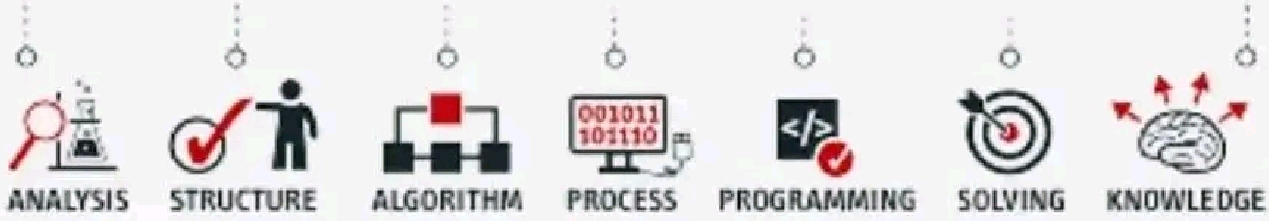
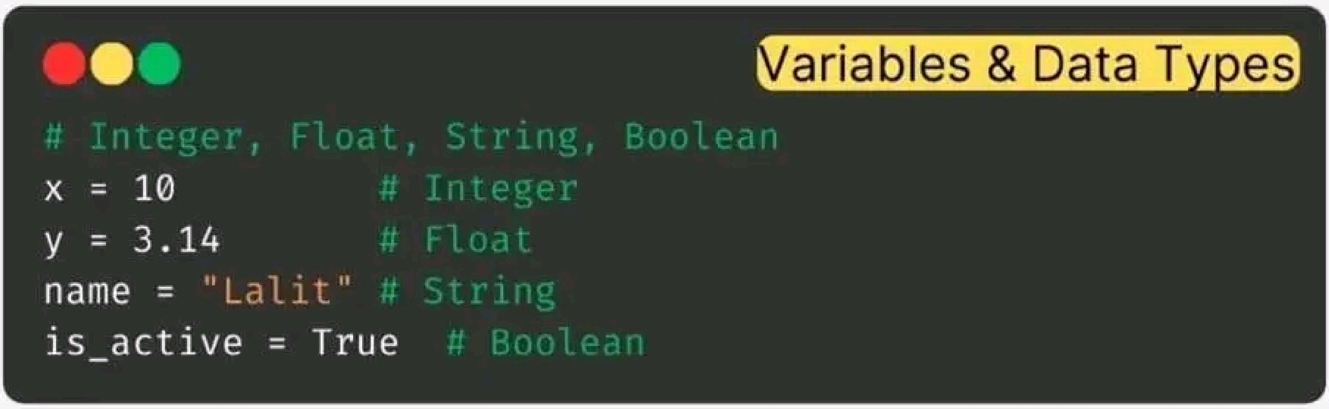
DATA SCIENCE



CHEAT SHEET

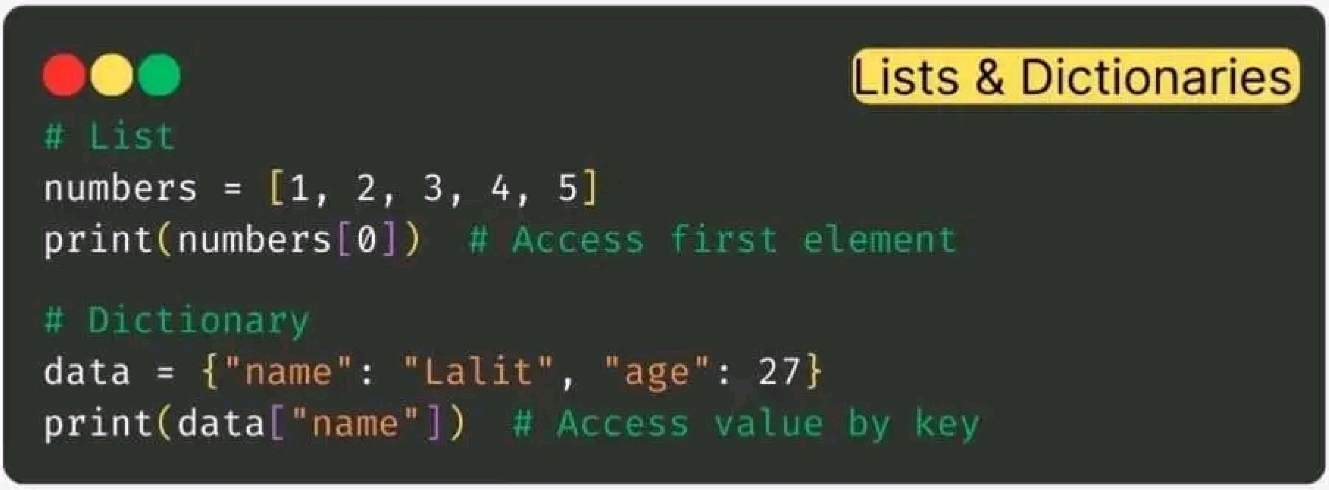
BASIC TO ADVANCE

1. Python Basics for Data Science



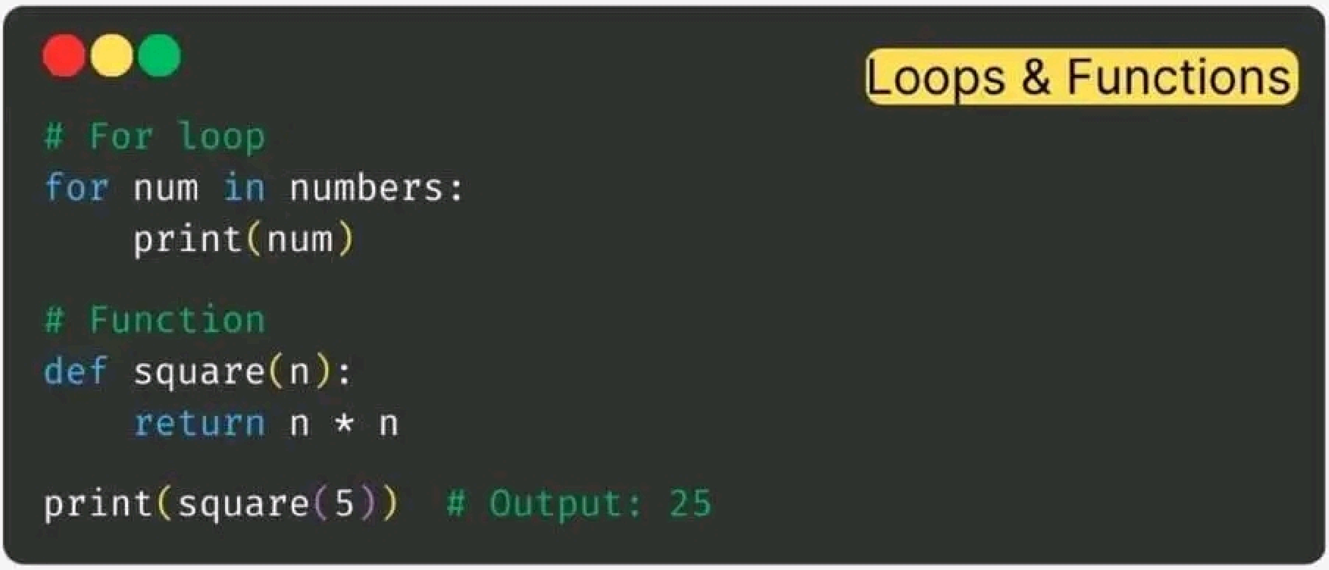
Variables & Data Types

```python
# Integer, Float, String, Boolean
x = 10          # Integer
y = 3.14        # Float
name = "Lalit" # String
is_active = True  # Boolean
```



