ERODE SENGUNTHAR
ENGINEERING COLLEGE

(APPROVED BY AICTE, NEW DELHI & PERMANENTLY AFFILIATED TO ANNA UNIVERSITY, CHENNAI.
ACCREDITED BY NBA, NEW DELHI, NAAC WITH GRADE "A" & IE(I),KOLKATA)

PERUNDURAI, ERODE – 638 057

An Autonomous Institution

BONAFIDE CERTIFICATE

Register No: .....

Certified that this is the Bonafide Record of Work Done By

Name of the Student :.....

Branch :.....

Lab Code/Name :.....

Year/Semester :.....

Faculty Incharge Head of the Department

Submitted for the End Semester Practical Examination held on .....

Internal Examiner *External Examiner*

INDEX

10		Build Chatbot	38		
		CONTENT BEYOND SYLLABUS			
		DATA VISUALIZATION			
11		Cryptocurrency Data Visualization using Power BI	41		
		NATURAL LANGUAGE PROCESSING			
12		Dependency Parsing using NLP	44		

DATA VISUALIZATION

EX NO: 1 DATE:	ACQUIRING AND PLOTTING DATA

AIM:

To acquire and plotting data using python libraries

PROCEDURE:

Step 1:Install the necessary Python libraries such as pandas, matplotlib, seaborn, etc. You can use the command pip install pandas matplotlib seaborn in your terminal or command prompt.

Step 2:Load the data into a Pandas dataframe. You can use the read_csv function from the Pandas library to load a CSV file or read_excel function to load an Excel file.

Step 3:Clean and pre-process the data as necessary using Pandas functions such as dropna, fillna, groupby, etc.(If required)

Step 4: Load that data to dataset of Power BI and Check that fields you want to visualize

Step 5:Add the custom visual to your PowerBI report and use it to display the plot.

PROGRAM:

In command prompt install the following libraries

Pip install seaborn

Pip install matplotlib

Pip install pandas

In powerbi,

getdata->more->Search for ‘Python script’

In python Script

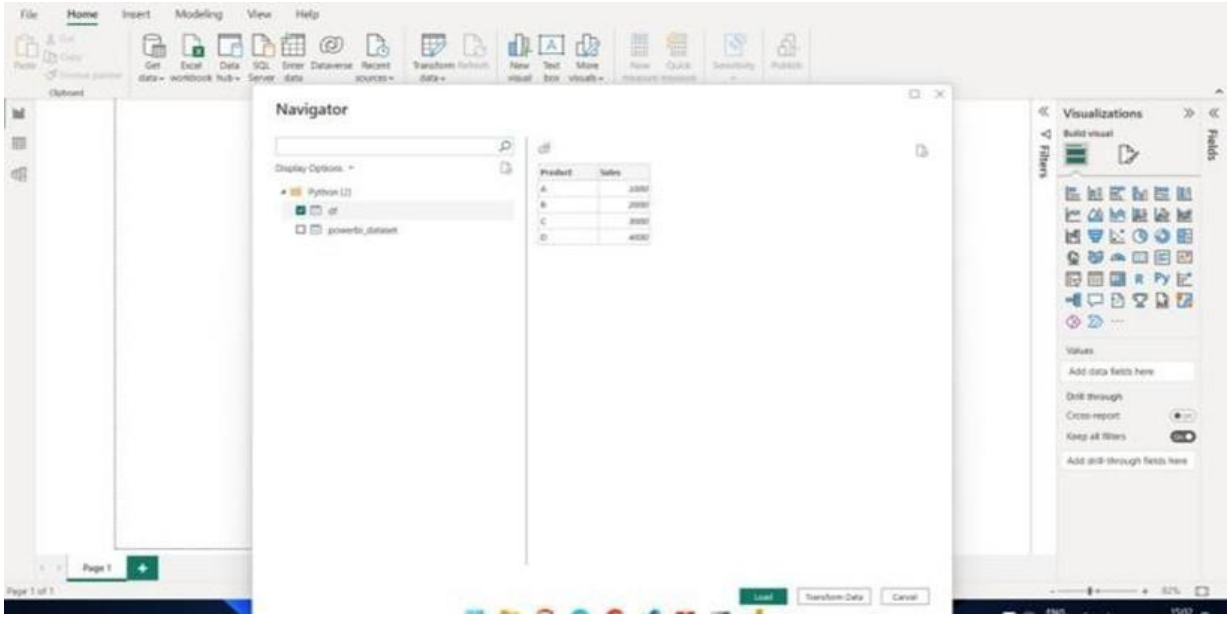
Import pandas as pd

data={‘Product’:[‘’A’,’B’,’C’,’D’],’Sale’:[1000,2000,3000,4000]}

df=pd.DataFrame(data)

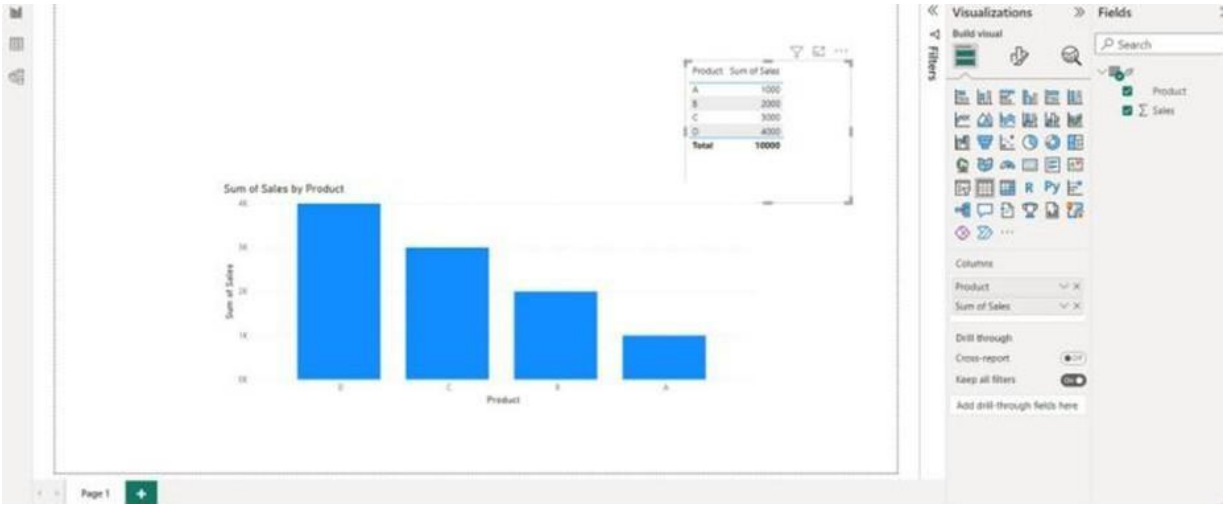
df.to_csv(‘powerbi_dataset.csv’,index=False)

Then goto-> try a sample dataset



Select ‘df’ datatset as navigator and click ‘Load’ to load data. In fields, check out the field which you want to visualize and add the custom visual on ‘Visulaizations’ tab on powerbi and use it to display the plot

OUTPUT:



Dept. of AI&DS

Description	Max Marks	Awarded
Aim	5	
Software/Tools Required & Algorithm	10	
Coding/ Programming & Execution	20	
Record	20	
Viva Voce	10	
Result	10	
TOTAL	75	

RESULT:

Acquiring and plotting data in Power BI with Python libraries enables users to leverage the power of Python for advanced data analysis and visualization, while utilizing the reporting capabilities of Power BI.

EX NO: 2 DATE:	TIME SERIES ANALYSIS - STOCK MARKET

AIM:

To perform time series analysis using stock market

PROCEDURE:

1. Import Data: Start by importing the stock market data into Power BI. The data can be sourced from various online data providers or through APIs.

2. Clean and transform the data: The next step is to clean and transform the data. This may involve removing null values, dealing with outliers, converting data types, and aggregating the data to the desired time intervals (daily, weekly, monthly, etc.).

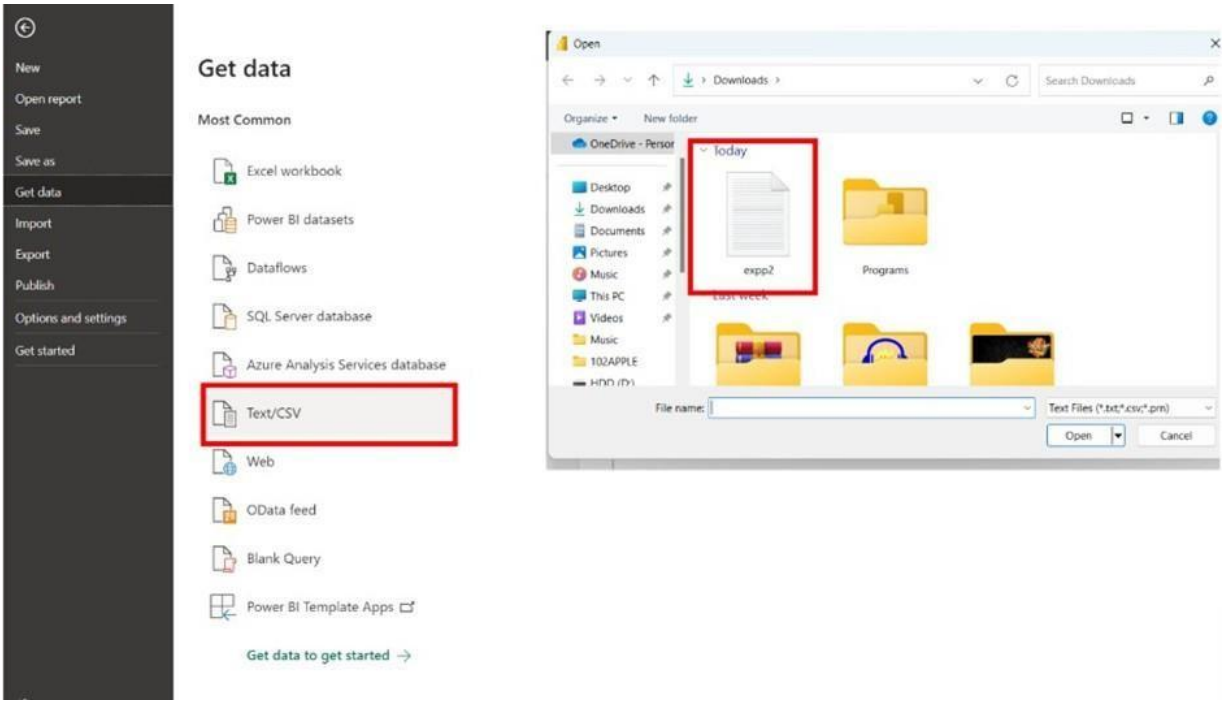
3. Create Time-Series Visualization: In Power BI, you can create a time-series visualization by selecting the "Line and Stacked Column Chart" visualization type and placing the date field on the x-axis and the stock market data on the y-axis.

