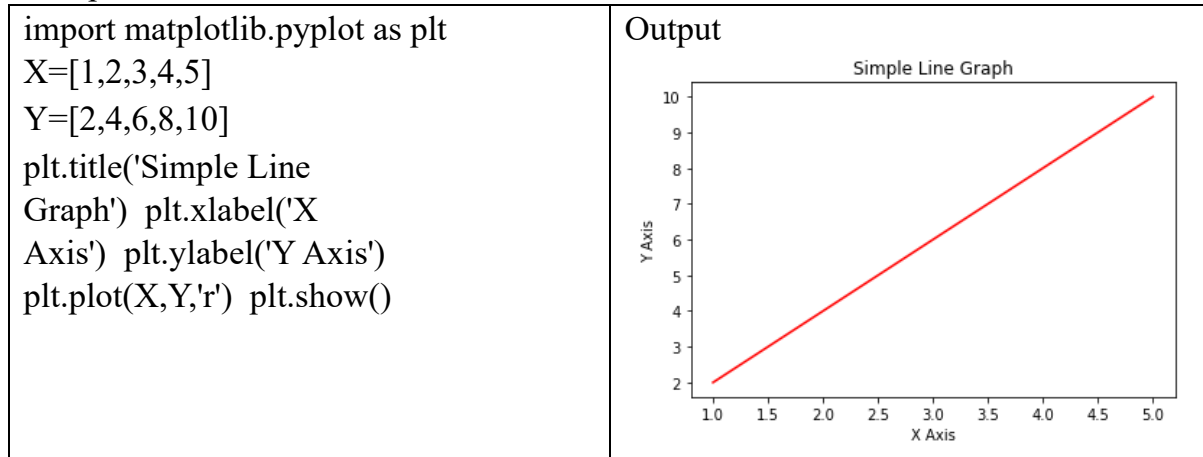# Chapter 3 Data Visualisation using pyplot

⚬ Data visualization is the technique to present the data in a pictorial or graphical format. It enables stakeholders and decision makers to analyze data visually. The data in a graphical format allows them to identify new trends and patterns easily.

⚬ The main benefits of data visualization are as follows:

  ✝ It simplifies the complex quantitative information
  ✝ It helps analyze and explore big data easily
  ✝ It identifies the areas that need attention or improvement
  ✝ It identifies the relationship between data points and variables
  ✝ It explores new patterns and reveals hidden patterns in the data

⚬ **matplotlib Library and pyplot Interface**
  ✝ The matplotlib is a python library that provides many interfaces functionally for 2D graphics
  ✝ In short, we can call mattplotlib as a high-quality plotting library of Python.
  ✝ The matplotlib library offers many different named collections of methods, pyplot is one such interface.
  ✝ pyplot is a collection of methods within matplotlib which allows user to construct 2D plots easily and interactively.
⚬ Installing matplotlib
  It is done using pip command in Command Prompt: **pip install matplotlib**
⚬ **Importing PyPlot**
  Syntax: import matplotlib.pyplot as plt
⚬ **Steps to plot in matplotlib:**
  ✝ Create a .py file & import matplotlib library to it using import statement   import

    matplotlib.pyplot as plt

  ✝ Set data points in plot( ) method of plt object
  ✝ Customize plot by setting different parameters
  ✝ Call the show() method to display the plot
  ✝ Save the plot/graph if required using function plt.savefig(file_path)

  ✝ ➢ Types of plot using matplotlib

  ✝ LINE PLOT   ❖ BAR GRAPH   ❖ HISTOGRAM etc.

⚬ **Line Plot customization**
  ✝ Custom line color
      ▢ plt.plot(x,y,'red')
      ▢ Change the value in color argument like 'b' for blue,'r','c',…..
  ✝ Custom line style and line width
      ▪ plt.plot(x,y, linestyle='solid' , linewidth=4).
  ✝ set linestyle to solid/dashed/dotted/dashdot   set linewidth as required   ❖ Title
    plt.title('DAY – TEMP Graph ') – Change it as per requirement   ❖ Label:
      ▢ plt.xlabel('Time') – to set the x axis label
      ▢ plt.ylabel('Temp') – to set the y axis label
  ✝ Changing Marker Type, Size and Color

❑ plt.plot(x,y,'blue',marker='*',markersize=10,markeredgecolor='magenta') ❖ Order of methods used in plot() function:

❑ Plt.plot(x,y,color,linewidth,linestyle,marker, markersize,markeredgecolor) ❖ Function used to show the graph – show()

Example

import matplotlib.pyplot as plt X=[1,2,3,4,5] Y=[2,4,6,8,10] plt.title('Simple Line Graph') plt.xlabel('X Axis') plt.ylabel('Y Axis') plt.plot(X,Y,'r') plt.show()	Output

○ Bar Graph
- ✞ A graph drawn using rectangular bars to show how large each value is.
- ✞ The bars can be horizontal or vertical.
- ✞ A bar graph makes it easy to compare data between different groups at a glance.
- ✞ Bar graph represents categories on one axis and a discrete value in the other.
- ✞ The goal of bar graph is to show the relationship between the two axes.
- ✞ Bar graph can also show big changes in data over time.
- ✞ Syntax : plt.bar(x,y) Bar graph customization