Lists & Dictionaries

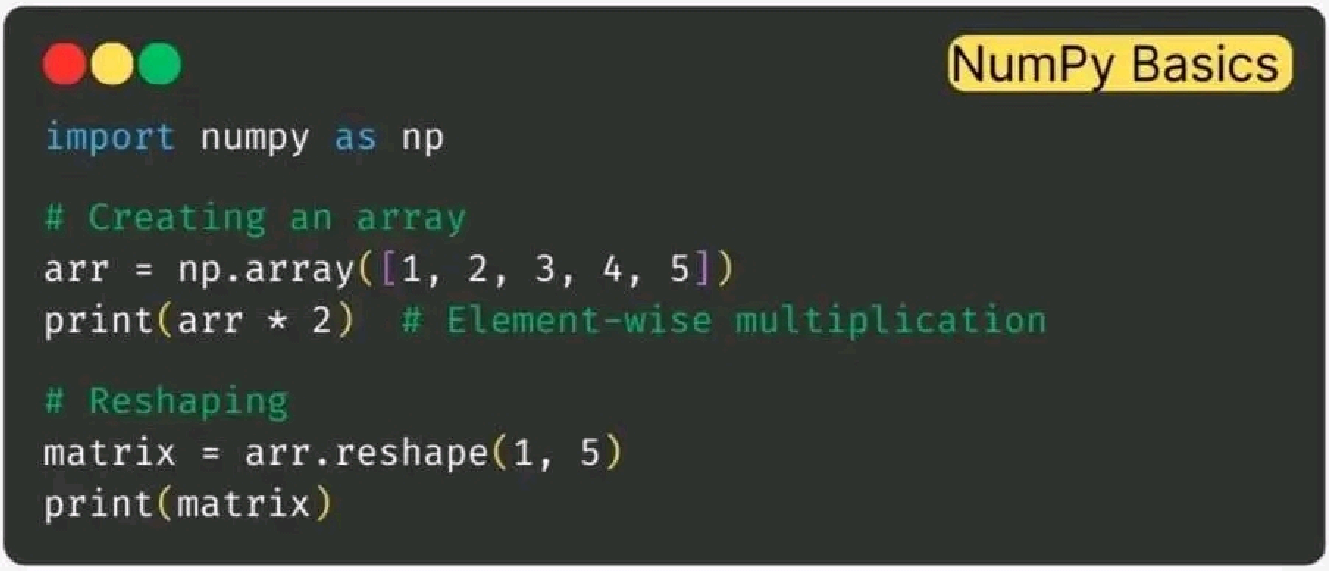
```python
# List
numbers = [1, 2, 3, 4, 5]
print(numbers[0])  # Access first element

# Dictionary
data = {"name": "Lalit", "age": 27}
print(data["name"])  # Access value by key
```



Loops & Functions

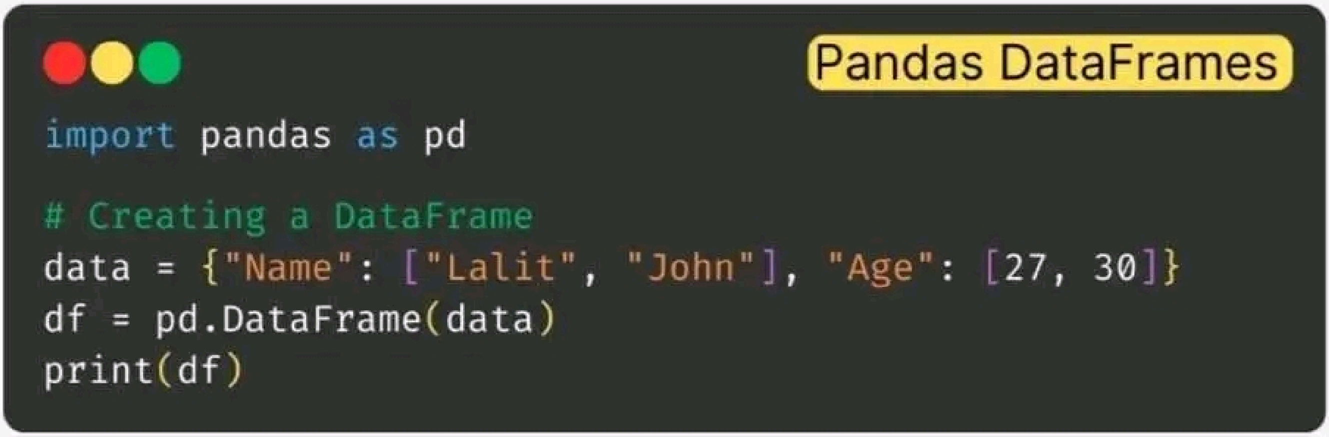
```python
# For loop
for num in numbers:
    print(num)

# Function
def square(n):
    return n * n

print(square(5))  # Output: 25
```

