



केन्द्रीय विद्यालय संगठन, एर्नाकुलम क्षेत्र
KENDRIYA VIDYALAYA SANGATHAN
ERNAKULAM REGION

CLASS XII

INFORMATICS PRACTICES(065)

STUDENT SUPPORT MATERIAL

TERM-I

2021-22



केन्द्रीय विद्यालय संगठन, एर्नाकुलम क्षेत्र

CHIEF PATRON



Mr R Senthil Kumar

Deputy Commissioner
KVSRO Ernakulam

PATRONS



Mrs. Deepti Nair
Assistant Commissioner
KVSRO Ernakulam



Mr. Santhosh Kumar N
Assistant Commissioner
KVSRO Ernakulam

CO-ORDINATOR

Mr.Jyothi Mohan N V

Principal

Kendriya Vidyalaya S A P, Peroorkada

PREPARED BY PGT COMPUTER SCIENCE

Mrs. SINI ALEX , KV PATTOM SHIFT-1

Mrs.DEEPA P R, KV SAP PEROORKADA

Mrs.BINDHYA N, KV KELTRON NAGAR

Mr. ARUN PRASANTH, KV KANJIKODE

Mr. SOJU.S, KV INS DRONACHARYA

Mrs.ANJU MATHEW, KV RB KOTTAYAM

Mrs.RESHMA SURENDRAN, KV No2, KOCHI

Mrs. HEMA C N, KV No1, KOCHI

आर सेन्डिल कुमार
उपायुक्त

R. Senthil Kumar
Deputy Commissioner



केन्द्रीय विद्यालय संगठन,
क्षेत्रीय कार्यालय, एरणाकुलम

KENDRIYA VIDYALAYA SANGATHAN
REGIONAL OFFICE,
ERNAKULAM,
KOCHI – 682 020
Ph. No.0484- 2205111(DC), 2203091(Fax))
Website: www.roernakulam.kvs.gov.in
Email : dcernakulamregion@gmail.com

F.31/Acad/KVS(EKM)

Dated: 01.11.2021

Message

I feel immense pleasure to publish the study material for class XII Informatics Practices (065). This support material is prepared incorporating all the recent changes in curriculum and assessment process made by CBSE. I am sure it will definitely be of great help to class XII students of all Kendriya Vidyalayas.

Getting acquainted with the latest changes will help students to prepare well for the board examination and enable students to face case based and Multiple-Choice Questions with confidence. This support material has been prepared by a team of dedicated and veteran teachers with expertise in their respective subjects.

The Support material contains all the important aspects required by the students- the design of question paper, term wise split up syllabus, summary of all the chapters, important formulas, Sample question papers, problem solving and Case study questions.

I hope that this Support Material will be used by students and teachers as well and will prove to be a good tool for quick revision.

I would like to express my sincere gratitude to the In-charge principal and all the teachers who have relentlessly worked for the preparation of this study material. Their enormous contribution in making this project successful is praiseworthy.

Meticulous planning blended with hard work, effective time management and sincerity will help the students to reach the pinnacle of success.

Wish you all the best

(R Senthil Kumar)

Sh. Jyothimohan N V
Principal
Kendriya Vidyalaya
SAP Peroorkada

Informatics Practices
CLASS XII
Code No. 065
2021-2022

1. **Prerequisite:** Informatics Practices – Class XI

2. **Learning Outcomes**

At the end of this course, students will be able to:

- Create Series, Data frames and apply various operations.
- Visualize data using relevant graphs.
- Design SQL queries using aggregate functions.
- Learn terminology related to networking and the internet.
- Identify internet security issues and configure browser settings
- Understand the impact of technology on society including gender and disability issues.

3. **Distribution of Marks and Periods**

Unit No	Unit Name	Marks	Periods		Total Period
			Theory	Practical	
1	Data Handling using Pandas and Data Visualization	25	25	25	50
2	Database Query using SQL	25	20	17	37
3	Introduction to Computer Networks	10	12	0	12
4	Societal Impacts	10	14	-	14
	Project	-	-	7	7
	Practical	30	-	-	-
	Total	100	71	49	120

Term - 1

Distribution of Theory Marks

Unit No	Unit Name	Marks
1	Data Handling using Pandas and Data Visualization	25
4	Societal Impacts	10
	Total	35

Unit 1:

Data Handling using Pandas and Data Visualization

Data Handling using Pandas -I

- Introduction to Python libraries- Pandas, Matplotlib.
- Data structures in Pandas - Series and data frames. Series: Creation of series from dictionary, scalar value; mathematical operations; series attributes, head and tail functions; selection, indexing and slicing.
- Data Frames: creation of data frames from dictionary of series, list of dictionaries, text/CSV files, display, iteration. Operations on rows and columns: add (insert /append) , select, delete (drop column and row), rename, Head and Tail functions, indexing using labels, Boolean indexing.

Data Visualization

- Data Visualization : Purpose of plotting, drawing and saving of plots using Matplotlib (line plot, bar graph, histogram). Customizing plots:: adding label, title, and legend in plots.

Unit 4:

Societal Impacts

- Digital footprint, net and communication etiquettes,
- Data protection, intellectual property rights (IPR), plagiarism, licensing and copyright,
- Free and open source software (FOSS),
- Cybercrime and cyber laws, hacking, phishing, cyber bullying, overview of Indian IT Act.
- E-waste: hazards and management. Awareness about health concerns related to the usage of technology.

Distribution of Practical Marks

Topic	Marks
Pandas program (pen and paper or Collab or any online idle or pyroid screen for mobile)	8
Practical File 15 Pandas Programs	3
Project synopsis	2
Viva	2
Total	15

Suggested Practical List

Data Handling

1. Create a panda's series from a dictionary of values and a ndarray
2. Given a Series, print all the elements that are above the 75th percentile.
3. Create a Data Frame quarterly sales where each row contains the item category, item name, and expenditure. Group the rows by the category and print the total expenditure per category.
4. Create a data frame for examination result and display row labels, column labels data types of each column and the dimensions
5. Filter out rows based on different criteria such as duplicate rows.
6. Importing and exporting data between pandas and CSV file

5.2 Visualization

1. Given the school result data, analyses the performance of the students on different parameters, e.g subject wise or class wise.
2. For the Data frames created above, analyze, and plot appropriate charts with title and legend.
3. Take data of your interest from an open source (e.g. data.gov.in), aggregate and summarize it. Then plot it using different plotting functions of the Matplotlib library.

Project Synopsis

The synopsis should cover the brief description about the project along with reasons for selection of the dataset. The learner should write the source of the dataset whether created or taken from any reliable source. The learner should write what analytics can be done on the project.

Term - 2

Distribution of Theory Marks

Unit No	Unit Name	Marks
2	Database Query using SQL	25
3	Introduction to Computer Networks	10
	Total	35

Unit 2:

Database Query using SQL

- Math functions: POWER (), ROUND (), MOD ().
- Text functions: UCASE ()/UPPER (), LCASE ()/LOWER (), MID ()/SUBSTRING ()/SUBSTR (), LENGTH (), LEFT (), RIGHT (), INSTR (), LTRIM (), RTRIM (), TRIM ().
- Date Functions: NOW (), DATE (), MONTH (), MONTHNAME (), YEAR (), DAY (), DAYNAME (). Aggregate Functions: MAX (), MIN (), AVG (), SUM (), COUNT (); using COUNT (*).
- Querying and manipulating data using Group by, Having, Order by.

Unit 3:

Introduction to Computer Networks

- Introduction to networks, Types of network: LAN, MAN, WAN.
- Network Devices: modem, hub, switch, repeater, router, gateway.
- Network Topologies: Star, Bus, Tree, Mesh.
- Introduction to Internet, URL, WWW and its applications- Web, email, Chat, VoIP.
- Website: Introduction, difference between a website and webpage, static vs dynamic web page, web server and hosting of a website.
- Web Browsers: Introduction, commonly used browsers, browser settings, add-ons and plug-ins, cookies.

Distribution of Practical Marks

Topic	Marks
SQL queries (pen and paper)	7
Practical File – 12 SQL Queries	2
Final Project Submission	3
Viva	3
Total	15

Suggested Practical List

Data Management

1. Create a student table with the student id, name, and marks as attributes where the student id is the primary key.
2. Insert the details of a new student in the above table.
3. Delete the details of a student in the above table.
4. Use the select command to get the details of the students with marks more than 80.
5. Find the min, max, sum, and average of the marks in a student marks table.
6. Find the total number of customers from each country in the table (customer ID, customer Name, country) using group by.
7. Write a SQL query to order the (student ID, marks) table in descending order of the marks.


Project Work

The aim of the class project is to create tangible and useful IT applications. The learner may identify a real-world problem by exploring the environment. e.g. Students can visit shops/business places, communities or other organizations in their localities and enquire about the functioning of the organization, and how data are generated, stored, and managed.

The learner can take data stored in csv or database file and analyze using Python libraries and generate appropriate charts to visualize. If an organization is maintaining data offline, then the learner should create a database using MySQL and store the data in tables.

Data can be imported in Pandas for analysis and visualization. Learners can use Python libraries of their choice to develop software for their school or any other social good. Learners should be sensitized to avoid plagiarism and violation of copyright issues while working on projects. Teachers should take necessary measures for this. Any resources (data, image etc.) used in the project must be suitably referenced.

The project can be done individually or in groups of 2 to 3 students. The project should be started by students at least 6 months before the submission deadline.




Class XII Informatics Practices

Term –I Study Material

Contents

	Page No
1. Data Handling using Series.....	<u>1-38</u>
2. Data Handling using Data Frames.....	<u>39-70</u>
3. Data Visualization.....	<u>71-94</u>
4. Societal Impacts.....	<u>95-103</u>

(Based on latest CBSE Exam Pattern for the Session 2021-22)



UNIT-1

Data Handling using Pandas –I

Python Pandas

- Python library developed by **Wes McKinney**
- Derived its name from “**PANel DAta System**”
- Two basic data structures- **Series and Dataframe**
- Series is one-dimensional
- Dataframe is two-dimensional
- installed using the command
pip install pandas
- imported to a python program using the command
import pandas
or
import pandas as pd
(where *pd* is an alias name for *pandas*)

Comparison between Series and Dataframes

Series	Dataframe
One-dimensional	Two-dimensional
Homogenous data i.e. all elements are of same type	Heterogeneous data i.e. elements of different datatypes
Value mutable i.e. element's value can be changed	Value mutable i.e. element's value can be changed
Size immutable i.e. once created, size of series cannot be changed	Size mutable i.e. size can be changed after creation

Series Datastructure

- Creating empty Series

<series object>=pandas.Series()

eg: s1=pandas.Series()

- **Creating Series from a List/Tuple**

<series object>=pandas.Series(<list/tuple>,index=<python sequence>)

Note: index argument is optional. If not given, **index is taken as 0,1,2,3,--- by default**

eg:

```
import pandas as pd
s1=pd.Series([12,10,14,16])
s2=pd.Series([12,10,14,16],index=['a','b','c','d'])
print("Series object with default index")
print(s1)
print("Series object with specified index")
print(s2)
```

Output:

Series object with default index

0	12
1	10
2	14
3	16

Series object with specified index

a	12
b	10
c	14
d	16

- **Creating Series from an ndarray**

<series object>=pandas.Series(<ndarray>, index=<python sequence>)

Note: index argument is optional. If not given, **index is taken as 0,1,2,3,--- by default**

eg:

```
import pandas as pd
import numpy as np
ar1=np.arange(10,20,3)
ar2=np.array([20,25,30])
```

```
s1=pd.Series(ar1)
s2=pd.Series(ar2,index=('Mark1','Mark2','Mark3'))
print('Series object from ndarray with default index')
print(s1)
print('Series object from ndarray with specified index')
print(s2)
```

Output:

```
Series object from ndarray with default index
0    10
1    13
2    16
3    19
Series object from ndarray with specified index
Mark1    20
Mark2    25
Mark3    30
```

- **Creating series from a Python dictionary**

<series object>=pandas.Series(<dictionary>,index=<Python sequence>)

Note: index argument is optional. If not given, **keys of the dictionary becomes the index values**

eg:

```
import pandas as pd
dict1={"Name":"Rajeev","Age":17,"Class":"XII"}
s1=pd.Series(dict1)
print('Series object from dictionary with keys as index')
print(s1)
```

Output:

```
Series object from dictionary with keys as index
Name    Rajeev
Age      17
Class   XII
```

- **Creating Series from a scalar value**

<series object>=pandas.Series(<scalar value>,index=<Python sequence>)

Note: While creating a Series from a scalar value, Index argument is mandatory

eg:

```
import pandas as pd
s1=pd.Series(15,index=['Mark1','Mark2','Mark3'])
print('Series object from scalar value')
print(s1)
```

Output:

```
Series object from scalar value
Mark1    15
Mark2    15
Mark3    15
```

MCQ questions

Section A	
1	<p>Which of the following command is used to install python pandas?</p> <p>a) install pandas b) pandas install python c) python install pandas d) pip install pandas</p> <p>Ans: d</p>
2	<p>Pandas Series is a -----array</p> <p>a) one dimensional b) two dimensional c) three dimensional d) None of the above</p> <p>Ans: a</p>
3	<p>Which of the following is the purpose of Python Pandas?</p> <p>a) To create a GUI programming b) To create a database</p>

	<p>c) To create a High level array d) All the above</p> <p>Ans: c</p>
4	<p>Identify the correct statement</p> <p>a) Standard marker for missing data in Pandas is NaN b) Series act in a way similar to that of an array c) Both of the above d) None of the above</p> <p>Ans: c</p>
5	<p>Minimum number of arguments required to pass in pandas Series function for creating a non-empty series-----</p> <p>a) 0 b) 1 c) 2 d) 3</p> <p>Ans: b</p>
6	<p>Pandas is a/an -----python library</p> <p>a) proprietary b) open source c) shareware d) None of the above</p> <p>Ans: b</p>
7	<p>Which of the following is not a feature of pandas series?</p> <p>a) Series values are mutable b) Series data is homogenous c) Series is a 1-D array d) Series is size mutable</p> <p>Ans: d</p>
8	<p>The label associated with a particular data value in Series is called.....</p> <p>a) Item b) Index c) Column d) Values</p> <p>Ans: b</p>
9	<p>Tabular data can be processed using-----</p>

	<p>a) Numpy b) Pandas c) Matplotlib d) All of these</p> <p>Ans: b</p>
10	<p>Which of the following datatype can be given as data in a pandas Series function?</p> <p>a) a python dictionary b) an ndarray c) a scalar value d) All the above</p> <p>Ans: d</p>
11	<p>Pandas series is a combination of _____</p> <p>a) Records arranged in row and column b) Collection of one dimensional data and associated index c) Collection of tabular data in two-dimension d) None of the above</p> <p>Ans: b</p>
12	<p>Which of the following is correct statement for creating empty series? (Assume that pandas library is already imported as pd)</p> <p>a) <code>ser = pd.Series(NaN)</code> b) <code>ser = pd.Series</code> c) <code>ser = pd.Series()</code> d) None of the above</p> <p>Ans: c</p>
13	<p>Which of the following condition raise a ValueError while creating a series?</p> <p>a) Data values are provides without indexes b) Scalar value is given as data c) Number of data values are not same as number of indexes d) All of the above</p> <p>Ans: c</p>
14	<p>How many values will be there in array1, if given code is not returning any error?</p> <pre>>>> series4 = pd.Series(array1, index = ["Jan", "Feb", "Mar", "Apr"])</pre> <p>a) 1 b) 2</p>

- c) 3
- d) 4

Ans: d

15 When we create a series from dictionary then the keys of dictionary become _____

- a) Index of the series
- b) Value of the series
- c) Caption of the series
- d) None of the series