○ Custom bar color
- ✞ To set same color to all bars
  plt.bar(x,y, color="color code/color name")
- ✞ To set different colors for different bars
  plt.bar(x,y, color="color code/color name sequence")

○ Custom bar width
- ✞ To set same color to all bars plt.bar(x,y, width=float value)
- ✞ To set different widths for different bars plt.bar(x,y, width=float value sequence)

○ Title
  plt.title(' Bar Graph ') – Change it as per requirement

○ Label
  plt.xlabel('Overs') – to set the x axis label plt.ylabel('Runs') –

  to set the y axis label

| # Code to plot a bar graph import matplotlib.pyplot as plt overs=['1-10','11-20','21-30','31-40','41-50'] runs=[65,55,70,60,90] plt.xlabel('Over Range') plt.ylabel('Runs Scored') plt.title('India Scoring Rate') plt.bar(overs,runs) plt.show( )	Output
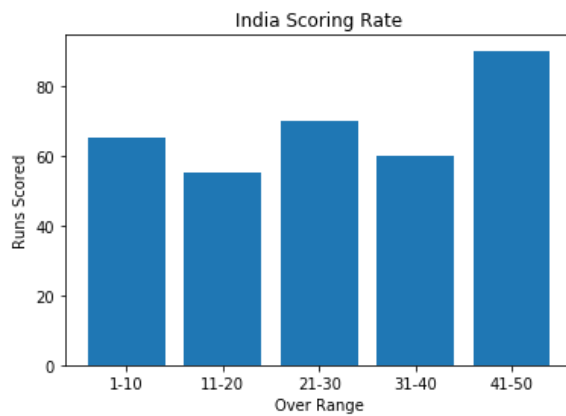
○ **Histogram**
- ✜ A histogram is a graphical representation which organizes a group of data points into user specified ranges.
- ✜ Histogram provides a visual interpretation of numerical data by showing the number of data points that fall within a specified range of values ("bins").
- ✜ It is similar to a vertical bar graph but without gaps between the bars.

○ **Difference between a histogram and a bar chart / graph** –

A bar chart majorly represents categorical data (data that has some labels associated with it), they are usually represented using rectangular bars with lengths proportional to the values that they represent. While histograms on the other hand, is used to describe distributions.
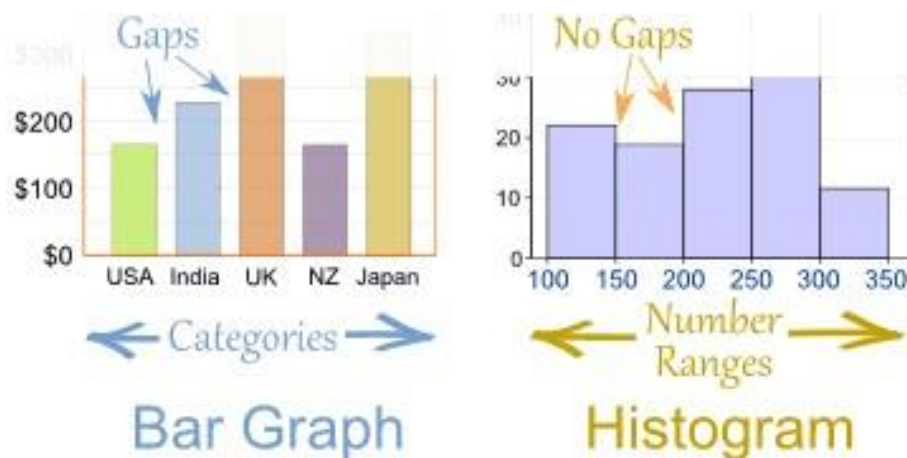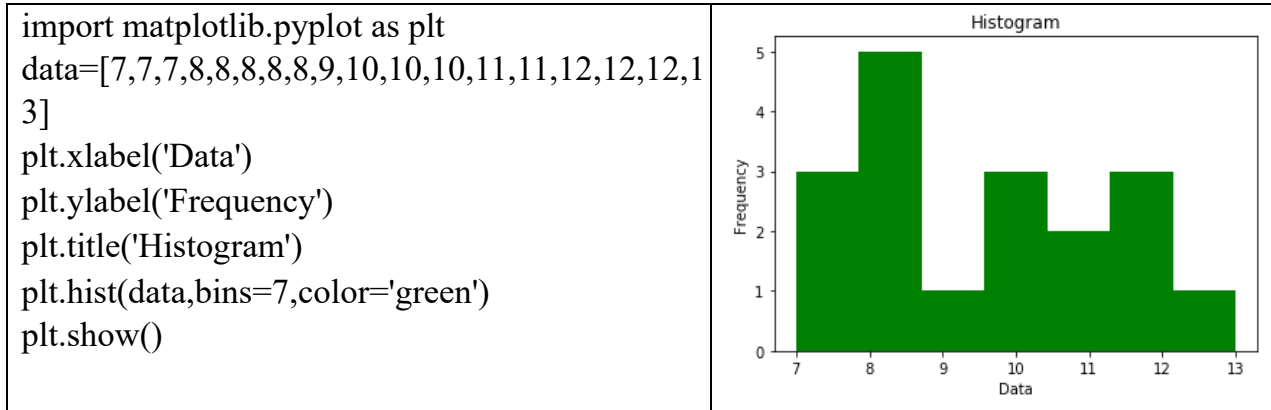