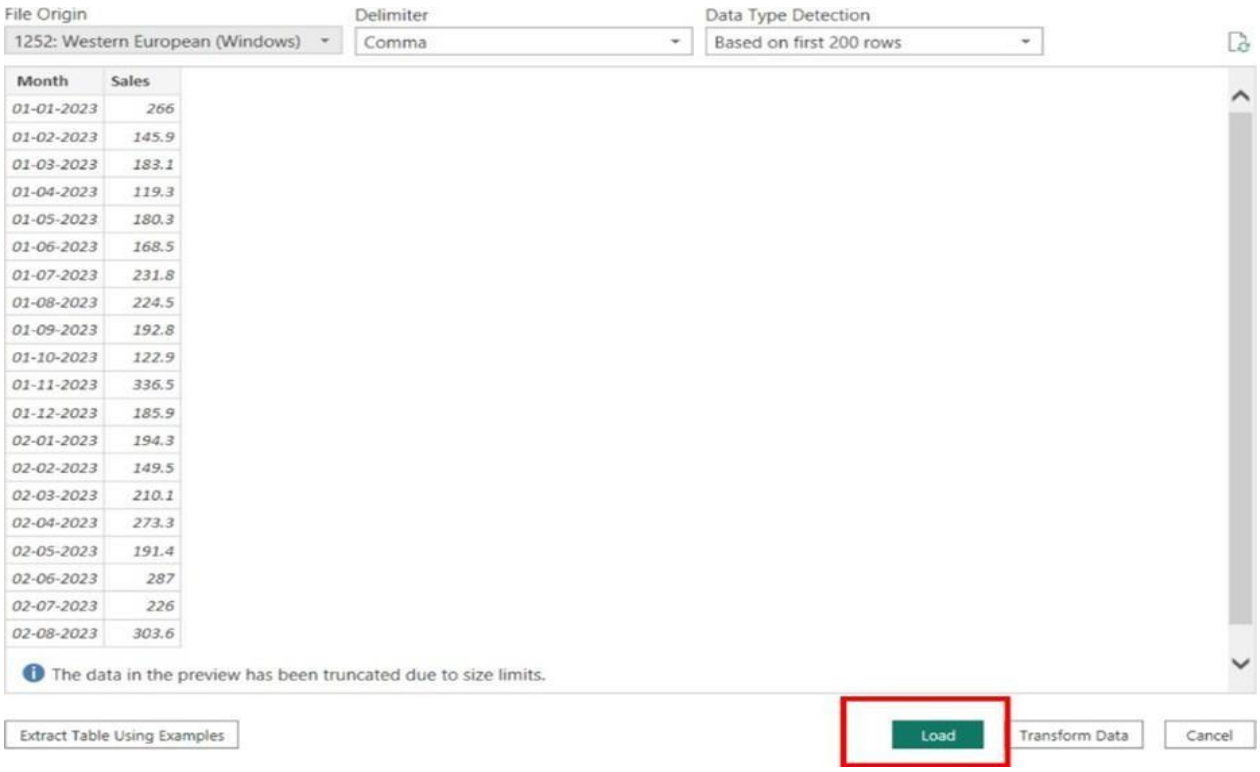
4. Analyze Trends and Patterns: Use the time-series visualization to analyse trends and patterns in the stock market data. You can use features such as trend lines, moving averages, and regression analysis to help identify patterns.

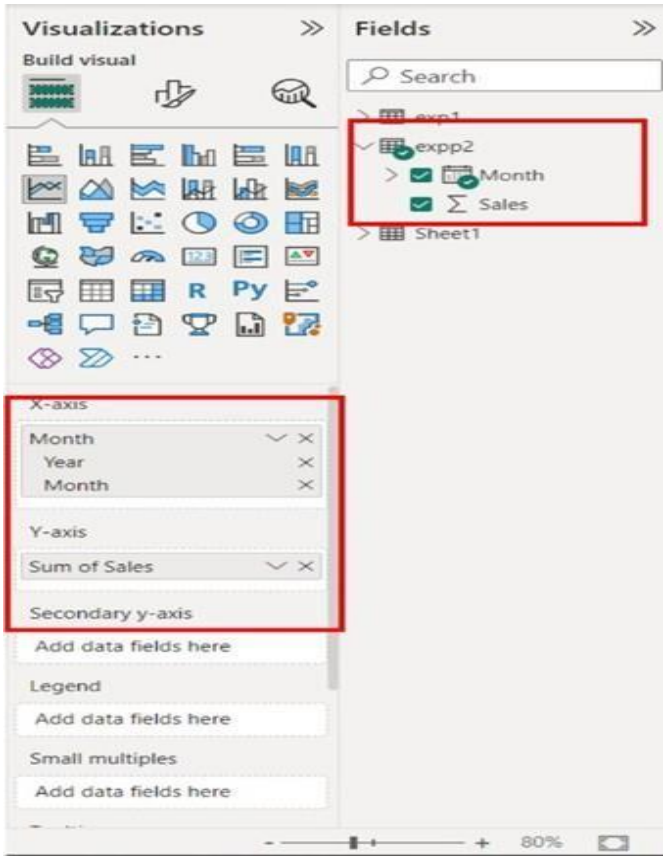
5. Forecast Future Values: Power BI has a built-in forecasting feature that can be used to forecast future stock market values

based on historical data. To use this feature, select the time-series visualization, click on the "Analytics" tab, and select "Forecast."

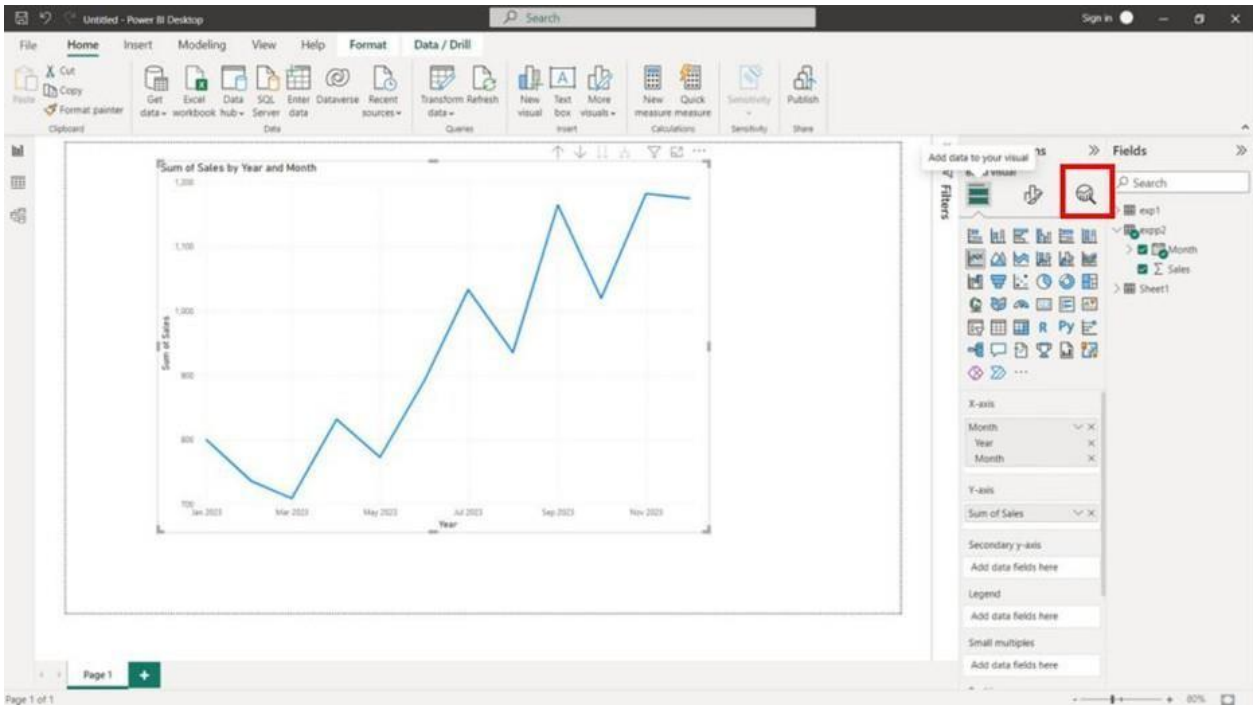
6. Share Insights: Once you have analysed the data and created visualizations, you can share your insights with others by publishing your report to the Power BI service or by sharing it directly through Power BI Desktop.

