

[illegible]**CBSE**



NCERT Informatics Practices

Class XII

DETAILED EXERCISE SOLUTIONS

With explanatory notes for average students



<https://pagenest.in/>

CONTENTS

CHAPTER 1: Querying and SQL Functions	1
CHAPTER 2: Data Handling Using Pandas - I	8
CHAPTER 3: Data Handling Using Pandas - II	14
CHAPTER 4: Plotting Data Using Matplotlib	22
CHAPTER 5: Internet and Web	31
CHAPTER 6: Societal Impacts	38

CHAPTER 1: Querying and SQL Functions

This chapter focuses on SQL (Structured Query Language) used to interact with Relational Database Management Systems (RDBMS). You will learn about SQL functions, queries, and joins.

Exercise Question 1

(a) Define RDBMS. Name any two RDBMS software.

RDBMS stands for Relational Database Management System. It is a type of database management system that stores data in a structured format using tables (called relations). Data in different tables can be related to each other using keys.

 **Note:** Think of RDBMS like an organised filing cabinet where each drawer is a table, and you can find connections between information in different drawers.

Two popular RDBMS software:

- MySQL – Free, open-source RDBMS widely used for web applications.
- Oracle – A commercial RDBMS used in large enterprise applications.

(b) Purpose of ORDER BY and HAVING clauses

(i) ORDER BY:

The ORDER BY clause is used to sort the result of a SELECT query in ascending (ASC) or descending (DESC) order based on one or more columns.

```
SELECT PName, UPrice FROM Product ORDER BY PName ASC;
```

 **Note:** By default, ORDER BY sorts in ascending order. Use DESC explicitly for descending order.

(ii) HAVING:

The HAVING clause is used to filter groups after the GROUP BY operation. It is like a WHERE clause but specifically for aggregate functions (like COUNT, SUM, AVG, etc.).

```
SELECT Manufacturer, COUNT(*) FROM Product GROUP BY Manufacturer HAVING  
COUNT(*) > 1;
```

 **Note:** Remember: WHERE filters individual rows BEFORE grouping. HAVING filters groups AFTER grouping.

(c) Two differences between Single-row functions and Aggregate functions

Single-Row Functions	Aggregate Functions
Operate on one row at a time and return one result per row.	Operate on a set of rows and return a single result for the group.
Examples: UPPER(), LENGTH(), ROUND()	Examples: SUM(), AVG(), COUNT(), MAX(), MIN()

Can be used in SELECT, WHERE, ORDER BY clauses.

Used with GROUP BY; cannot be used in WHERE (use HAVING instead).

(d) What do you understand by Cartesian Product?

A Cartesian Product (also called a Cross Join) is the result of combining every row from one table with every row from another table without any condition.

If Table A has 'm' rows and Table B has 'n' rows, then the Cartesian product will have m×n rows.

 **Note:** For example, if the Student table has 6 rows and the Stream table has 3 rows, the Cartesian product will produce 6×3 = 18 rows. In practice, Cartesian products are rarely useful; we use proper JOIN conditions to link related rows.

(e) Functions for specified operations

(i) To display the day like 'Monday', 'Tuesday' from India's independence date (1947-08-15):

```
SELECT DAYNAME('1947-08-15');
```

(ii) To display the specified number of characters from a particular position of a string:

```
SELECT SUBSTR(string, position, length); -- OR MID(string, position, length)
```

(iii) To display the name of the month in which you were born (e.g., if born on 2005-03-15):

```
SELECT MONTHNAME('2005-03-15');
```

(iv) To display your name in capital letters:

```
SELECT UPPER('your_name');
```

Exercise Question 2

Write the output produced by the following SQL commands:

(a) SELECT POW(2,3);

```
Output: 8
```

 **Note:** POW(base, exponent) raises the base to the power of exponent. $2^3 = 8$.

(b) SELECT ROUND(123.2345, 2), ROUND(342.9234, -1);

```
Output: 123.23 | 340
```

 **Note:** ROUND(number, decimal_places): With positive decimal_places, it rounds to that many decimal places. With negative decimal_places, it rounds to the left of the decimal point. -1 rounds to the nearest 10, so 342.9234 rounds to 340.

(c) SELECT LENGTH('Informatics Practices');

```
Output: 21
```

 **Note:** LENGTH() counts all characters including spaces. Count: I-n-f-o-r-m-a-t-i-c-s (11) + space (1) + P-r-a-c-t-i-c-e-s (9) = 21.

(d) SELECT YEAR('1979/11/26'), MONTH('1979/11/26'), DAY('1979/11/26'), MONTHNAME('1979/11/26');

```
Output: 1979 | 11 | 26 | November
```

 **Note:** These date functions extract specific parts from a date value.

(e) SELECT LEFT('INDIA', 3), RIGHT('Computer Science', 4);

Output: IND | ence

 **Note:** LEFT(string, n) returns the first n characters. RIGHT(string, n) returns the last n characters.

(f) SELECT MID('Informatics', 3, 4), SUBSTR('Practices', 3);

Output: form | actices

 **Note:** MID('Informatics', 3, 4): starts at position 3 (character 'f') and picks 4 characters—form'. SUBSTR('Practices', 3): starts at position 3 and goes to end—actices'.