○ **Creating a Histogram :**
- ✜ It is a type of bar plot where X-axis represents the bin ranges while Y-axis gives information about frequency.
- ✜ To create a histogram the first step is to create bin of the ranges, then distribute the whole range of the values into a series of intervals, and count the values which fall into each of the intervals.
- ✜ Bins are clearly identified as consecutive, non-overlapping intervals of variables.
- ✜ The hist() function is used to create histogram
- ✜ Syntax: **plt.hist(x,other parameters)**

○ **Optional Parameters**

x	array or sequence of array
bins	optional parameter contains integer or sequence or strings default =5

histtype	optional parameter used to create type of histogram [bar, barstacked, step, stepfilled], default is "bar"
align	optional parameter controls the plotting of histogram [left, right, mid]
orientation	Optional. Possible values are 'horizontal' or 'vertical'
color	optional parameter used to set color or sequence of color specs

Program to plot histogram

```
import matplotlib.pyplot as plt
data=[7,7,7,8,8,8,8,8,9,10,10,10,11,11,12,12,12,13]
plt.xlabel('Data')
plt.ylabel('Frequency')
plt.title('Histogram')
plt.hist(data,bins=7,color='green')
plt.show()
```

- Title (title)
    - plt.title('Histogram ') – Change it as per requirement • Labelplt.xlabel('Data') – to set the x axis label plt.ylabel('Frequency') – to set the y axis label
- Legend –
    - A legend is an area describing the elements of the graph.
    - In the matplotlib library there is a function named legend() which is used to place a legend on the axes .
    - When we plot multiple ranges in a single plot, it becomes necessary that legends are specified. It is a color or mark linked to a specific data range plotted .
    - To plot a legend you need to do two things.
        - In the plotting function like bar() or plot() , give a specific label to the data range using label
        - Add legend to the plot using legend ( ) as per the sytax given below .
        Syntax : - plt.legend((loc=position number or string)
    - position number can be 1,2,3,4 specifying the position strings upper right/'upper left/'lower left/lower right respectively.
    - Default position is upper right or 1
- **Saving the Plot**
    - To save any plot savefig() method is used.
    - Plots can be saved in various formats like pdf,png,eps etc . plt.savefig('line_plot.pdf') // save plot in the current directory plt.savefig('d:\\plot\\line_plot.pdf') // save plot in the given path

MCQ

1. Which of the following is not a valid function pyplot?
    a) line()   b) plot()   c) hist()   d) bar()
2. Which of the following is use to create a line chart?
    a) line()   b) chart()   c) plot()   d) drawline
3. Which of the following functions will create a horizontal bar chart>

a) bar()   b) Horizontal   c) barplot()   d) barh()

4. Which argument will be used to specify the size of data points in a scatter() function?
    a) marker   b) size   c) ms   d) msize

5. choose the correct function to be used to show legend in a chart.
    a) show()   b) display()   c) legend()   d) show_legend()

6. The_____ function makes a plotted chart visible.
    a) plot()   b) visible()   c) display()   d) show()

7. From which library pyplot belongs to?
    a) matplotlib library b) pandas library   c) Numpy library   d) none of these

8. In data visualisation, the color code 'k' represents _____ color.
    a) Blue   b) Green   c) Black   d) Red

9. In chart, the statement to display 'Student Names' as x-asis title will be …………
    a) plt.xtitle='Student Names'   b) xlabel='Student Names'   c)
plt.xlabel('Students Name')   d) None of these

10. Choose the appropriate statement to show the title 'student's Performance' from the following.
    a) import matplotlit.pyplot as pyt   b) import matplotlit.pyplot as pyt
    pyt.title('student's Performance')   pyt.show ('student's Performance')
    c) import matplotlit.pyplot as pyt   d) None of these
    pyt.heading('student's Performance')

11. Which of the following plots will display the five-number summary of a set of data values?
    a) Bar graph   b) Histogram   c) Box plot   d) scatter plot

12. Which of the following statements will you choose to plot line graph from a DataFrame?
    a) <DataFrame>.plot()   b) < DataFrame>.plot(kind='line')
    c) Either (a) or (b)   d) None of these

13 . Which of the following functions is used to import data from Data from DataFrame to CSV file?
    a) to_csv()   b) read_csv()   c) call_csv()   d) dfto_csv()

14 . To use the field 'Rollno' of the CSV file as row indecis/labels of the DataFrame, which of the following arguments will be used in the read_csv()function?
    a) index='Rollno'   b) index_no='Rollno'
    c) index_col='Rollno'   d) row_index='Rollno'

15 . Which of the following arguments will be used to apply default column labels while transferring data from CSV file to the DataFrame?
    a) Head=None   b) header=None   c) column=None   d) header=none

Ans. 1. A) line(), 2. C) plot(), 3.d) barh(), 4. A)marker, 5. Legend(), 6. D)show(), 7. A) matplotlib library, 8. C)black, 9. C) Plt.xlabel('Students Name'), 10 a) import matplotlit.pyplot as pytpyt.title('student's Performance')   11. C)Box plot, 12. c) Either (a) or (b), 13. b) read_csv() , 14. c) index_col='Rollno', 15. b) header=None

-----

Importing/Exporting data between CSV files and Data Frames

CSV FILES
- CSV stands for Comma-separated values.
- This type of file can be opened with excel file or notepad.
- CSV file is nothing more than a simple text file. However, it is the most common,simple and easiest method to store tabular data.
- This particular format arranges tables by a specific structure divided into rows and columns.