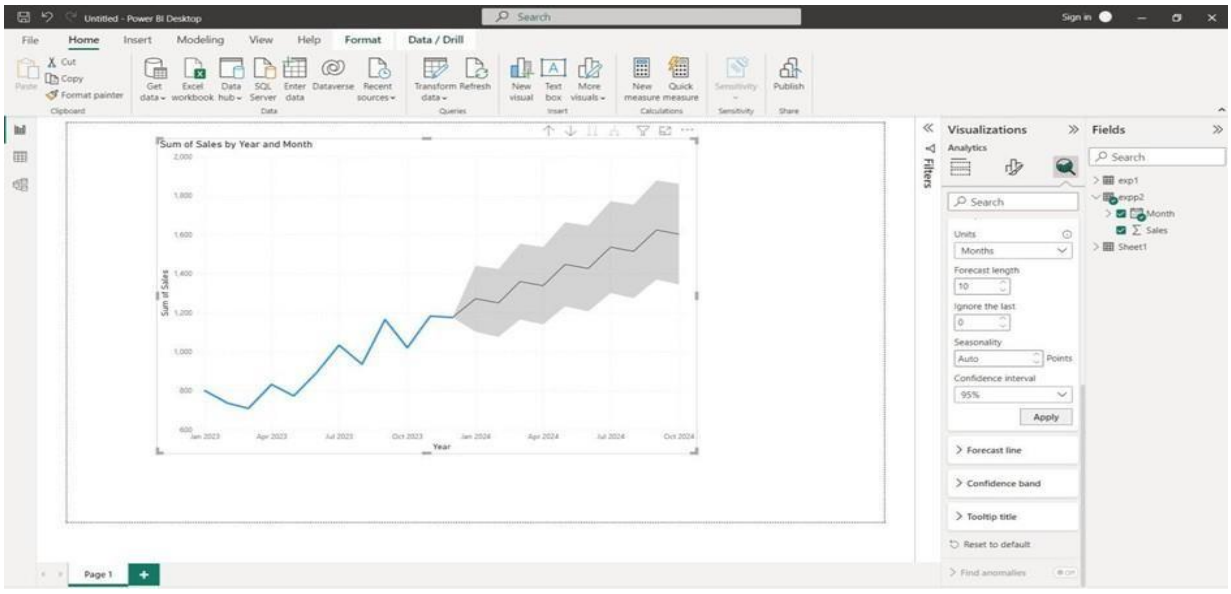



File Origin
1252: Western European (Windows)
Delimiter
Comma
Data Type Detection
Based on first 200 rows
Month
Sales
01-01-2023 266
01-02-2023 145.9
01-03-2023 183.1
01-04-2023 119.3
01-05-2023 180.3
01-06-2023 168.5
01-07-2023 231.8
01-08-2023 224.5
01-09-2023 192.8
01-10-2023 122.9
01-11-2023 336.5
01-12-2023 185.9
02-01-2023 194.3
02-02-2023 149.5
02-03-2023 210.1
02-04-2023 273.3
02-05-2023 191.4
02-06-2023 287
02-07-2023 226
02-08-2023 303.6
The data in the preview has been truncated due to size limits.
Extract Table Using Examples
Load
Transform Data
Cancel


Visualizations
Build visual
Fields
Search
expp2
Month
Sales
Sheet1
X-axis
Month
Year
Month
Y-axis
Sum of Sales
Secondary y-axis
Add data fields here
Legend
Add data fields here
Small multiples
Add data fields here
80%

OUTPUT:


Untitled - Power BI Desktop
Search
Sign in
File
Home
Insert
Modeling
View
Help
Format
Data / Drill
Sum of Sales by Year and Month
Sum of Sales
Year
Add data to your visual
Filters
Fields
Search
exp1
expp2
Month
Sales
Sheet1
X-axis
Month
Year
Month
Y-axis
Sum of Sales
Secondary y-axis
Add data fields here
Legend
Add data fields here
Small multiples
Add data fields here
Page 1
Page 1 of 1


Untitled - Power BI Desktop
Search
Sign in
File
Home
Insert
Modeling
View
Help
Format
Data / Drill
Sum of Sales by Year and Month
Sum of Sales
Year
Visualizations
Analytics
Fields
Search
exp1
expp2
Month
Sales
Sheet1
Filters
Search
Units
Months
Forecast length
10
Ignore the last
0
Seasonality
Auto
Points
Confidence interval
95%
Apply
Forecast line
Confidence band
Tooltip title
Reset to default
Find anomalies
Page 1

Dept. of AI&DS

Description	Max Marks	Awarded
Aim	**5**	
Software/Tools Required & Algorithm	**10**	
Coding/ Programming & Execution	**20**	
Record	**20**	
Viva Voce	**10**	
Result	**10**	
TOTAL	**75**	

RESULT:

Successfully completed time series analysis on stock market dataset identifying key trends and patterns using statistical modelling techniques.

EX NO: 3 DATE:	VISUALIZATION OF MASSIVE DATASET-FINANCE

AIM:

To gain insights from a massive finance dataset and inform business decisions through a visually appealing and interactive dashboard.

PROCEDURE:

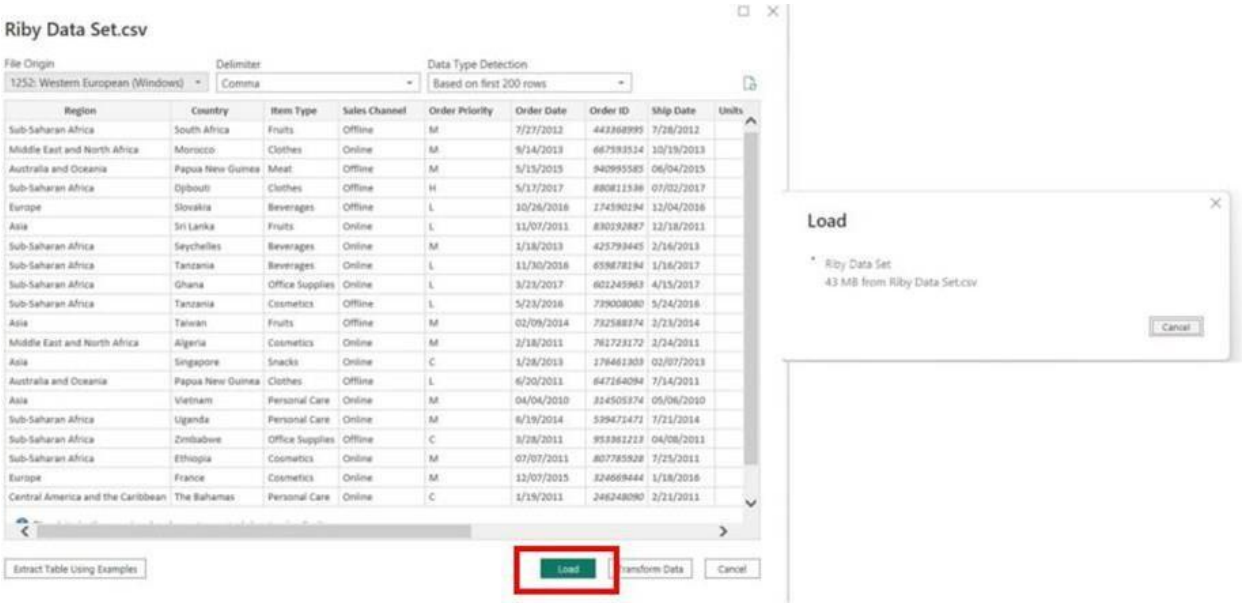
1. Data Gathering: Collect the finance dataset from a reliable source and clean it to remove any irrelevant or duplicate data.

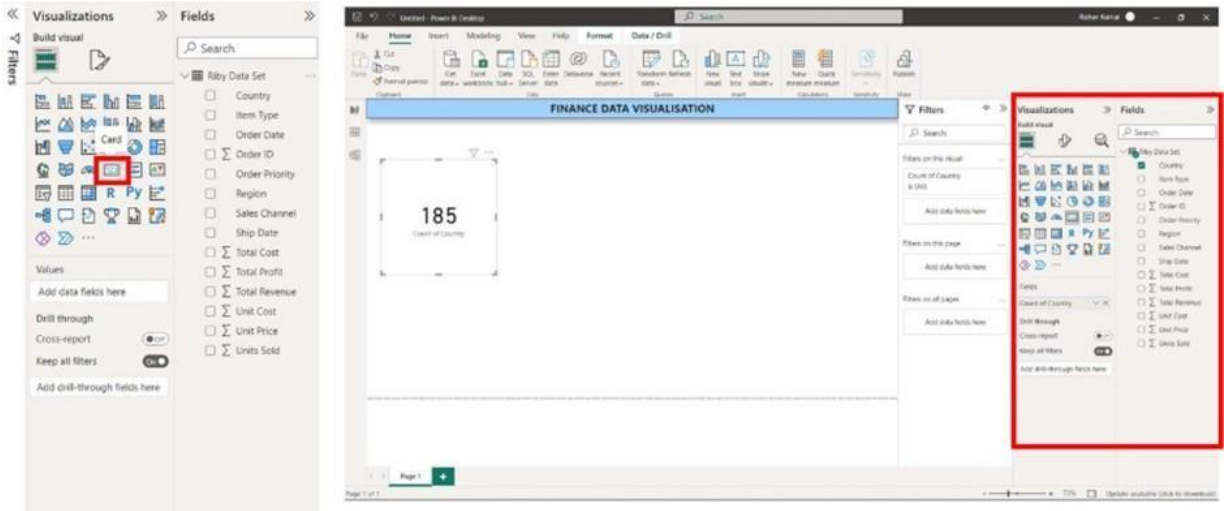
2. Data Modelling: Develop a data model that can efficiently handle the large volume of data.

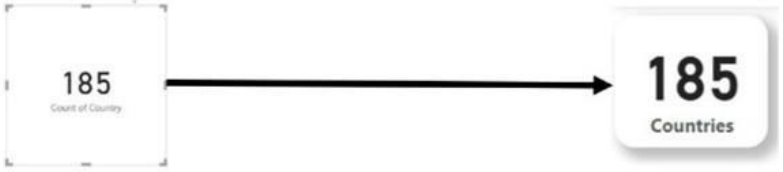
3. Visual Design: Create a visually appealing and interactive dashboard that presents the data in a clear and easy-to-understand manner.