Exercise Question 3

Table: Product

PCode	PName	UPrice	Manufacturer
P01	Washing Powder	120	Surf
P02	Tooth Paste	54	Colgate
P03	Soap	25	Lux
P04	Tooth Paste	65	Pepsodant
P05	Soap	38	Dove
P06	Shampoo	245	Dove

(a) SQL Queries:

(i) Create the table Product with appropriate data types and constraints:

```
CREATE TABLE Product (
    PCode VARCHAR(5) PRIMARY KEY,
    PName VARCHAR(30) NOT NULL,
    UPrice DECIMAL(8,2) NOT NULL,
    Manufacturer VARCHAR(30)
);
```

 **Note:** PCode is the Primary Key (unique identifier). We use VARCHAR for text fields and DECIMAL for price to handle decimal values.

(ii) Primary Key in Product:

The Primary Key is PCode, because each product has a unique code (P01, P02, etc.).

(iii) List Product Code, Product Name and Price in descending order of PName; if PName is same, ascending order of price:

```
SELECT PCode, PName, UPrice
FROM Product
ORDER BY PName DESC, UPrice ASC;
```

 **Note:** When sorting by multiple columns, list them separated by commas. The first column is sorted first, and for ties, the second column decides the order.

(iv) Add a new column Discount to the table Product:

```
ALTER TABLE Product ADD Discount DECIMAL(8,2);
```

(v) Calculate discount as 10% of UPrice where UPrice > 100, otherwise 0:

```
UPDATE Product
SET Discount = CASE
    WHEN UPrice > 100 THEN UPrice * 0.10
    ELSE 0
END;
```

 **Note:** The CASE statement works like an IF-ELSE condition in SQL. Here: if UPrice is more than 100, discount = 10% of UPrice; otherwise discount = 0.

(vi) Increase price by 12% for all products manufactured by Dove:

```
UPDATE Product
SET UPrice = UPrice * 1.12
WHERE Manufacturer = 'Dove';
```

(vii) Display the total number of products manufactured by each manufacturer:

```
SELECT Manufacturer, COUNT(*) AS TotalProducts
FROM Product
GROUP BY Manufacturer;
```

(b) Outputs of queries:

(i) SELECT PName, Average(UPrice) FROM Product GROUP BY PName;

 **Note:** Note: The correct function is AVG() not Average(). Assuming AVG():

PName	AVG(UPrice)
Shampoo	245.00
Soap	31.50 (avg of 25 and 38)
Tooth Paste	59.50 (avg of 54 and 65)
Washing Powder	120.00

(ii) SELECT DISTINCT Manufacturer FROM Product;

```
Output: Surf Colgate Lux Pepsodant Dove
```

 **Note:** DISTINCT removes duplicate values. Though Dove appears for Soap and Shampoo, it is listed only once.

(iii) SELECT COUNT(DISTINCT PName) FROM Product;

```
Output: 4
```

 **Note:** There are 4 distinct product names: Washing Powder, Tooth Paste, Soap, Shampoo.

(iv) SELECT PName, MAX(UPrice), MIN(UPrice) FROM Product GROUP BY PName;

PName	MAX(UPrice)	MIN(UPrice)
Shampoo	245	245
Soap	38	25
Tooth Paste	65	54

Washing Powder	120	120
----------------	-----	-----

Exercise Question 4

Using CARSHOWROOM database (INVENTORY table)

 **Note:** The CARSHOWROOM database has an INVENTORY table with car details. LXI, VXi are model variants.

(a) Add a new column Discount in the INVENTORY table:

```
ALTER TABLE INVENTORY ADD Discount DECIMAL(8,2);
```

(b) Set discount values:

```
UPDATE INVENTORY
SET Discount = CASE
    WHEN Model = 'LXI' THEN 0
    WHEN Model = 'VXI' THEN Price * 0.10
    ELSE Price * 0.12
END;
```

 **Note:** This uses a CASE statement to apply different discount rates based on car model.

(c) Display the name of the costliest car with fuel type 'Petrol':

```
SELECT Car_Name
FROM INVENTORY
WHERE Fuel_Type = 'Petrol'
ORDER BY Price DESC
LIMIT 1;
```

 **Note:** Alternatively: `SELECT Car_Name FROM INVENTORY WHERE Fuel_Type='Petrol' AND Price = (SELECT MAX(Price) FROM INVENTORY WHERE Fuel_Type='Petrol');`

(d) Calculate the average discount and total discount available on Car4:

```
SELECT AVG(Discount) AS AvgDiscount, SUM(Discount) AS TotalDiscount
FROM INVENTORY
WHERE Car_Name = 'Car4';
```