Ans: a

Section B

1 For creating the below series, S1, which of the following command(s) can be used?

Series(S1)

```
0 10
1 12
2 14
```

- a) S1=pandas.Series([10,12,14])
- b) S1=pandas.Series([10,12,14],index=[0,1,2])
- c) S1=pandas.Series(index=[0,1,2],data=[10,12,14])
- d) All of the above

Ans: d

2 Write the output of the following :

```
>>> S1=pandas.Series("Hello", index = ['One', 'Two', 'Three'])
>>> print(S1)
```

- a)
One Hello
Two Hello
Three Hello
- b)
One Hello
- c) Error
- d) None of the above

	Ans: a
3	<p>Choose correct option :</p> <pre>import pandas as p1 #line1 Lst = [11,12,13,14] #line2 s1=p1.Series(Lst , index = ('a','b','c')) #line3 print(s1) #line4</pre> <p>Which line of above code will generate error?</p> <p>a) line1 b) line2 c) line3 d) line4</p> <p>Ans: line3</p>
4	<p>Which of the following code will generate the following output?</p> <pre>January 31 February 28 March 31</pre> <p>a) import pandas as pd S1 = pd.Series(data = [31,28,31], index=["January","February","March"]) print(S1)</p> <p>b) import pandas as pd S1 = pd.Series([31,28,31], index=["January","February","March"]) print(S1)</p> <p>c) Both of the above d) None of the above</p> <p>Ans: c</p>
5	<p>Read the statements given below and identify the right option</p> <p>Statement 1: Series is a one-dimensional labeled array capable of holding any data type</p> <p>Statement 2: If data is an ndarray, index must be the same length as data.</p> <p>a) Statement 1 is correct, statement 2 is wrong b) Statement 1 is wrong, Statement 2 is correct c) Both statement 1 and statement 2 are correct d) Both statements are incorrect</p>

	<p>Ans: c</p>
6	<p>Read the statements given below and identify the right option</p> <p>Assertion (A): You need to install the pandas library using the pip install command.</p> <p>Reason (R): You can also access pandas without installation.</p> <p>a) Both A and R are true and R is the correct explanation of A b) Both A and R are true but R is not the correct explanation of A c) A is true but R is false d) A is false but R is true</p> <p>Ans: c</p>
7	<p>Read the statements given below and identify the right option</p> <p>Assertion (A): We cannot modify the values of Series elements once created.</p> <p>Reason (R): Series is an immutable object.</p> <p>a) Both A and R are true and R is the correct explanation of A. b) Both A and R are true and R is not the correct explanation of A. c) A is true but R is false. d) Both A and R are false</p> <p>Ans: d</p>
8	<p>Ananya wants to store her Term-I mark in a Series which is already stored in a NumPy array. Choose the statement which will create the series with Subjects as indexes and Marks as elements.</p> <pre>import pandas as pd import numpy as np Marks =np.array([30,32,34,28,30]) subjects = ['English','Maths','Chemistry','Physics','IP'] Series1= _____</pre> <p>a) pd.Series(Marks,index=subjects) b) pd.Series(np.Marks,index=subjects) c) pd.Series(index=Marks, subjects) d) pd.Series(Marks,index)</p> <p>Ans: a</p>
9	<p>Write the output of the following:</p> <pre>import pandas as pd S1 = pd.Series(data = range(31, 2, -6), index = [x for x in "aeiou"]) print(S1)</pre> <p>a) a 31</p>

e 25
i 19
o 13
u 7
dtype: int64

b) a 31
e 25
i 19
o 13
dtype: int64

c) Error
d) None of the above

Ans: a

10 Tushar is a new learner for the python pandas series. He learned some of the concepts of python in class 11 with NumPy module. He wants to create a series with the following code. The index should be from 20 to 30 and data value is obtained by multiplying each index value by 7. Help him to create series by following code:

```
import pandas as pd
import numpy as np
s=np.arange(20,30)
```

Choose the correct code to fill in the blank above:

a) sm7= pd.Series(s, index=s*7)
b) sm7=pd.Series(s*7,index=s)
c) sm7=pd.Series([s*7],index=s)
d) All of the above

Ans: b

Section C

1 Ms. Priya is a python developer and she created a series using the following code, but she missed some of the lines given as blank. Fill the blanks and help her to complete the code:

```
import pandas as _____ #statement 1
import _____ as np      #statement2
s1=pd.Series([3,4,_____,44,67]) #statement 3
print(_____) #statement 4
```

Output:

	0 3 1 4 2 NaN 3 44 4 67
i)	Identify the missing code in statement 1 a) p b) pd c) pandas d) pdy Ans: b
ii)	Name the library to be imported in statement2 for the code to execute correctly a) numpy b) pandas c) matplotlib d) pyplot Ans: a
iii)	Complete statement 3 to obtain the output shown in the code a) NaN b) np.NaN c) np.None d) none of the above Ans: b
iv)	Fill the missing code to display the Series a) np b) pd c) s1 d) Series Ans: c

Mathematical Operations on Series

a) Vector operations on Series objects

Any operation on Series object will be applied to each item of the Series. This is known as *Vector Operation*

eg: Consider the Series S1

```
0  5
1 10
2 11
3 25
```

All the following examples are based on the Series S1

Operation	Output
>>> S1+3	0 8 1 13 2 14 3 28
>>> S1*2	0 10 1 20 2 22 3 50
>>> S1/2	0 2.5 1 5.0 2 5.5 3 12.5
>>> S1%2	0 1 1 0 2 1 3 1

b) Arithmetic on Series Objects

All arithmetic operations like addition, subtraction, multiplication, division etc. can be done on Series objects

The arithmetic operation is performed only on matching indexes. If the indexes are not matching, NaN will be produced as output.

Eg;

```

import pandas as pd
s1=pd.Series([15,20,21], index=['A','B','C'])
s2=pd.Series([10,10,6], index=['A','B','D'])
print('Series object 1(s1)')
print(s1)
print('Series object 2(s2)')
print(s2)

```

Output

Series object 1(s1)

A 15

B 20

C 21

Series object 2(s2)

A 10

B 10

D 6

Arithmetic operation	Operator	Example
Addition	+ or <i>add</i>	<p>>>> s1+s2 or >>> s1.add(s2)</p> <p><u>Output</u></p> <p>A 25.0 B 30.0 C NaN D NaN</p>
Subtraction	- or <i>sub</i>	<p>>>> s1-s2 or >>> s1.sub(s2)</p> <p><u>Output</u></p> <p>A 5.0 B 10.0 C NaN D NaN</p>
Multiplication	* or <i>mul</i>	<p>>>> s1*s2 or >>> s1.mul(s2)</p> <p><u>Output</u></p> <p>A 150.0 B 200.0 C NaN D NaN</p>

Division	/ or <i>div</i>	>>> s1/s2 or >>> s1.div(s2) <u>Output</u> A 1.5 B 2.0 C NaN D NaN
Modulus	% or <i>mod</i>	>>> s1 % s2 or >>> s1.mod(s2) <u>Output</u> A 5.0 B 0.0 C NaN D NaN

MCQ

Section A	
1	The result of an operation between unaligned Series will have the ----- ---of the indexes involved a) intersection b) union c) total d) all of the above Ans: b
2	We can perform _____ on two series in Pandas a) Addition b) Subtraction c) Multiplication d) All of the above Ans: d
3	Which of the following method is used to add two series? a) sum() b) addition() c) add() d) None of the above Ans: c
4	Which of the following statement will display the difference of two Series 'A' and 'B'? a) A – B b) A.sub(B) c) Both a and b d) None of the above

	Ans: c
5	<p>Which of the following are valid operations on Series 'S1'?</p> <p>a) S1 + 2 b) S1 ** 2 c) S1 * 2 All of the above</p> <p>Ans: d</p>
6	<p>Which of the following function is used for basic mathematical operations in Series?</p> <p>a) add() b) mul() c) div() d) All of the above</p> <p>Ans: d</p>
Section B	
1	<p>Consider the following two series objects S1 , S2</p> <p>Series - S1 0 10 1 18</p> <p>Series - S2 a 5 b 6</p> <p>What will be the output of S1+S2</p> <p>a) 0 NaN 1 NaN a NaN b NaN b) 0 10 1 18 a 5 b 6 c) 0 15 1 24 d) a 15 b 24</p> <p>Ans: a</p>
2	<p>Choose the correct option:</p> <p>Assertion (A): We can add two series objects using addition operator (+) or calling explicit function add() .</p>

	<p>Reason (R): While adding two series objects index matching is implemented and missing values are filled with NaN by default.</p> <p>a) Both A and R are true and R is the correct explanation of A. b) Both A and R are true and R is not the correct explanation of A. c) A is true but R is false. d) A is false but R is true.</p> <p>Ans: a</p>
3	<p>Assume there is a series S1 having data elements as 11, 12, and 13 respectively. Programmer 'Ravi' wrote <code>print(s1*2)</code> in his python program.</p> <p>Statement 1: A series will data elements as 22, 24, 26 will get printed.</p> <p>Statement 2: Series supports vectorized operation.</p> <p>a) Only Statement 1 is true. b) Only Statement 2 is true. c) Both Statement 1 and 2 are true, Statement 2 is not correct reasoning of Statement 1. d) Both Statement 1 and 2 are true, Statement 2 is correct reasoning of Statement 1.</p> <p>Ans: d</p>
4	<p>Identify the correct option</p> <p>Assertion (A): We can perform mathematical operations on two series objects of different size but not on two 1 D arrays of different size.</p> <p>Reason (R) : if two series are not aligned NaN are generated but in case of arrays no concept of NaN and hence operations fail to perform.</p> <p>a) Both A and R are true and R is the correct explanation of A. b) Both A and R are true and R is not the correct explanation of A. c) A is true but R is false. d) A is false but R is true.</p> <p>Ans: a</p>
5	<p>Assuming the given series, named Salary, which command will be used to increase 2000 in every employee's salary?</p> <p>Om 35000 Vinay 35000</p>

	<p>Simi 50000 Nitin 54000 Nandi 60000 dtype: int64</p> <p>a) Salary*2000 b) Salary.add(2000) c) Salary+2000 d) Salary.count()</p> <p>Ans: c</p>
6	<p>Write the output of the given program:</p> <pre>import pandas as pd S1=pd.Series([3,6,9,12],index=['a','b','c','e']) S2=pd.Series([2,4,6,8],index=['c','d','b','f']) print(S1*S2)</pre> <p>a) a 6.0 b 24.0 c 54.0 d 96.0 e NaN f NaN dtype: float64</p> <p>b) a NaN b 36.0 c 18.0 d NaN e NaN f NaN dtype: float64</p> <p>c) a 6.0 b 36.0 c 18.0 d 24.0 e NaN f NaN dtype: float64</p> <p>d) Error</p> <p>Ans: b</p>
7	<p>Predict the output of the following code:</p> <pre>import pandas as pd stationary=['pencils','notebooks','scales','erasers']</pre>

```
S1=pd.Series([20,33,52,10],index=stationary)
S2=pd.Series([17,13,31,32],index=stationary)
S1=S1+S2
print(S1+S2)
```

a) pencils 37
notebooks 46
scales 83
erasers 42
dtype: int64

b) pencils 54
notebooks 59
scales 114
erasers 74
dtype: int64

c) pencils 20
notebooks 33
scales 52
erasers 10
dtype: int64

d) Error

Ans: b

8 Write the output of the following:
import pandas as pd
S1 = pd.Series(data = (31, 2, -6))
print(S1*2)

a) 0 31
1 2
2 -6
3 31
4 2
dtype: int64

b) 0 31
1 2
2 -6
dtype: int64

c) 0 62
1 4
2 -12
dtype: int64

	<p>d) Error</p> <p>Ans: c</p>
9	<p>Write the output of the following :</p> <pre>import pandas as pd S1=pd.Series([1,2,3,4]) S2=pd.Series([7,8,9,10]) S2.index=['a','b','c','d'] print((S1+S2).count())</pre> <p>a) 8 b) 4 c) 0 d) 6</p> <p>Ans: c</p>
10	<p>What will be the output of the following code?</p> <pre>import pandas as pd s1=pd.Series([4,5,7,8,9],index=['a','b','c','d','e']) s2=pd.Series([1,3,6,4,2],index=['a','p','c','d','e']) print(s1-s2)</pre> <p>a) a 3.0 b 0 c 1.0 d 4.0 e 7.0 p 0 dtype: float64</p> <p>b) a 3.0 b NaN c 1.0 d 4.0 e 7.0 p NaN dtype: float64</p> <p>c) a 3.0 c 1.0 d 4.0 e 7.0 dtype: float64</p> <p>d) a 3.0 b - c 1.0 d 4.0 e 7.0</p>

	<p>p – dtype: float64</p> <p>Ans: b</p>
Section C	
1	<p>Answer the following questions(i to iv) based on the series given below:</p> <pre>import _____ as pd #statement1 nstud1 = [10,2,6,4,5] event1 = ['swimming', 'skating','kho kho', 'chess', 'football'] nstud2 = [3,6,5] event2 = ['swimming', 'chess', 'football'] school1=pd.Series(nstud1, index= event1) school2=pd.Series(nstud2, index= event2) print (_____) #statement 2 print(school1+school2) #statement3 print(school1. _____ (school2)) #statement4</pre>
i)	<p>Name the library to be imported in the program in statement1</p> <p>a) numpy b) pandas c) matplotlib d) math</p> <p>Ans: b</p>
ii)	<p>Complete code in statement2 to obtain the following output:</p> <pre>swimming 6 chess 12 football 10</pre> <p>a) school2 * 2 b) school1 * 2 c) school1+2 d) school1+school2</p> <p>Ans: a</p>
iii)	<p>Predict the output of statement 3</p> <pre>a) swimming 10 skating 2 kho kho 6 chess 4</pre>

	<pre> football 5 swimming 3 chess 6 football 5 b) chess True football True kho kho False skating False swimming True c) chess 10.0 football 10.0 kho kho NaN skating NaN swimming 13.0 d) Error </pre> <p>Ans: c</p>
iv)	<p>Which method is to be used in statement4 to produce the following output?</p> <pre> chess 24.0 football 25.0 kho kho NaN skating NaN swimming 30.0 a) add b) sub c) div d) mul </pre> <p>Ans: d</p>

TOPIC-Attributes of Pandas Series

EXAMPLES ARE BASED ON THE GIVEN SERIES.

```

>>> seriesCapCntry
India   NewDelhi
USA     WashingtonDC
UK      London
France  Paris
dtype: object

```

Attribute Name	Purpose	Syntax	Example
name	assigns a name to the Series	<Seriesname>.name =<"name">	seriesCapCntry.name = 'Capitals' >>> print(seriesCapCntry) India NewDelhi USA WashingtonDC UK London France Paris Name: Capitals, dtype: object
index.name	assigns a name to the index of the series	<Seriesname>.index.name=<"name">	>>>seriesCapCntry.index.name = 'Countries' >>> print(seriesCapCntry) Countries India NewDelhi USA WashingtonDC UK London France Paris Name: Capitals, dtype: object
values	prints a list of the values in the series	<Seriesname>.values	>>> print(seriesCapCntry.values) ['NewDelhi' 'WashingtonDC' 'London', 'Paris']
size	prints the number of values in the Series object	<Seriesname>.size	>>> print(seriesCapCntry.size) 4
empty	prints True if the series is empty, and False otherwise	<Seriesname>.empty	>>> seriesCapCntry.empty False # Create an empty series seriesEmpt=pd.Series() >>> seriesEmpt.empty True
ndim	prints the dimension of the Series object	<Seriesname>.ndim	d1={'a':9, 'b':1, 'c':7, 'd':2} s1=pd.Series(d1) print(s1.ndim) o/p: 1
shape	shape property returns a tuple (n,) containing a single element	<Seriesname>.shape	d1={'a':9, 'b':1, 'c':7, 'd':2} s1=pd.Series(d1) print(s1.shape) o/p: (4,)

	which is the number of elements in the Series object.		
--	---	--	--

MCQ TYPE QUESTIONS

SECTION A	
1	<p>Which of the following is not an attribute of pandas Series?</p> <p>a.name b.size c.shape d.Series.T</p> <p>Ans.d</p>
2	<p>.....attribute will display the total number of elements in a given Series.</p> <p>a.shape b.size c.values d.ndim</p> <p>Ans c</p>
3	<p>Which of the following attribute is used to assigns a name to the index of the Series.</p> <p>a.name b.index c.index.name d.All of the above</p> <p>Ans c</p>
4	<p>.....property returns a tuple (n,) containing a single element which is the number of elements in the Series object.</p> <p>a.size b.shape c.values d.dim</p> <p>Ans shape</p>
5	<p>Choose the correct syntax to get the dimension of series named SR:</p> <p>a.SR.dimension b.SR.ndim c.SR.dim d.SR.ndimension</p> <p>Ans b</p>
SECTION B	

