LECTURE - 27

WEB SCRAPING:

It is a term used to describe the use of a program or algorithm to extract and Process large amounts of data from the web.

In this you will learn about:

1. DATA EXTRACTION from the web using PYTHON's BEAUTIFULSOUP module.
2. DATA MANIPULATION & CLEANING using PYTHON'S PANDAS library.
3. DATA VISUALIZATION using PYTHON's MATPLOTLIB library.

The step by step procedure of Web Scraping are as follows:-

STEP-1: IMPORTING NECESSARY LIBRARIES

```
import numpy as np
import pandas as pd
import re
import requests
from bs4 import BeautifulSoup
import matplotlib as plt
import seaborn as sns
import plotly.express as px
```

↳ request : This is used to extract the HTML code of the given URL

↳ Beautifulsoup : Scrape and format the data from the HTML

STEP-2: ACCESSING THE HTML CONTENT FROM WEBPAGE

```
URL = '.....'
page = requests.get(URL)
```

```
page.status_code

htmlCode = page.text

htmlcode
```

STEP-3: PARSING THE HTML CONTENT

```
soup = BeautifulSoup(htmlCode)

soup

Print(soup.prettify())
```

OR EXTRACT

STEP-4: FIND THE TABLES FROM THE URL

```
all_tables = soup.find_all('table')
```

STEP-5: SELECT THE REQUIRED TABLE FROM ALL ACQUIRED TABLES

```
my_table = soup.find('table', {'class': ' '})
```

STEP-6: SCRAPING THE DATA FROM WEBPAGE

STEP-7: CONCATENATE THE TWO DATAFRAMES ALONG COLUMNS

STEP-8: VIEW OF FINAL DATA FRAME

STEP-9: EXPORT DATAFRAME TO EXCEL/CSV

STEP-10: FILLING THE NULL VALUES (NaN/NA)

df.isnull().sum()

STEP-11: REMOVING - CLEANING THE DATA FRAME USING METACHARACTERS & PRE-DEFINED CHARACTER CLASSES

STEP-12: EXPLORATORY DATA ANALYSIS

STEP-13: PLOTTING THE GRAPHS FROM THE FINAL DATA FRAME. (HIST, HEXA, BOX... etc)

STEP-14: CONCLUSIONS - OBSERVATIONS OF THE GRAPHS OBTAINED