(e) List the total number of cars having no discount:

```
SELECT COUNT(*) AS NoCarsNoDiscount
FROM INVENTORY
WHERE Discount = 0 OR Discount IS NULL;
```

Exercise Question 5

Tables: Student and Stream (Streams_of_Students database)

AdmNo	Name	StCode
211	Jay	NULL

241	Aditya	S03
290	Diksha	S01
333	Jasqueen	S02
356	Vedika	S01
380	Ashpreet	S03

StCode	Stream
S01	Science
S02	Commerce
S03	Humanities

(a) Create the database:

```
CREATE DATABASE Streams_Of_Students;
USE Streams_Of_Students;
```

(b) Create the Student table:

```
CREATE TABLE Student (
    AdmNo INT(4) PRIMARY KEY,
    Name VARCHAR(20) NOT NULL,
    StCode VARCHAR(4),
    FOREIGN KEY (StCode) REFERENCES Stream(StCode)
);
```

 **Note:** AdmNo is an integer (3-digit number), Name is a text field. StCode references the Stream table.

(c) Primary keys and foreign keys:

- Primary Key of Student table: AdmNo
- Primary Key of Stream table: StCode
- Foreign Key in Student table: StCode (references Stream table's StCode)

 **Note:** Note: The question says 'identify the foreign key from the table Stream' which seems to be an error in the book. The foreign key is in the Student table, not the Stream table.

(d) Jay has changed his stream to Humanities (S03):

```
UPDATE Student
SET StCode = 'S03'
WHERE AdmNo = 211;
```

(e) Display names ending with 'a', arranged alphabetically:

```
SELECT Name FROM Student
WHERE Name LIKE '%a'
ORDER BY Name ASC;
```

 **Note:** The % wildcard matches any sequence of characters. '%a' means 'any text ending with a'. Expected output: Aditya, Diksha, Vedika (and now Jay—Humanities, but Jay doesn't end in 'a').

(f) Names of students in Science and Humanities, ordered by name then AdmNo:

```
SELECT S.Name FROM Student S
```

```
JOIN Stream ST ON S.StCode = ST.StCode
WHERE ST.Stream IN ('Science', 'Humanities')
ORDER BY S.Name ASC, S.AdmNo ASC;
```

(g) Number of students in each stream having more than 1 student:

```
SELECT StCode, COUNT(*) AS StudentCount
FROM Student
GROUP BY StCode
HAVING COUNT(*) > 1;
```

(h) Names of students enrolled in different streams, in descending order of AdmNo:

```
SELECT S.Name, ST.Stream FROM Student S
JOIN Stream ST ON S.StCode = ST.StCode
ORDER BY S.AdmNo DESC;
```

(i) Cartesian product on Student and Stream:

```
SELECT * FROM Student, Stream;
```

 **Note:** Degree (number of columns) = columns in Student + columns in Stream = 3 + 2 = 5 columns. Cardinality (number of rows) = rows in Student × rows in Stream = 6 × 3 = 18 rows.

(j) Add TeacherIncharge column to Stream and insert data:

```
ALTER TABLE Stream ADD TeacherIncharge VARCHAR(30);
UPDATE Stream SET TeacherIncharge = 'Mr. Sharma' WHERE StCode = 'S01';
UPDATE Stream SET TeacherIncharge = 'Mrs. Gupta' WHERE StCode = 'S02';
UPDATE Stream SET TeacherIncharge = 'Mr. Verma' WHERE StCode = 'S03';
```

(k) List the names of teachers and students (using UNION):

```
SELECT Name FROM Student
UNION
SELECT TeacherIncharge FROM Stream;
```

 **Note:** UNION combines results from two SELECT queries, removing duplicates. This gives us all names from both tables.

(l) Degree and cardinality after adding TeacherIncharge and applying Cartesian product again:

 **Note:** Stream table now has 3 columns (StCode, Stream, TeacherIncharge). Student still has 3 columns. Degree = 3 + 3 = 6 columns. Cardinality = 6 (students) × 3 (streams) = 18 rows. (Cardinality remains same since we only added a column, not rows.)

CHAPTER 2: Data Handling Using Pandas - I

This chapter introduces Python's Pandas library, which is used for data manipulation and analysis. Key data structures: Series (1D) and DataFrame (2D).

Exercise Question 1

What is a Series and how is it different from a 1-D array, a list and a dictionary?

A Pandas Series is a one-dimensional labelled array capable of holding data of any type (integers, strings, floats, etc.). Each element has an associated label called an index.

 **Note:** Think of a Series as a column in an Excel spreadsheet with row labels (index) on the left.

Differences:

Feature	1-D Array / List	Pandas Series
Index	Only numeric indices (0, 1, 2...)	Custom labels (can be text, numbers, dates)
Data Types	List allows mixed types; NumPy array is homogeneous	Homogeneous (like NumPy array)
Operations	Limited math operations on lists	Supports vectorised operations and alignment by index
Dictionary Comparison	Dictionary has key-value pairs but no positional access	Series is like an ordered dictionary with label-based AND position-based access

Exercise Question 2

What is a DataFrame and how is it different from a 2-D array?

