Python A-Z Cheatsheet for Data Analysts and Data Engineers

A - Arrays

- NumPy Arrays:
  - numpy.array([1, 2, 3]): Creates a NumPy array.
  - numpy.zeros((2, 3)): Creates an array filled with zeros.
  - numpy.ones((2, 3)): Creates an array filled with ones.
  - numpy.arange(0, 10, 2): Creates an array with a range of values.
  - numpy.linspace(0, 1, 5): Creates an array with a specified number of evenly spaced values.

B - Boolean Operations

- Comparison Operators:
  - ==, !=, <, >, <=, >=: Used to compare values.
- Logical Operators:
  - and, or, not: Used to combine Boolean expressions.

C - Conditional Statements

- If-Else:

```
if condition:
    # code
elif condition:
    # code
else:
    # code
```

D - Data Frames

- Pandas DataFrames:

  - pandas.DataFrame(data): Creates a DataFrame from data.

  - df.head(): Shows the first few rows of the DataFrame.

  - df.describe(): Provides statistical summary.

  - df.info(): Displays DataFrame information.

  - df.drop(columns=['column_name']): Removes specified columns.

  - df.groupby('column_name'): Groups data based on column values.

E - Exception Handling

- Try-Except:

```
  try:
      # code that may raise an exception
  except Exception as e:
      # code to handle the exception
  finally:
      # code that will run no matter what
```

F - Functions

- Defining Functions:

```
  def func_name(params):
      # code
      return value
```

- Lambda Functions:

  lambda x: x + 1: A short anonymous function.

G - Graphs and Plotting

- Matplotlib:

matplotlib.pyplot.plot(x, y): Plots data.

matplotlib.pyplot.show(): Displays the plot.

- Seaborn:

seaborn.heatmap(data): Creates a heatmap.

seaborn.scatterplot(x, y): Creates a scatter plot.

H - Handling Missing Data

- Pandas:

df.fillna(value): Replaces missing values with specified value.

df.dropna(): Removes rows with missing values.

I - Iteration

- Loops:

for item in iterable:: Iterates over an iterable object.

while condition:: Repeats as long as the condition is true.

J - JSON Handling

- JSON Operations:

json.load(file): Reads JSON data from a file.

json.dump(data, file): Writes JSON data to a file.

json.loads(string): Parses JSON from a string.

json.dumps(data): Converts Python data to a JSON string.

K - Key Libraries

- NumPy: For numerical operations.

- Pandas: For data manipulation and analysis.

- Matplotlib: For data visualization.

- Seaborn: For statistical data visualization.

- Scikit-Learn: For machine learning.

L - Lists

- Basic Operations:

  list.append(item): Adds an item to the end.

  list.remove(item): Removes the first occurrence of an item.

  list.sort(): Sorts the list in place.

M - Merging Data

- Pandas Merge:

  pd.merge(df1, df2, on='key'): Merges two DataFrames on a key.

  pd.concat([df1, df2]): Concatenates two DataFrames.

N - NumPy Operations

- Array Operations:

  numpy.mean(array): Calculates the mean of array elements.

  numpy.median(array): Calculates the median.

  numpy.std(array): Calculates the standard deviation.

  numpy.sum(array): Sums up the array elements.

O - Object-Oriented Programming

- Classes:

```
  class ClassName:
      def __init__(self, attribute):
          self.attribute = attribute
      def method(self):
```

```
    # code
```

P - Plotting

- Plot Types:

  plt.plot(x, y): Creates a line plot.

  plt.bar(x, height): Creates a bar plot.

  plt.hist(data): Creates a histogram.

Q - Querying Data

- Pandas Query:

  df.query('column > value'): Filters DataFrame based on a condition.

  df.loc[]: Accesses a group of rows and columns by labels.

  df.iloc[]: Accesses a group of rows and columns by integer position.

R - Reading/Writing Files

- CSV Files:

  pd.read_csv('file.csv'): Reads CSV data into a DataFrame.

  df.to_csv('file.csv'): Writes DataFrame to a CSV file.

- Excel Files:

  pd.read_excel('file.xlsx'): Reads Excel data into a DataFrame.

  df.to_excel('file.xlsx'): Writes DataFrame to an Excel file.

S - Statistics

- Basic Statistics:

  numpy.mean(): Calculates the mean.

  numpy.median(): Calculates the median.

  pandas.describe(): Provides statistical summary of DataFrame.

T - Time Series

- Pandas Time Series:

  pd.to_datetime(data): Converts data to datetime.

  df.resample('M').mean(): Resamples time series data.

  df.shift(): Shifts data by a specified period.

U - User-Defined Exceptions

- Creating Exceptions:

  class CustomException(Exception):

      pass

V - Variables

- Scope:

  global variable_name: Accesses a global variable within a function.

  nonlocal variable_name: Accesses a variable from an outer scope.

W - Web Scraping

- Requests Library:

  requests.get('url'): Sends a GET request to a URL.

  response.text: Retrieves the content of the response.

X - XML Handling

- XML Parsing:

  xml.etree.ElementTree.parse('file.xml'): Parses XML from a file.

  xml.etree.ElementTree.Element(tag): Creates an XML element.

Y - YAML Handling

- PyYAML:

  yaml.load(file, Loader=yaml.FullLoader): Loads YAML data from a file.

  yaml.dump(data, file): Dumps Python data to a YAML file.

Z - Zipping Files

- Zip Files:

  zipfile.ZipFile('file.zip', 'w'): Creates a new zip file.

  zipfile.extractall('path'): Extracts all contents of the zip file.