2. NumPy & Pandas (Data Manipulation)



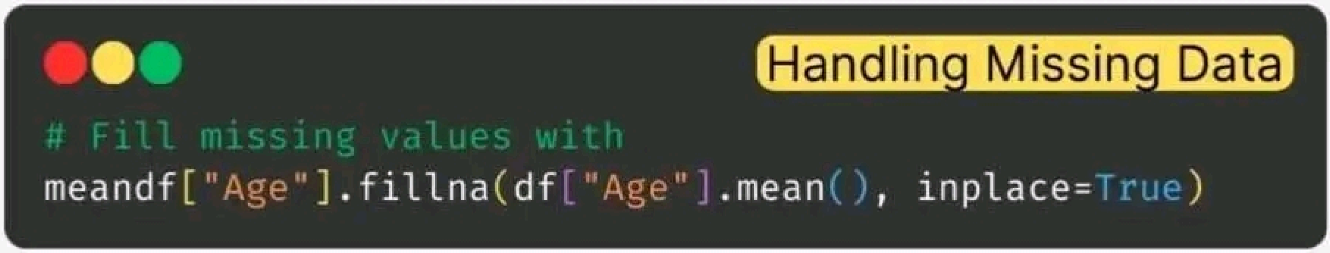
NumPy Basics

```
import numpy as np

# Creating an array
arr = np.array([1, 2, 3, 4, 5])
print(arr * 2)  # Element-wise multiplication

# Reshaping
matrix = arr.reshape(1, 5)
print(matrix)
```



Pandas DataFrames

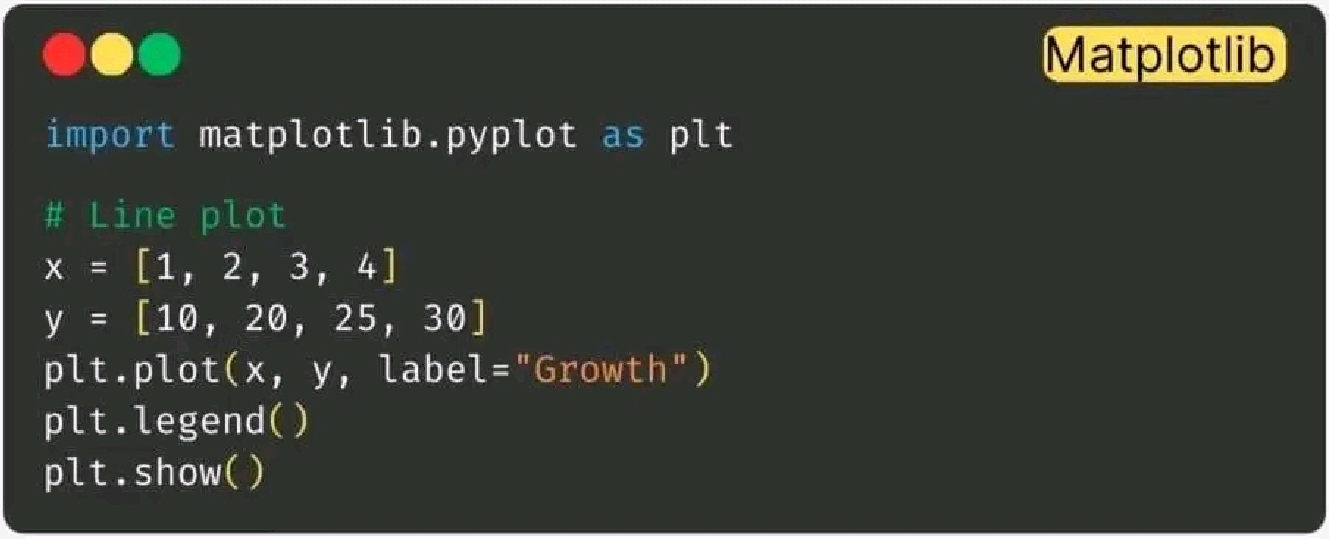
```
import pandas as pd

# Creating a DataFrame
data = {"Name": ["Lalit", "John"], "Age": [27, 30]}
df = pd.DataFrame(data)
print(df)
```



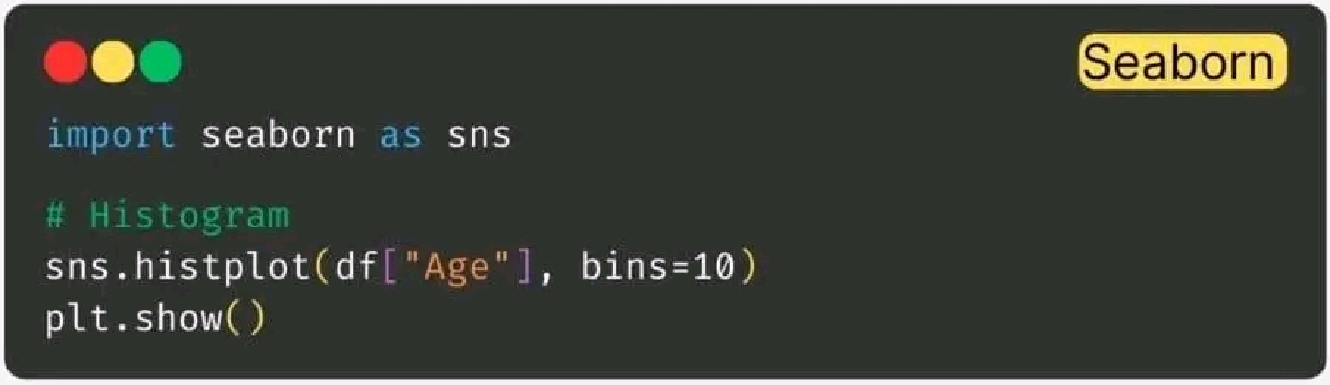
Handling Missing Data

```
# Fill missing values with
meandf["Age"].fillna(df["Age"].mean(), inplace=True)
```