A Pandas DataFrame is a two-dimensional, table-like data structure with labelled rows (index) and labelled columns. It can hold columns of different data types.

Feature	2-D NumPy Array	Pandas DataFrame
Row Labels	Only numeric indices (0, 1, 2...)	Custom row labels (index)
Column Labels	No column names	Named columns (like a spreadsheet)
Data Types	All elements same type	Each column can have a different type
Flexibility	Fixed structure	Can add/remove rows and columns easily

 **Note:** A DataFrame is essentially a collection of Series sharing the same index.

Exercise Question 3

How are DataFrames related to Series?

A DataFrame can be thought of as a dictionary of Series objects, where each column is a Series.

- All Series in a DataFrame share the same index (row labels).
- Selecting a single column from a DataFrame returns a Series.
- A DataFrame can be created from multiple Series objects.

 **Note:** Analogy: If a Series is like a single column in a spreadsheet, then a DataFrame is like the entire spreadsheet with multiple columns.

Exercise Question 4

What do you understand by the size of (i) a Series, (ii) a DataFrame?

(i) Size of a Series: The total number of elements in the Series.

```
s = pd.Series([10, 20, 30, 40])
print(s.size) # Output: 4
```

(ii) Size of a DataFrame: The total number of elements = number of rows × number of columns.

```
df = pd.DataFrame({'A': [1,2,3], 'B': [4,5,6]})
print(df.size) # Output: 6 (3 rows × 2 columns)
```

Exercise Question 5

Create the following Series:

First, import necessary libraries:

```
import pandas as pd
import numpy as np
```

(a) EngAlph – 26 alphabets with default index:

```
import string
EngAlph = pd.Series(list(string.ascii_lowercase))
print(EngAlph)
```

 **Note:** `string.ascii_lowercase` gives 'abcdefghijklmnopqrstuvwxyz'. `list()` converts it to individual characters. Default index = 0, 1, 2, ... 25

(b) Vowels – 5 elements with vowel labels, all values zero:

```
Vowels = pd.Series([0, 0, 0, 0, 0], index=['a', 'e', 'i', 'o', 'u'])
print(Vowels.empty) # Output: False (it has elements, just all zeros)
```

 **Note:** A Series is considered empty only when it has NO elements, not when values are zero.

(c) Friends – from a dictionary (roll numbers as data, names as keys):

```
friends_dict = {'Ravi': 101, 'Priya': 102, 'Rohan': 103, 'Sneha': 104,
'Amit': 105}
Friends = pd.Series(friends_dict)
```

(d) MTseries – empty Series:

```
MTseries = pd.Series([], dtype='float64')
print(MTseries.empty) # Output: True
```

(e) MonthDays – from numpy array, labels = month numbers 1 to 12:

```
days = np.array([31, 28, 31, 30, 31, 30, 31, 31, 30, 31, 30, 31])
```

```
MonthDays = pd.Series(days, index=range(1, 13))
```

 **Note:** *range(1, 13) gives numbers 1 through 12 as the index.*

Exercise Question 6

Operations on Series:

(a) Set all values of Vowels to 10:

```
Vowels[:] = 10
print(Vowels)
```

(b) Divide all values of Vowels by 2:

```
Vowels = Vowels / 2
print(Vowels) # All values will be 5.0
```

(c) Create Vowels1 with values [2, 5, 6, 3, 8]:

```
Vowels1 = pd.Series([2, 5, 6, 3, 8], index=['a', 'e', 'i', 'o', 'u'])
```

(d) Add Vowels and Vowels1, assign to Vowels3:

```
Vowels3 = Vowels + Vowels1
```

 **Note:** *After step (b), Vowels has all values = 5. So Vowels3 = [5+2, 5+5, 5+6, 5+3, 5+8] = [7, 10, 11, 8, 13]*

(e) Subtract, Multiply, Divide Vowels by Vowels1:

```
print(Vowels - Vowels1) # Subtraction
print(Vowels * Vowels1) # Multiplication
print(Vowels / Vowels1) # Division
```

(f) Alter labels of Vowels1 to uppercase:

```
Vowels1.index = ['A', 'E', 'I', 'O', 'U']
```

Exercise Question 7

Finding dimensions, size and values of Series:

(a) Dimensions, size and values:

```
# Dimensions (ndim)
print(EngAlph.ndim) # Output: 1 (all Series are 1-dimensional)
print(Vowels.ndim) # Output: 1

# Size (number of elements)
print(EngAlph.size) # Output: 26
print(Vowels.size) # Output: 5
print(Friends.size) # Output: 5 (your 5 friends)
print(MTseries.size) # Output: 0
print(MonthDays.size) # Output: 12

# Values
```

```
print(EngAlph.values)    # array of 26 alphabets
print(Vowels.values)    # array([5., 5., 5., 5., 5.]) after step 6b
```