1	<p>Assuming the given series, named stud, which command will be used to print 5 as output?</p> <pre>Amit 90 Ramesh 100 Mahesh 50 John 67 Abdul 89 Name: Student, dtype: int64</pre> <p>a. stud.index b. stud.length c. stud.values d. stud.size</p> <p>Ans d</p>
2	<p>What will be the output of the following code given:</p> <pre>import pandas as pd seriesEmpt=pd.Series() >>> seriesEmpt.empty</pre> <p>a.Error b.0 c.True d.False</p> <p>Ans c</p>
3	<p>Assuming the given series, named 'capital', which command will be used to print the following output?</p> <pre>['NewDelhi' 'WashingtonDC' 'London', 'Paris']</pre> <pre>India NewDelhi USA WashingtonDC UK London France Paris</pre> <p>a.capital.index b.capital.elements c.capital.values d.capital.size</p> <p>Ans c</p>
4	<p>Choose the correct name of Series from the given python code.</p> <pre>import pandas as pd dict1 = {'India': 'NewDelhi', 'UK':'London', 'Japan': 'Tokyo'} series8 = pd.Series(dict1) print(series8) #Display the series series8.name='capital'</pre> <p>a.dict1</p>

	<p>b.series8 c.capital d.name</p> <p>Ans.c</p>
5	<p>Write the correct python statement to assign name to the index of the given series to 'State'.</p> <pre>import pandas as pd dict1 = {'India': 'NewDelhi', 'UK':'London', 'Japan': 'Tokyo'} series8 = pd.Series(dict1) print(series8) series8. _____ ='state'</pre> <p>a.series8.index b.series8.index.name c.series8.name.index d All of the above.</p> <p>Ans.b</p>
ASSERTION AND REASONING TYPE	
6	<p>Choose correct option :</p> <pre>import pandas as p1 import numpy as np a1=np.arange(2,11,2) s1=p1.Series(a1,index=list('ABCDE')) print(s1.ndim)</pre> <p>Statement 1: Above code will give output as 1.</p> <p>Statement 2: Series is a one dimensional data structure.</p> <p>a) Only Statement 1 is True b) Only Statement 2 is True c) Both Statement 1 and 2 are true, but Statement 2 is not correct reasoning of Statement 1. d) Both Statement 1 and 2 are true, but Statement 2 is correct reasoning of Statement 1.</p> <p>Ans:d</p>

SECTION C	
1	Nidhi has created Series S1 as following , help her to perform following tasks and write the code to help her to S1 India NewDelhi USA WashingtonDC UK London France Paris dtype: object
a	Display the number of values in the series s1 i.print(S1.shape) ii.print(S1.size) iii.print(S1.values) iv.print(S1.number)
b.	Returns True/Flase if the Series S1 is empty i.print(S1.empty()) ii.print(S1.empty) iii.print(S1.null) iv.print(S1.nan)
c	Displays the list of values in the series S1 i.print(S1.values) ii.print(S1.value) iii.print(S1.values()) iv.All of the above
d	Display the ouput as (1,) i.print(S1.ndim) ii.print(S1.shape) iii.print(S1.size) iv.print(S1.ndim())
e	The command which will change the name of Series S1 to States. i.S1.name='state' ii.name.S1='state' iii.S1.name(state) iv.AI of the above.

TOPIC:Methods of Series

Head and Tail functions

LET US CONSIDER THE FOLLOWING EXAMPLE.

```
>>> seriesTenTwenty=pd.Series(np.arange( 10, 20, 1 ))
```

```
>>> print(seriesTenTwenty)
```

```
0 10
1 11
2 12
3 13
4 14
5 15
6 16
```

7 17
 8 18
 9 19
 dtype: int32

Method	Explanation	Example
head(n)	Returns the first n members of the series. If the value for n is not passed, then by default n takes 5 and the first five members are displayed.	<pre>>>> seriesTenTwenty.head(2) 0 10 1 11 dtype: int32 >>> seriesTenTwenty.head() 0 10 1 11 2 12 3 13 4 14 dtype: int32</pre>
count()	Returns the number of non-NaN values in the Series	<pre>>>> seriesTenTwenty.count() 10</pre>
tail(n)	Returns the last n members of the series. If the value for n is not passed, then by default n takes 5 and the last five members are displayed.	<pre>>>> seriesTenTwenty.tail(2) 8 18 9 19 dtype: int32 >>> seriesTenTwenty.tail() 5 15 6 16 7 17 8 18 9 19 dtype: int32</pre>

MCQ TYPE QUESTIONS

SECTION A

1	<p>Which of the following statement shows first five values of Series 'S1'?</p> <p>a. S1.head() b. S1.head(5) c. Both of the above d. None of the above</p> <p>Ans c</p>
2	<p>Which of the following returns number of non-NaN values of Series?</p> <p>a. count b. size c. index d. values</p>

	Ans a
3	Which of following statement will return 10 values from the end of the Series 'S1'? a. S1.tail() b. S1.tail(10) c. S1.head(10) d. S1(10) Ans b
4	Function to display the first n rows in the Series: a. tail (n) b. head (n) c. top (n) d. first (n) Ans b
5	To get bottom three rows of a Series, you may use _____ function: 1 a. tail() b. bottom(3) c. bottom(3) d. tail(3) Ans d
SECTION B	
1	Write the output of the following: import pandas as pd S1=pd.Series([1,2,3,4]) S2=pd.Series([7,8]) print((S1+S2).count()) a. 6 b. 4 c. 2 d. 0 Ans b
2	Which of the following returns number of non-NaN values of Series? a. count b. size c. index d. values Ans a
3	Write the output of the following: import pandas as pd S1=pd.Series([1,2,3,4]) S2=pd.Series([7,8]) S3=S1+S2 print(S3.head(3)) a 0 8.0 1 10.0

	<pre> 2 NaN b. 0 1.0 1 2.0 2 NaN c. 0 7.0 1 8.0 2 NaN d 0 1.0 1 7.0 2 NaN </pre> <p>Ans a</p>
4	<p>Write the output of the following:</p> <pre> import pandas as pd S1=pd.Series([1,2,3,4]) S2=pd.Series([7,8]) print((S1+S2).tail(2)) </pre> <pre> a 2 NaN 3 NaN b 0 8.0 1 10.0 c 2 3 3 4 d 0 7 1 8 </pre> <p>Ans a</p>

Indexing/Slicing a Series object-

The index [] operator can be used to perform indexing and slicing operations on a Series object. The index[] operator can accept either-

- a) Index/labels
- b) Integer index positions

a) Using the index operator with labels-

The index operator can be used in the following ways-

- i) **Using a single label inside the square brackets-** Using a single label/index inside the square brackets will return only the corresponding element referred to by that label/index.

```
# indexing a Series object single label
import pandas as pd

d={'a':101, 'b':102, 'c':103, 'd':104, 'e':105, 'f':106}
s=pd.Series(d)
t=s['b']
print(t)

o/p:102
```

ii) **Using multiple labels-** We can pass multiple labels in any order that is present in the Series object. The multiple labels must be passed as a list i.e. the multiple labels must be separated by commas and enclosed in double square brackets. Passing a label is passed that is not present in the Series object, should be avoided as it right now gives NaN as the value but in future will be considered as an error by Python.

```
# indexing a Series object
multiple labels
import pandas as pd

d={'a':101, 'b':102, 'c':103, 'd':104, 'e':105, 'f':106}
s=pd.Series(d)
u=s[['b', 'a', 'f']]
print(u)

o/p:
b      102
a      101
f      106
dtype: int64
```

iii) **Using slice notation startlabel:endlabel-** Inside the index operator we can pass startlabel:endlabel. Here contrary to the slice concept all the items from startlabel values till the endlable values including the endlable values is returned back.

```
# indexing a Series object using
startlabel:endlabel
import pandas as pd

d={'a':101, 'b':102, 'c':103, 'd':104, 'e':105, 'f':106}
s=pd.Series(d)
u=s['b':'e']
print(u)
```

Output

```
b 102
c 103
d 104
e 105
dtype: int64
```

b) Slicing a Series object using Integer Index positions-

The concept of slicing a Series object is similar to that of slicing python lists, strings etc. Even though the data type of the labels can be anything each element of the Series object is associated with two integer numbers:

- In forward indexing method the elements are numbered from 0,1,2,3, ... with 0 being assigned to the first element, 1 being assigned to the second element and so on.
- In backward indexing method the elements are numbered from -1,-2, -3, ... with -1 being assigned to the last element, -2 being assigned to the second last element and so on.

For example consider the following Series object-

```
d={'a':101, 'b':102, 'c':103, 'd':104, 'e':105, 'f':106}
s=pd.Series(d)
```

The Series object is having the following integer index positions-

forward indexing--->	0	1	2	3	4	5	
	a	b	c	d	e	f	
	101	111	121	131	141	151	
	-6	-5	-4	-3	-2	-1	<---- backward indexing

Slice concept-

The basic concept of slicing using integer index positions are common to Python object such as strings, list, tuples, Series, Dataframe etc. Slice creates a new object using elements of an existing object. It is created as: *ExistingObjectName*[start : stop : step] where start, stop, step are integers

```
# Slicing a Series object import pandas as pd

d={'a':101, 'b':111, 'c':121, 'd':131, 'e':141, 'f':151}
s=pd.Series(d)
x=s[1: :2]
print('x=\n', x)
y=s[-1: :-1]
```



```

print('y=\n', y)
z=s[1: -2: 2]
print('z=\n', z)
o/p:
x=
b 111 d 131 f 151
dtype: int64
y=
f 151 e 141 d 131 c 121 b 111 a 101
dtype: int64
z=
b 111 d 131

```

Modifying elements of Series object-

The elements of a Series object can be modified using any of the following methods-

a. Using index [] operator to modify single/multiple values

```

# Modifying a Series object index [] method import pandas as pd
d={'a':101, 'b':111, 'c':121, 'd':131, 'e':141, 'f':151}
s=pd.Series(d)
s['c'] = 555
s[['f','a']] = [666,777]
print('s=\n', s)
s['b':'d']=[0,1,2]
print('s=\n', s)

```

Output

```

s=
a 777
b 111
c 555
d 131
e 141
f 666
dtype: int64

s=
a 777
b 0
c 1
d 2
e 141
f 666
dtype: int64

```

b. sing at/iat property to modify a single value

```

# Modifying a Series object at iat property import pandas as pd
d={'a':101, 'b':111, 'c':121, 'd':131, 'e':141, 'f':151}

```

```
s=pd.Series(d)
s.at['d'] = 999
s.iat[-1] = 777
print('s=\n', s)
```

Output

```
s=
a  101
b  111
c  121
d  999
e  141
f  777
dtype: int64
```

c. Using loc, iloc property to modify single /multiple values

```
#Modifying a Series object loc iloc property import pandas as pd
d={'a':101, 'b':111, 'c':121, 'd':131, 'e':141, 'f':151}
s=pd.Series(d)
s.loc['b'] = 9
s.loc['e':'f'] = [8,7]
print('s=\n', s)
s.iloc[1: :2] = [33,44,55]
print('s=\n', s)
```

Output

```
s=
a  101
b    9
c  121
d  131
e    8
f    7
dtype: int64
s=
a  101
b   33
c  121
d   44
e    8
f   55
dtype: int64
```

c. Using slice method to modify multiple values

```
# Modifying a Series object slice method
import pandas as pd
d={'a':101, 'b':111, 'c':121, 'd':131, 'e':141, 'f':151}
s=pd.Series(d)
s[1: :2] = [1,2,3]
print('s=\n', s)
```

Output

```
s=
a    101
b     1
c    121
d     2
e    141
f     3
dtype: int64
```

Changing indexes of Series object-

The index property can be used to change the indexes of a Series object

```
import pandas as pd
# Changing indexes of Series object
import pandas as pd
d={'a':101, 'b':111, 'c':121, 'd':131}
s=pd.Series(d)
s.index = ['have','a','nice', 'day']
print('s=\n', s)
```

Output

```
s=
have    101
a        111
nice    121
day     131
dtype: int64
```

MCQ	
1	<p>What will be the output of the given code?</p> <pre>import pandas as pd s = pd.Series([1,2,3,4,5], index=['akram','brijesh','charu','deepika','era']) print(s['charu'])</pre> <p>a 1 b 2</p>

	<p>c 3 d 4 Ans C</p>
2	<p>Consider the following series named animal:</p> <pre>L Lion B Bear E Elephant T Tiger W Wolf dtype: object</pre> <p>Write the output of the command:</p> <pre>print(animal[::-3])</pre> <p>a L Lion T Tiger dtype: object</p> <p>b. B Bear E Elephant dtype: object</p> <p>c. W Wolf B Bear dtype: object</p> <p>d. W Wolf T Tiger dtype: object</p> <p>Ans C</p>
3	<p>Write the output for the following Python code.</p> <pre>import pandas as pd s=pd.Series([1,2,3,4,5,6],index=['A','B','C','D','E','F']) print(s[s%2==0])</pre> <p>a. B 2 D 4 F 6</p> <p>b. A 1 C 3 E 5</p> <p>c. B 2 D 4 F 5</p> <p>d. B 3 D 4 F 6</p> <p>Ans a</p>

4	<p>Write the output of the following code ?</p> <pre>import pandas as pd seriesMnths=pd.Series([2,3,4],index=['Feb','Mar','Apr']) print(seriesMnths[1])</pre> <p>a. 2 b. Mar c. Feb d. 3</p> <p>Ans d</p>
5	<p>Choose the correct output of the following code?</p> <pre>import pandas as pd seriesCapCntry=pd.Series(['New Delhi','WashingtonDC','London','Paris'],index= ['India','USA','UK','France']) print(seriesCapCntry[[3,2]])</pre> <p>a. France Paris France Paris b. USA WashingtonDC France Paris c. France Paris UK London d. USA WashingtonDC UK London</p> <p>Ans c</p>
6	<p>Assertion (A) : We cannot access more than one element of Series without slicing . Reason (R) :More than one element of series can be accessed using a list of positional index or labeled index.</p> <p>(A) Both A and R are true and R is the correct explanation of A. (B) Both A and R are true and R is not the correct explanation 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.</p> <p>Ans D</p>
7	<p>Assertion (A) : Elements of Series can be accessed using positional index. Reason (R) : positional index values ranges from 1 to n if n is the size of the series.</p> <p>(A) Both A and R are true and R is the correct explanation of A. (B) Both A and R are true and R is not the correct explanation 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</p> <p>Ans A</p>
8	<p>Answer the following based on the series given below.</p>

	<pre>import pandas as pd list1=[1,2,3,4,5,6,7,8] list2=['swimming','tt','skating','kho kho','bb','chess','football','cricket'] school=pd.Series(list1,index=list2) school.name=("little") print(school*2) # statement 1 print(school.tail(3)) # statement 2 print(school['tt']) # statement 3 print(school[2:4])</pre>
i	<p>Choose the correct name of the Series</p> <p>a) list1 b) list2 c) school d) little</p> <p>Ans: c</p>
ii	<p>Choose the correct output of the statement</p> <p>print(school.tail(3)) # statement 2</p> <p>a. swimming 1 tt 2 skating 3 b. chess 6 football 7 cricket 8 c. 4 d. kho kho 4 bb 5 chess 6 football 7 cricket 8</p> <p>Ans b</p>
iii	<p>Choose the correct output of the statement</p> <p>print(school['tt']) # statement 3</p> <p>a. 2 b. 3 c. tt 2 d. true</p> <p>Ans c</p>
9	<p>Write the output of the following:</p> <pre>import pandas as pd S1 = pd.Series(['NewDelhi', 'WashingtonDC', 'London', 'Paris'], index=['India', 'USA', 'UK', 'France']) print(S1['India', 'UK'])</pre> <p>a. India NewDelhi UK London dtype: object</p>

	<p>b. India NewDelhi UK Washington dtype: object</p> <p>c. Error</p> <p>d. None of the above</p> <p>Ans a</p>
10	<p>What will be the output of the above given code?</p> <pre>import pandas as pd s=pd.Series([1,2,3,4,5],index=["ajay", "pankaj","deepti","rajesh","ritika"]) print(s["rajesh"])</pre> <p>a) 1 b) 2 c) 3 d) 4</p> <p>Ans 4</p>

UNIT I- DATA FRAMES

❖ DataFrame Data Structure

- It is **two dimensional (tabular) heterogeneous data labeled array**.
- It has two indices or two axes : **a row index (axis=0)** and a **column index (axis=1)**
- The **row index is known as index** and the **column index is called the column name**.
- The indices can be of any data type.
- It is both **value mutable** and **size mutable**.
- We can perform arithmetic operations on rows and columns.