3. Data Visualization (Matplotlib & Seaborn)

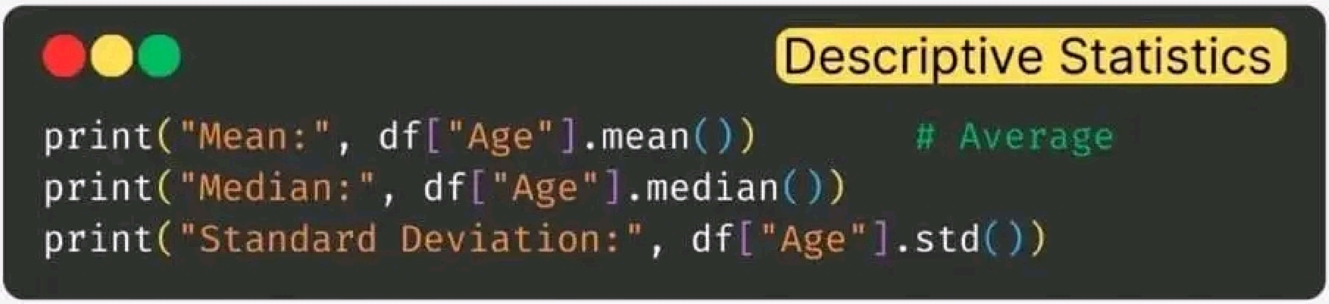


```python
import matplotlib.pyplot as plt

# Line plot
x = [1, 2, 3, 4]
y = [10, 20, 25, 30]
plt.plot(x, y, label="Growth")
plt.legend()
plt.show()
```

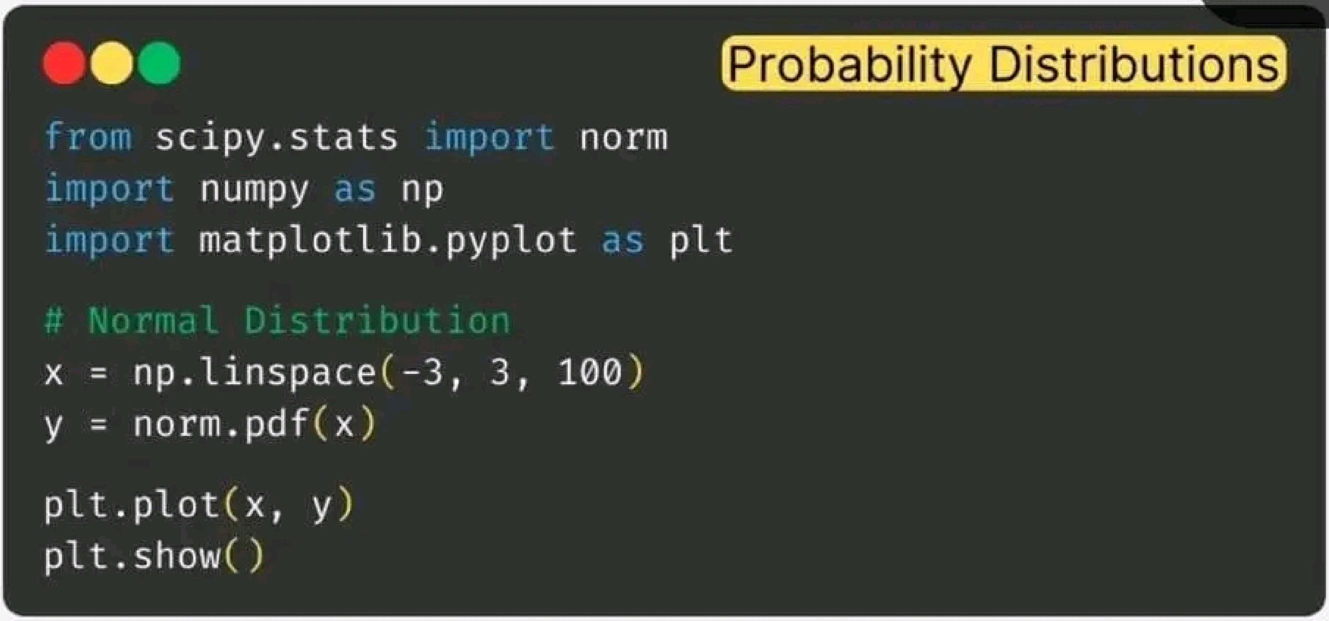


```python
import seaborn as sns

# Histogram
sns.histplot(df["Age"], bins=10)
plt.show()
```

4. Statistics & Probability

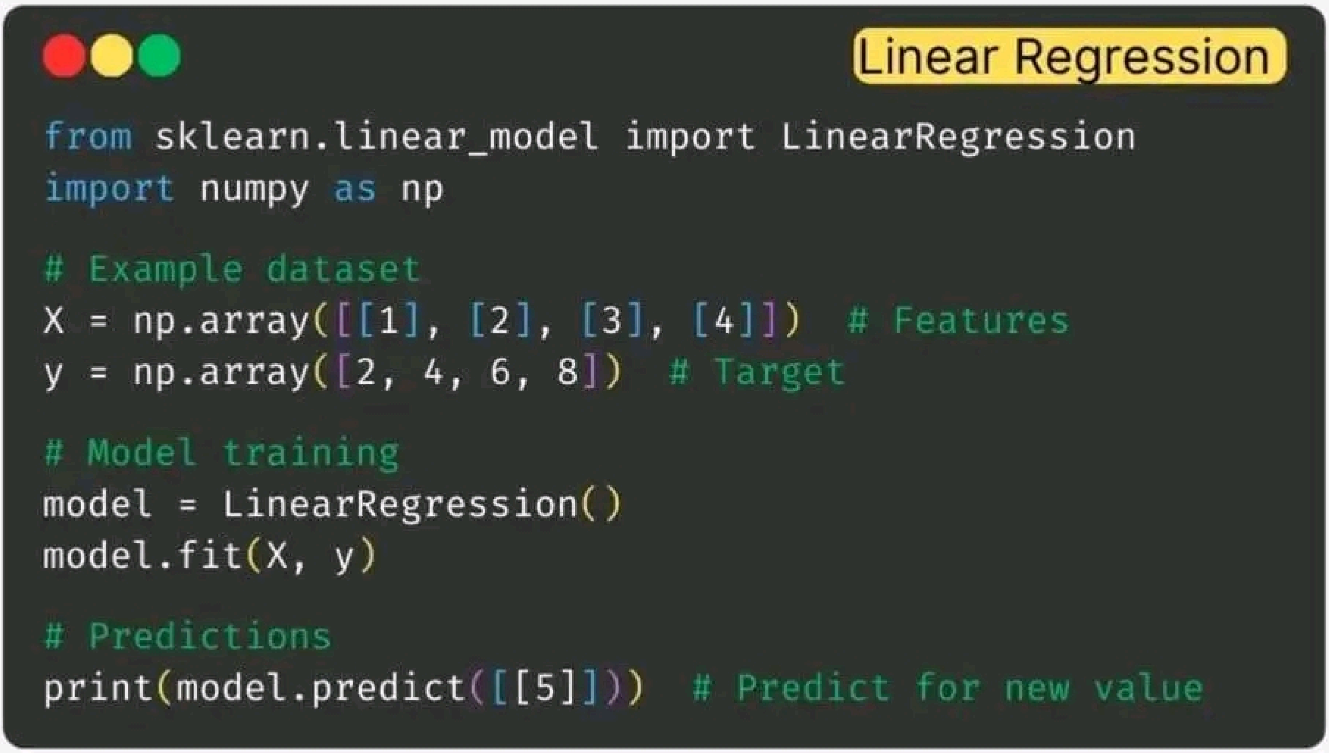


```python
print("Mean:", df["Age"].mean())        # Average
print("Median:", df["Age"].median())
print("Standard Deviation:", df["Age"].std())
```

