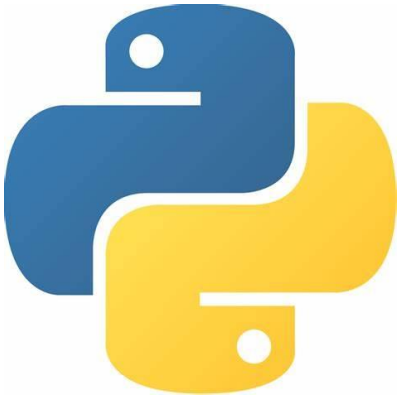
Python Essential libraries



1.NumPy: Fundamental package for numerical computing, providing support for large, multi-dimensional arrays and matrices.

Create a NumPy array:

```
import numpy as np
my_array = np.array([1, 2, 3, 4, 5])
print(my_array)
```

```
[1 2 3 4 5]
```

Perform operations on the array:

```
result = my_array * 2
print(result)
```

```
[ 2  4  6  8 10]
```

Syed Afroz Ali
https://www.linkedin.com/in/syed-afroz-70939914/

2.Pandas: Powerful data analysis and manipulation library, offering data structures like Series and DataFrames.

Create a Pandas DataFrame:

```
import pandas as pd

data = {'Name': ['Alice', 'Bob', 'Charlie'],
        'Age': [25, 30, 28]}
df = pd.DataFrame(data)
df
```

	Name	Age
0	Alice	25
1	Bob	30
2	Charlie	28

Access and manipulate data:

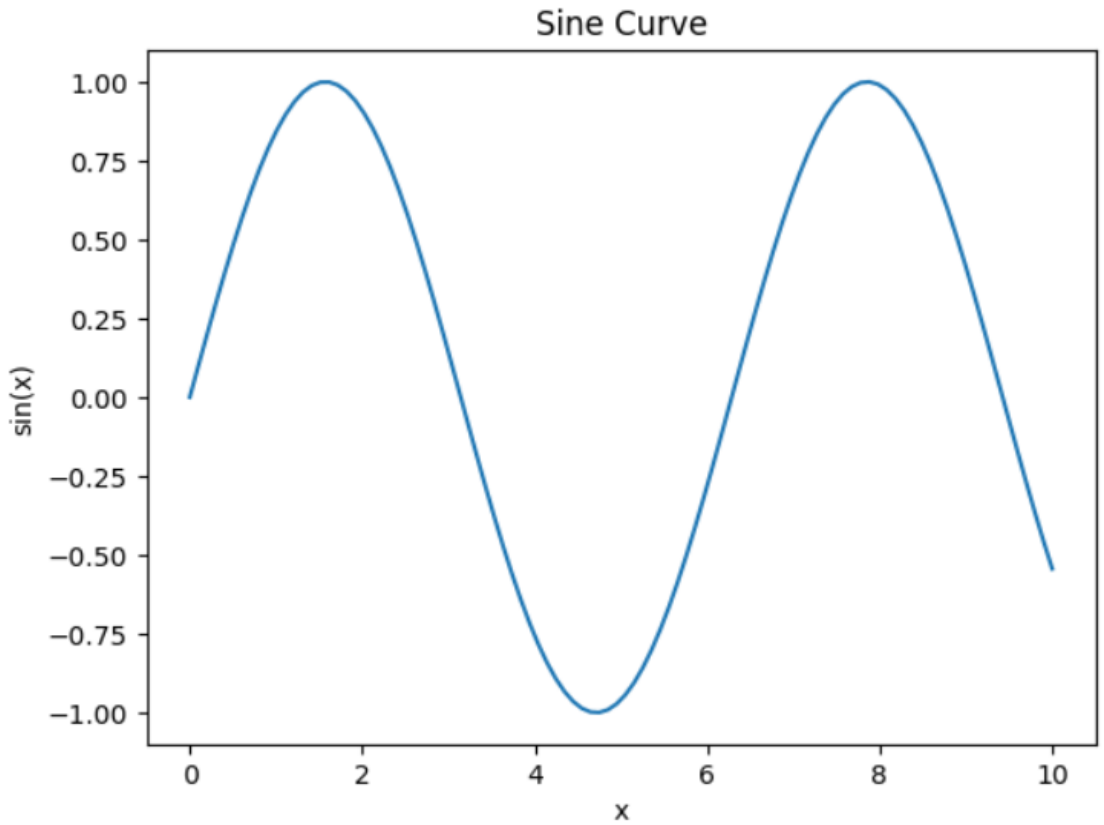
```
df['Age'] = df['Age'] * 2
df
```

	Name	Age
0	Alice	50
1	Bob	60
2	Charlie	56

Syed Afroz Ali
https://www.linkedin.com/in/syed-afroz-70939914/

3.Matplotlib: Comprehensive plotting library for creating static, animated, and interactive visualizations.

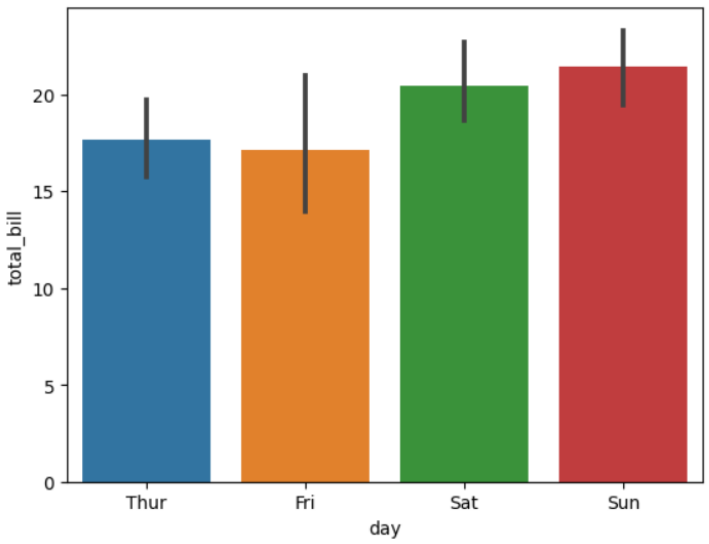
```
import matplotlib.pyplot as plt
import numpy as np

x = np.linspace(0, 10, 100)
y = np.sin(x)

plt.plot(x, y)
plt.xlabel('x')
plt.ylabel('sin(x)')
plt.title('Sine Curve')
plt.show()
```