❖ Creating and Displaying a DataFrame

To create a DataFrame object, we can use the syntax:

```
<dataframe object> = pandas.DataFrame( <a 2D datastructure> , [columns=<column sequence>] , [index=<index sequence>] )
```

where the 2D data structure passed to it, contains the data values.

➤ Empty DataFrame

```
import pandas as pd          Empty DataFrame
df=pd.DataFrame()          Columns: []
print(df)                  Index: []
```

➤ DataFrame from 2D dictionary

A 2D dictionary is a dictionary having items as (key : value) where value part is a data structure of any type : a list, a series, a dictionary etc. But the value parts of all the keys should have similar structure and equal lengths.

✓ Creating a DataFrame from 2D dictionary having values as lists:

```
dict1={'Students':['Neha','Maya','Reena'],
'Marks':[20,40,30],
'Sports':['Cricket','Football','Badminton']}
df1=pd.DataFrame(dict1)
print(df1)
```

	Students	Marks	Sports
0	Neha	20	Cricket
1	Maya	40	Football
2	Reena	30	Badminton

- The keys of the dictionary has become columns.
- The columns are placed in sorted order.

- The index is assigned automatically (0 onwards).

We can specify our own index too by using the index argument.

```
df2=pd.DataFrame(dict1,index=['I','II','III'])
print(df2)
```

	Students	Marks	Sports
I	Neha	20	Cricket
II	Maya	40	Football
III	Reena	30	Badminton

- The number of indexes given in the index sequence must match the length of the dictionary's values, otherwise Python will give error.

✓ **Creating a DataFrame from 2D dictionary having values as Series objects.**

- DataFrames are two dimensional representation of series.

```
smarks=pd.Series({'Neha':80,'Maya':90,'Reena':70})
sage=pd.Series({'Neha':25,'Maya':30,'Reena':29})
dict={'Marks':smarks,'Age':sage}
df3=pd.DataFrame(dict)
print(df3)
```

	Marks	Age
Neha	80	25
Maya	90	30
Reena	70	29

or

```
smarks=pd.Series([80,90,70],index=['Neha','Maya','Reena'])
sage=pd.Series([25,30,29],index=['Neha','Maya','Reena'])
dict={'Marks':smarks,'Age':sage}
df3=pd.DataFrame(dict)
print(df3)
```

	Marks	Age
Neha	80	25
Maya	90	30
Reena	70	29

- DataFrame object created has **columns assigned** from the **keys of the dictionary object** and its **index assigned** from the **indexes of the Series object** which are the values of the dictionary object.

➤ **Creating a DataFrame from list of dictionaries**

```
student=[{'Neha':50,'Manu':40},{'Neha':60,'Maya':45}]
df4=pd.DataFrame(student,index=['term1','term2'])
print(df4)
```

	Neha	Manu	Maya
term1	50	40.0	NaN
term2	60	NaN	45.0

- NaN is automatically added in missing places.

❖ Selecting or Accessing Data

```
import pandas as pd
dict={'BS':[80,98,100,65,72],'ACC':[88,67,93,50,90],
      'ECO':[100,75,89,40,96],'IP':[100,98,92,80,86]}
df5=pd.DataFrame(dict,index=['Ammu','Achu','Manu','Anu','Abu'])
print(df5)
```

	BS	ACC	ECO	IP
Ammu	80	88	100	100
Achu	98	67	75	98
Manu	100	93	89	92
Anu	65	50	40	80
Abu	72	90	96	86

➤ Selecting / Accessing a column

Syntax :

<dataframe object>[<column name>] Or <dataframe object>.<column name>

- In the dot notation make sure not to put any quotation marks around the column name.

```
print(df5.BS)
or
print(df5['BS'])
```

Ammu	80
Achu	98
Manu	100
Anu	65
Abu	72

Name: BS, dtype: int64

➤ Selecting / Accessing multiple columns

Syntax :

<dataframe object>[[<column name>,<column name>,.....]]

- Columns appear in the order of column names given in the list inside square brackets.

```
print(df5[['BS','IP']])
```

	BS	IP
Ammu	80	100
Achu	98	98
Manu	100	92
Anu	65	80
Abu	72	86

➤ Selecting / Accessing a subset from a DataFrame using Row/Column names

<dataframe object>.loc[<start row>:<end row>,<start column>:<end column>]

➤ To access a row:

<dataframe object>.loc[<row label>, :]

- Make sure not to miss the colon after comma.

```
print(df5.loc['Ammu', :])
```

BS	80
ACC	88
ECO	100
IP	100

Name: Ammu, dtype: int64

➤ **To access multiple rows:**

<dataframe object>.loc[<start row>:<end row> , :]

- Python will return all rows falling between start row and end row; along with start row and end row.

```
print(df5.loc['Ammu':'Manu', : ])

```

	BS	ACC	ECO	IP
Ammu	80	88	100	100
Achu	98	67	75	98
Manu	100	93	89	92

- Make sure not to miss the colon after comma.

➤ **To access selective columns:**

<dataframe object>.loc[: , <start column> : <end column>]

- Lists all columns falling between start and end column.

```
print(df5.loc[:, 'ACC':'IP'])

```

	ACC	ECO	IP
Ammu	88	100	100
Achu	67	75	98
Manu	93	89	92
Anu	50	40	80
Abu	90	96	86

- Make sure not to miss the colon before comma.

➤ **To access range of columns from a range of rows:**

**<dataframe object>.loc[<start row> : <end row> ,
<start column> : <end column>]**

```
print(df5.loc['Manu':'Abu', 'ACC':'ECO'])

```

	ACC	ECO
Manu	93	89
Anu	50	40
Abu	90	96

➤ **Selecting / Accessing a subset from a DataFrame using Row/Column numeric index/position**

Sometimes our dataframe object does not contain row or column labels or even we may not remember, then to extract subset from dataframe we can use iloc.

<dataframe object>.iloc[<start row index> : <end row index> ,

[<start column index> : <end column index>]

- When we use iloc, then end index is excluded.

```
print(df5.iloc[1:3,1:3])
```

	ACC	ECO
Achu	67	75
Manu	93	89

➤ **Selecting / Accessing individual value**

- (i) **Either give name of row or numeric index in square bracket of column name**

<dataframe object>.<column>[<row name or row numeric index>]

```
print(df5.ACC['Achu'])
```

67

or

```
print(df5.ACC[1])
```

- (ii) **Using at or iat**

<dataframe object>.at[<row label>,<column label>]

Or

**<dataframeobject>.iat[<numeric row index>,
<numeric column index>]**

```
print(df5.at['Achu','ACC'])
```

67

or

```
print(df5.iat[1,1])
```

❖ **Assigning / Modifying Data Values in DataFrame**

➤ **To change or add a column**

<dataframe object>[<column name>]=<new value>

- If the given column name does not exist in dataframe then a new column with the name is added.

```
df5['ENG']=60
```

```
print(df5)
```

	BS	ACC	ECO	IP	ENG
Ammu	80	88	100	100	60
Achu	98	67	75	98	60
Manu	100	93	89	92	60
Anu	65	50	40	80	60
Abu	72	90	96	86	60

- If you want to add a column that has different values for all its rows, then we can assign the data values for each row of the column in the form of a list.

```
df5['ENG']=[50,60,40,30,70]
```

- There are some other ways for adding a column to a database.

<dataframe object>.at[: , <column name>]=value

Or

<dataframe object>.loc[: ,<column name>]=value

```
df5.at[ : , 'ENG']=60
```

```
print(df5)
```

or

```
df5.loc[ : , 'ENG']=60
```

```
print(df5)
```

➤ **To change or add a row**

<dataframe object>.at[*rowname* , :]=value

or

<dataframe object>.loc[*rowname* , :]=value

```
df5.at['Sabu', : ]=50
```

```
print(df5)
```

or

```
df5.loc['Sabu', : ]=50
```

```
print(df5)
```

	BS	ACC	ECO	IP	ENG
Ammu	80.0	88.0	100.0	100.0	60.0
Achu	98.0	67.0	75.0	98.0	60.0
Manu	100.0	93.0	89.0	92.0	60.0
Anu	65.0	50.0	40.0	80.0	60.0
Abu	72.0	90.0	96.0	86.0	60.0
Sabu	50.0	50.0	50.0	50.0	50.0

- If there is no row with such row label, then adds new row with this row label and assigns given values to all its columns.

➤ **To change or modify a single data value**

<dataframe object>.<column>[<row label or row index>] = value

```
df5.BS['Ammu']=100
```

```
print(df5)
```

or

```
df5.BS[0]=100
```

```
print(df5)
```

	BS	ACC	ECO	IP	ENG
Ammu	100.0	88.0	100.0	100.0	60.0
Achu	98.0	67.0	75.0	98.0	60.0
Manu	100.0	93.0	89.0	92.0	60.0
Anu	65.0	50.0	40.0	80.0	60.0
Abu	72.0	90.0	96.0	86.0	60.0
Sabu	60.0	60.0	60.0	60.0	60.0

❖ **Deleting columns in DataFrame**

- We can use **del** statement, to delete a column

del <dataframeobject>[<column name>]

e.g.: `del df5['ENG']`

- We can use **drop()** also to delete a column. **By default axis=0.**

`<dataframe object> = <dataframeobject>.drop([<columnname or index>],axis=1)`

Or

`<dataframe object> = <dataframeobject>.drop(columns=[<columnnames or indices>])`

```
df5=df5.drop(['ECO'], axis =1)
```

```
df5=df5.drop(columns=['ECO','IP'])
```

- We can use **pop()** to delete a column. The deleted column will be returned as Series object.

```
bstud=df5.pop('BS')
```

```
print(bstud)
```

❖ Deleting rows in DataFrame

`<dataframe object>=<dataframe object>.drop([index or sequence of index], axis=0)`

```
df5=df5.drop(['Ammu','Achu'])
```

or

```
df5=df5.drop(index=['Ammu','Achu'])
```

❖ Iterating over a DataFrame

➤ Using pandas.iterrows() Function

- The method <DF>.iterrows() views a dataframe in the form of **horizontal** subset ie **row-wise**.
- Each horizontal subset is in the form of (**row-index, Series**) where **Series contains all column values for that row –index**.
- We can iterate over a Series object just as we iterate over other sequences.

```
import pandas as pd
dict={'BS':[80,98],'ACC':[88,67]}
df5=pd.DataFrame(dict,index=['Ammu','Achu'])
print(df5,"\n")
```

```
      BS  ACC
Ammu  80   88
Achu  98   67
```

```
Row index: Ammu
containing
At position 0 : 80
At position 1 : 88
```

```
for (row,rowseries) in df5.iterrows():
    print("Row index:",row)
    print("containing")
    i=0
    for val in rowseries:
        print("At position ",i,":",val)
        i=i+1
    print()
```

```
Row index: Achu
containing
At position 0 : 98
At position 1 : 67
```

➤ Using pandas.iteritems() Function

- The method <DF>.iteritem() views a dataframe in the form of **vertical** subset ie **column-wise**.
- Each vertical subset is in the form of (**col-index, Series**) where **Series contains all row values for that column index**.

```
import pandas as pd
dict={'BS':[80,98],'ACC':[88,67]}
df5=pd.DataFrame(dict,index=['Ammu','Achu'])
```

```
print(df5,"\n")
```

```
for (column,columnseries) in df5.iteritems():
```

```
    print("Column index:",column)
```

```
    print("containing")
```

```
    i=0
```

```
    for val in columnseries:
```

```
        print("At row ",i,":",val)
```

```
        i=i+1
```

```
    print()
```

```
        BS  ACC
Ammu  80  88
Achu  98  67
```

```
Column index: BS
containing
At row  0 : 80
At row  1 : 98
```

```
Column index: ACC
containing
At row  0 : 88
At row  1 : 67
```

❖ Head and Tail Functions

➤ head()

<DF>.head([n=5])

- To retrieve 5, top rows of a dataframe.
- We can change the number of rows by specifying value for n.

```
df5.head(5)
```

```
df5.head(2)
```

➤ tail()

- To retrieve 5, bottom rows of a dataframe.
- We can change the number of rows by specifying value for n.

```
df5.tail(5)
```

```
df5.tail(2)
```


❖ Renaming index / column labels

- **rename()** renames the existing index or column labels in a dataframe/series.
- The old and new index/column labels are to be provided **in the form of a dictionary where keys are the old indexes/row labels and the values are the new names for the same.**

Syntax:

<DF>.rename(index=None, columns=None, inplace=False)

where index and columns are dictionary like.

inplace, a boolean by default False (which returns a new dataframe with renamed index/labels).

If True then changes are made in the current

	p_id	p_name
0	101	Hard disk
1	102	Pen Drive

	Product_ID	product_name
0	101	Hard disk
1	102	Pen Drive

dataframe.

```
import pandas as pd
dict={'p_id':[101,102],'p_name':['Hard disk','Pen Drive']}
df=pd.DataFrame(dict)
print(df,"\n")
#df.rename(columns={'p_id':'Product_ID','p_name':'product_name'},inplace=True)
#or
df=df.rename(columns={'p_id':'Product_ID','p_name':'product_name'})
print(df)
```

- Columns can also be renamed by using the **columns attribute** of dataframe.

```
import pandas as pd
dict={'p_id':[101,102],'p_name':['Hard disk','Pen Drive']}
df=pd.DataFrame(dict)
df.columns=["Product_ID","product_name"]
print(df,"\n")
```

	Product_ID	product_name
0	101	Hard disk
1	102	Pen Drive

❖ Reindexing

- **reindex()** used to change the order of the rows or columns in DataFrame/Series and returns DataFrame/Series after changes.