(b) Rename MTseries as SeriesEmpty:

```
MTseries.name = 'SeriesEmpty'
# OR: SeriesEmpty = MTseries.rename('SeriesEmpty')
```

(c) Name the index of MonthDays as 'monthno' and Friends as 'Fname':

```
MonthDays.index.name = 'monthno'
Friends.index.name = 'Fname'
```

(d) Display 3rd and 2nd value of Friends, in that order:

```
print(Friends.iloc[[2, 1]])
```

 **Note:** *iloc uses position-based indexing (0-based). So iloc[2] = 3rd element, iloc[1] = 2nd element.*

(e) Display alphabets 'e' to 'p' from EngAlph:

```
print(EngAlph[4:16])
```

 **Note:** *'e' is at index 4, 'p' is at index 15. Slicing with [4:16] gives positions 4 to 15 (16 is excluded).*

(f) Display first 10 values of EngAlph:

```
print(EngAlph.head(10))
```

(g) Display last 10 values of EngAlph:

```
print(EngAlph.tail(10))
```

(h) Display MTseries:

```
print(MTseries)    # Output: Series([], dtype: float64)
```

Exercise Question 8

Using MonthDays Series:

(a) Display names of months 3 through 7:

```
print(MonthDays[3:8])
```

 **Note:** *Since index is 1 to 12, MonthDays[3:8] gives months 3,4,5,6,7 (March to July) with their day counts.*

(b) Display MonthDays in reverse order:

```
print(MonthDays[::-1])
```

 **Note:** *The [::-1] slicing trick reverses any sequence in Python.*

Exercise Question 9

Create DataFrame Sales:

```
import pandas as pd

data = {
```

```

2014: [100.5, 150.8, 200.9, 30000, 40000],
2015: [12000, 18000, 22000, 30000, 45000],
2016: [20000, 50000, 70000, 100000, 125000],
2017: [50000, 60000, 70000, 80000, 90000]
}

Sales = pd.DataFrame(data,
    index=['Madhu', 'Kusum', 'Kinshuk', 'Ankit', 'Shruti'])
print(Sales)

```

 **Note:** The dictionary keys become column labels (2014, 2015, 2016, 2017). The index list provides row labels (salesperson names).

Exercise Question 10

Operations on Sales DataFrame:

(a) Display row labels:

```

print(Sales.index)
# Output: Index(['Madhu', 'Kusum', 'Kinshuk', 'Ankit', 'Shruti'],
dtype='object')

```

(b) Display column labels:

```

print(Sales.columns)
# Output: Int64Index([2014, 2015, 2016, 2017], dtype='int64')

```

(c) Display data types of each column:

```

print(Sales.dtypes)

```

(d) Display dimensions, shape, size and values:

```

print(Sales.ndim)      # Output: 2
print(Sales.shape)     # Output: (5, 4) → 5 rows, 4 columns
print(Sales.size)      # Output: 20 → 5 × 4
print(Sales.values)    # 2D numpy array of all values

```

(e) Display last two rows:

```

print(Sales.tail(2))

```

(f) Display first two columns:

```

print(Sales.iloc[:, 0:2])

```

 **Note:** `iloc[:, 0:2]` means all rows (:) and columns at position 0 and 1 (0:2).

(g) Create Sales2 DataFrame:

```

sales2_dict = {2018: {'Madhu': 160000, 'Kusum': 110000,
                      'Kinshuk': 500000, 'Ankit': 340000, 'Shruti':
                      900000}}
Sales2 = pd.DataFrame(sales2_dict)

```

(h) Check if Sales2 is empty:

```

print(Sales2.empty)  # Output: False (it contains data)

```

Exercise Question 11

Advanced DataFrame Operations:

(a) Append Sales2 to Sales:

```
Sales = pd.concat([Sales, Sales2], axis=1)
```

 **Note:** *axis=1 adds Sales2 as a new column (year 2018) to the existing Sales DataFrame.*

(b) Transpose Sales:

```
Sales = Sales.T
```

 **Note:** *Transposing swaps rows and columns: rows become columns and vice versa.*

(c) Display sales of all salespeople in 2017:

```
print(Sales[2017])
```

(d) Display sales by Madhu and Ankit in 2017 and 2018:

```
print(Sales.loc[['Madhu', 'Ankit'], [2017, 2018]])
```

(e) Display sales by Shruti in 2016:

```
print(Sales.loc['Shruti', 2016])
```

(f) Add data for Sumeet:

```
Sales.loc['Sumeet'] = [196.2, 37800, 52000, 78438, 38852]
```

(g) Delete column 2014 (year 2014 data):

```
Sales = Sales.drop(columns=[2014])  
# OR: del Sales[2014]
```