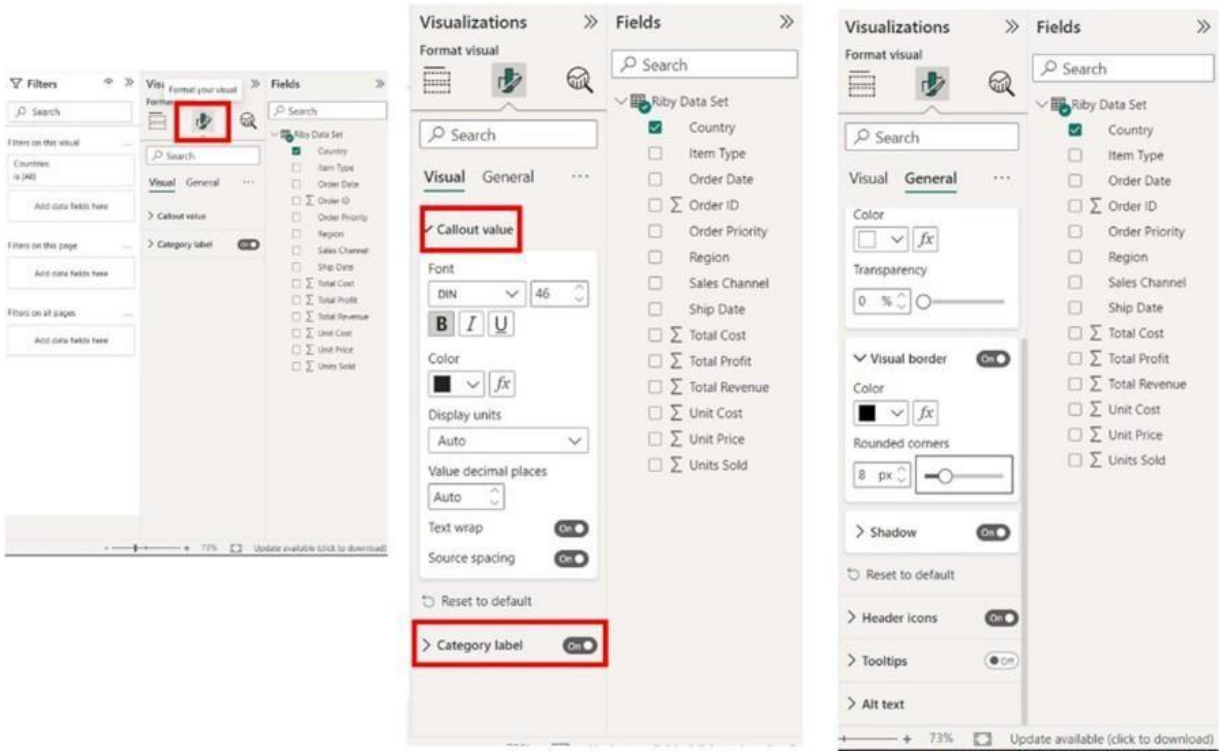
4. Analysis: Explore the dataset using various analytical techniques to uncover trends, patterns, and insights.

5. Reporting: Summarize the findings in a concise report and share it with the relevant stakeholders.

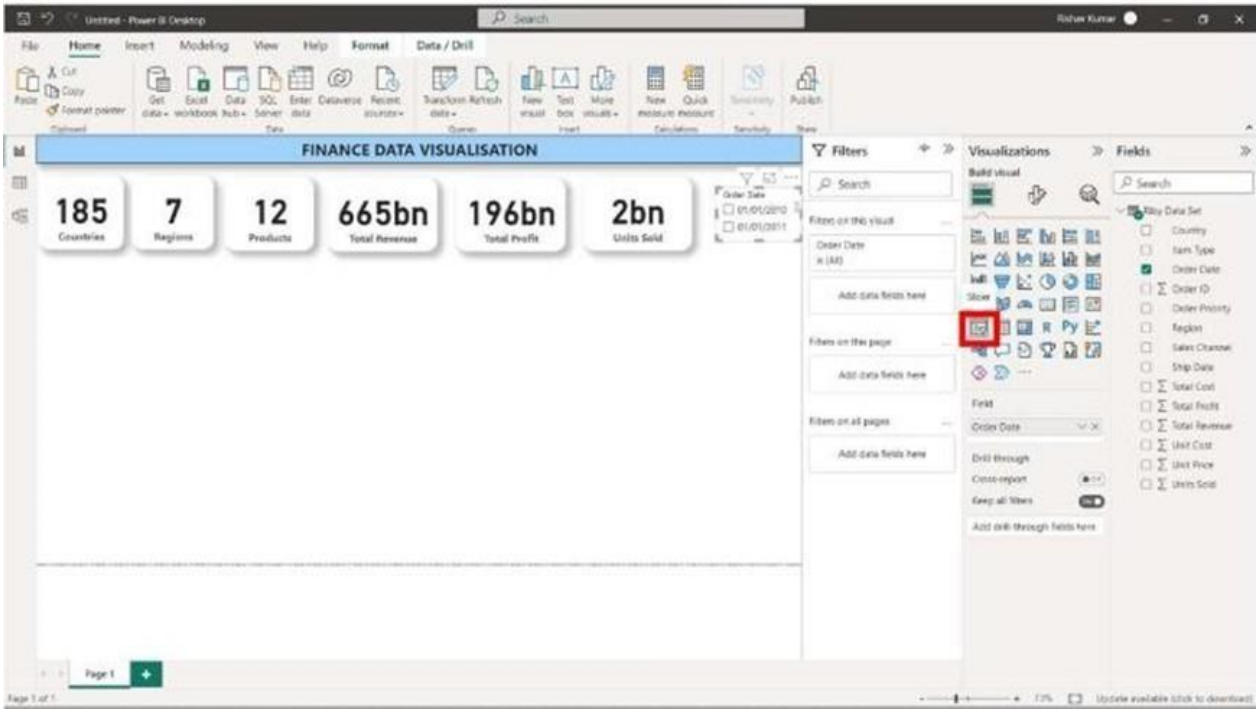

Riby Data Set.csv
File Origin
1252: Western European (Windows)
Delimiter
Comma
Data Type Detection
Based on first 200 rows
Region | Country | Item Type | Sales Channel | Order Priority | Order Date | Order ID | Ship Date | Units
Sub-Saharan Africa | South Africa | Fruits | Offline | M | 7/27/2012 | 443368995 | 7/28/2012
Middle East and North Africa | Morocco | Clothes | Online | M | 9/14/2013 | 667593514 | 10/19/2013
Australia and Oceania | Papua New Guinea | Meat | Offline | M | 5/15/2015 | 940995585 | 06/04/2015
Sub-Saharan Africa | Djibouti | Clothes | Offline | H | 5/17/2017 | 880811536 | 07/02/2017
Europe | Slovakia | Beverages | Offline | L | 10/26/2016 | 174590194 | 12/04/2016
Asia | Sri Lanka | Fruits | Online | L | 11/07/2011 | 830192887 | 12/18/2011
Sub-Saharan Africa | Seychelles | Beverages | Online | M | 1/18/2013 | 425793445 | 2/16/2013
Sub-Saharan Africa | Tanzania | Beverages | Online | L | 11/30/2016 | 659878194 | 1/16/2017
Sub-Saharan Africa | Ghana | Office Supplies | Online | L | 3/23/2017 | 602245963 | 4/15/2017
Sub-Saharan Africa | Tanzania | Cosmetics | Offline | L | 5/23/2016 | 739008080 | 5/24/2016
Asia | Taiwan | Fruits | Offline | M | 02/09/2014 | 732588374 | 2/23/2014
Middle East and North Africa | Algeria | Cosmetics | Online | M | 2/18/2011 | 761723172 | 2/24/2011
Asia | Singapore | Snacks | Online | C | 1/28/2013 | 178461303 | 02/07/2013
Australia and Oceania | Papua New Guinea | Clothes | Offline | L | 6/20/2011 | 647164094 | 7/14/2011
Asia | Vietnam | Personal Care | Online | M | 04/04/2010 | 314505374 | 05/06/2010
Sub-Saharan Africa | Uganda | Personal Care | Online | M | 8/19/2014 | 539471471 | 7/21/2014
Sub-Saharan Africa | Zimbabwe | Office Supplies | Offline | C | 3/28/2011 | 953361213 | 04/08/2011
Sub-Saharan Africa | Ethiopia | Cosmetics | Online | M | 07/07/2011 | 807785928 | 7/25/2011
Europe | France | Cosmetics | Online | M | 12/07/2015 | 324669444 | 1/18/2016
Central America and the Caribbean | The Bahamas | Personal Care | Online | C | 1/19/2011 | 246248090 | 2/21/2011
Extract Table Using Examples
Load
Transform Data
Cancel
Load
Riby Data Set
43 MB from Riby Data Set.csv
Cancel