Syntax:

<DF>.reindex(index=None, columns=None, fill_value=NaN)

```
df=df.reindex(columns=['product_name','Product_ID'])
print(df)
```

	product_name	Product_ID
0	Hard disk	101
1	Pen Drive	102

- **If the mentioned indexes/columns do not exist in dataframe, these will be added as per the mentioned order with NaN values.**

```
df=df.reindex(columns=['product_name','Product_ID','product_category'])
print(df)
```

	product_name	Product_ID	product_category
0	Hard disk	101	NaN
1	Pen Drive	102	NaN

- By using **fill_value**, we can specify which will be filled in the newly added row/column.

```
df=df.reindex(columns=['product_name','Product_ID','product_category'],
              index=[1,0],fill_value='Home')
print(df)
```

	product_name	Product_ID	product_category
1	Pen Drive	102	Home
0	Hard disk	101	Home

❖ Boolean indexing

- Like default indexing (0,1,2...) or labeled indexing, there is one more way to index – Boolean Indexing (Setting row index to True/ False etc.) .
- This helps in displaying the rows of Data Frame, according to True or False as specified in the command.

```
import pandas as pd
dict={'p_id':[101,102,103],'p_name':['Hard disk','Pen Drive','Camera']}
df=pd.DataFrame(dict)
df.index=[True,False,True]
print(df,"\n")
print(df.loc[True])
```

	p_id	p_name
True	101	Hard disk
False	102	Pen Drive
True	103	Camera

	p_id	p_name
True	101	Hard disk
True	103	Camera

❖ **DataFrame attributes**

All information related to a DataFrame object is available through attributes.

<DataFrane object> . <attribute name>

Attribute	Description
index	Returns the index (row labels) of the DataFrame
columns	Returns the column labels of the DataFrame
axes	Returns a list representing both the axes of the Data Frame (axis=0 i.e. index and axis=1 i.e. columns)
values	Returns a Numpy representation of the DataFrame
dtypes	Returns the dtypes of data in the DataFrame
shape	Returns tuple of the shape of the DataFrame
ndim	Returns number of dimensions of the dataframe
size	Returns the number of elements in the dataframe
empty	Returns True if the DataFrame object is empty, otherwise False
T	Transpose index and columns of DataFrame

Case study questions:

1. Consider the following Data Frame df and answer questions

	A	B	C
DEPT	CS	PROD	MEDICAL
EMPNO	101	102	103
ENAME	ABC	PQR	LMN
SALARY	200000	100000	20000

- i. Write code to delete column B
- ii. Write the output of the below code
`print(df.tail(2))`
- iii. Write code to delete row salary
- iv. Change the value of column A to 100
- v. Change the value of DEPT of B to MECH
- vi. Display DEPT and SALARY of column A and B

- vii. Write code to rename column 'A' to 'D' which will not effect original dataframe
- viii. Write code to add a column E with values [CS, 104,XYZ, 300000]
- ix. Write code to add a row COMM with values [3000,4000,5000]
- x. Write code to rename DEPT to DEPARTMENT which will effect the original dataframe
- xi. Write code to display DEPT in A
 - i. `print(df.A['DEPT'])`
 - ii. `print(df['A','DEPT'])`
 - iii. `print(df.iloc[1:2,1:2])`
 - iv. `print(df.iat[3,2])`
- xii. Write the output of the statement `print(len(df))`
 - i. 3
 - ii. 4
 - iii. (4,3)
 - iv. (3,4)

Answers :=

- i. `del df['A']`
- ii.

	A	B	C
ENAME	ABC	PQR	LMN
SALARY	200000	100000	20000
- iii. `df=df.drop(['SALARY'],axis=0)`
- iv. `df['A']=100`
- v. `df.B['DEPT']='MECH'`
- vi. `print(df.loc[['DEPT','SALARY'],['A','B']])`
- vii. `df.rename(columns={"A":"D"},inplace=False)`
- viii. `df['E']=["CS",104,"XYZ",300000]`
- ix. `df.loc['COMM']=[3000,4000,5000]`
- x. `df.rename(index={"DEPT":"DEPARTMENT"},inplace=True)`
- xi. `print(df.A['DEPT'])`
- xii. 4

2. Consider the following Data Frame df and answer questions

	ACC	BST	ECO	IP
S1	90	91	92	93
S2	94	95	96	97
S3	98	99	100	100
S4	91	92	93	94

- i. Create a new column total TOT by adding marks
- ii. Find the highest marks scored by student s1

- iii. Find the lowest marks scored by student s1
- iv. Find the highest marks in ACC
- v. Find the lowest marks in IP

Answers:=

- i. `df['TOT']=df['ACC']+df['BST']+df['ECO']+df['IP']`
- ii. `print(max(df.loc['S1',:]))`
- iii. `print(min(df.loc['S1',:]))`
- iv. `print(max(df['ACC']))`
- v. `print(min(df['IP']))`

3. Consider the following Data Frame df and answer questions

	delhi	mumbai	kolkatta	chennai
hospitals	200	300	100	50
population	10	20	30	40
schools	250	350	400	200

- i. Display details of city delhi and chennai
- ii. Display hospitals in delhi
- iii. Display shape of dataframe
- iv. Change the population in kolkatta as 50
- v. Rename the column population as “pop”

Answers:=

- i. `print(df[['delhi','chennai']])`
- ii. `print(df.delhi["hospitals"])`
- iii. `print(df.shape)`
- iv. `df.kolkatta['population']=50`
- v. `df.rename(index={"population":"pop"},inplace=True)`

4. Consider the following Data Frame df and answer questions

```

      population schools hospitals
chennai      40    200     500
delhi        10    250     200
kolkatta     30    400     100
mumbai       20    350     300
>>>|

```

- i. Display the name of city whose population ≥ 20 range of 12 to 20
- ii. Write command to set all values of df as 0
- iii. Display the df with rows in the reverse order
- iv. Display the df with only columns in the reverse order
- v. Display the df with rows & columns in the reverse order

answers:-

- i. `print(df[df.population \geq 20])`
- ii. `df[:]=0`
- iii. `print(df.iloc[::-1])`
- iv. `print(df.iloc[:,::-1])`
- v. `print(df.iloc[::-1,::-1])`

5. Consider the following Data Frame df and answer questions

```

      A      B      C
DEPT  CS  PROD MEDICAL
EMPNO  101  102  103
ENAME  ABC  PQR  LMN
SALARY 200000 100000 20000
>>>|

```

Write the output of the following

- i. `print(len(df))`
- ii. `print(df.count())`
- iii. `print(df.count(1))`
- iv. `print(min(df.loc['SALARY']))`
- v. `print(max(df.loc['ENAME']))`

Answers

- i. 4
- ii. A 4
B 4
C 4
dtype: int64
- iii. DEPT 3
EMPNO 3
ENAME 3
SALARY 3
dtype: int64
- iv. 20000
- v. PQR

SI No	MCQ QUESTIONS
1	<p>To display the 3rd, 4th and 5th columns from the 6th to 9th rows of a dataframe you can write</p> <p>(a) DF.loc[6:9, 3:5] (b) DF.loc[6:10, 3:6] (c) DF.iloc[6:10, 3:6] (d) DF.iloc[6:9, 3:5]</p> <p>ANS: c) DF.iloc[6:10, 3:6]</p>
2	<p>We can add a new row to a DataFrame using the _____ method</p> <p>(i) rloc[] (ii) loc[] (iii)iloc[] (iv)None of the above</p> <p>ANS: (ii) loc[]</p>

3	<p>The head() function of dataframe will display how many rows from top if no parameter is passed.</p> <p>(i) 1 (ii) 3 (iii) 5 (iv) None of these</p> <p>ANS : (iii) 5</p>
4	<p>To change the 5th column's value at 3rd row as 35 in dataframe DF, you can write</p> <p>(a) DF[4, 6] = 35 (b) DF.iat[4, 6] = 35 (c) DF[3, 5] = 35 (d) DF.iat[3, 5] = 35</p> <p>ANS:- d) DF.iat[3, 5] = 35</p>
5	<p>Which function is used to find values from a DataFrame D using the index number?</p> <p>a) D.loc b) D.iloc c) D.index d) None of these</p> <p>ANS: b) D.iloc</p>
6	<p>In a DataFrame, Axis= 0 represents the elements</p> <p>a.rows b.columns c.both d.None of these.</p> <p>ANS: a.rows</p>

7	<p>In DataFrame, by default new column added as the _____ column</p> <p>(i) First (Left Side) (ii) Second (iii) Last (Right Side) (iv) Any where in dataframe</p> <p>ANS: (iii) Last (Right Side)</p>
8	<p>Which of the following is correct Features of DataFrame?</p> <p>a. Potentially columns are of different types b. Can Perform Arithmetic operations on rows and columns c. Labeled axes (rows and columns) d. All of the above</p> <p>ANS: d. All of the above</p>
9	<p>Write the code to append df2 with df1</p> <p>a. <code>Df2=Df2.append(Df1)</code> b. <code>Df2=Df2+Df1</code> c. <code>Df2=Df2.appendwith.Df1</code> d. <code>Df2=Df1.append(Df1)</code></p> <p>ANS: a. <code>Df2=Df2.append(Df1)</code></p>
10	<p>When we create DataFrame from List of Dictionaries, then number of columns in DataFrame is equal to the _____</p> <p>a. maximum number of keys in first dictionary of the list b. maximum number of different keys in all dictionaries of the list c. maximum number of dictionaries in the list d. None of the above</p> <p>ANS: b. maximum number of different keys in all dictionaries of the list</p>
11	<p>When we create DataFrame from List of Dictionaries, then dictionary keys will become _____</p>

	<p>(i) Column labels (ii) Row labels (iii) Both of the above (iv) None of the above</p> <p>ANS: (i) Column labels</p>
12	<p>Which method is used to access vertical subset of a dataframe?</p> <p>(i) iterrows() (ii) iteritems() (iii) itercolumns() (iv) itercols()</p> <p>ANS: (ii) iteritems()</p>
13	<p>Write statement to transpose dataframe DF.</p> <p>(i) DF.t (ii) DF.transpose (iii)DF.T (iv)DF.T()</p> <p>ANS: (iii)DF.T</p>
14	<p>In DataFrame, by default new column added as the _____ column</p> <p>a. First (Left Side) b. Second c. Last (Right Side) d. Any where in dataframe</p> <p>ANS: Last (Right Side)</p>
15	<p>We can add a new row to a DataFrame using the _____ method</p> <p>(i) rloc[] (ii) loc[] (iii) iloc[] (iv) None of the above</p>

	ANS: (ii) loc[]
16	<p>Which of the following function is used to load the data from the CSV file to DataFrame?</p> <p>(i) read.csv() (ii) readcsv() (iii) read_csv() (iv) Read_csv()</p> <p>ANS: (iii) read_csv()</p>
17	<p><i>Which of the following function is not a Boolean reduction function</i></p> <p>(i) Empty (ii) Any() (iii) All() (iv) Fillna()</p> <p>ANS: (iv) Fillna()</p>
18	<p>Which among the following options can be used to create a DataFrame in Pandas ?</p> <p>(a) A scalar value (b) An ndarray (c) A python dict (d) All of these</p> <p>ANS:- (d) All of these</p>
19	<p>Which attribute of a dataframe is used to convert row into columns and columns into rows in a dataframe?</p> <p>a) T b) ndim c) empty d) shape</p>

	ANS: a) T
20	<p>When we create DataFrame from List of Dictionaries, then number of columns in DataFrame is equal to the _____</p> <p>(i) maximum number of keys in first dictionary of the list (ii) maximum number of different keys in all dictionaries of the list (iii) maximum number of dictionaries in the list (iv) None of the above</p> <p>ANS: (ii) maximum number of different keys in all dictionaries of the list</p>
21	<p>Which of the following is/are characteristics of DataFrame?</p> <p>a) Columns are of different types b) Can Perform Arithmetic operations c) Axes are labeled (rows and columns) d) All of the above</p> <p>ANS: d) All of the above</p>
22	<p>Write short code to show the information having city="Delhi" from dataframe SHOP.</p> <p>(a) <code>print(SHOP[City=='Delhi'])</code> (b) <code>print(SHOP[SHOP.City=='Delhi'])</code> (c) <code>print(SHOP[SHOP.'City']=='Delhi')</code> (d) <code>print(SHOP[SHOP[City]=='Delhi'])</code></p> <p>ANS: (b) <code>print(SHOP[SHOP.City=='Delhi'])</code></p>
23	<p>Which of the following commands is used to install pandas?</p> <p>(i) <code>pip install python -pandas</code> (ii) <code>pip install pandas</code> (iii) <code>python install python</code> (iv) <code>python install pandas</code></p> <p>ANS: (ii) <code>pip install pandas</code></p>
24	<p>Which attribute of a dataframe is used to get number of axis?</p>

	<p>a.T b.Ndim c.Empty d.Shape</p> <p>ANS: b.Ndim</p>
25	<p>Display first row of dataframe 'DF'</p> <p>(i) print(DF.head(1)) (ii) print(DF[0 : 1]) (iii)print(DF.iloc[0 : 1]) (iv)All of the above</p> <p>ANS: (iv)All of the above</p>
26	<p>To delete a column from a DataFrame, you may use statement.</p> <p>(a) remove (b) del (c) drop (d) cancel statement.</p> <p>ANS:- (b) del</p>
27	<p>In given code dataframe 'Df1' has _____ rows and _____ columns</p> <pre>import pandas as pd dict= [{'a':10, 'b':20}, {'a':5, 'b':10, 'c':20},{ 'a':7, 'd':10, 'e':20}] Df1 = pd.DataFrame(dict)</pre> <p>(i) 3, 3 (ii) 3, 4 (iii)3, 5 (iv)None of the above</p> <p>ANS: (iii)3, 5</p>

28	<p>To delete a row from a DataFrame, you may use</p> <ul style="list-style-type: none"> (a) remove (b) del (c) drop (d) cancel <p>ANS:- (c) drop</p>
29	<p>In the following statement, if column 'mark' already exists in the DataFrame 'Df1' then the assignment statement will _____ Df1['mark'] = [95,98,100] #There are only three rows in DataFrame Df1</p> <ul style="list-style-type: none"> (i) Return error (ii) Replace the already existing values. (iii) Add new column (iv) None of the above <p>ANS: (ii) Replace the already existing values.</p>
30	<p>To skip first 5 rows of CSV file, which argument will you give in read_csv() ?</p> <ul style="list-style-type: none"> (a) skip_rows = 5 (b) skiprows = 5 (c) skip - 5 (d) noread - 5 <p>ANS:- (a) skip_rows = 5</p>
31	<p>. Which of the following statement is false:</p> <ul style="list-style-type: none"> i. DataFrame is size mutable ii. DataFrame is value mutable iii. DataFrame is immutable iv. DataFrame is capable of holding multiple types of data <p>ANS:- iii. DataFrame is immutable</p>

32	<p>Which of the following statements is false?</p> <p>(i) Dataframe is size mutable (ii) Dataframe is value mutable (iii) Dataframe is immutable (iv) Dataframe is capable of holding multiple type of data</p> <p>ANS: (iii) Dataframe is immutable</p>
33	<p>To delete a row, the parameter axis of function drop() is assigned the value _____</p> <p>(i) 0 (ii) 1 (iii) 2 (iv) 3</p> <p>ANS: (i) 0</p>
34	<p>Which of the following function is used to load the data from the CSV file to DataFrame?</p> <p>(i) read.csv() (ii) readcsv() (iii)read_csv() (iv)Read_csv()</p> <p>ANS: (iii)read_csv()</p>
35	<p>Write code to delete rows those getting 5000 salary.</p> <p>(a) df=df.drop[salary==5000] (b) df=df[df.salary!=5000] (c) df.drop[df.salary==5000,axis=0] (d) df=df.drop[salary!=5000]</p> <p>ANS: (b) df=df[df.salary!=5000]</p>

36	<p>DF1.loc[] method is used to _____ # DF1 is a DataFrame</p> <p>(i) Add new row in a DataFrame 'DF1'</p> <p>(ii) To change the data values of a row to a particular value</p> <p>(iii)Both of the above</p> <p>(iv)None of the above</p> <p>ANS: (iii)Both of the above</p>
37	<p>To iterate over horizontal subsets of dataframe,</p> <p>(a) iterate()</p> <p>(b) iterrows() function may be used.</p> <p>(c) itercols()</p> <p>(d) iteritems()</p> <p>ANS:- (b) iterrows() function may be used.</p>
38	<p>Write code to delete the row whose index value is A1 from dataframe df.</p> <p>(a) df=df.drop('A1')</p> <p>(b) df=df.drop(index='A1')</p> <p>(c) df=df.drop('A1,axis=index')</p> <p>(d) df=df.del('A1')</p> <p>ANS: (a) df=df.drop('A1')</p>
39	<p>A two-dimension labeled array that is an ordered collection of columns to store heterogeneous data type is</p> <p>i. Series</p> <p>ii. Numpy array</p> <p>iii. Dataframe</p> <p>iv. Panel</p> <p>ANS:- iii. Dataframe</p>
40	<p>To skip 1st, 3rd and 5th rows of CSV file, which argument will you give in read_csv() ?</p>