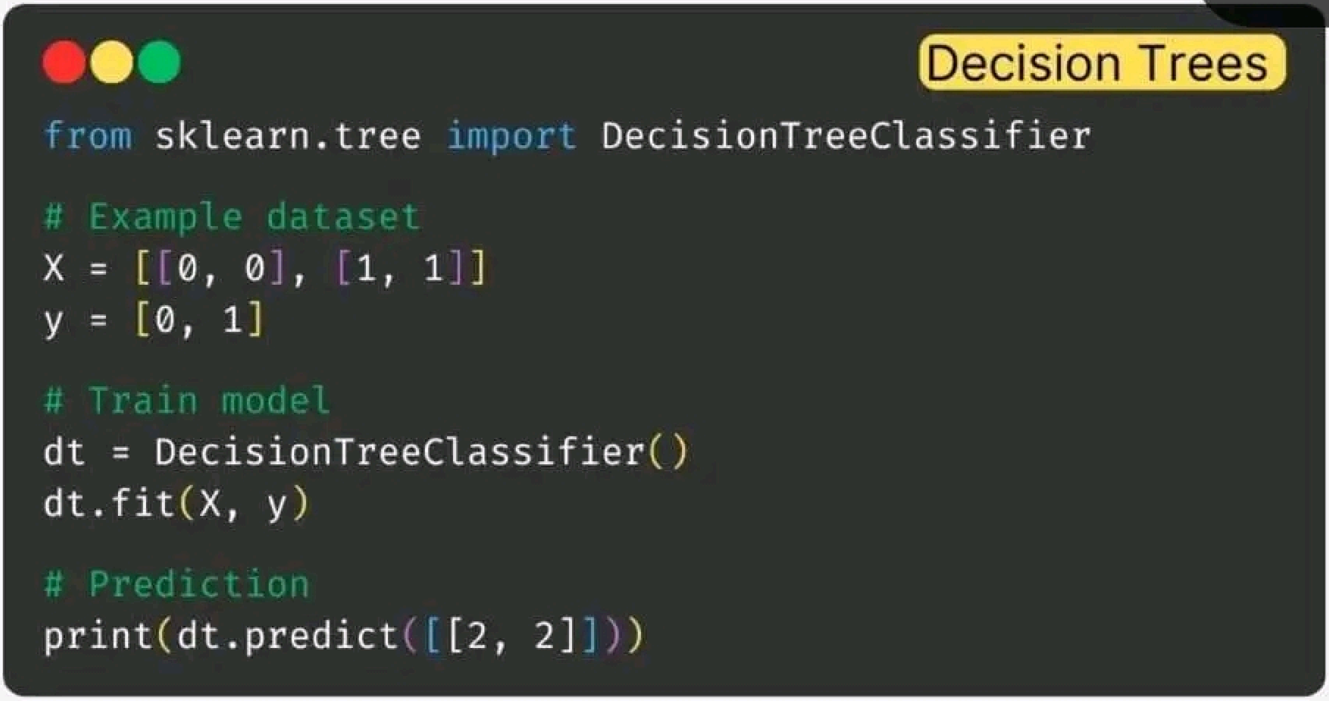


```python
from scipy.stats import norm
import numpy as np
import matplotlib.pyplot as plt

# Normal Distribution
x = np.linspace(-3, 3, 100)
y = norm.pdf(x)

plt.plot(x, y)
plt.show()
```

5. Machine Learning Algorithms

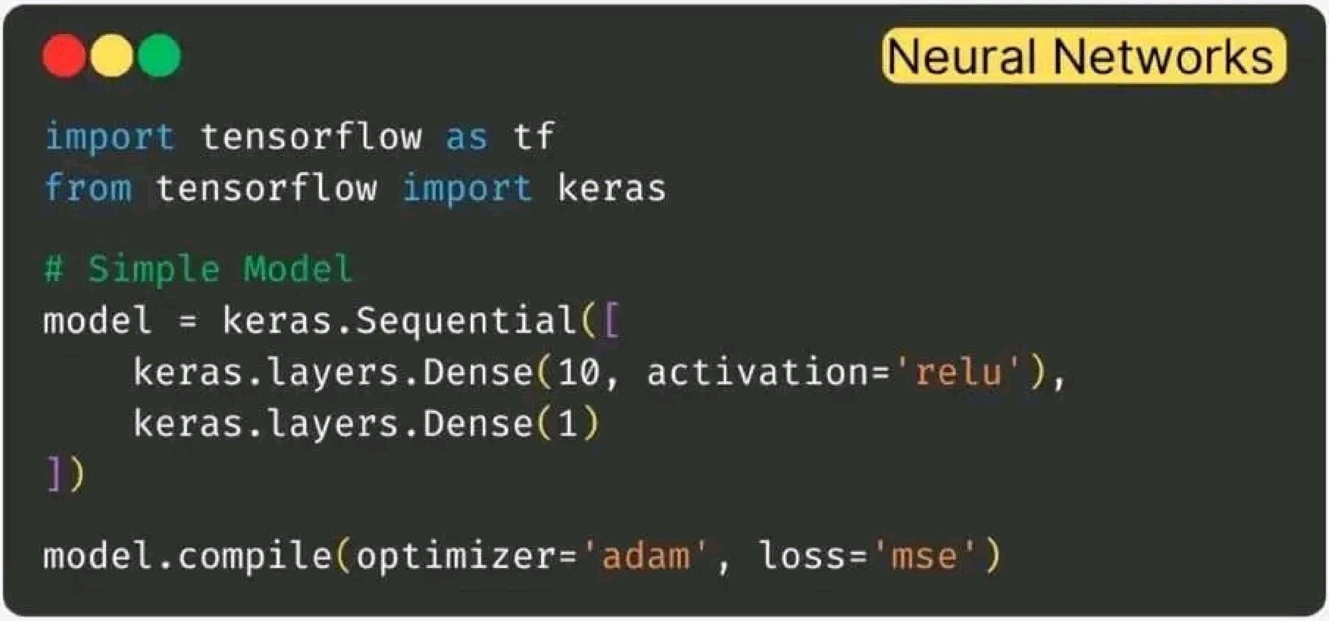


```python
from sklearn.linear_model import LinearRegression
import numpy as np

# Example dataset
X = np.array([[1], [2], [3], [4]])  # Features
y = np.array([2, 4, 6, 8])  # Target

# Model training
model = LinearRegression()
model.fit(X, y)

# Predictions
print(model.predict([[5]]))  # Predict for new value
```