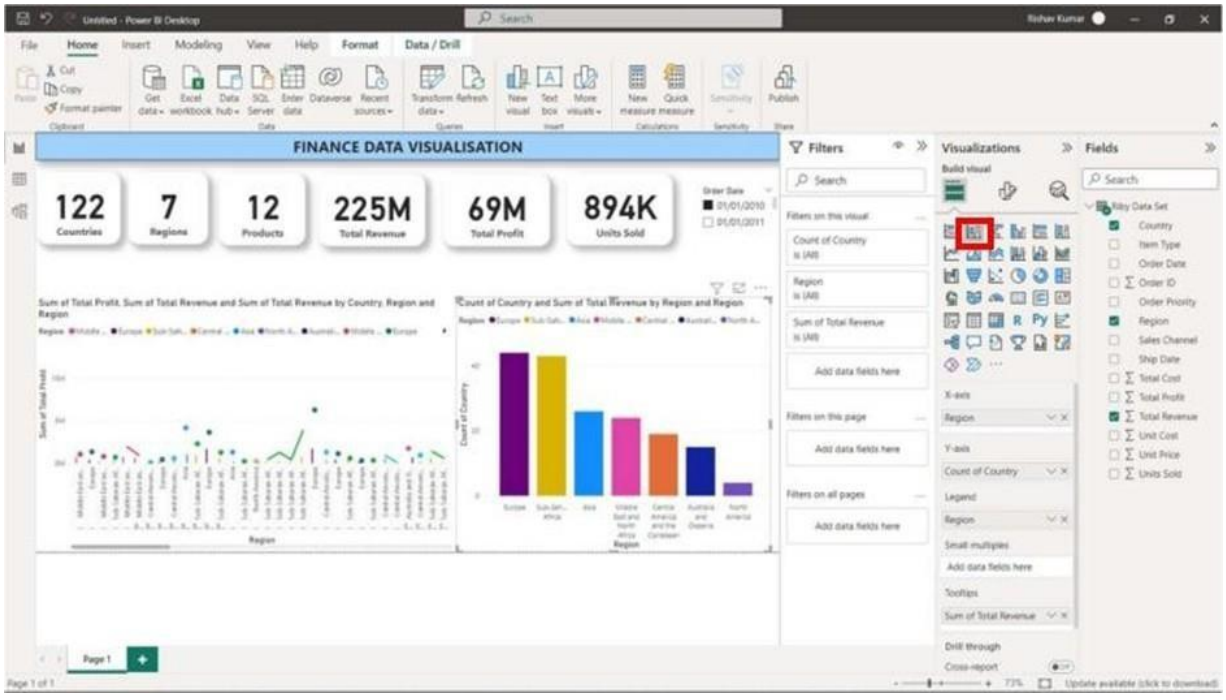

Visualizations
Build visual
Fields
Search
Riby Data Set
Country
Item Type
Order Date
Order ID
Order Priority
Region
Sales Channel
Ship Date
Total Cost
Total Profit
Total Revenue
Unit Cost
Unit Price
Units Sold
Values
Add data fields here
Drill through
Cross-report
Keep all filters
Add drill-through fields here
FINANCE DATA VISUALISATION
185
Filters


185
Count of Country
185
Countries


Visualizations
Format visual
Fields
Search
Visual
General
Callout value
Font
DIN
46
Color
Display units
Auto
Value decimal places
Text wrap
Source spacing
Reset to default
Category label
Ruby Data Set
Country
Item Type
Order Date
Order ID
Order Priority
Region
Sales Channel
Ship Date
Total Cost
Total Profit
Total Revenue
Unit Cost
Unit Price
Units Sold
Transparency
Visual border
Rounded corners
Shadow
Header icons
Tooltips
Alt text
73%
Update available (click to download)

OUTPUT:


FINANCE DATA VISUALISATION
185
Countries
7
Regions
12
Products
665bn
Total Revenue
196bn
Total Profit
2bn
Units Sold
Filters
Visualizations
Fields
Page 1


FINANCE DATA VISUALISATION
122
Countries
7
Regions
12
Products
225M
Total Revenue
69M
Total Profit
894K
Units Sold
Filters
Visualizations
Fields
Page 1

Dept. of AI&DS

Description	Max Marks	Awarded
Aim	5	
Software/Tools Required & Algorithm	10	
Coding/ Programming & Execution	20	
Record	20	
Viva Voce	10	
Result	10	
TOTAL	75	

RESULT:

The finance dataset was analysed with precision, and the resulting insights were used to create a visually appealing and interactive dashboard, which was successfully deployed to inform business decisions.

EX NO: 4 DATE:	VISUALIZATION ON STREAMING DATASET

AIM:

The aim of visualizing a streaming dataset in Power BI is to create real-time data visualizations that update dynamically as new data is received from a streaming source. This can help users quickly identify trends, patterns, and anomalies in their data.

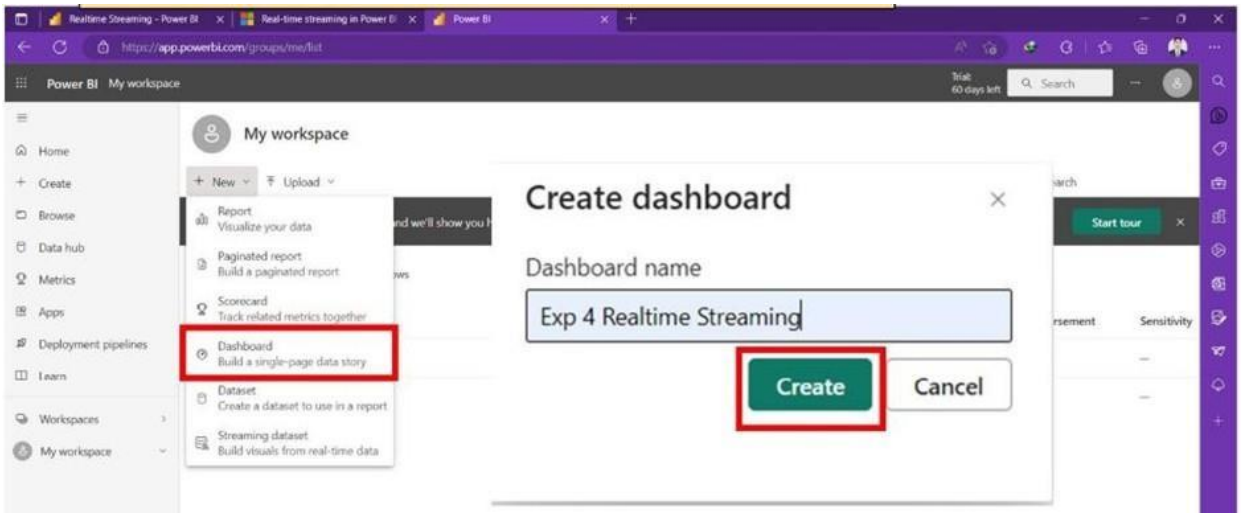
PROCEDURE:

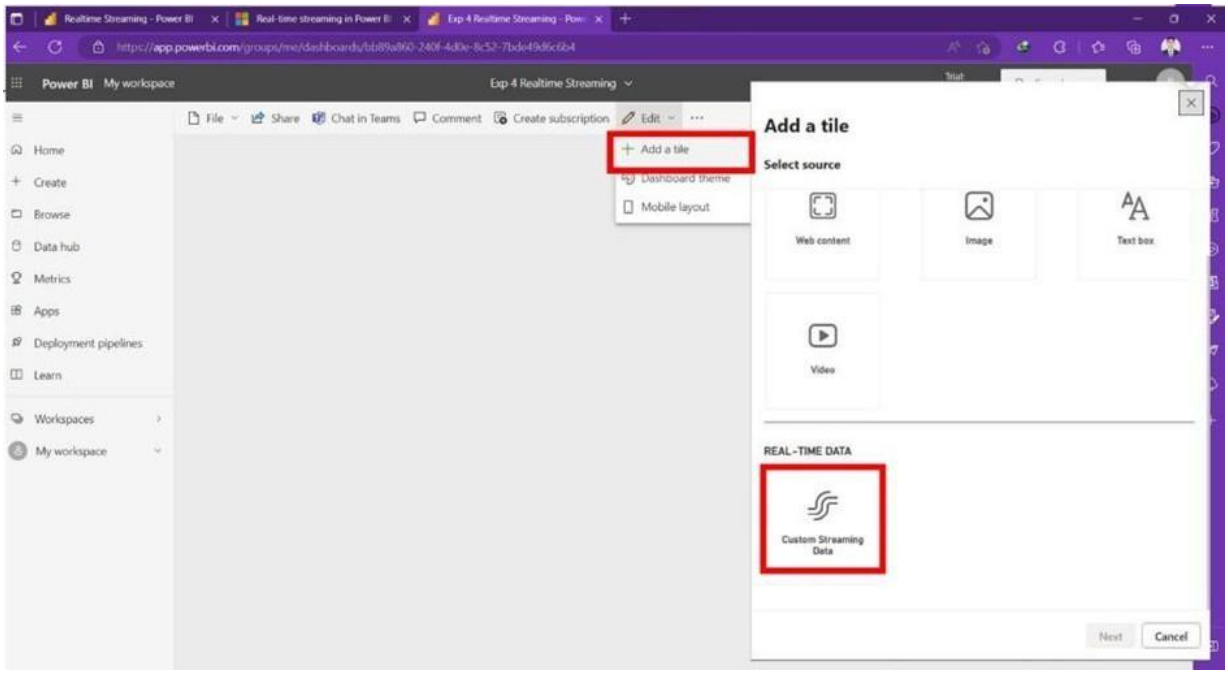
1. Connect to your streaming data source: In Power BI, go to the "Home" tab and click on "Get Data". Select "Streaming" under the "Other" category and choose your streaming data source.