- To import data from a CSV file into a DataFrame, you can use the read_csv() function.
- This function takes the path to the CSV file as its input, and returns a DataFrame object.
- For example, the following code imports the data from the data.csv file into a DataFrame:

```
import pandas as pd
df = pd.read_csv('data.csv')
```

- Parameter of read_csv() function
  - **sep**: str, default ',' - Delimiter to use as column separator
  - **header :** int, list of int, None, default 'infer' - Row number(s) to use as the column names, and the start of the data. Default behavior is to infer the column names.
  - **index_col** : int, str, sequence of int / str, or False, optional, default None - Specifies the column that should be used as the index of the DataFrame
  - **skiprows**: list-like, int or callable, optional - Line numbers to skip (0-indexed) or number of lines to skip (int) at the start of the file.
  - **nrows**: int, optional -Number of rows of file to read.

- To export a DataFrame to a CSV file, you can use the to_csv() function.

```
import pandas as pd
df=  {'Name': ['Pankaj', 'Meghna'], 'ID': [1, 2], 'Role': ['CEO', 'CTO']}
df.to_csv('employee.csv')
```

- Parameters of to_csv() function:
  - **sep:** str, default ',' - String of length 1. Field delimiter for the output file.
  - **columns**: sequence, optional - Columns to write.
  - **header**: bool or list of str, default True - Write out the column names. If a list of strings is given it is assumed to be aliases for the column names.
  - **index**: bool, default True - Write row names (index).

- This function takes the path to the CSV file as its output, and writes the DataFrame to the file. For example, the following code exports the df DataFrame to the output.csv file:

```
df.to_csv('output.csv')
```

MCQ

1. Which of the following functions is used to import data from a CSV file into a DataFrame?
   - a) read_csv()
   - b) to_csv()
   - c) write_csv()
   - d) import_csv()

   Answer: read_csv().

2. Which of the following functions is used to export a DataFrame to a CSV file?
   - a) read_csv()
   - b) to_csv()
   - c) write_csv()
   -) import_csv()

   Answer: to_csv().

3. Which of the following parameters is used to specify whether the first row of the CSV file contains column names?
   - a) header
   - b) sep
   - c) index_col
   - d) na_values

   Answer: header.

4. Which of the following parameters is used to specify the separator character used in the CSV file?
  a) header   b) sep   c) index_col   d) na_values
  Answer: sep.
5. Which of the following parameters is used to specify the column that should be used as the index of the DataFrame?
  a) header   b) sep   c) index_col   d) na_values
  Answer: index_col.

Questions answers

Q.1 define the following terms with reference to a CSV file. 1. Table , 2. Field
Ans. A CSV (comma-separated values) file is a text file that has a specific format which allows data to be saved in a table structured format.

Table:- one row for a new data point, each column for the value of an attribute. The first line contains the header: the values are the 'titles' of the columns.
Field:- A field in a comma-separated values (CSV) file is a column of data that is separated from other columns by commas.
Q.2 Write name the necessary modules to be imported while transferring data from a DataFrame to the CSV file.
Ans. Pandas library
Q.3 Wha is meant by data visualisation?
Ans. Data visualization is the representation of data through use of common graphics, such as charts, plots, infographics and even animations. These visual displays of information communicate complex data relationships and data-driven insights in a way that is easy to understand.
Q.3 Write down three advantages of CSV file.
Ans. The files have a simple structure and are human-readable;
- The data can be easily imported or exported into other programs;
- CSV file format can be easily and accurately compressed;
- The running costs are not high;
- Perfect fit for getting data out of one application and into another one;
Q.4 What are the different types of histogram that can be plotted? Name them.
Ans. Uniform histogram
- Symmetric histogram
- Bimodal histogram
- Probability histogram
Q.5 The following data represent the number of Indian touries went to various countries.
number=[1600, 1800, 2100, 1400, 2500]
country=[UK, USA, Japan, Canada, New Zealand]
write a code in python to plot a pie chart from the above data.
The code also includes the following:
 i. Title   ii. Creating an explode for japan
Ans. number=[1600, 1800, 2100, 1400, 2500]
country=['UK', 'USA', 'Japan', 'Canada', 'New Zealand']
plt.pie(number,labels=country,explode=[0,0,0.3,0,0])
plt.title('country')
plt.show()

Q.6 Consider the following DataFrame citydf showing the rainfall recorded for Kolkata, Mumbai and Delhi for Day 1 to Day4.