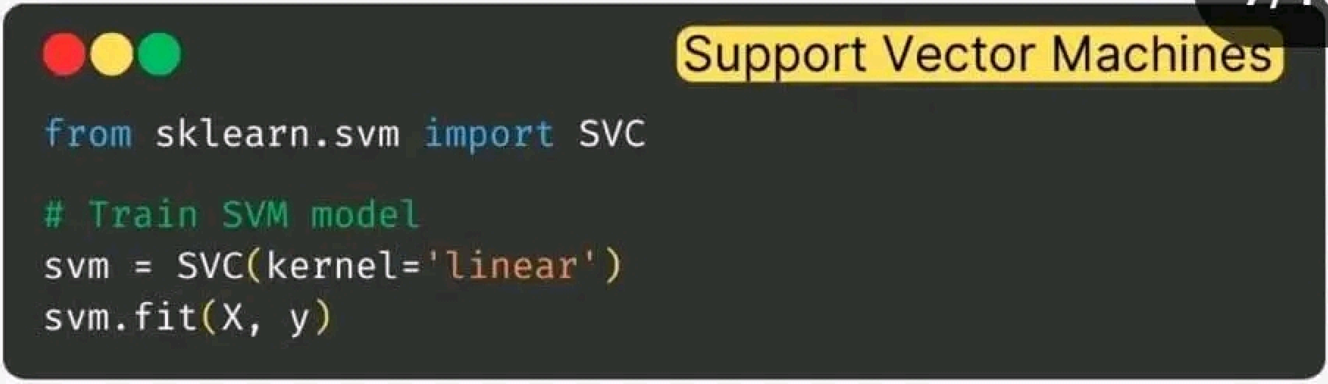
```python
from sklearn.tree import DecisionTreeClassifier

# Example dataset
X = [[0, 0], [1, 1]]
y = [0, 1]

# Train model
dt = DecisionTreeClassifier()
dt.fit(X, y)

# Prediction
print(dt.predict([[2, 2]]))
```

6. Deep Learning Basics

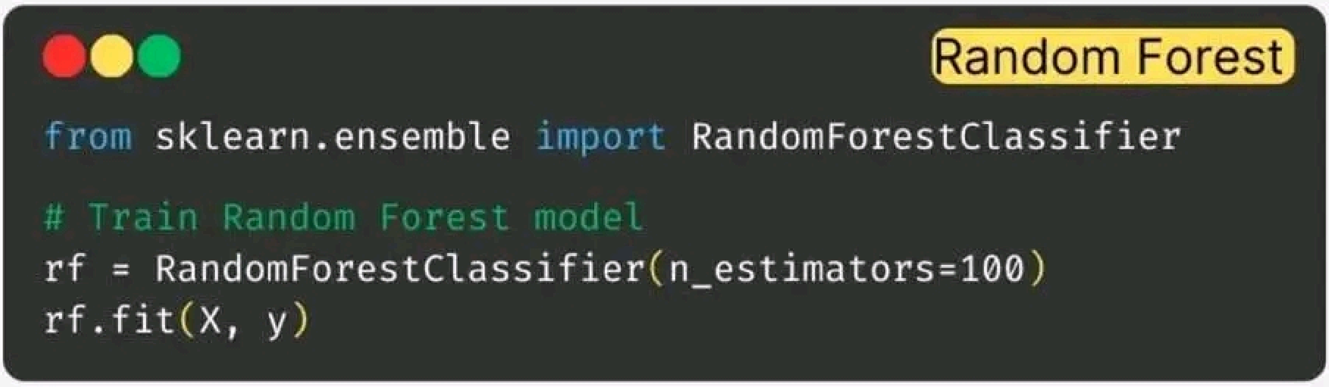


```python
import tensorflow as tf
from tensorflow import keras

# Simple Model
model = keras.Sequential([
    keras.layers.Dense(10, activation='relu'),
    keras.layers.Dense(1)
])

model.compile(optimizer='adam', loss='mse')
```

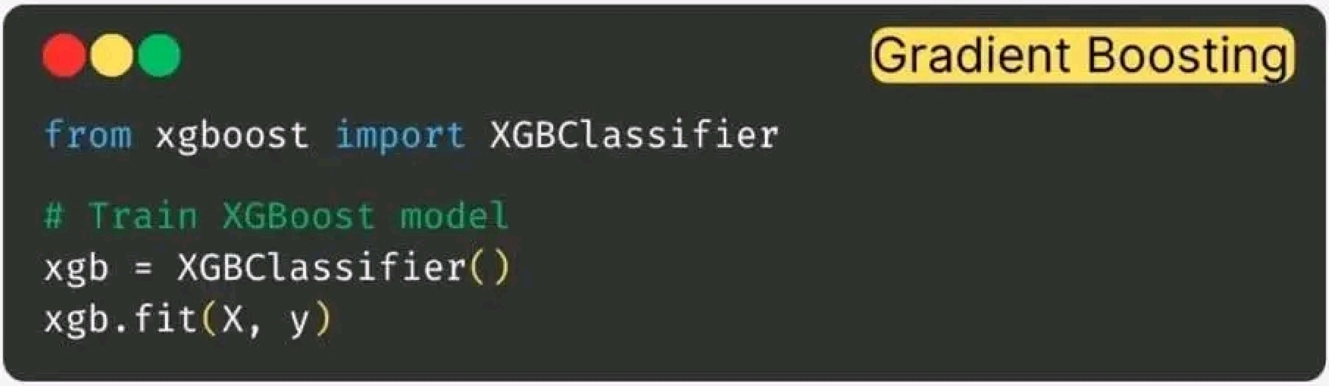
7. Advanced Machine Learning



```
from sklearn.svm import SVC

# Train SVM model
svm = SVC(kernel='linear')
svm.fit(X, y)
```