(h) Delete row for Kinshuk:

```
Sales = Sales.drop('Kinshuk')
```

(i) Rename Ankit to Vivaan and Madhu to Shailesh:

```
Sales = Sales.rename(index={'Ankit': 'Vivaan', 'Madhu': 'Shailesh'})
```

(j) Update sale by Shailesh in 2018 to 100000:

```
Sales.loc['Shailesh', 2018] = 100000
```

(k) Write Sales to CSV without row/column labels:

```
Sales.to_csv('SalesFigures.csv', index=False, header=False)
```

(l) Read CSV back and update labels:

```
SalesRetrieved = pd.read_csv('SalesFigures.csv', header=None)  
SalesRetrieved.index = Sales.index  
SalesRetrieved.columns = Sales.columns  
print(SalesRetrieved)
```

 **Note:** *Since we saved without headers, we need to manually set the row and column labels after reading.*

CHAPTER 3: Data Handling Using Pandas - II

This chapter covers advanced Pandas features including descriptive statistics, data aggregation, handling missing values, pivoting, and connecting with MySQL.

Exercise Question 1

Statement to install pymysql connector:

```
pip install pymysql
```

 **Note:** This command is run in the terminal/command prompt, not inside Python. pymysql is a Python connector that allows Python to communicate with MySQL databases.

Exercise Question 2

Difference between pivot() and pivot_table() function:

pivot()	pivot_table()
Reshapes data based on column values. Requires unique index-column combinations.	More flexible version; handles duplicate entries using an aggregation function.
Does NOT handle duplicate values; raises ValueError if duplicates exist.	Handles duplicates by applying aggregate functions like mean, sum, count, etc.
Syntax: df.pivot(index, columns, values)	Syntax: df.pivot_table(values, index, columns, aggfunc)
Best for simple reshaping with unique data.	Best for summarising data with duplicates (like an Excel PivotTable).

 **Note:** Remember: pivot_table() is like the Excel PivotTable feature you may have seen. It summarises large data sets easily.

Exercise Question 3

What is SQLAlchemy?

SQLAlchemy is a Python library that provides a powerful and flexible interface for connecting Python programs to relational databases like MySQL, PostgreSQL, SQLite, etc.

- It acts as a bridge between Python's Pandas library and SQL databases.
- It is required when using pd.read_sql() or DataFrame.to_sql() to transfer data between Pandas and MySQL.
- It provides an Object Relational Mapper (ORM) which lets you interact with databases using Python objects instead of writing raw SQL.

```
# Installation
pip install sqlalchemy
```

```
# Usage example:
from sqlalchemy import create_engine
engine =
create_engine('mysql+pymysql://username:password@localhost/dbname')
```

Exercise Question 4

Can you sort a DataFrame with respect to multiple columns?

Yes, you can sort a DataFrame with respect to multiple columns using the `sort_values()` function.

```
df.sort_values(by=['Column1', 'Column2'], ascending=[True, False])
```

 **Note:** In this example, the DataFrame is first sorted by Column1 in ascending order. For rows where Column1 is the same, they are sorted by Column2 in descending order. The ascending parameter can take a list of True/False values for each column.

Exercise Question 5

What are missing values? What are the strategies to handle them?

Missing values are data points that are absent in a dataset. In Pandas, they are typically represented as NaN (Not a Number) or None.

Strategies to handle missing values:

1. Drop them: Remove rows or columns with missing values.

```
df.dropna()          # Drop rows with any NaN
df.dropna(axis=1)    # Drop columns with any NaN
```

2. Fill with a constant value:

```
df.fillna(0)         # Replace all NaN with 0
df.fillna('Unknown') # Replace with text
```

3. Fill with statistical values:

```
df.fillna(df.mean()) # Replace NaN with column mean
df.fillna(df.median()) # Replace with median
```

4. Forward/Backward fill (use previous/next valid value):

```
df.fillna(method='ffill') # Forward fill
df.fillna(method='bfill') # Backward fill
```

 **Note:** The best strategy depends on the data. For example, for continuous numerical data, replacing with mean/median is common. For categorical data, the mode (most frequent value) is often used.

Exercise Question 6

Define Median, Standard Deviation and Variance:

Median:

The median is the middle value when data is arranged in ascending order. If there are an even number of values, the median is the average of the two middle values.

```
df['column'].median()
```

Standard Deviation:

Standard deviation measures how spread out the values are from the mean. A low standard deviation means values are close to the mean; a high standard deviation means they are spread far.

```
df['column'].std()
```

Variance:

Variance is the square of the standard deviation. It quantifies the degree of variation in the dataset.

```
df['column'].var()
```

 **Note:** Simple analogy: If students' test scores are 90, 91, 89, 90 – standard deviation is low (clustered). If scores are 40, 60, 80, 100 – standard deviation is high (spread out).

Exercise Question 7

What is MODE? Name the function to calculate it.