	Day	Kolkata	Mumbai	Delhi
0	Day1	6.2	4.2	4.5
1	Day2	4.8	5.8	5.5
2	Day3	6.0	5.0	4.0
3	Day4	5.5	6.0	5.4

Output of the above details is shown alongside and the incomplete code is given below.

```
#a code to plot bar graph from a DataFrame
import pandas as pd
import matplotlib.pyplot as plt
citydf=…………..({'Day':['Day1',' Day2','Day3','Day4'],'Kolkata':[ 6.2, 4.8, 6.0, 5.5],'Mumbai':[
4.2, 5.8, 5.0, 6.0],' Delhi':[ 4.5, 5.5, 4.0, 5.4]})
citydf.plot(x=…………,kind=………….)                    #to plot vertical bar graph
plt.ylabel(………………………….)
……………………………………………
plt.show()
```

fill in the blanks to the complete the code to produce the desired output.

Ans.

```
import pandas as pd
   import matplotlib.pyplot as plt
   citydf=pd.DataFrame({'Day':['Day1',' Day2','Day3','Day4'],'Kolkata':[ 6.2, 4.8, 6.0, 5.5],
               'Mumbai':[ 4.2, 5.8, 5.0, 6.0],'Delhi':[ 4.5, 5.5, 4.0, 5.4]})
   print(citydf)
   citydf.plot(x='Day',kind='bar')
   plt.ylabel('Rainfall')
   plt.xlabel('Rainfall in metro cities')
   plt.show()
```

DATABASE MANAGEMENT SYSTEM
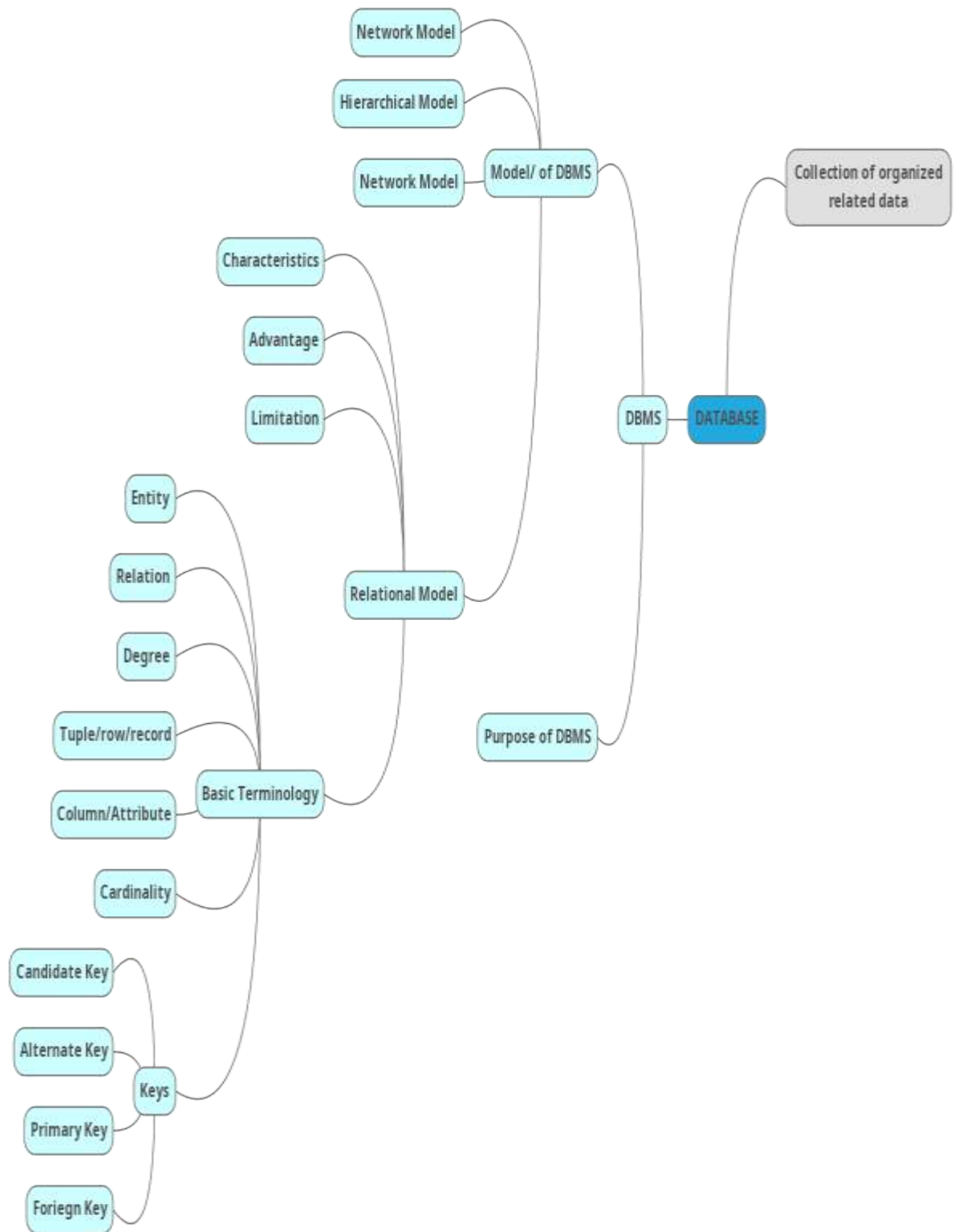
DATABASE:
- May be defined as a collection of interrelated data stored together to serve multiple application
- It is computer based record keeping system.
- It not only allows to store but also allows us modification of data as per requirements