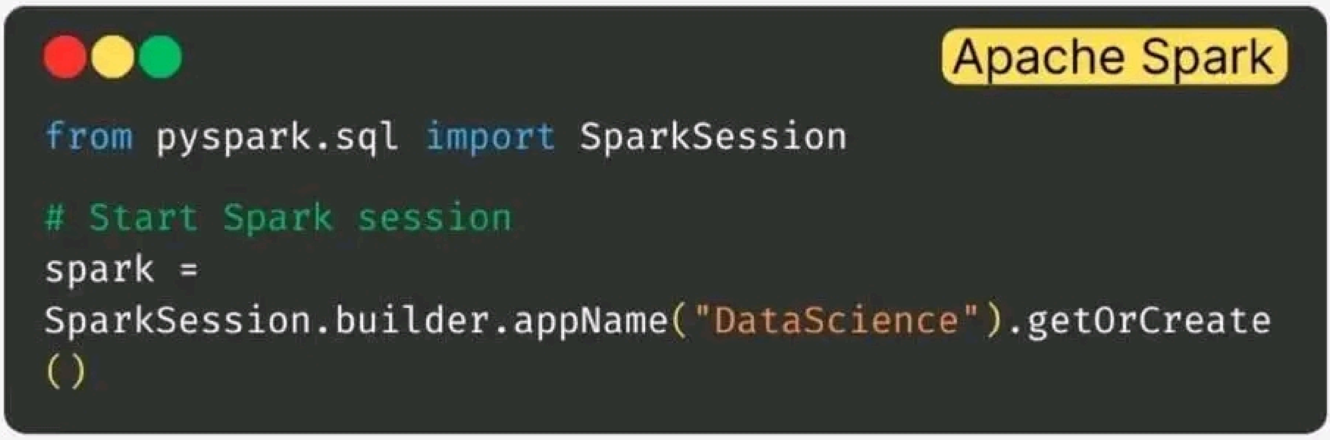


```
from sklearn.ensemble import RandomForestClassifier

# Train Random Forest model
rf = RandomForestClassifier(n_estimators=100)
rf.fit(X, y)
```

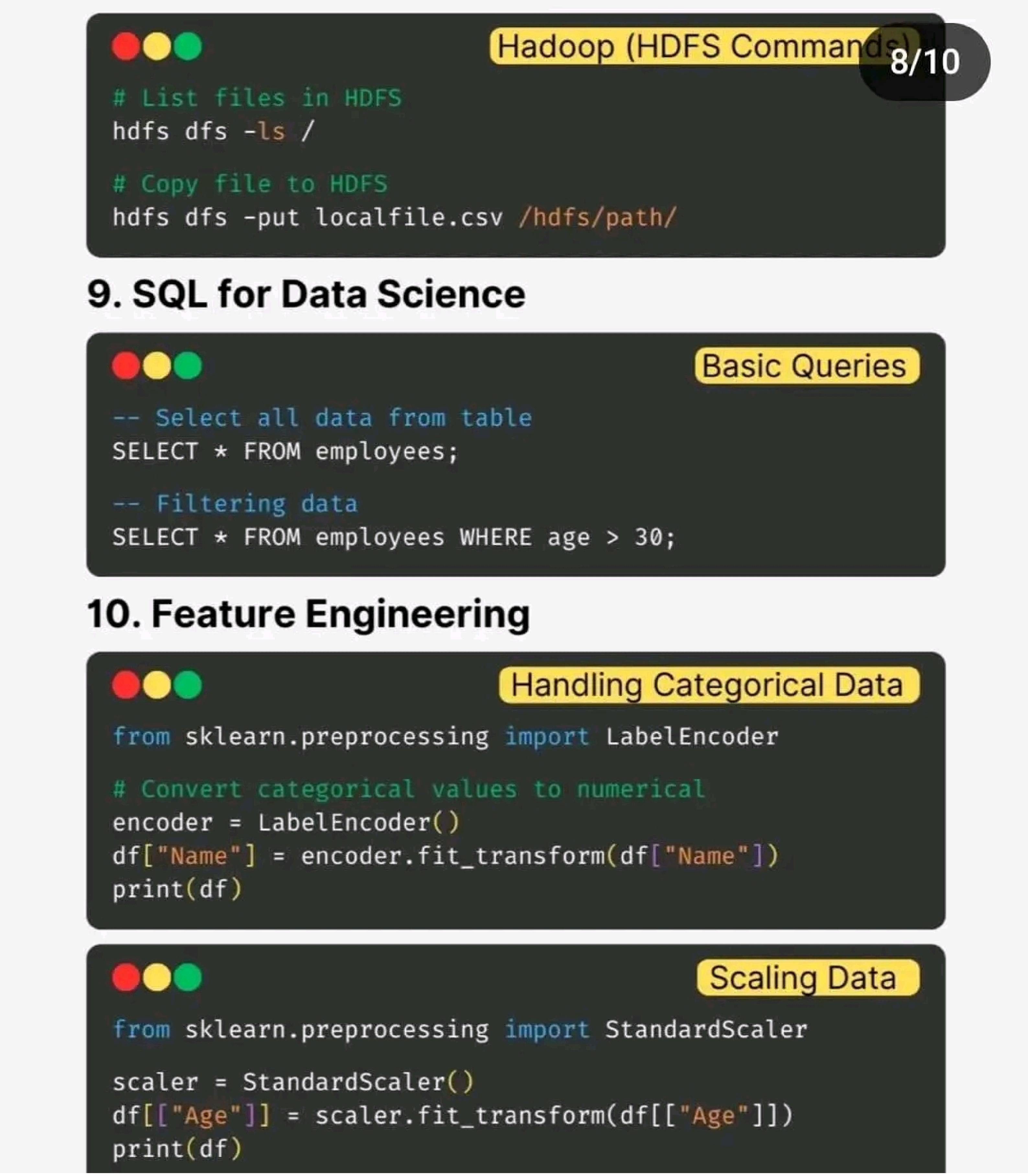


```
from xgboost import XGBClassifier

# Train XGBoost model
xgb = XGBClassifier()
xgb.fit(X, y)
```