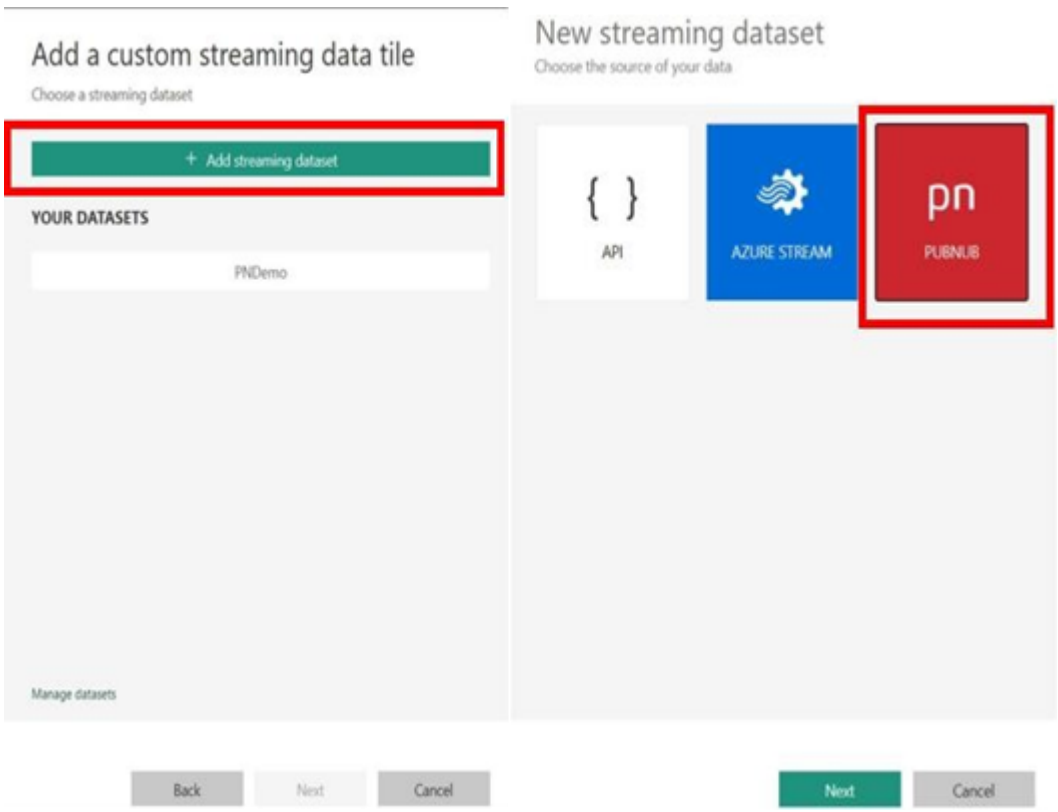
2. Configure the streaming dataset: In the "Configure Streaming Dataset" window, enter the required information about your data source, such as the streaming URL, schema, and data types.

3. Create visuals: Once your streaming dataset is configured, you can create visualizations by dragging and dropping fields onto the canvas. You can choose from various visualization types, including charts, maps, and tables.

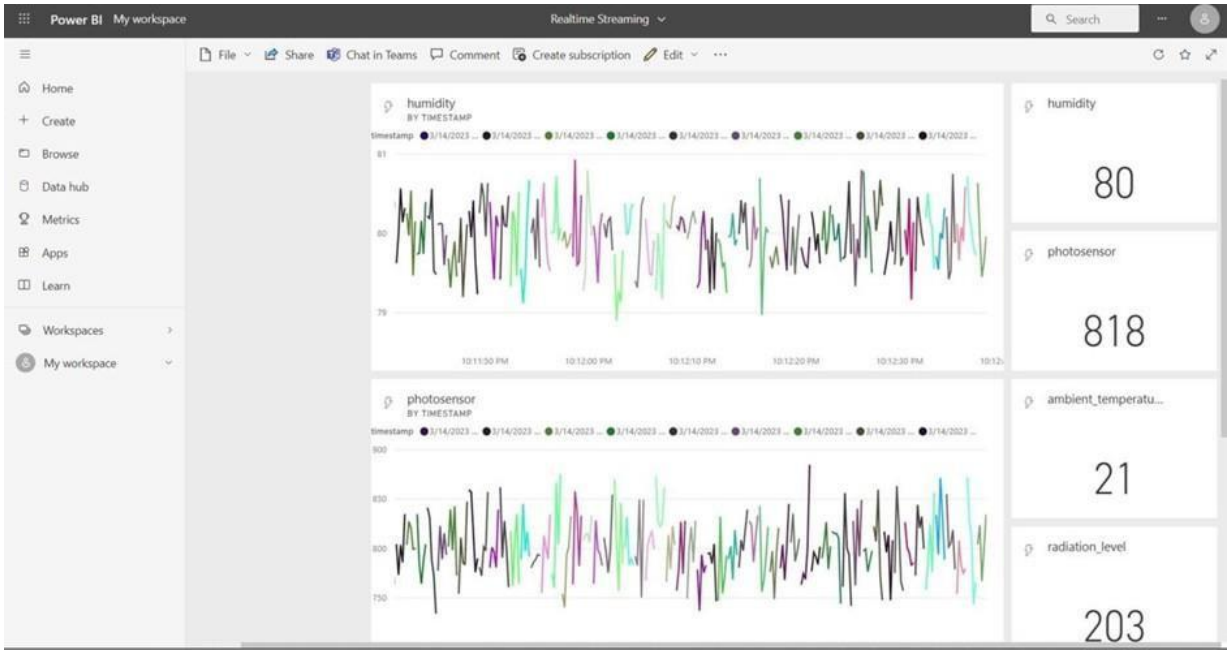
4. Publish your report: Once you have created your visuals, you can publish your report to the Power BI service. This will allow you to view and share your real-time data visualizations with others.


Power BI My workspace
My workspace
Home
Create
Browse
Data hub
Metrics
Apps
Deployment pipelines
Learn
Workspaces
My workspace
New
Upload
Report
Visualize your data
Paginated report
Build a paginated report
Scorecard
Track related metrics together
Dashboard
Build a single-page data story
Dataset
Create a dataset to use in a report
Streaming dataset
Build visuals from real-time data
Create dashboard
Dashboard name
Exp 4 Realtime Streaming
Create
Cancel
Start tour
Sensitivity


Power BI My workspace
Exp 4 Realtime Streaming
File
Share
Chat in Teams
Comment
Create subscription
Edit
Add a tile
Dashboard theme
Mobile layout
Add a tile
Select source
Web content
Image
Text box
Video
REAL-TIME DATA
Custom Streaming Data
Next
Cancel


Add a custom streaming data tile
Choose a streaming dataset
+ Add streaming dataset
YOUR DATASETS
PNDemo
Manage datasets
New streaming dataset
Choose the source of your data
API
AZURE STREAM
pn
PUBNUB
Back
Next
Cancel
Next
Cancel

OUTPUT:


Power BI
My workspace
Realtime Streaming
Search
File
Share
Chat in Teams
Comment
Create subscription
Edit
Home
Create
Browse
Data hub
Metrics
Apps
Learn
Workspaces
My workspace
humidity
BY TIMESTAMP
photosensor
BY TIMESTAMP
humidity
80
photosensor
818
ambient_temperatu...
21
radiation_level
203

Dept. of AI&DS

Description	Max Marks	Awarded
Aim	5	
Software/Tools Required & Algorithm	10	
Coding/ Programming & Execution	20	
Record	20	
Viva Voce	10	
Result	10	
TOTAL	75	

RESULT:

Successfully visualize streaming data in Power BI to create real-time data visualizations that provide valuable insights into patterns, trends, and anomalies, and make data-driven decisions on the fly.

EX NO: 5 DATE:	TEXT VISUALIZATION USING WEB ANALYTICS

AIM:

The aim of this project is to use web analytics data to create a visualization in PowerBI that provides insights into website performance.

PROCEDURE:

1. Data collection: The first step is to collect web analytics data from your website. This can be done using tools such as Google Analytics or Adobe Analytics.

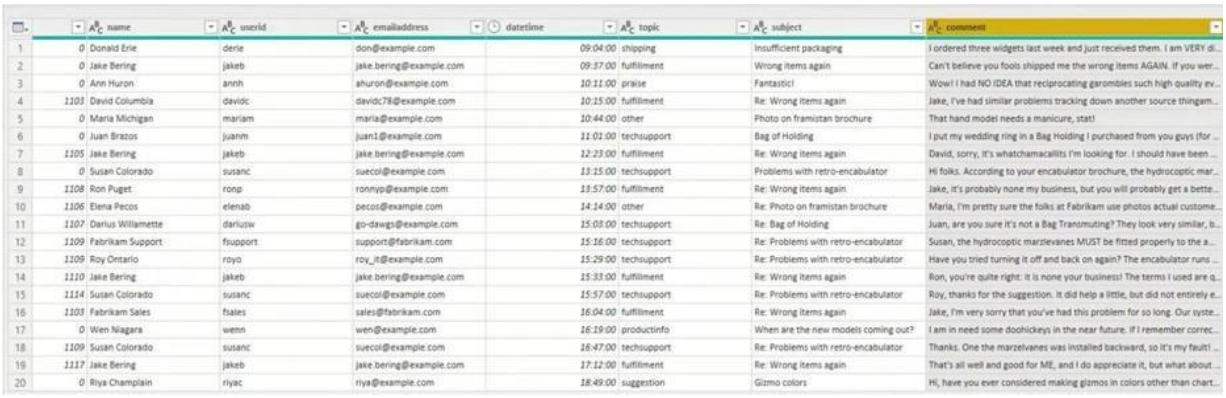
2. Data preparation: Once you have collected the data, you will need to clean and prepare it for analysis. This may involve removing duplicates, fixing errors, and formatting data in a way that is compatible with Power BI.

3. Data modeling: The next step is to create a data model in Power BI that will allow you to analyze and visualize the data effectively. This involves creating relationships between different tables and defining calculations and measures that will be used in the visualization.