DBMS:

- A DBMS refers to Database Management System
- It is a software that is responsible for storing, manipulating, maintaining and utilizing database.
- A database along with a DBMS is referred to as a database system.
- There are various DBMS software available in the market like :- Oracle, MS SQL Server, MySQL, Sybase, PostgreSQL, SQLite

Purpose of DBMS:
- Reduced Data redundancy – ➢ Control Data Inconsistency
- Sharing of data
- Ensure data integrity
- Enforce standard

Relational Database Model:
- In relational database model data is organized into table (i.e. rows and columns).
- These tables are also known as relations.
- A row in a table represent relationship among a set of values.
- A column represent the field/attributes related to relation under which information will be stored.
- For example if we want to store details of students then : Roll, Name, Class, Section, etc. will be the column/attributes and the collection of all the column information will become a

Row/Record **Sample Tables:**

EMPLOYEE

EMPNO	ENAME	GENDER	DEPTNO	SALARY	COMM
1	ANKITA	F	10	20000	1200
2	SUJEET	M	20	24000	
3	VIJAYA	F	10	28000	2000
4	NITIN	M	30	18000	3000
5	VIKRAM	M	30	22000	1700

DEPT

DEPTNO	DNAME	LOC
10	Accounting	New York
20	Research	Dallas
30	Sales	Chicago
40	Operations	Bostan

Component of a table:
- Byte: group of 8 bits and is used to store a character.
  ata Item: smallest unit of named data. It represent one type of information and often referred to as a field or column information
- Record : collection of data items which represent a complete unit of information ➢ Table: collection of all Rows and Columns.

Field/Attribiute/Column

deptno	dname	loc
10	Accounting	New York
20	Research	Dallas
30	Sales	Chicago
40	Operations	Bostan

Tuple/Row/record

Data Item
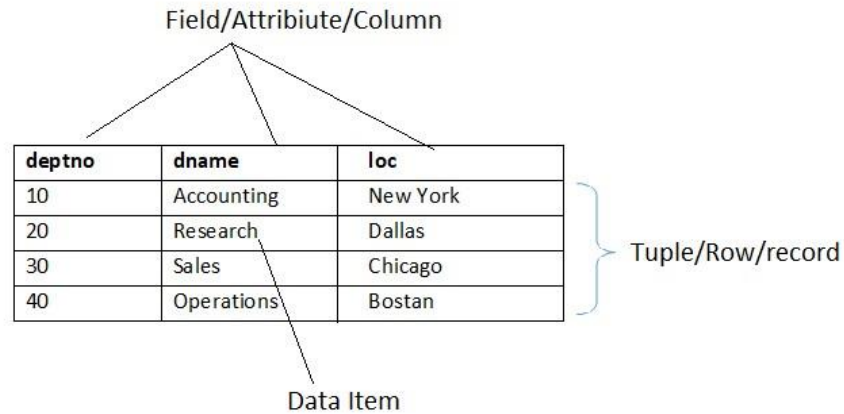
Common RDBMS:
- Oracle, MS SQL Server, MySQL, IBM DB2, IBM Informix, SAP Sybase, Adaptive Server Enterprise, SAP Sybase IQ, Teradata, PostgreSQL, SQLite, etc.
- Out of these MySQL, PostgreSQL and SQLite are Open source implementation.

MySQL
- Runs on virtually all platforms including Linux, Unix and Windows. Popular for web based application and online publishing. It is a part of LAMP (Linux, Apache, MySQL, PHP) stack

SQLite
- Relational DBMS but it is not client-server database engine rather, it is embedded into end program. Arguably the most widely deployed database engine as it is used by several browsers, OS and embedded systems (Mobiles).

PostgreSQL
- General purpose object-relational DBMS.
- It is the most advanced open source database system. It is free and open source i.e. source code is available under PostgreSQL license, a liberal open source license.

Common DBMS Tools for Mobile Devices:
- SQL Anywhere, DB2 Everywhere, IBM Mobile Database, SQL Server Compact, SQL Server Express, Oracle DatabaseLite, SQLite, SQLBase etc.
- Out of these SQLite is public domain open source implementation.

SQL and MYSQL:
- SQL stands for Structured Query Language.
- It is a language that enables you to create and operate on relational databases.
- MySQL uses SQL in order to access databases.
- It is the standard language used by almost all the database s/w vendors.

MYSQL Elements ➤ Literals
- Data types ➤ Nulls
- Comments

Literals
- It means the fixed value or constant value. It may be of character, numeric or date time type.