8. Big Data Tools



```
from pyspark.sql import SparkSession

# Start Spark session
spark =
SparkSession.builder.appName("DataScience").getOrCreate
()
```



8/10

Hadoop (HDFS Commands)

```
# List files in HDFS
hdfs dfs -ls /

# Copy file to HDFS
hdfs dfs -put localfile.csv /hdfs/path/
```

9. SQL for Data Science

Basic Queries

```
-- Select all data from table
SELECT * FROM employees;

-- Filtering data
SELECT * FROM employees WHERE age > 30;
```

10. Feature Engineering

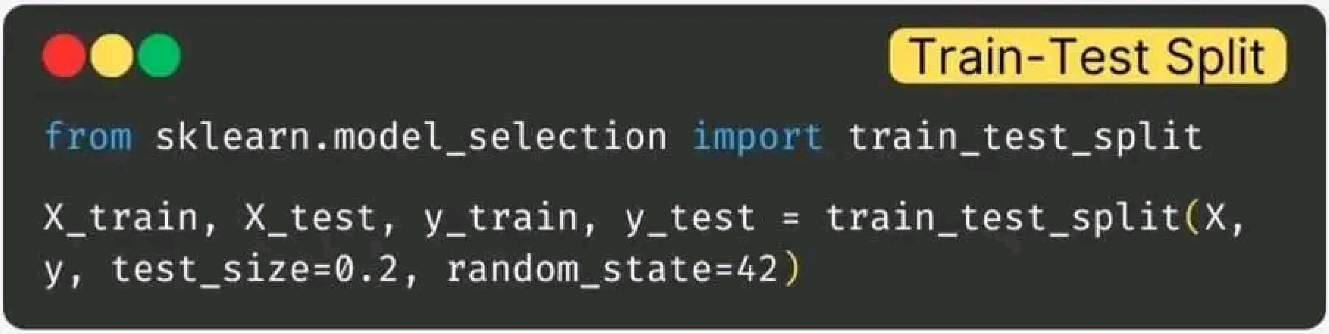
Handling Categorical Data

```
from sklearn.preprocessing import LabelEncoder

# Convert categorical values to numerical
encoder = LabelEncoder()
df["Name"] = encoder.fit_transform(df["Name"])
print(df)
```

Scaling Data

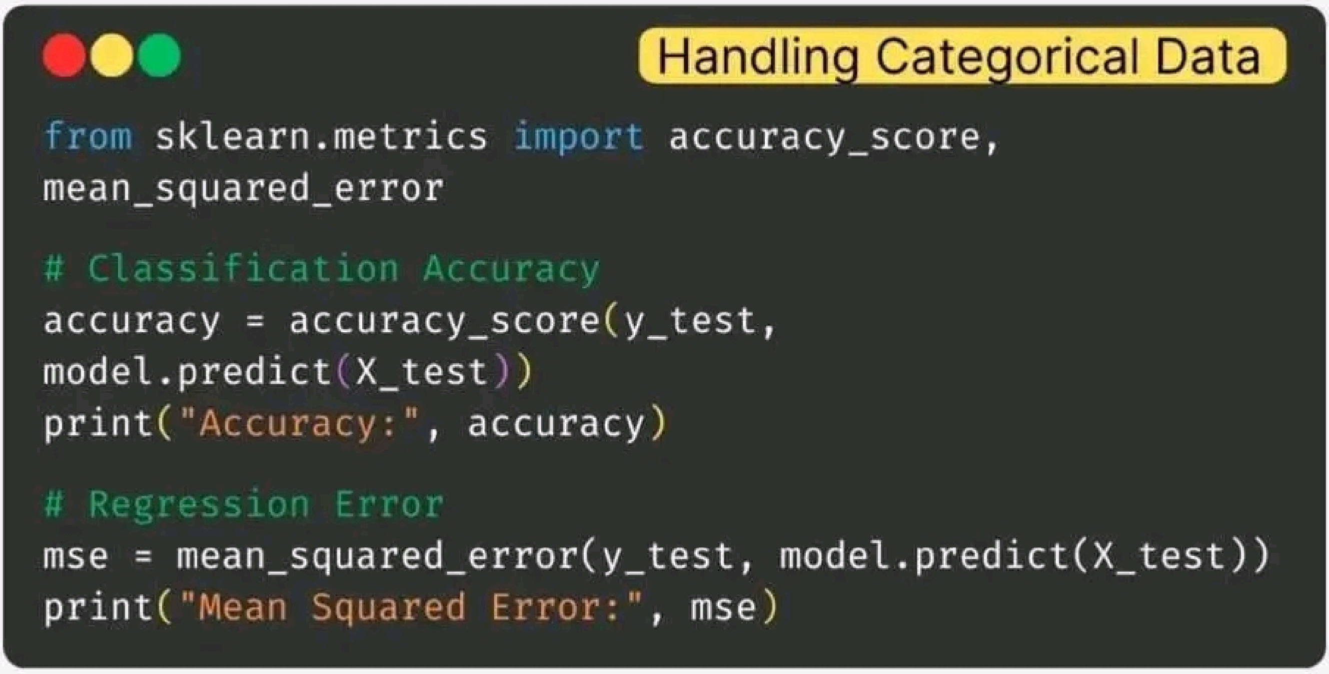
```
from sklearn.preprocessing import StandardScaler

scaler = StandardScaler()
df[["Age"]] = scaler.fit_transform(df[["Age"]])
print(df)
```

11. Model Evaluation



Train-Test Split

```python
from sklearn.model_selection import train_test_split

X_train, X_test, y_train, y_test = train_test_split(X,
y, test_size=0.2, random_state=42)
```



Handling Categorical Data

```python
from sklearn.metrics import accuracy_score,
mean_squared_error

# Classification Accuracy
accuracy = accuracy_score(y_test,
model.predict(X_test))
print("Accuracy:", accuracy)

# Regression Error
mse = mean_squared_error(y_test, model.predict(X_test))
print("Mean Squared Error:", mse)
```

Conclusion

This cheat sheet provides essential concepts for data science, covering Python basics, data visualization, machine learning, deep learning, big data tools, SQL, and model evaluation. Keep practicing these concepts to gain expertise!