The mode is the value(s) that appear most frequently in a dataset. There can be more than one mode if multiple values appear the same number of times.

```
df['column'].mode()
```

 **Note:** Example: In the series [1, 2, 2, 3, 3, 3, 4], the mode is 3 because it appears most often (3 times). In [1, 1, 2, 2, 3], both 1 and 2 are modes.

Exercise Question 8

Purpose of Data Aggregation:

Data aggregation refers to the process of collecting and summarising data into a single, meaningful value or set of values.

Purpose/Importance:

- Summarise large datasets into useful statistics (sum, average, count, etc.)
- Enables comparison across groups (e.g., average sales by region)
- Helps in identifying patterns and trends in data
- Reduces data complexity while retaining key information

```
df.groupby('Category')['Sales'].sum()    # Total sales per category
df.groupby('Region')['Sales'].mean()    # Average sales per region
```

Exercise Question 9

Concept of GROUP BY with an example:

GROUP BY is used to group rows that have the same values in a specified column(s). It is typically used with aggregate functions to compute summary statistics for each group.

 **Note:** Think of it like sorting students into groups by their class, and then calculating the average marks for each class group.

```
# Example: Product DataFrame
import pandas as pd
data = {'Product': ['Apple', 'Banana', 'Apple', 'Banana', 'Cherry'],
        'Sales': [100, 80, 120, 90, 50]}
df = pd.DataFrame(data)

# Group by Product and find total sales
result = df.groupby('Product')['Sales'].sum()
print(result)

# Output:
# Product
# Apple      220
# Banana     170
# Cherry      50
```

Exercise Question 10

Steps to read data from a MySQL database to a DataFrame:

5. Install required libraries (pymysql and sqlalchemy):

```
pip install pymysql sqlalchemy
```

6. Import the libraries in Python:

```
import pandas as pd
from sqlalchemy import create_engine
```

7. Create a database connection engine:

```
engine =
create_engine('mysql+pymysql://username:password@localhost/database_name')
```

8. Write and execute the SQL query to fetch data:

```
query = 'SELECT * FROM table_name'
df = pd.read_sql(query, engine)
```

9. Display or work with the data:

```
print(df)
```

 **Note:** Replace 'username', 'password', 'localhost', 'database_name', and 'table_name' with your actual credentials and table name.

Exercise Question 11

Importance of reshaping data with an example:

Data reshaping refers to transforming the structure (layout) of data without changing its content. It helps in converting data from one format to another for better analysis and visualisation.

Importance:

- Helps in creating pivot tables for multi-dimensional analysis
- Makes data suitable for specific algorithms or visualisations
- Enables comparison across categories easily

```
import pandas as pd
data = {'Date': ['Jan', 'Jan', 'Feb', 'Feb'],
        'Product': ['A', 'B', 'A', 'B'],
        'Sales': [200, 150, 300, 250]}
df = pd.DataFrame(data)

# Pivot to reshape: Date as rows, Products as columns
pivoted = df.pivot(index='Date', columns='Product', values='Sales')
print(pivoted)

# Output:
# Product      A      B
# Date
# Feb          300    250
# Jan          200    150
```

Exercise Question 12

Why is estimation an important concept in data analysis?

Estimation is the process of making reasonable guesses or approximations about population parameters (like mean, total) based on sample data.

Importance of estimation:

- In real-world problems, it is impossible to collect data from every single member of a population. We use samples to estimate population characteristics.
- Statistical estimation helps in forecasting future trends (e.g., predicting next year's sales based on past 5 years).
- It helps in making decisions with incomplete data under uncertainty.
- Used in fields like economics, healthcare, marketing, and government surveys.

 **Note:** Example: Instead of measuring heights of all 1 billion Indians, we measure heights of 10,000 people and estimate the average height for the entire population.

Exercise Question 13

Python code for Product table:

Item	Company	Rupees	USD
TV	LG	12000	700
TV	VIDEOCON	10000	650

TV	LG	15000	800
AC	SONY	14000	750

(a) Create the DataFrame:

```
import pandas as pd
data = {
    'Item': ['TV', 'TV', 'TV', 'AC'],
    'Company': ['LG', 'VIDEOCON', 'LG', 'SONY'],
    'Rupees': [12000, 10000, 15000, 14000],
    'USD': [700, 650, 800, 750]
}
df = pd.DataFrame(data)
```

(b) Add new rows to the DataFrame:

```
new_row = {'Item': 'Fridge', 'Company': 'Samsung', 'Rupees': 18000, 'USD': 900}
df = pd.concat([df, pd.DataFrame([new_row])], ignore_index=True)
```

(c) Display the maximum price of LG TV:

```
lg_tv = df[(df['Item'] == 'TV') & (df['Company'] == 'LG')]
print(lg_tv['Rupees'].max())
# Output: 15000
```

(d) Display the sum of all products:

```
print(df[['Rupees', 'USD']].sum())
```

(e) Display the median of USD for Sony products:

```
sony = df[df['Company'] == 'SONY']
print(sony['USD'].median())
# Output: 750.0
```