- Character and date/time literals are always in single quotation marks whereas numeric literals must be without single quotation marks ➢ For example – 'Virat', 12, 12.56, '04-20-2018'
- Date and time values are always in the format YYYY-MM-DD   HH:MI:SS
- Special character like quotes are always written be preceding it back-slash(\). For example if we want to store value as Tom's Cat then it should be written as Tom\'s Cat **Data Type**
- Means the type of value and type of operation we can perform on data. For example on numeric value we can store numbers and perform all arithmetic operations and so on.
- Numeric
- Date and time
- String types

Numeric Data Types

DATA TYPE	DESCRIPTION
INT	Numbers without decimal. Store up to 11 digits. -2147483648 to 2147483647
TINYINT	Small integer value between 0 – 255 (4 digits)
SMALLINT	More than TINYINT between -32768 to 32767 (5 digit)
MEDIUMINT	Integer values up to 9 digits
BIGINT	Very large integer value up to 11 digits
FLOAT(M,D)	Real numbers i.e. number with decimal. M specify length of numeric value including decimal place D and decimal symbol. For example if it is given as FLOAT(8,2) then 5 integer value 1 decimal symbol and 2 digit after decimal TOTAL – 8. it can work on 24 digits after decimal.
DOUBLE(M,D)	Real numbers with more precision up to 53 place after decimal.
DECIMAL	It is used to store exact numeric value that preserve exact precision for e.g. money data in accounting system. **DECIMAL(P,D) means P no. of significant digits (1-65), D represent no. of digit after decimal(0-30), for e.g DECIMAL(6,2) means 4 digit before decimal and 2 digit after decimal. Max will be 9999.99**

- Difference between CHAR & VARCHAR

CHAR	VARCHAR
Fixed length string	Variable length string
Fast, no memory allocation every time	Slow, as it take size according to data so every time memory allocation is done
It takes more memory	It takes less space

Simple Queries in SQL

- **SHOW DATABASES**- This command is used to list all databases on MySql Server
- **USE <DATABASENAME>**- This command is used to change/select/open given database e.g. To open a database named 'test'
  >>> **use test**
- **SHOW TABLES** – This command will list all the tables from current database. If no database is selected it will generate error.

- **SELECT DATABASE ()** – This command will display the name of current database.
- **DESC <TABLENAME> OR DESCRIBE <TABLENAME>** - This command will be used to display the structure of the table. For Eg: The above figure display the use of **describe** command


```
mysql> desc emp;
+----------+-------------+------+-----+---------+-------+
| Field    | Type        | Null | Key | Default | Extra |
+----------+-------------+------+-----+---------+-------+
| empno    | decimal(4,0)| NO   |     | NULL    |       |
| ename    | varchar(10) | YES  |     | NULL    |       |
| job      | varchar(9)  | YES  |     | NULL    |       |
| mgr      | decimal(4,0)| YES  |     | NULL    |       |
| hiredate | date        | YES  |     | NULL    |       |
| sal      | decimal(7,2)| YES  |     | NULL    |       |
| comm     | decimal(7,2)| YES  |     | NULL    |       |
| deptno   | decimal(2,0)| YES  |     | NULL    |       |
+----------+-------------+------+-----+---------+-------+
8 rows in set (0.00 sec)
```

- **CREATE DATABASE <DATABASENAME>** - This command is use to create a new database.
  For example – **create database mydb** will new database **mydb**
  Above command will generate error, if database already exist. To supress the error following command can be used **create database if not exist mydb**

- **DROP DATABASE <DATABASENAME>** - This command is used to remove existing database.
  For example – **drop database mydb** will permanently delete **mydb** database.
  Above command will generate error, if database already exist. To supress the error following command can be used **drop database if exist mydb**

- **CREATE TABLE <TABLENAME>** - This command / statement is used to create a new table in a database. The syntax the is
  CREATE TABLE table_name
  (
  column1 datatype[(size)] [constraint] , column2
  datatype[(Size)] [constraint], column3
  datatypeI[(size)] [constraints] ,
  ....);
  Here size and constraints are optional. e. g.
  CREATE TABLE PET
  (
  **   NAME VARCHAR(20),**
  **   OWNER VARCHAR(20),**
  **   SPECIES VARCHAR(20),**
  **   SEX CHAR(1),**
  **   BIRTH DATE, DEATH DATE**
  );

- **DROP TABLE <TABLENAME>** - This command is used to permanently delete the table from database.
  For example, **drop table pet**; will delete **pet** table from database
- **ALTER TABLE <TABLENAME>** - This command is used to modify the structure of existing table such as adding new column, removing existing column, rename or changing data type, size and constraints.
    o **Adding new column to exiting table Syntax** :

Alter table <tablename>
Add [column] column_name datatype [(size)];

- o **Removing new existing column from the table** Syntax:
  Alter table <tablename>
  drop column <column_name>;

- o **Changing datatype/size of the column** Systax:
  alter table <tablename>modify column <colname> datatype(size);

- o **Renaming column name**
  Alter table tablename
  Change old_columnname new_column_name   datatype (size)

⊙ **UPDATE <TABLENAME>** - This command is used to update data from the table Syntax:

UPDATE table_name SET column_name=new_value, column2_name=new_value
WHERE condition;
   e.g. UPDATE emp set sal=sal+100 where ename ='Scot'

The above query will update salary of Scot by 100.
If where clause is not given, it will update/modify the value of column from each row.