	<p>(a) skiprows = 11315 (b) skiprows = (1, 3, 5] (c) skiprows = [1, 5, 1] (d) Any of these</p> <p>ANS:- (b) skiprows - (1, 3, 5]</p>
41	<p>In Pandas _____ is used to store data in multiple columns.</p> <p>(i) Series (ii) DataFrame (iii) Both of the above (iv) None of the above</p> <p>ANS: (ii) DataFrame</p>
42	<p>What is dataframe?</p> <p>a. 2 D array with heterogeneous data b. 1 D array with homogeneous data c. 2 D array with homogeneous data d. 1 D array with heterogeneous data</p> <p>ANS: a. 2 D array with heterogeneous data</p>
43	<p>In a DataFrame, Axis= 1 represents the _____ elements</p> <p>(a) Row (b) Column (c) True (d) False</p> <p>ANS: (b) Column</p>
44	<p>Which of the following is not an attribute of a DataFrame Object ?</p> <p>a. index</p>

	<p>b. Index c. size d. value</p> <p>ANS: b. Index</p>
45	<p>To get top 5 rows of a dataframe, you may use</p> <p>(a) head() (b) head(5) (c) top() (d) top(5)</p> <p>ANS:- (a) head() , b) head(5)</p>
46	<p>27. To iterate over horizontal subsets of dataframe,</p> <p>(a) iterate() (b) iterrows() function may be used. (c) itercols() (d) iteritems()</p> <p>ANS:- (b) iterrows() function may be used.</p>
47	<p>Write code to delete the row whose index value is A1 from dataframe df.</p> <p>(a) df=df.drop('A1') (b) df=df.drop(index='A1') (c) df=df.drop('A1,axis=index') (d) df=df.del('A1')</p> <p>ANS: (a) df=df.drop('A1')</p>
48	<p>A two-dimension labelled array that is an ordered collection of columns to store heterogeneous datatype is</p> <p>v. Series vi. ii. Numpy array vii. iii. Dataframe viii. iv. Panel</p>

	<p>ANS:- iii. Dataframe</p>
49	<p>To skip 1st, 3rd and 5th rows of CSV file, which argument will you give in read_csv() ?</p> <p>(a) skiprows = 11315 (b) skiprows - (1, 3, 5] (c) skiprows = [1, 5, 1] (d) Any of these</p> <p>ANS:- (b) skiprows - (1, 3, 5]</p>
50	<p>In a DataFrame, Axis= 1 represents the _____ elements</p> <p>(a) Row (b) Column (c) True (d) False</p> <p>ANS: (b) Column</p>
51	<p>NaN stands for:</p> <p>a. Not a Number b. None and None c. Null and Null d. None a Number</p> <p>ANS: a. Not a Number</p>
52	<p>To get top 5 rows of a dataframe, you may use</p> <p>(a) head() (b) head(5) (c) top() (d) top(5)</p>

	<p>ANS:- (a) head() , b) head(5)</p>
53	<p>The correct statement to read from a CSV file in a dataframe is :</p> <p>(a) .read_csv() (b) . read_csv()() (c) = pandas.read() (d) = pandas.read_csv()</p> <p>ANS:- (d) = pandas.read_csv()</p>
54	<p>To delete a column from a dataframe, you may use _____ statement.</p> <p>i. remove() ii. ii. del() iii. iii. drop() iv. iv. cancel()</p> <p>ANS:- iii. drop()</p>
55	<p>The following code create a dataframe named 'Df1' with _____ columns.</p> <pre>import pandas as pd Df1 = pd.DataFrame([10,20,30])</pre> <p>(i) 1 (ii) 2 (iii) 3 (iv) 4</p> <p>ANS: (i) 1</p>
56	<p>To delete a row from dataframe, you may use _____ statement.</p> <p>i. remove() ii. ii. del() iii. iii. drop() iv. iv. cancel()</p> <p>ANS:- ii. del()</p>

57	<p>In a Data-Frame, Axis= 0 represents the elements along the_____</p> <p>a. Row b. Column c. Row and Column Both d. None of the above</p> <p>ANS: a. Row</p>
58	<p>_____ method in Pandas can be used to change the index of rows and columns of a Series or Dataframe</p> <p>(a) rename() (b) reindex() (c) reframe() (d) none of these</p> <p>ANS: (b) reindex()</p>
59	<p>Write the single line command to delete the column “marks” from dataframe df using drop function.</p> <p>(a) df=df.drop(col='marks') (b) df=df.drop('marks',axis=col) (c) df=df.drop('marks',axis=0) (d) df=df.drop('marks',axis=1)</p> <p>ANS: (d) df=df.drop('marks',axis=1)</p>
60	<p>Which of the following is used to give user defined column index in DataFrame?</p> <p>(i) index (ii) column (iii) columns (iv) colindex</p> <p>ANS: (iii) columns</p>

61	<p>The following statement will _____</p> <pre>df = df.drop(['Name', 'Class', 'Rollno'], axis = 1) #df is a DataFrame object</pre> <p>a. delete three columns having labels 'Name', 'Class' and 'Rollno'</p> <p>b. delete three rows having labels 'Name', 'Class' and 'Rollno'</p> <p>c. delete any three columns</p> <p>d. return error</p> <p>ANS:- a. delete three columns having labels 'Name', 'Class' and 'Rollno'</p>
62	<p>Difference between loc() and iloc().:</p> <p>a. Both are Label indexed based functions.</p> <p>b. Both are Integer position-based functions.</p> <p>c. loc() is label based function and iloc() integer position based function.</p> <p>d. loc() is integer position based function and iloc() index position based function.</p> <p>ANS: c. loc() is label based function and iloc() integer position based function.</p>
63	<p>Which command will be used to delete 3 and 5 rows of the data frame. Assuming the data frame name as DF.</p> <p>a. DF.drop([2,4],axis=0)</p> <p>b. DF.drop([2,4],axis=1)</p> <p>c. DF.drop([3,5],axis=1)</p> <p>d. DF.drop([3,5])</p> <p>ANS: a DF.drop([2,4],axis=0)</p>
64	<p>Assuming the given structure, which command will give us the given output:</p> <p>Output Required: (3,4)</p>

	EmpCode	Name	Desig
0	1405	VINAY	Clerk
1	1985	MANISH	Works Manager
2	1636	SMINA	Sales Manager
3	1689	RINU	Cleark

- a. print(df.shape())
- b. print(df.shape)
- c. print(df.size)
- d. print(df.size()).

ANS: b. print(df.shape)

Write the output of the given command: df1.loc[:0,'Name'] Consider the given dataframe.

	EmpCode	Name	Desig
0	1405	VINAY	Clerk
1	1985	MANISH	Works Manager
2	1636	SMINA	Sales Manager
3	1689	RINU	Clerk

65

- a. 0 1405 VINAY Clerk
- b. **VINAY**
- c. Works Manager
- d. Clerk

ANS : VINAY

UNIT I- Data Visualization

What is Data Visualization ?

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

Purpose of Data visualization:

- Better analysis
- Quick action
- Identifying patterns
- Finding errors
- Understanding the story
- Exploring business insights
- Grasping the Latest Trends

matplotlib Library and pyplot Interface

- The matplotlib is a python library that provides many interfaces functionally for 2D graphics
- In short we can call matplotlib 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

To import Pyplot following syntax is

```
import matplotlib.pyplot  
or  
import matplotlib.pyplot as plt
```

After importing matplotlib in the form of **plt** we can use **plt** for accessing any function of matplotlib

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

Types of plot using matplotlib

- **LINE PLOT**
- **BAR GRAPH**
- **HISTOGRAM etc.**

Line Plot:

A line plot/chart is a graph that shows the frequency of data occurring along a number line. The line plot is represented by a series of data points called **markers** connected with a straight line. Generally line plots are used to display trends over time. A line plot or line graph can be created using the plot() function available in pyplot library.

We can, not only just plot a line but we can explicitly define the grid, the x and y axis scale and labels, title and display options etc.

Line chart: displaying data in form of lines.

- We can create line graph with x coordinate only or with x and y coordinates.
- Function to draw line chart – plot()

- Default colour of line- blue
- Syntax: plt.plot(x,y)

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()

plt.show()

PROGRAM

```
import matplotlib.pyplot as plt
```

```
X=[1,2,3,4,5]
```

```
Y=[2,4,6,8,10]
```

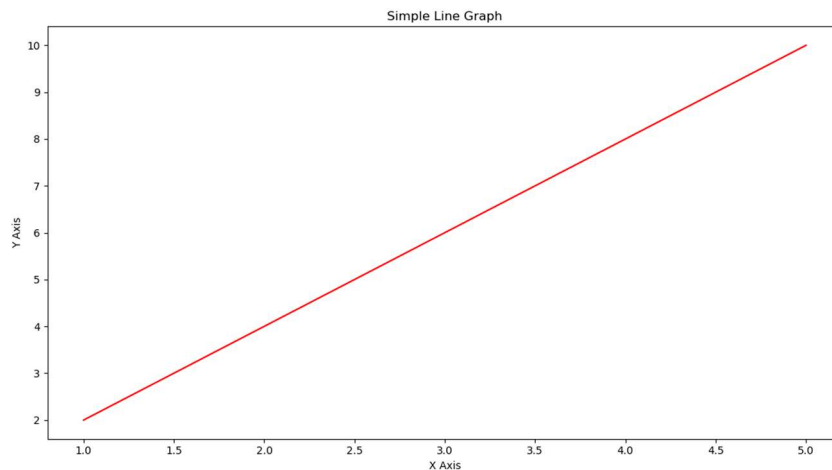
```
plt.title('Simple Line Graph')
```

```
plt.xlabel('X Axis')
```

```
plt.ylabel('Y Axis')
```

```
plt.plot(X,Y,'r')
```

```
plt.show()
```



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 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**

```
plt.bar(x,y, color="color code/color name")
```

To see different colors for different bars

```
plt.bar(x,y, color="color code/color name sequence")
```

- **Custom bar width**

```
plt.bar(x,y, width=float value)
```

To se 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

PROGRAM :

```
import matplotlib.pyplot as plt
```

```
overs=['1-10','11-20','21-30','31-40','41-50']
```

```
runs=[65,55,70,60,90]
```

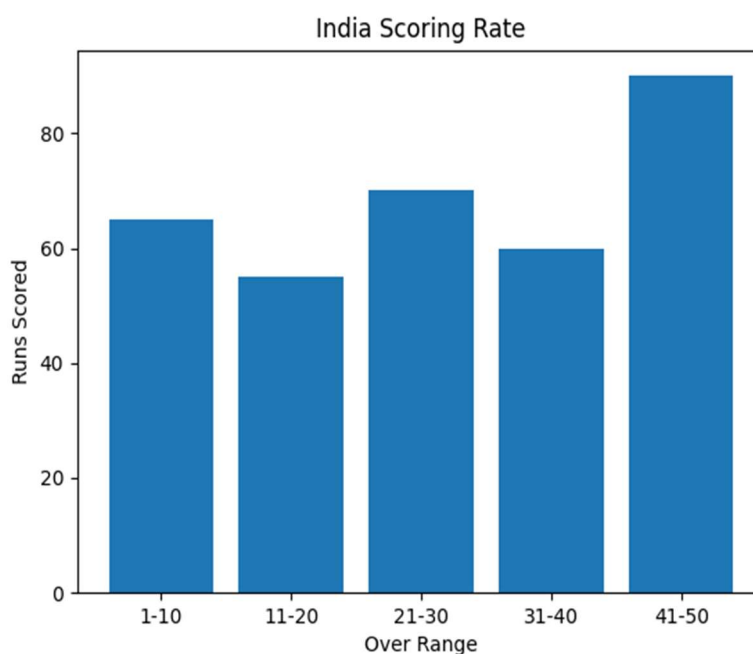
```
plt.xlabel('Over Range')
```

```
plt.ylabel('Runs Scored')
```

```
plt.title('India Scoring Rate')
```

```
plt.bar(overs,runs)
```

```
plt.show( )
```



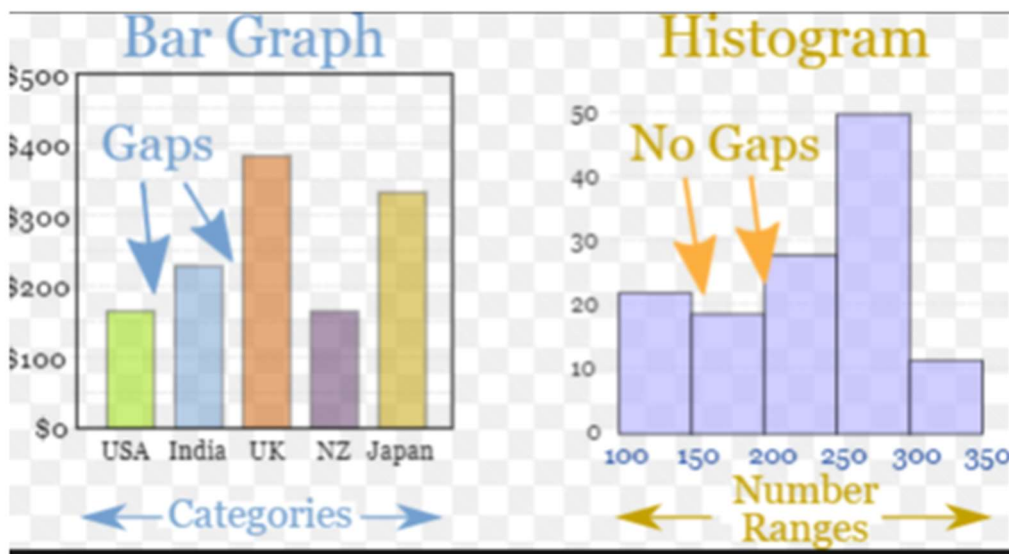
HISTOGRAM

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:**

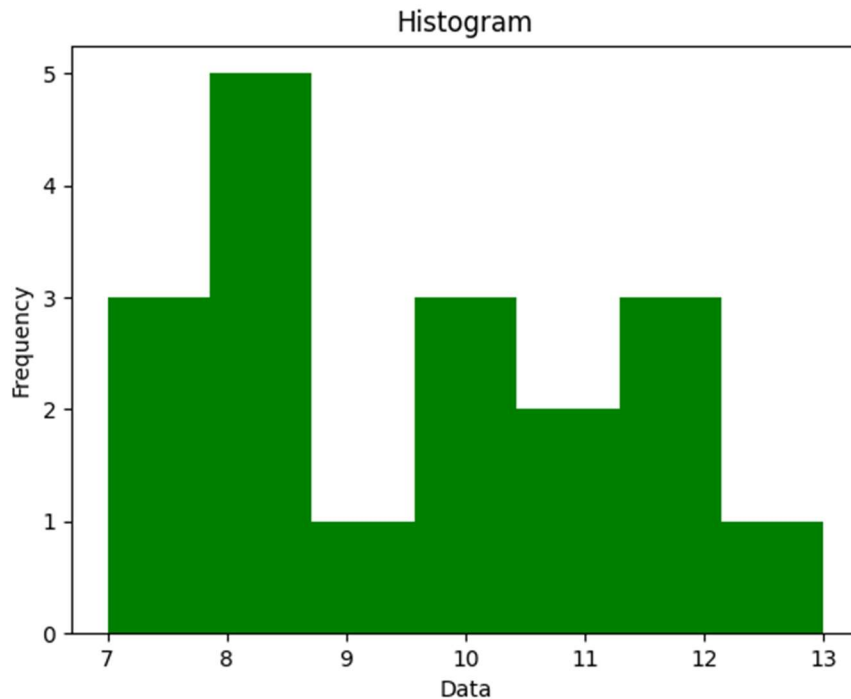
plt.hist(x,other parameters)

Optional Parameters

x	array or sequence of array
bins	optional parameter contains integer or sequence or strings
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 :

```
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**

`plt.title('Histogram')` – Change it as per requirement

- **Label-**

`plt.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.

i) In the plotting function like `bar()` or `plot()`, give a specific label to the data range using label

ii) Add legend to the plot using `legend()` as per the syntax given below.

Syntax : - `plt.legend((loc=position number or string))`

position number **can be** u1,2,3,4 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
```

Multiple Choice Questions and answers

SECTION B

1. What is data visualization?

- a) It is the numerical representation of information and data
- b) It is the graphical representation of information and data
- c) It is the character representation of information and data
- d) None of the above

Ans : b) It is the graphical representation of information and data

2. Which is a python package used for 2D graphics?

- a) matplotlib.pyplot
- b) matplotlib.pip
- c) matplotlib.numpy
- d) mathplotlib.pyplot

Ans: a) matplotlib.pyplot

3. The command used to give a heading to a graph is _____

- (a) plt.show()
- (b) plt.plot()
- (c) plt.xlabel()

(d) plt.title()

Ans: (d) plt.title()

4. Using Python Matplotlib _____ can be used to count how many values fall into each interval.

(a) line plot

(b) bar graph

(c) histogram

(d) None of these

Ans : (c) histogram

5.Fill the missing statement

```
import matplotlib.pyplot as plt
```

```
marks=[30,10,55,70,50,25,75,49,28,81]
```

```
plt.____(marks, bins='auto', color='green')
```

```
plt.show()
```

(a) plot

(b) bar

(c)hist

(d)draw

Ans : (c)hist

6.Which module of matplotlib library is required for plotting of graph ?