4. Visualization design: With the data model in place, you can now start designing your visualization. This may involve creating charts, graphs, and tables that display the data in a meaningful way. You

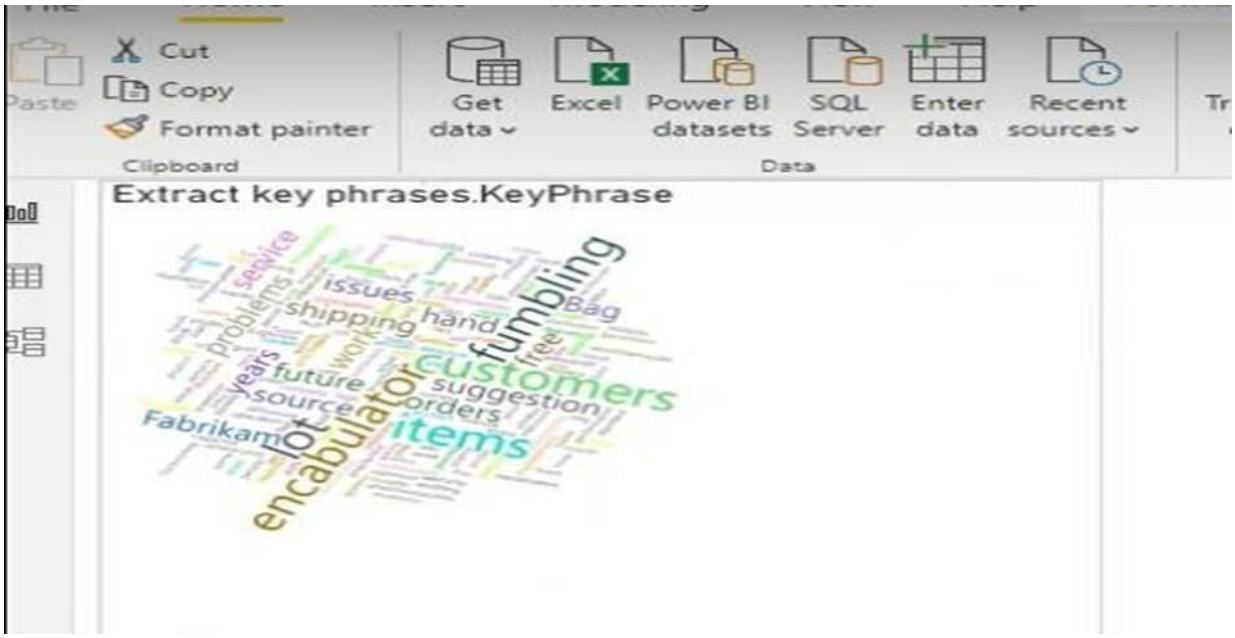
may also want to add interactive elements such as filters and slicers that allow users to explore the data in more detail.

5. Report creation: Once you have designed your visualization, you can create a report in Power BI that brings everything together. This report can be shared with others in your organization, allowing them to gain insights into website performance.



		name	userid	emailaddress	datetime	topic	subject	comment
1	0	Donald Erie	derie	don@example.com	09:04:00	shipping	Insufficient packaging	I ordered three widgets last week and just received them. I am VERY di...
2	0	Jake Bering	jakeb	jake.bering@example.com	09:37:00	fulfillment	Wrong items again	Can't believe you fools shipped me the wrong items AGAIN. If you wer...
3	0	Ann Huron	annh	ahuron@example.com	10:11:00	praise	Fantastic!	Wow! I had NO IDEA that reciprocating garombles such high quality ev...
4	1103	David Columbia	davidc	davidc78@example.com	10:15:00	fulfillment	Re: Wrong items again	Jake, I've had similar problems tracking down another source thingam...
5	0	Maria Michigan	mariam	maria@example.com	10:44:00	other	Photo on framistan brochure	That hand model needs a manicure, stat!
6	0	Juan Brazos	juanm	juan1@example.com	11:01:00	techsupport	Bag of Holding	I put my wedding ring in a Bag Holding I purchased from you guys (for ...
7	1105	Jake Bering	jakeb	jake.bering@example.com	12:23:00	fulfillment	Re: Wrong items again	David, sorry, it's whatchamacallits I'm looking for. I should have been ...
8	0	Susan Colorado	susanc	suecol@example.com	13:15:00	techsupport	Problems with retro-encabulator	Hi folks. According to your encabulator brochure, the hydrocoptic mar...
9	1108	Ron Puget	ronp	ronnyp@example.com	13:57:00	fulfillment	Re: Wrong items again	Jake, it's probably none my business, but you will probably get a bette...
10	1106	Elena Pecos	elenab	pecos@example.com	14:14:00	other	Re: Photo on framistan brochure	Maria, I'm pretty sure the folks at Fabrikam use photos actual custome...
11	1107	Darius Willamette	dariusw	go-dawgs@example.com	15:03:00	techsupport	Re: Bag of Holding	Juan, are you sure it's not a Bag Transmuting? They look very similar, b...
12	1109	Fabrikam Support	fsupport	support@fabrikam.com	15:16:00	techsupport	Re: Problems with retro-encabulator	Susan, the hydrocoptic marzlevanes MUST be fitted properly to the a...
13	1109	Roy Ontario	royo	roy_it@example.com	15:29:00	techsupport	Re: Problems with retro-encabulator	Have you tried turning it off and back on again? The encabulator runs ...
14	1110	Jake Bering	jakeb	jake.bering@example.com	15:33:00	fulfillment	Re: Wrong items again	Ron, you're quite right: it is none your business! The terms I used are q...
15	1114	Susan Colorado	susanc	suecol@example.com	15:57:00	techsupport	Re: Problems with retro-encabulator	Roy, thanks for the suggestion. It did help a little, but did not entirely e...
16	1103	Fabrikam Sales	fsales	sales@fabrikam.com	16:04:00	fulfillment	Re: Wrong items again	Jake, I'm very sorry that you've had this problem for so long. Our syste...
17	0	Wen Niagara	wenn	wen@example.com	16:19:00	productinfo	When are the new models coming out?	I am in need some doohickeys in the near future. If I remember correc...
18	1109	Susan Colorado	susanc	suecol@example.com	16:47:00	techsupport	Re: Problems with retro-encabulator	Thanks. One the marzelvanes was installed backward, so it's my fault! ...
19	1117	Jake Bering	jakeb	jake.bering@example.com	17:12:00	fulfillment	Re: Wrong items again	That's all well and good for ME, and I do appreciate it, but what about ...
20	0	Riya Champlain	riyac	riya@example.com	18:49:00	suggestion	Gizmo colors	Hi, have you ever considered making gizmos in colors other than chart...

OUTPUT:


Cut
Copy
Format painter
Clipboard
Get data
Excel
Power BI datasets
SQL Server
Enter data
Recent sources
Data
Extract key phrases.KeyPhrase
service
issues
shipping
hand
Bag
fumbling
customers
suggestion
orders
items
encabulator
future
years
source
Fabrikam

Dept. of AI&DS

Description	Max Marks	Awarded
Aim	5	
Software/Tools Required & Algorithm	10	
Coding/ Programming & Execution	20	
Record	20	
Viva Voce	10	
Result	10	
TOTAL	75	

RESULT:

Successfully use web analytics data to create a Power BI visualization that provides insights into website performance, allowing for data-driven decisions and improved user experience.

NATURAL LANGUAGE PROCESSING

EX NO: 6 DATE:	WORD ANALYSIS & MORPHOLOGY

AIM:

The aim of this program is to analyze and examine the morphology of words using the Python Natural Language Toolkit (NLTK) library.

PROCEDURE:

1. Import necessary libraries, such as the NLTK library and the word_tokenize function.
2. Define the input text to be analyzed.
3. Tokenize the input text into individual words using the word_tokenize function from the NLTK library.
4. Apply morphology analysis using the PorterStemmer algorithm from the NLTK library.
5. Store the stemmed words in a new list.
6. Print out the original text, the tokenized words, and the stemmed words.

PROGRAM:

```
import nltk
from nltk.tokenize import word_tokenize
# input text
text = "The quick brown fox jumps over the lazy dog."
 # tokenizing the text into words
words = word_tokenize(text)
# applying morphology analysis
morph = nltk.PorterStemmer()
stemmed_words = [morph.stem(word) for word in words]
# output
print("Original Text: ", text)
print("Tokenized Words: ", words)
print("Stemmed Words: ", stemmed_words)
```

OUTPUT:

Original Text: The quick brown fox jumps over the lazy dog.

Tokenized Words: ['The', 'quick', 'brown', 'fox', 'jumps', 'over', 'the', 'lazy', 'dog', '.']

Stemmed Words: ['the', 'quick', 'brown', 'fox', 'jump', 'over', 'the', 'lazi', 'dog', '.']

Dept. of AI&DS

Description	Max Marks	Awarded
Aim	**5**	
Software/Tools Required & Algorithm	**10**	
Coding/ Programming & Execution	**20**	
Record	**20**	
Viva Voce	**10**	
Result	**10**	
TOTAL	**75**	

RESULT:

Thus the Program for Word Analysis & Morphology written and executed successfully.

EX NO: 7 DATE:	N-GRAMS

AIM:

The aim of this code is to generate the bi-gram and tri-gram sequences of words from a given input sentence.

PROCEDURE:

1. Import the required module - "ngrams" from the NLTK package.

2. Define a string variable "s" that contains the input sentence.

3. Split the sentence into individual words using the "split()" function and store it in a variable called "w".

4. Generate the bi-gram sequences of words using the "ngrams()" function from the NLTK package, by passing the word list "w" and the value "2" to specify the length of the sequence as a parameter. Store the result in a variable called "bi_gram".

5. Generate the tri-gram sequences of words using the same "ngrams()" function, but by passing the value "3" as a parameter. Store the result in a variable called "tri_gram".

6. Print the bi-gram and tri-gram sequences using the "print()" function, along with appropriate messages.

PROGRAM:

```
from nltk.util import ngrams
s = "This My laptop it's very comfortable for me"
w = s.split()
bi_gram = list(ngrams(w, 2))
tri_gram = list(ngrams(w, 3))
print("Bi-gram:", bi_gram)
print("Tri-gram:", tri_gram)
```

OUTPUT:

Bi-gram: [('This', 'My'), ('My', 'laptop'), ('laptop', "it's"), ("it's", 'very'),('very', 'comfortable'), ('comfortable', 'for'), ('for', 'me')]

Tri-gram: [('This', 'My', 'laptop'), ('My', 'laptop', "it's"), ('laptop', "it's", 'very'), ("it's", 'very', 'comfortable'), ('very', 'comfortable', 'for'), ('comfortable', 'for', 'me')]

Dept. of AI&DS

Description	Max Marks	Awarded
Aim	5	
Software/Tools Required & Algorithm	10	
Coding/ Programming & Execution	20	
Record	20	
Viva Voce	10	
Result	10	
TOTAL	75	

RESULT:

Thus the Program for N-grams written and executed successfully.