Syed Afroz Ali
https://www.linkedin.com/in/syed-afroz-70939914/

4.Seaborn: High-level data visualization library built on Matplotlib, offering attractive statistical graphics.

```
import seaborn as sns
import matplotlib.pyplot as plt
tips = sns.load_dataset("tips")
sns.barplot(x="day", y="total_bill", data=tips)
plt.show()
```



Syed Afroz Ali
https://www.linkedin.com/in/syed-afroz-70939914/

5.Scikit-learn: Machine learning library providing various algorithms for classification, regression, clustering, and more.

Load a dataset and create a model:

```
from sklearn import datasets
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LinearRegression
from sklearn.metrics import mean_squared_error

# Load the diabetes dataset
diabetes = datasets.load_diabetes()
X = diabetes.data
y = diabetes.target

# Split the data into training and testing sets
X_train, X_test, y_train, y_test = train_test_split(X, y,
test_size=0.2, random_state=42)

# Create a linear regression model
model = LinearRegression()

model.fit(X_train, y_train)

# Make predictions
y_pred = model.predict(X_test)

# Evaluate the model
mse = mean_squared_error(y_test, y_pred)
print("Mean Squared Error:",mse)
```

```
Mean Squared Error: 2900.1936284934804
```

Syed Afroz Ali
https://www.linkedin.com/in/syed-afroz-70939914/

6.TensorFlow: Open-source platform for machine learning, used for building and training complex models.

Create a simple TensorFlow model:

```
import tensorflow as tf

# Create some data
x = tf.constant([[1, 2], [3, 4]])
y = tf.constant([[5], [7]])

# Create a simple linear model
model = tf.keras.Sequential([tf.keras.layers.Dense(1)])

# Compile the model
model.compile(loss='mean_squared_error',
optimizer='adam')

# Train the model
model.fit(x,y,epochs=10)
```

```
1/1 [==============================] - 1s 1s/step - loss: 2.6562
Epoch 2/10
1/1 [==============================] - 0s 4ms/step - loss: 2.6426
Epoch 3/10
1/1 [==============================] - 0s 25ms/step - loss: 2.6292
Epoch 4/10
1/1 [==============================] - 0s 16ms/step - loss: 2.6158
Epoch 5/10
1/1 [==============================] - 0s 4ms/step - loss: 2.6025
Epoch 6/10
1/1 [==============================] - 0s 24ms/step - loss: 2.5893
Epoch 7/10
1/1 [==============================] - 0s 0s/step - loss: 2.5762
Epoch 8/10
1/1 [==============================] - 0s 7ms/step - loss: 2.5631
Epoch 9/10
1/1 [==============================] - 0s 19ms/step - loss: 2.5502
Epoch 10/10
1/1 [==============================] - 0s 1ms/step - loss: 2.5373
```

Syed Afroz Ali
https://www.linkedin.com/in/syed-afroz-70939914/

7. Keras: High-level API built on top of TensorFlow or Theano for deep learning.

Create a Keras model:

```
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Dense

# Create a simple sequential model
model = Sequential([
    Dense(32, activation='relu', input_shape=(784,)),
    Dense(10, activation='softmax')
])

# Compile the model
model.compile(optimizer='adam',
              loss='categorical_crossentropy',
              metrics=['accuracy'])
```

8. PyTorch: Popular open-source machine learning library for deep learning applications.

Create a PyTorch model:

```
import torch
import torch.nn as nn

# Create a simple neural network
class Net(nn.Module):
    def __init__(self):
        super(Net, self).__init__()
        self.fc1 = nn.Linear(784, 120)
        self.fc2 = nn.Linear(120, 84)
        self.fc3 = nn.Linear(84, 10)

    def forward(self, x):
        x = torch.relu(self.fc1(x))
        x = torch.relu(self.fc2(x))
        x = self.fc3(x)
        return x

net = Net()
```

9. LightGBM: Gradient boosting framework for efficient machine learning.

```
import lightgbm as lgb

# Create training and validation datasets
train_data = lgb.Dataset(X_train, label=y_train)
val_data = lgb.Dataset(X_val, label=y_val)

# Specify parameters
params = {
    'objective': 'regression',
    'metric': 'rmse',
    'num_leaves': 31,
    'learning_rate': 0.05,
    'feature_fraction': 0.9
}

# Train the model
gbm = lgb.train(params,
                train_data,
                valid_sets=[val_data],
                num_boost_round=100,
                early_stopping_rounds=10)
```

10. NLTK: Natural Language Toolkit for processing and analyzing human language data.

```
import nltk
from nltk.corpus import stopwords
from nltk.tokenize import word_tokenize

# Download stopwords if not already downloaded
nltk.download('stopwords')

# Sample text
text = "This is a sample sentence for tokenization."

# Tokenize the text
words = word_tokenize(text)

# Remove stop words
stop_words = set(stopwords.words('english'))
filtered_words = [word for word in words if word
not in stop_words]

print(filtered_words)
```

```
['This', 'sample', 'sentence', 'tokenization', '.']
```

Syed Afroz Ali
https://www.linkedin.com/in/syed-afroz-70939914/