Artificial Intelligence

BS (CS) \_SP\_2024

# Lab\_02Manual



Learning Objectives:

1. Lambda Functions and Regular Expressions
2. Introduction to Numpy
3. Introduction to Pandas
4. Matplotlib

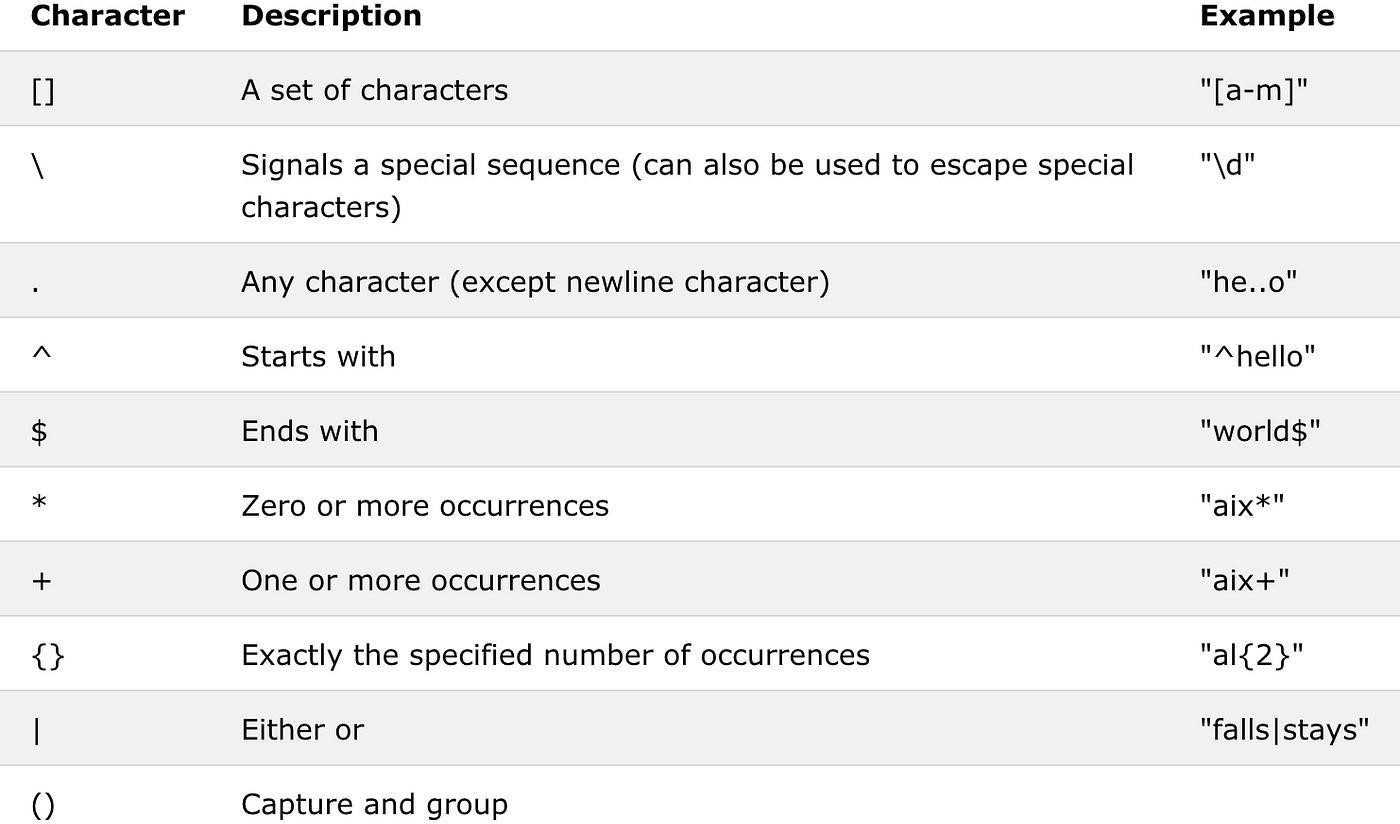
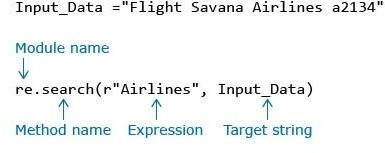
# **Lab Manual**

**Regular expression:**

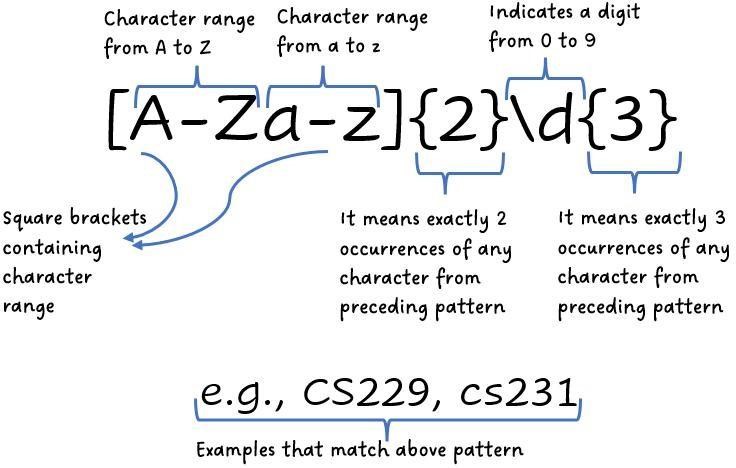
Many times a lot of data would be stored in files and we may have to pick and change only relevant portions from a file. Even though there are string functions that allow us to manipulate strings, when dealing with more complicated requirements, we would need more powerful tools.