EX NO: 8 DATE:	POS TAGGING

AIM:

To perform POS tagging on a given sentence using NLTK in Python.

PROCEDURE:

1. First, we import the nltk library and specifically the word_tokenize function which tokenizes the input sentence into individual words.

2. We define the input sentence as a string variable.

3. We tokenize the sentence into individual words using word_tokenize.

4. We then use the pos_tag function from nltk to perform POS tagging on the tokenized words.

5. Finally, we print out the POS tagged words.

PROGRAM:

```
import nltk
from nltk.tokenize import word_tokenize
sentence = "The quick brown fox jumps over the lazy dog."
tokens = word_tokenize(sentence)
pos_tags = nltk.pos_tag(tokens)
print(pos_tags)
```

OUTPUT:

```
[('The', 'DT'), ('quick', 'JJ'), ('brown', 'NN'), ('fox', 'NN'), ('jumps', 'VBZ'),
('over', 'IN'), ('the', 'DT'), ('lazy', 'JJ'), ('dog', 'NN'), ('.', '.')]
```

Dept. of AI&DS

Description	Max Marks	Awarded
Aim	5	
Software/Tools Required & Algorithm	10	
Coding/ Programming & Execution	20	
Record	20	
Viva Voce	10	
Result	10	
TOTAL	75	

RESULT:

Thus the program for POS tagging written and executed successfully.

EX NO: 9 DATE:	BUILDING CHUNKER

AIM:

The aim of the above code is to parse a given input sentence using regular expressions and chunk it into phrases.

PROCEDURE:

1. Import the necessary module nltk which provides natural language processing functionalities.
2. Define a regular expression-based grammar pattern that specifies the structure of the phrases to be extracted.
3. Initialize the nltk.RegexpParser() object with the defined grammar pattern.
4. Provide an input sentence that needs to be parsed and chunked into phrases.
5. Tokenize the input sentence into individual words using nltk.word_tokenize().
6. Tag each word with its corresponding Part-of-Speech (POS) tag using nltk.pos_tag().
7. Parse the tagged sentence using the initialized parser object, which creates a parse tree with phrases as nodes and words as leaves.
8. Traverse the parsed tree and extract the Noun Phrases using the subtrees() method.

9. Check the label of each subtree and extract the ones labeled as "NP".

Print the extracted Noun Phrases.

PROGRAM:

```
import nltk

# Define a grammar pattern using regular expressions

grammar_pattern = r"""

        NP: {<DT>?<JJ>*<NN>} # chunking Noun Phrases

        VP: {<VB.*><NP|PP|CLAUSE>+$} # chunking Verb Phrases

        PP: {<IN><NP>} # chunking Prepositional Phrases

        CLAUSE: {<NP><VP>} # chunking Clauses

        """

# Initialize the parser with the grammar pattern parser =
nltk.RegexpParser(grammar_pattern).

# Input sentence

s = "The quick brown fox jumped over the lazy dog." # Tokenize the input
sentence

tokens = nltk.word_tokenize(s)

# Tag the tokens with Part-of-Speech tags

pos_tags = nltk.pos_tag(tokens)

# Parse the tagged sentence using the parser
```

```
tree = parser.parse(pos_tags)
# Traverse the parsed tree and print the leaves
for subtree in tree.subtrees():
    if subtree.label() == 'NP':
        print(' '.join(word for word, tag in subtree.leaves()))
```

OUTPUT:

```
The
quick brown fox
the lazy dog
```

Dept. of AI&DS

Description	Max Marks	Awarded
Aim	5	
Software/Tools Required & Algorithm	10	
Coding/ Programming & Execution	20	
Record	20	
Viva Voce	10	
Result	10	
TOTAL	75	

RESULT:

Thus the program for building chunker was written and executed successfully.

EX NO: 10 DATE:	BUILD CHATBOT

AIM:

The given code implements a simple chatbot using the NLTK library in Python. The chatbot responds to a few pre-defined patterns of user input.

PROCEDURE:

1. First, the code imports the necessary modules from the NLTK library - ' ‘nltk' and 'Chat' from 'nltk.chat.util'.

2. The code defines a set of input-output patterns using regular expressions and corresponding responses. For example, if the user inputs 'Hi there!', the chatbot will respond with 'Hello!'.

3. These input-output patterns are stored in a list of tuples, where each tuple contains a regular expression pattern and a response for that pattern.

4. The code initializes a Chat object using the input-output patterns and a dictionary of reflections (used to map user pronouns to chatbot pronouns).

5. Finally, the code starts a conversation with the chatbot using the 'converse()' method of the Chat object.

PROGRAM:

```
import nltk
from nltk.chat.util import Chat, reflections
input_output_patterns = [
        (r'Hi there!', ['Hello!']),
        (r'How are you?', ['I am doing well, thank you. How are you?']),
        (r'What is your name?', ['My name is Chatbot.']),
        (r'Quit|Bye|Thank you', ['Goodbye!', 'Bye! Nice talking to you.'])
]
chatbot = Chat(input_output_patterns, reflections)
chatbot.converse()
```

OUTPUT:

```
>Hi there!
Hello!
>What is your name?
My name is Chatbot.
>How are you?
I am doing well, thank you. How are you?
>Quit
Goodbye!
```

Dept. of AI&DS

Description	Max Marks	Awarded
Aim	5	
Software/Tools Required & Algorithm	10	
Coding/ Programming & Execution	20	
Record	20	
Viva Voce	10	
Result	10	
TOTAL	75	

RESULT:

Thus the program for building a chatbot was written and executed successfully.

EX NO: 11 DATE:	CRYPTOCURRENCY DATA VISUALIZATION USING POWER BI

AIM:

To fetch real-time cryptocurrency market data and visualize trends using Power BI to analyze market fluctuations, price trends, and correlations.

PROCEDURE:

Step 1: Install Power BI and Required Tools

- Ensure Power BI Desktop is installed.
- Install Power Query (comes with Power BI).
- Use APIs (e.g., CoinGecko, CoinMarketCap) to fetch real-time cryptocurrency data.

Step 2: Fetch Cryptocurrency Data using Power BI

- Open Power BI Desktop → Click on Get Data → Choose Web.
- Enter the API URL:

  https://api.coingecko.com/api/v3/coins/markets?vs_currency=usd&order=market_cap_desc&per_page=10&page=1&sparkline=false

- Click ok then transform data to clean and format the dataset.

Step 3: Data Preprocessing

- Convert price values to a numerical format.
- Rename columns for better readability.
- Remove unnecessary columns.

Step 4: Data Visualization in Power BI

Bar Chart – Price Change in the Last 24 Hours

1. Select Bar Chart from the Visualizations pane.
2. Drag `id` to the X-axis and `price_change_percentage_24h` to the Y-axis.
3. Apply conditional formatting (green for positive, red for negative).

Line Graph – Bitcoin Price Trend Over Time

1. Select Line Chart from the Visualizations pane.
2. Drag `Date` to the X-axis and `current_price` to the Y-axis.
3. Filter for Bitcoin only.

Heatmap – Correlation Between Cryptocurrencies

1. Load additional cryptocurrency data with historical price changes.
2. Use a Matrix visualization to plot the correlation between different cryptocurrencies.
3. Apply color gradients to indicate correlations.

OUTPUT:

Dept. of AI&DS

Description	Max Marks	Awarded
Aim	5	
Software/Tools Required & Algorithm	10	
Coding/ Programming & Execution	20	
Record	20	
Viva Voce	10	
Result	10	
TOTAL	75	

RESULT:

Thus fetching real time crypto currency data was visualized and provided insights into market trends and showed relationships.

EX NO: 12 DATE:	DEPENDENCY PARSING USING NLP

AIM:

To analyze the grammatical structure of sentences by identifying relationships between words using Dependency parsing in NLP

PROCEDURE:

Step 1: Install require libraries

We use the spacy library which provides efficient dependency parsing models. Install it using

pip install spacy

python -m spacy download en_core_web_sm

Step 2: Load NLP Model

Load the spacy english model to process text

Step 3: Perform dependency parsing

- Tokenize the text
- Identify dependencies between words
- Visualize relationships using displacy

Step:4 Display results

- Print dependencies and head words
- Generate a dependency tree visualization

PROGRAM:

```
import spacy
from spacy import displacy

#Load English NLP model
nlp = spacy.load("en_core_web_sm")
sentence = "The autonomous robot efficiently navigates through obstacles."

# Process the sentence using spacy
doc = nlp(sentence)
print("Token\t\tDependency\t\tHead")
print("="*40)
for token in doc:
    print(f"{token.text}\t\t{token.dep_}\t\t{token.head.text}")

# Visualize dependency parsing
displacy.serve(doc, style="dep")
```

OUTPUT:

```
Token           Dependency      Head
========================================
The             det             robot
autonomous      amod            robot
robot           nsubj           navigates
efficiently     advmod          navigates
navigates       ROOT            navigates
through         prep            navigates
obstacles       pobj            through
.               punct           navigates
```

Dept. of AI&DS

Description	Max Marks	Awarded
Aim	5	
Software/Tools Required & Algorithm	10	
Coding/ Programming & Execution	20	
Record	20	
Viva Voce	10	
Result	10	
TOTAL	75	

RESULT:

Thus we successfully implemented dependency parsing using spacy and the output was verified.