprietary software is more accessible and includes technical support. Most companies offering these programs provide dedicated sources, 24/7 assistance, live chat and user manuals.

3) Lower Maintenance Costs

As a small-business owner, you may prefer open-source software due to its low cost. Most programs are free or cost next to nothing. The downside is that you may end up paying a lot more for setup, maintenance and customization than you'd pay with proprietary software.
Some open-source programs are difficult to install and set up, so you may need to call an expert to do the job.

4) Stronger Competitive Advantage

Proprietary technology enables organizations to be more profitable, productive and innovative. This is particularly true for software-development companies, which often use custom programs at the core of their business model.

5)Secure Financing for Your Business

Nearly one-third of startups close their doors because they run out of capital. Developing proprietary technology doesn't guarantee success, but it could make it easier to secure financing for your small business. Plus, you will be able to charge higher prices because no other company offers the same product as you do.

Examples:

Atom Editor

Purpose:

Atom is a free and open source code editor and source code editor for macOS, linux and windows with support for plugin written in JavaScript and embedded git control.

Properties:

i. Customizable
ii. Atom is a "hackable" text editor
iii. One can even make a package to wrap all of this functionality into a single package
iv. enables users to install third-party packages
v. Syntactic highlighting support for other languages than the default
vi. Easily extensible through extra packages that provide among other things code autocompletion for a wide variety of languages
vii. FTP capabilities, and built-in browser preview.

NextCloud:

NetCloud simplifies network management by making it easy to understand and apply licenses, identify and push out software updates, upgrade software feature sets, amass education tools, and connect with other customers.

Properties:

End-to-end encryption
Virtual Data Rooms
User management and authentication (LDAP, SAML, 2FA)
Online document editing with Collabora Online
Mobile and desktop clients
Secure file sharing abilities
Scalability
Customer file upload (file drop)
virtual data rooms
Monitoring and handling

Celestia:

Purpose:

Celestia allows users to navigate at different speeds, and allow users to orbit stars, planets, moons, and other space objects, track space objects such as spacecraft, asteroids, and comets as they fly by, or travel to and/or fly through galaxies.

Properties:

The user can change Celestia's field of view
allows users to split the window into multiple panes
Celestia allows users to navigate at different speeds
allow users to orbit stars, planets, moons, and other space objects
Screenshots and movies can be captured in classic or HD resolutions.
track space objects such as spacecraft, asteroids, and comets as they fly by
Information about the objects that Celestia draws can also be displayed.
The user can change Celestia's field of view

FreeRDP:

Purpose:

FreeRDP is a free implementation of the Remote Desktop Protocol (RDP), released under the Apache license. It enables network administrators to remotely diagnose problems that individual users encounter as well as gives users remote access to their physical work desktop computers.

Properties:

smart card authentication
bandwidth reduction;
the ability to use multiple displays;
the ability to disconnect temporarily without logging off;
128-bit encryption for mouse and keyboard data using RC4 encryption;
directs audio from a remote desktop to the user's computer;
redirects local files to a remote desktop;
local printers can be used in remote desktop sessions;
applications in the remote desktop session can access local ports;

shares clipboard between local and remote computers;

Blender

Purpose:

Blender is the free and open source 3D creation suite. It supports the entirety of the 3D pipeline-modelling, rigging, animation, simulation, rendering, compositing and motion tracking, even video editing and game creation

Properties:

3D modelling
UV unwrapping
Texturing
raster graphics
editing
rigging and skinning
fluid and smoke simulation
particle simulation
soft body simulation
animating