(a) Plot

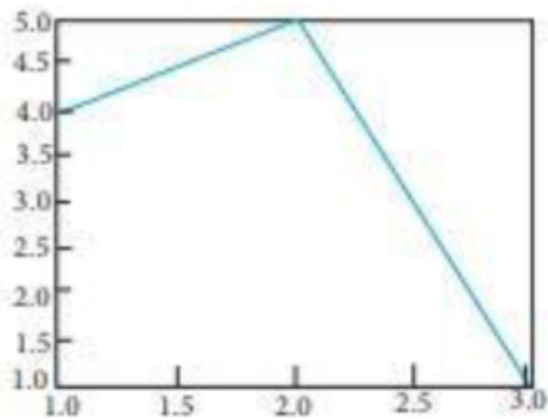
(b) Matplot

(c) pyplot

(d) graphics

Ans : (c) pyplot

7.Observe the output figure. Identify the code for obtaining this output.



- a) `import matplotlib.pyplot as plt`
`plt.plot([1,2],[4,5])`
`plt.show()`
- b) `import matplotlib.pyplot as plt`
`plt.plot([2,3],[5,1])`
`plt.show()`
- c) `import matplotlib.pyplot as plt`
`plt.plot([1,2,3],[4,5,1])`
`plt.show()`
- d) `import matplotlib.pyplot as plt`
`plt.plot([1,3],[4,1])`
`plt.show()`

Ans: c) import matplotlib.pyplot as plt

`plt.plot([1,2,3],[4,5,1])`

`plt.show()`

8. Identify the right type of chart using the following hints.

Hint 1: This chart is often used to visualize a trend in data over intervals of time.

Hint 2: The line in this type of chart is often drawn chronologically.

- a) Line chart
- b) Bar chart
- c) Pie chart
- d) Scatter plot

Ans : a) Line chart

9. Which of the following is/are correct statement for plot method?

- a) `plt.plot(x,y,color,others)`
- b) `pl.plot(x,y)`

- c) `plt.plot(x,y,color)`
- d) All the above

Ans: d) All the above

10. To give a title to x-axis, which of the following method is used?

- a) `plt.xtitle("title")`
- b) `plt.xlabel("title")`
- c) `plt.xheader("title")`
- d) `plt.xlabel.show("title")`

Ans: b) `plt.xlabel("title")`

11. To change the width of bars in bar chart, which of the following argument with a float value is used?

- a) `thick`
- b) `thickness`
- c) `width`
- d) `barwidth`

Ans: c) `width`

12. What is the purpose of legend?

- a) A legend is an area describing the elements of the graph.
- b) A legend is top area with information about graph
- c) A legend is additional information of x and y labels
- d) A legend is a mini box with bars data

Ans: a) A legend is an area describing the elements of the graph.

13. Which function can be used to export generated graph in matplotlib to png

- a) `savefigure ()`
- b) `savefig()`
- c) `save()`
- d) `export ()`

Ans: b) `savefig()`

14. which one of these is not a valid line style in matplotlib

- a) '-'
- b) '--'
- c) '-.'
- d) '<'

Ans: d) '<'

15. How can we make bar chart horizontal?

- a) plt.bar()
- b) plt.hbar()
- c) plt.barh()
- d) plt.rightbar()

Ans: c) plt.barh()

16. A histogram is used:

- a) for continuous data
- b) for grouped data
- c) for time series data
- d) to compare two sets of data

Ans: a) for continuous data

17. Which function is used to show legend ?

- a) display ()
- b) show()
- c) legend()
- d) legends()

Ans: c) legend()

18. The datapoints plotted on a graph are called _____

- a) Markers
- b) Values
- c) Ticks
- d) Pointers

Ans : a) Markers

19. To specify the style of line as dashed, which argument of plot() needs to be set ?

- a) line
- b) width
- c) Style
- d) linestyle

Ans: d) linestyle

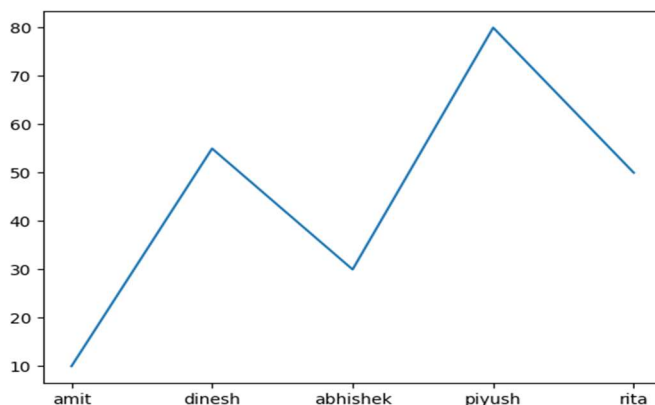
20. Which of the following is not a valid plotting function in pyplot?

- a) bar()
- b) hist()
- c) **hith()**
- d) barh()

Ans: c) hith()

SECTION B

1. Observe the following figure. Identify the coding for obtaining this as output.



- a)

```
import matplotlib.pyplot as plt
eng_marks=[10,55,30,80,50]
st_name=["amit","dinesh","abhishek","piyush","rita"]
plt.plot(st_name,eng_marks)
plt.show()
```

b) `import matplotlib.pyplot as plt`
`eng_marks=[10,55,30,80,50]`
`st_name=["amit","dinesh","abhishek","piyush","rita"]`
`plt.plot(st_name,eng_marks)`

c) `import matplotlib.pyplot as plt`
`eng_marks=[10,55,30,80,50]`
`st_name=["amit","dinesh","abhishek","piyush","rita"]`
`plt.plot(eng_marks, st_name)`
`plt.show()`

d) `import matplotlib.pyplot as plt`
`eng_marks=[10,55,30,80,50]`
`st_name=["amit","dinesh","abhishek","piyush","rita"]`
`plt.plot(eng_marks, st_name)`
`plt.show()`

Ans : **`import matplotlib.pyplot as plt`**
`eng_marks=[10,55,30,80,50]`
`st_name=["amit","dinesh","abhishek","piyush","rita"]`
`plt.plot(st_name,eng_marks)`
`plt.show()`

2. Read the statements given below and identify the right option to draw a histogram.

Statement A: To make a Histogram with Matplotlib, we can use the `plt.hist()` function.

Statement B: The bin parameter is compulsory to create histogram.

- a) Statement A is correct
- b) Statement B is correct
- c) Statement A is correct, but Statement B is incorrect
- d) d. Statement A is incorrect, but Statement B is correct

Ans: Statement A is correct, but Statement B is incorrect

3. Which graph should be used where each column represents a range of values, and the height of a column corresponds to how many values are in that range?

- a) plot
- b) line

- c) bar
- d) histogram**

Ans: d). **histogram**

4. Statement A : Data visualization refers to the graphical representation of information and data using visual elements like charts, graphs and maps etc.

Statement B : To install matplotlib library we can use the command ***pip install matplotlib***.

- a. Both statements are correct.
- b. Both statements are incorrect.
- c. Statement A is correct, but Statement B is incorrect
- d. Statement A is incorrect, but Statement B is correct

Ans : a. Both statements are correc

5. Fill the missing statement

```
import matplotlib.pyplot as plt
```

```
marks=[30,10,55,70,50,25,75,49,28,81]
```

```
plt.____(marks, bins='auto', color='green')
```

```
plt.show()
```

- (a) plot
- (b) bar
- (c) hist
- (d) barh

Ans : (c) hist

ASSERTION BASED QUESTIONS:

In each of the questions given below, there are two statements marked as Assertion (A) and Reason (R). Mark your answer as per the codes provided below:

- (A) A is true but R is false.
- (B) Both A and R are true
- (C) A is false but R is true.
- (D) Both A and R are false.

1. ASSERTION(A) :A histogram is basically used to represent data provided in the form of groups spread in non-continuous ranges

REASON(R) : matplotlib.pyplot.hist() function is used to compute and create histogram of a variable.

Ans: C

2.ASSERTION(A) : legend (labels = ['Text']) is used to give title to the graph

REASON(R) : plt.savefig("path") will save the current graph in png or jpeg format

Ans: C

3.ASSERTION(A) : plt.plot(x,y,'g',label="Students participating in CCA competition") will plot a

Line chart

REASON(R) : 'g' in plot() function is colour of the marker

Ans: A

4.ASSERTION(A) : linestyle, linewidth are used to customize line graph

REASON(R) : In the following example markers, line style and colour are mentioned exclusively

```
emp_count = [3, 20, 50, 200, 350, 400]
year = [2014, 2015, 2016, 2017, 2018, 2019]
plt.plot(year, emp_count, 'o', '-.', 'g')
```

Ans: B

5. ASSERTION(A) : In histogram X-axis is about bin ranges where Y-axis talks about frequency

REASON(R) : The bins (intervals) must be adjacent, and are often (but are not required to be) of equal size.

Ans: B

6. ASSERTION(A) : matplotlib.pyplot.show() is a method used to plot a line graph

REASON(R) : show() is method is defined in the library matplotlib.pip

Ans: D

7. ASSERTION(A) : pyplot is a sub-library of matplotlib

REASON(R) : line() is not a valid plotting function of pyplot

Ans: B

8. ASSERTION(A) : legend of the graph reflects the data displayed on the graph's Y-axis

REASON(R) : Location of the legend can be changed by using loc attribute

Ans: B

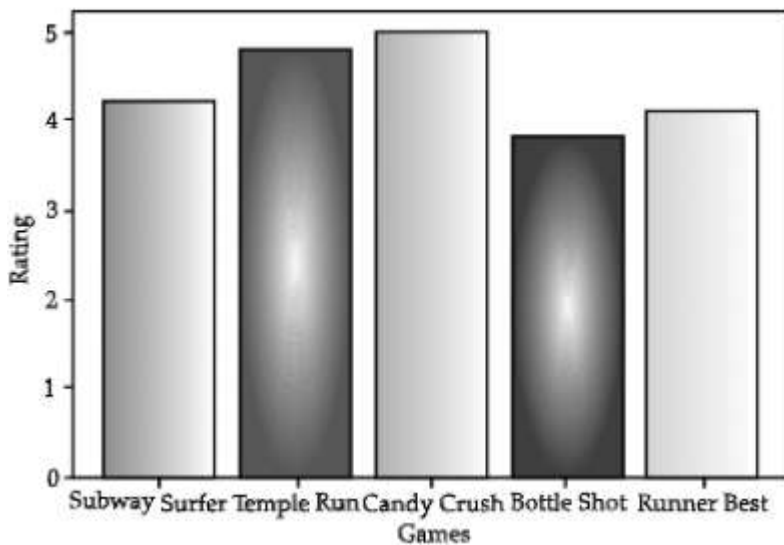
9. ASSERTION(A): Bar graph and histogram are same

REASON(R): A bar graph represents categorical data using rectangular bars. A histogram represents data which is grouped into continuous number ranges and each range correspond to a vertical bar.

Ans: C

Case Study based questions:

1. Mr. Sharma is working in a game development industry and he was comparing the given chart on the basis of the rating of the various games available on the play store. He is trying to write a code to plot the graph. Help Mr. Sharma to fill in the blanks of the code and get the desired output.



```
import _____ #Statement 1
```

```
Games=["Subway Surfer","Temple Run","Candy Crush","Bottle hot","Runner Best"]
```

Rating=[4.2,4.8,5.0,3.8,4.1]

plt. _____ (Games,Rating) #Statement 2

plt.xlabel("Games")

plt. _____ ("Rating") #Statement 3

plt. _____ #Statement 4

(i) Choose the right code from the following for statement 1.

- (a) matplotlib as plt
- (b) pyplot as plt
- (c) matplotlib.pyplot as plt
- (d) matplotlib.pyplot as pyplot

Ans: (c) matplotlib.pyplot as plt

(ii) Identify the name of the function that should be used in statement 2 to plot the above graph.

- (a) line()
- (b) bar()
- (c) hist()
- (d) barh()

Ans: (b) bar()

(iii) Choose the correct option for the statement 3.

- (a) title("Rating")
- (b) ytitle("Rating")
- (c) ylabel("Rating")
- (d) yaxis("Rating")

Ans: (c) ylabel("Rating")

(iv) Choose the right function/method from the following for the statement 4.

(a) display()

(b) print()

(c) bar()

(d) show()

Ans: (d) show()

(v) In case Mr. Sharma wants to change the above plot to any other shape, which statement, should he change.