⊙ **INSERTING INTO TABLE_NAME** – This command is used to add new row
to the table Syntax :
INSERT INTO table_name VALUES (list of values)

⊙ **SELECT DATA FROM TABLE USING SELECT STATEMENT**

Syntax:

    SELECT   */ COLUMN LIST
    FROM   TABLE(S)
    WHERE   CLAUSE/CONDITION
     GROUP BY CLAUSE/COLUMN NAME
    HAVING   CLAUSE/CONDITION
     ORDER BY COLUMN NAME ASC/DESC

⊙ **Selecting/displaying entire data from the table** Syntax:
        SELECT * From tablename;

⊙ **Selecting/displaying data from specific column** Syntax:
        SELECT column1, column2, column3, …. FROM tablename;
⊙ **Giving descripting name for column using column alias** Syntax:
        SELECT */col_name as 'alias1' FROM tablename;

- **Removing duplicate value from the column using distinct clause** Syntax:
  DISTINCT column_name FROM tablename;

- **Display all data from column using all clause** Syntax:
  SELECT ALL column_name FROM tablename;

- **Inserting text in the query** Syntax:
  SELECT 'text' FROM tablename;

MCQ

1. Assertion (A): Database commands are case sensitive.

Reason (R): MySql makes no difference whether you type the commands in lowercase or uppercase while creating databases.

Based on the above discussion, choose an appropriate statement from the options given below:

a) Both A and R are true and R is the correct explanation of A.

b) Both A and R true and R is not the correct explanations of A.

c) A is true but R is false.

d) A is false but R is true.

e) Both A and R are false.

Ans. d) A is false but R is true.

2 . What does the letter S signify in the word SQL?

a) Structure   b) Structured c) semi-sturcture   d) semi-sturctured

ans. b) Structured

Questions and answers

1. What is Database? What are the advantages of Database System?

Ans. A database is a collection of data that is organized and stored electronically. A database management system (DBMS) is software that allows users to store, retrieve, and edit the data in a database.

Here are some advantages of a database management system:

- Improved data security: A DBMS provides a framework for enforcing data privacy and security policies.
- Data integrity: A relational database can perform data typing and validity checks to ensure that data is entered correctly.
- Data backup: A DBMS keeps a backup of all data, making it easier to recover data in case of a failure.
- Data sharing: A DBMS allows users to share data within or outside an organization, but users must follow the correct authorization protocol.
- Better data integration: A DBMS provides wider access to well-managed data, which can lead to a better understanding of an organization's operations.
- Improved decision making: A DBMS can help improve decision making.
- Increased productivity: A DBMS can help increase productivity.
- Enhanced data consistency: A DBMS can help enhance data consistency.
- Improved data recovery: A DBMS can help improve data recovery.

Q.2 How are SQL commands classified?

Ans. These SQL commands are mainly categorized into five categories: DDL – Data Definition Language. DQL – Data Query Language. DML – Data Manipulation Language.

Q.3 Define RDBMS, name any three RDBMS software.

Ans. A relational database management system (RDBMS) is a program that lets you create, update, and manage relational databases. Relational databases are the most popular type of database and store data in tables made up of rows and columns. Here are some examples of RDBMS software:

- MySQL: A popular RDBMS
- PostgreSQL: A popular RDBMS
- MariaDB: A popular RDBMS
- Microsoft SQL Server: A popular RDBMS
- Oracle Database: A popular RDBMS
- SQLite: A popular RDBMS
- SQLBase: A proprietary RDBMS
- SQream DB: A proprietary RDBMS
-

Q.4 Abhik has written a code to read a table 'Company' available in MySQL database (MyDatabase) to fetch the rows into a DataFrame using (mysql.connector/pymysql) any one package and write output of 5 record of dataframe.

Ans.

import pymysql as sql
conn=sql.connect(host='localhost',user='root',password='1234',port=3377,database='abhi')
df1=pd.read_sql('select * from tlogin',conn)
df2=df1.head(5)
print(df2)

Q5. The table 'Grocery' Showing details of the products being sold in a grocery shop is given below:

P_Code	P_Name	Manu_Compy	Price	Qty
P01	Shampoo	Head & Shoulders	190	90
P02	Body Wash	Nivea	230	85
P03	Washing Powder	Ariel	265	45
P04	Washing Liquid	Surf	220	80
P05	Tooth Paste	Sensodyne	190	90
P06	Hand Wash	Santoor	180	40

a. Create the table 'Grocery' with appropriate data types and constraints (if any).
b. Identify the 'Primary key' in the table 'Grocery'.
c. Add a new column 'Discount' in the table 'Grocery' with the requisite data type.
d. Calculate the value of the discount in the table 'Grocery' as 10% of the Price and display in the column 'Discount'.

Ans.

a. create table grocery(p_code varchar(20) primary key,p_name varchar(20), manu_compy varchar(20), price int, qty int);
b. P_Code varchar(20) primary key
c. alter table grocery add column discount int;
d. UPDATE GROCERY SET DISCOUNT=(PRICE*10)/100;
   FOR SHOW TABLE select * from grocery;