**Regular expressions (regex)** in Python provide a powerful way to search, manipulate, and validate strings based on specific patterns. The re module in Python allows you to work with regular expressions.

Python has a module named ‘re‘ for regular expressions.



**Example:**



**Lambda Functions:**

Lambdas are functions without names, in other words they are anonymous functions. They take inputs and return outputs but do not have a name. They are shortcuts to create simple temporary functions.

**Syntax:**

lambda arguments : expression

**Example:**

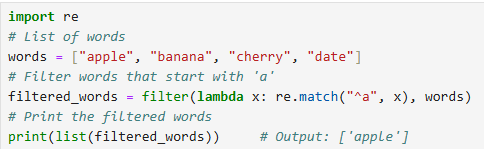
A screenshot of a computer code

Description automatically generated

**Lambda Functions(regular expression basis):**

In the context of regular expressions (regex), lambda functions can be used in languages that support them to define custom behavior for matching patterns. For example, in Python, you can use lambda functions with the re module to perform regex operations.

**Example:**

****

**Introduction to NumPy:**

**Numpy:**

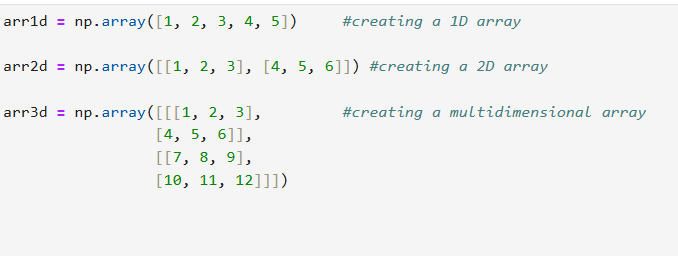
Numpy is a python library that deals with numerical data and arrays. Python lists same purpose and arrays but numpy arrays are far faster than the traditional python lists. The array object in nunpy is called ndarray which is created by np.array()

1. **Install numpy:**

pip install numpy

1. **Import numpy:**

import numpy as np

1. **Create numpy** **array:**

**Different Numpy Operations:**

A white sheet with black text

Description automatically generated

A cross-section of a graph

Description automatically generated

**Accessing and Slicing:**

**Access through their index:**

A group of numbers and symbols

Description automatically generated

**Slicing***:*

Slicing general pattern goes like: [start : end].

We can also define the step, like this: [start : end : step].

* If we don't pass start it considered 0
* If we don't pass end its considered length of array in that dimension.
* If we don't pass step it considered 1

A screenshot of a computer code

Description automatically generated

## Pandas

Pandas is a powerful Python library for data manipulation and analysis. It provides data structures like Series and DataFrame, which are designed for handling structured data efficiently.

**Installing pandas:**

If you haven't installed pandas yet, you can do so using pip:

pip install pandas

**Importing pandas:**

In your Python script or notebook, import pandas as follows:

import pandas as pd

**Creating Pandas Dataframe:**

data = {'Name': ['Alice', 'Bob', 'Charlie', 'David'], 'Age': [25, 30, 35, 40]}

df = pd.DataFrame(data)

* Saving my dataframe into a csv file: df.to\_csv("Abc.csv")
* Reading csv and storing it into dataframe: df1=pd.read\_csv("Abc.csv")
* Assign values to a specific column: values=[18,19, 17, 18] df["age"]=values
* Use to select specific rows by their index labels: df.loc[[3]]
* Display details of specific column: df["column\_name"]
* Consice summary of dataframe: df.info()
* Display first 5 elements of the dataframe: df.head(5)
* Statistical details of the dataframe: df.describe()

## MatplotLib

Matplotlib is a popular Python library for creating static, animated, and interactive visualizations. It provides a wide range of plotting functions and customization options for creating high-quality plots.

**Installing Matplotlib:**

If you haven't installed matplotlib yet, you can do so using pip: pip install matplotlib

**Importing Matplotlib:**

In your Python script or notebook, import matplotlib's pyplot module, which provides a MATLABlike interface for plotting:

import matplotlib.pyplot as plt

**Simple Line Plot:**

Use the plot function to create a simple line plot:

A screenshot of a computer program

Description automatically generated

**Customizing Plots:**

A black and purple text

Description automatically generatedYou can customize various aspects of the plot, such as colors, line styles, markers, and more:

**Multiple Plots:**

You can create multiple plots in the same figure using subplot:

A screenshot of a computer program

Description automatically generated

**Scatter Plot**

A screenshot of a computer

Description automatically generated

A graph with blue dots

Description automatically generated

**Other Plots**

* **Bar Plot:** plt.bar(x, height)
* **Histogram:** plt.hist(data, bins)
* **Box Plot:** plt.boxplot(data)
* **Scatter Plot:** plt.scatter(x, y)
* **Pie Chart:** plt.pie(data, labels)
* **Heatmap:** sns.heatmap(data) # using seaborn library