(a) Statement 1

(b) Statement 2

(c) Statement 3

(d) Statement 4

Ans: (b) Statement 2

2. ABC Enterprises is selling its products through three salesmen and keeping the records of sales done quarterly of each salesman as shown below:

	Quarter 1	Quarter 2	Quarter 3	Quarter 4
Salesman 1	23000	18000	30000	35000
Salesman 2	11000	15000	20000	22000
Salesman 3	60000	40000	35000	55000

Company is storing the above information in a CSV file “Qtrly_Sales.csv”. Mr. Rohit is a programmer. Company has given him the responsibility to create the

program to visualise the above data. He wrote Python code but he is facing some difficulties. Help him by giving the solutions of following situation: Python code:

```
1 import pandas as pd
2 import _____ as plt
3 df=_____("Qtrly_Sales.csv")
4 df.plot(_____='bar', color=['red','blue','brown','green'])
5 plt._____('Quarterly Report')
6 plt.xlabel('Salesman')
7 plt.ylabel('Sales')
8 plt._____()
```

1. Choose the correct Python library out of following options in line 2

- (a). matplotlib
- (b). matplotlib.plot
- (c) . py.plot
- (d). matplotlib.pyplot

Ans. (d). matplotlib.pyplot

2. Choose the correct option to read the csv file in line 3

- (a). read_csv
- (b). pd.read_csv
- (c). pd.get_csv
- (d). get_csv

Ans B

3. Choose the correct option to select the type of graph in line 4

- (a). type
- (b). kind
- (c). style
- (d). graph

Ans : (b). kind

4. Choose the correct word to give the heading in line 5

- (a). label
- (b). heading
- (c). title
- (d). caption

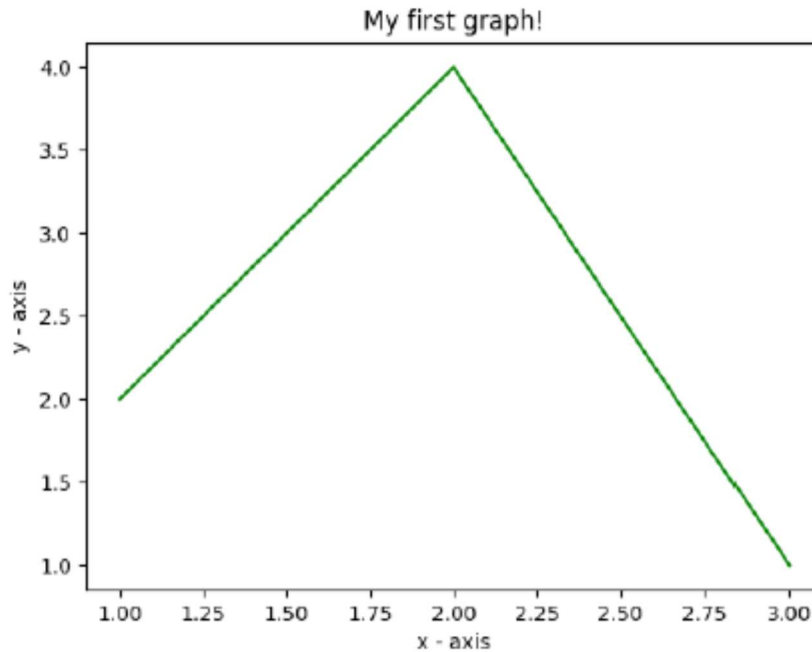
Ans : (c). title

5. Choose the correct word to display the graph in line 8

- (a). plot()
- (b). display()
- (c) . showgraph()
- (d). show()

Ans : (d). show()

3. Mr.Sharma is trying to write a code to plot line graph shown in fig-1. Help Mr. Sharma to fill in the blanks of the code and get the desired output.



```
import matplotlib.pyplot as plt # statement 1
x = [1,2,3] # statement 2
y = [2,4,1] # statement 3
plt.plot(x, y, color='g') #statement 4
_____ # statement 5
_____ # statement 6

# giving a title to my graph
plt._____('My first graph!') # statement 7
# function to show the plot
_____ # statement 8
```

- i) Which of the above statement is responsible for plotting the values on canvas.
- Statement 8
 - Statement 4
 - Statement 1
 - None of the above

Ans: **b) Statement 4**

ii) Statements 5 & 6 are used to give names to x-axis and y-axis as shown in fig.1. Which of the following can fill those two gaps

- plt.xlabel('x - axis') plt.ylabel('y - axis')**
- plt.xtitle('x - axis') plt.ytitle('y - axis')
- plt.xlabel('x - axis') plt.ylable('x - axis')

d) `plt.xlabel('x axis')` `plt.ylabel('y axis')`

Ans : **d) `plt.xlabel('x axis')` `plt.ylabel('y axis')`**

iii) Raman has executed code with first 7 statements. But No output displayed. which of the following statements will display the graph?

- a) `plt.display()`
- b) `plt.show()`
- c) `matplotlib.pyplot.show()`
- d) Both b & c]

Ans : **d) Both b & c**

iv) The number of markers in the above line chart are

- a) zero
- b) three
- c) Infinite
- d) One

Ans: b) three

v) Which of the following methods will result in displaying 'My first graph!' in the above graph

- a) `legend()`
- b) `label()`
- c) `title()`
- d) Both a & c

Ans : **c) `title()`**

UNIT 4: SOCIETAL IMPACTS

- Digital footprint, net and communication etiquettes,
 - Data protection, intellectual property rights (IPR), plagiarism, licensing and copyright,
 - Free and open source software (FOSS),
 - Cybercrime and cyber laws, hacking, phishing, cyber bullying, overview of Indian IT Act.
 - E-waste: hazards and management. Awareness about health concerns related to the usage of technology.
-

DIGITAL FOOTPRINT

A digital footprint – refers to the trail of data you leave while using the internet. It includes websites you visit, emails you send, and information you submit online. A digital footprint can be used to track a person's online activities and devices.

Internet users create their digital footprint either actively or passively. A passive footprint is made when information is collected from the user without the person knowing this is happening. An active digital footprint is where the user has deliberately shared information about themselves either by using social media sites or by using websites

Digital footprint examples

Online shopping

- Making purchases from e-commerce websites

Online banking

- Using a mobile banking app

Social media

- Using social media on your computer or devices
- Sharing information, data, and photos with your connections

Reading the news

- Subscribing to an online news source

Health and fitness

- Using fitness trackers
- Using apps to receive healthcare

NETIQUETTE

It is the abbreviation of **Internet etiquette** or **network etiquette**, refers to online manners while using internet or working online. While online you should be courteous, truthful and respectful of others. It includes proper manners for sending [e-mail](#), conversing online, and so on.

Some basic rules of netiquette are:

- Be respectful
- Think about who can see what you have shared.
- Read first, then ask
- Pay attention to grammar and punctuation
- Respect the privacy of others
- Do not give out personal information

DATA PROTECTION

Data protection is a set of strategies and processes you can use to secure the privacy, availability, and integrity of your data. It is sometimes also called data security or information privacy. A data protection strategy is vital for any organization that collects, handles, or stores sensitive data.

Data Privacy v/s Data Protection

For data privacy, users can often control how much of their data is shared and with whom. For data protection, it is up to the companies handling data to ensure that it remains private. Data privacy is focused on defining who has access to data while data protection focuses on applying those restrictions.

How we can protect our personal data online

- Through Encrypt our Data
- Keep Passwords Private
- Don't Overshare on Social Networking Sites
- Use Security Software
- Avoid Phishing Emails
- Be Wise About Wi-Fi
- Be Alert to Impersonators
- Safely Dispose of Personal Information

INTELLECTUAL PROPERTY RIGHTS (IPR)

Intellectual Property (IP) – is a property created by a person or group of persons using their own intellect for ultimate use in commerce and which is already not available in the public domain.

Examples of Intellectual Property :- an invention relating to a product or any process, a new design, a literary or artistic work and a trademark (a word, a symbol and / or a logo, etc.)

Intellectual Property Right (IPR) is the statutory right granted by the Government, to the owner(s) of the intellectual property or applicant(s) of an intellectual property (IP) to exclude others from exploiting the IP commercially for a given period of time, in lieu of the disclosure of his/her IP in an IPR application.

Copyright laws protect intellectual property

Copyright It is a legal concept, enacted by most governments giving creator of original work exclusive rights to it, usually for a limited period.

Copyright infringement – When someone uses a copyrighted material without permission, it is called Copyright infringement.

Patent – A patent is a grant of exclusive right to the inventor by the government. Patent give the holder a right to exclude others from making, selling, using or importing a particular product or service, in exchange for full public disclosure of their invention.

Trademark – A Trademark is a word, phrase, symbol, sound, colour and/or design that identifies and distinguishes the products from those of others.

PLAGIARISM

Plagiarism It is stealing someone's intellectual work and representing it as your own work without citing the source of information.

Any of the following acts would be termed as Plagiarism:

- Using some other author's work without giving credit to the author
- Using someone else's work in incorrect form than intended originally by the author or creator.
- Modifying /lifting someone's production such as music composition etc. without attributing it to the creator of the work.
- Giving incorrect source of information.

LICENSING AND COPYRIGHT

Licenses are the permissions given to use a product or someone's creation by the copyright holder.

Copyright is a legal term to describe the rights of the creator of an original creative work such as a literary work, an artistic work, a design, song, movie or software etc.

FREE AND OPEN-SOURCE SOFTWARE (FOSS)

OSS refers to Open Source Software, which refers to software whose source code is available to customers and it can be modified and redistributed without any limitation.

Free and open-source software (FOSS) is software that can be classified as both free software and open-source software. That is, anyone is freely licensed to use, copy, study, and change the software in any way, and the source code is openly shared so that people are encouraged to voluntarily improve the design of the software.

❖ **CYBER CRIME:**

Any criminal or illegal activity through an electric channel or through any computer network is considered as cyber crime.

Eg: Cyber harassment and stalking, distribution of child pornography, types of spoofing, credit card fraud ,. etc

❖ **CYBER LAW:**

It is the law governing cyberspace which includes freedom of expression, access to and usage of internet and online privacy.

The issues addressed by cyber law include cybercrime, e-commerce, IPR and Data protection.

❖ **HACKING:**

It is an act of unauthorised access to a computer, computer network or any digital system.

Hackers usually are technical expertise of hardware and software.

- Hacking when done with a positive intent is called as **Ethical hacking or White hat**.
- Hacking when done with a negative intent is called as **Unethical hacking or Black hat**.

❖ **PHISHING:**

It is an unlawful activity where fake websites or emails appear as original or authentic .This sites when clicked by the user will collect sensitive and personal details like usernames, password, credit card details etc.

❖ **CYBER BULLYING:**

It is the use of technology to harass , threaten or humiliate a target .

Example: sharing of embarrassing photos or videos, posting false information, sending mean text., etc.

❖ **OVERVIEW OF INDIAN IT ACT:**

The Government of India's – Information Technology Act, 2000 (also known as IT Act) , amended in 2008, provides guidelines to the user on the processing , storage and transmission of sensitive information

❖ **E-waste - HAZARDS AND MANAGEMENT:**

Various forms of electric and electronic equipment which no longer satisfy their original purpose are termed as Ewaste. This includes Desktop, Laptop, Projectors, Mobiles,etc

- **HAZARDS:**It consists of mixtures of various hazardous organic and inorganic materials which when mixed with water/soil may create threat to the environment.
- **MANAGEMENT:** Sell back, gift/donate, reuse the parts giveaway to a certified e-waste recycler

❖ **AWARENESS ABOUT HEALTH CONCERNS RELATED TO THE USE OF TECHNOLOGY:**

There are positive as well as negative impact on health due to the use of these technologies.

- **POSITIVE IMPACT**
 - Various health apps and gadgets are available to monitor and alert
 - Online medical records can be maintained
- **NEGATIVE IMPACT**

- One may come across various health issues like eye strain, muscle problems, sleep issues, etc
- Anti social behaviour, isolation, emotional issues, etc.

ASSERTION AND REASONING BASED QUESTIONS

Assertion: (A) Plagiarism is stealing someone else's intellectual work and representing it as your own work.

Reason : (R) Using someone else's work and giving credit to the author or creator.

- a) Both A and R are true and R is the correct explanation of A.
- b) Both A and R are true but R is not the correct explanation of A.
- c) A is true but R is false.
- d) A is false but R is true.
- e) Both A and B are false

Ans: c) A is true but R is false.

MUTIPLE CHOICE QUESTIONS

1. Online posting of rumours, giving threats online, posting the victim's personal information, comments aimed to publicly ridicule a victim is termed as _____
 - a. Cyber bullying
 - b. Cyber crime
 - c. Cyber insult
 - d. All of the above

Ans: Cyber bullying

2. Ankit made a ERP - Enterprise resource planning solution for a renowned university and registered and Copyrights for the same. Which of the most important option; Ankit got the copyrights.
 - a. To get society status
 - b. To get fame
 - c. To get community welfare
 - d. To secure finance protection

Ans: To secure finance protection

3. Which of the following is not an example of Social media platform?
 - a. Facebook
 - b. Pinterest
 - c. Google+
 - d. Social channel

Ans: Social channel

4. A responsible netizen must abide by _____
- a. Net etiquettes
 - b. Communication etiquettes
 - c. Social media etiquettes
 - d. All of the above

Ans: All of the above

5. A _____ is some lines of malicious code that can copy itself and can have detrimental effect on the computers, by destroying data or corrupting the system.
- a. Cyber crime
 - b. Computer virus
 - c. Program
 - d. Software

Ans: Computer virus

6. Which of the following activity is an example of leaving Active digital footprints?
- a) Surfing internet
 - b) Visiting a website
 - c) Sending an email to a friend
 - d) None of the above

Ans: Sending an email to a friend

7. You are planning to go for a vacation. You surfed the internet to get answers for following queries.
- a) Places to visit
 - b) Availability of air tickets and fares
 - c) Best hotel deals
 - d) All of these

Which of the above-mentioned actions might have created a digital footprint?

Ans: All of these

8. Legal term to describe the rights of a creator of original creative or artistic work is called.....
- a) Copyright
 - b) Copyleft
 - c) GPL
 - d) BSD

Ans: Copyright

9. Intellectual Property is legally protected through _____
- a) copyright
 - b) patent
 - c) registered trademark

- d) All of the above
Ans: All of the above

10. _____ includes any visual symbol, word, name, design, slogan, label, etc., that distinguishes the brand from other brands.
- a) Trademark
 - b) Patent
 - c) Copyright
 - d) None of the above
- Ans:** Trademark

CASE STUDY BASED QUESTION:

1. Naveen received an email warning him of closure of his bank accounts if he did not update his banking information as soon as possible. He clicked the link in the email and entered his banking information. Next he got to know that he was duped.

- a) This is an example of _____ .
 - i. Online Fraud
 - ii. Identity Theft
 - iii. Phishing
 - iv. Plagiarism

- b) Someone steals Naveen's personal information to commit theft or fraud, it is called _____
 - i. Online Fraud
 - ii. Identity Theft
 - iii. Phishing
 - iv. Plagiarism

- c) Naveen receiving an Unsolicited commercial emails is known as _____
 - i. Spam
 - ii. Malware
 - iii. Virus
 - iv. Worms

- d) Naveen's Online personal account, personal website are the examples of?
 - i. Digital wallet
 - ii. Digital property
 - iii. Digital certificate
 - iv. Digital signature

- e) Sending mean texts, posting false information about a person online, or sharing embarrassing photos or videos to harass, threaten or humiliate a target person, is called _____
 - i. Eavesdropping
 - ii. Cyberbullying
 - iii. Spamming
 - iv. Phishing

Solution:

- a) iii. Phishing
- b) ii. Identity theft
- c) i. spam
- d) ii. Digital Property
- e) ii. Cyberbullying

2. Prathyush has to prepare a project on “**Cyber Jaagrookta Diwas**”. He decides to get information from the Internet. He downloads three web pages (webpage1, webpage 2, webpage 3) containing information on the given topic.

1. He read a paragraph from webpage 1 and rephrased it in his own words. He finally pasted the rephrased paragraph in his project. And he put a citation about the website he visited and its web address also.
2. He downloaded three images of from webpage 2. He made a collage for his project using these images.
3. He also downloaded an icon from web page 3 and pasted it on the front page of his project report.

(i) Step1 is an act of.....

- (a) Plagiarism
- (b) copyright infringement
- (c) Intellectual Property right
- (d) None of the above

(ii) Step 2 is an act of _____.

- (a) plagiarism
- (b) copyright infringement
- (c) Intellectual Property right
- (d) Digital Footprints

(iii) Step 3 is an act of _____.

- (a) Plagiarism
- (b) Paraphrasing
- (c) copyright infringement
- (d) Intellectual Property right

(iv) _____ is a small piece of data sent from a website and stored in a user’s web browser while a user is browsing a website.

- (a) Hyperlinks
- (b) Web pages
- (c) Browsers
- (d) Cookies

(v) The process of getting web pages, images and files from a web server to local computer is called

- (a) FTP
- (b) Uploading
- (c) Downloading
- (d) Remote access

Solution:

- I. (d)None of the above
- II. (a) plagiarism
- III. (c) copyright infringement
- IV. (d) Cookies
- V. (c) Downloading