(f) Sort by Rupees and transfer to MySQL:

```
from sqlalchemy import create_engine
df_sorted = df.sort_values(by='Rupees')
engine = create_engine('mysql+pymysql://user:password@localhost/dbname')
df_sorted.to_sql('Product', engine, if_exists='replace', index=False)
```

(g) Transfer a new DataFrame with new values to MySQL:

```
new_data = {'Item': ['Washing Machine'], 'Company': ['LG'],
            'Rupees': [25000], 'USD': [1200]}
new_df = pd.DataFrame(new_data)
new_df.to_sql('NewProduct', engine, if_exists='replace', index=False)
```

Exercise Question 14

Python statements for student dataset:

 **Note:** The question references a dataset with Name, Degree, Stream, and Marks columns (with some NaN values). A sample dataset is assumed below.

```
import pandas as pd
```

```
import numpy as np

data = {
    'Name': ['Amit', 'Priya', 'Ravi', 'Sneha', 'Kunal', 'Pooja'],
    'Degree': ['BCA', 'MBA', 'BCA', 'MBA', 'BCA', 'MBA'],
    'Stream': ['Science', 'Commerce', 'Science', 'Arts', 'Commerce',
'Science'],
    'Marks': [75, 80, np.nan, 90, 85, np.nan]
}
df = pd.DataFrame(data)
```

(a) Create the DataFrame: (already shown above)

(b) Print Degree and maximum marks in each stream:

```
result = df.groupby(['Stream', 'Degree'])['Marks'].max()
print(result)
```

(c) Fill NaN with 76:

```
df['Marks'].fillna(76, inplace=True)
```

(d) Set index to Name:

```
df.set_index('Name', inplace=True)
```

(e) Display name and degree-wise average marks:

```
print(df.groupby('Degree')['Marks'].mean())
```

(f) Count the number of students in MBA:

```
print(df[df['Degree'] == 'MBA'].shape[0])
```

(g) Print the mode marks in BCA:

```
bca = df[df['Degree'] == 'BCA']
print(bca['Marks'].mode())
```

Solved Case Study: Auto-MPG Open Dataset

 **Note:** The UCI auto-mpg dataset has 398 rows and 9 attributes about car fuel efficiency.

1) Load auto-mpg.data into a DataFrame autodf:

```
import pandas as pd
cols = ['mpg', 'cylinders', 'displacement', 'horsepower',
        'weight', 'acceleration', 'model_year', 'origin', 'car_name']
autodf = pd.read_csv('auto-mpg.data', sep='\s+',
                    names=cols, na_values='?')
```

2) Description of the DataFrame:

```
print(autodf.describe())
```

 **Note:** describe() gives count, mean, std, min, 25%, 50%, 75%, max for numerical columns.

3) Display first 10 rows:

```
print(autodf.head(10))
```

4) Find attributes with missing values and handle them:

```
# Find missing values
```

```
print(autodf.isnull().sum()) # horsepower has 6 missing values

# Method i: Replace with previous valid value
autodf_ffill = autodf.fillna(method='ffill')

# Method ii: Remove rows with missing values
autodf_dropped = autodf.dropna()
```

5) Print details of the car with maximum mileage:

```
max_mpg_row = autodf[autodf['mpg'] == autodf['mpg'].max()]
print(max_mpg_row)
```

6) Find average displacement given number of cylinders:

```
print(autodf.groupby('cylinders')['displacement'].mean())
```

7) Average number of cylinders:

```
print(autodf['cylinders'].mean())
```

8) Number of cars with weight greater than average weight:

```
avg_weight = autodf['weight'].mean()
count = autodf[autodf['weight'] > avg_weight].shape[0]
print(f'Cars heavier than average: {count}')
```

CHAPTER 4: Plotting Data Using Matplotlib

Matplotlib is a comprehensive Python library for creating static, animated, and interactive visualisations. It helps in making data insights visual and easy to understand.

Exercise Question 1

What is the purpose of the Matplotlib library?

Matplotlib is a Python library used to create various types of data visualisations (plots, charts, and graphs). It helps data analysts and scientists:

- Visualise data patterns and trends that are not obvious from raw numbers.
- Create publication-quality charts like line graphs, bar charts, histograms, scatter plots, and pie charts.
- Communicate data insights effectively to others.
- Compare datasets visually for better understanding.

 **Note:** In short, Matplotlib turns data (numbers/tables) into visual stories that are easier to understand. It integrates perfectly with Pandas and NumPy.

```
import matplotlib.pyplot as plt # Standard import alias is 'plt'
```

Exercise Question 2

What are some of the major components of any graph or plot?

A typical Matplotlib plot has the following major components:

Component	Description
Figure	The entire area/window where the plot is drawn. It can contain one or more subplots.
Axes	The actual plot area inside the figure where data is drawn. It contains the x-axis and y-axis.
X-axis / Y-axis	The horizontal and vertical reference lines along which data is plotted.
Title	A heading displayed at the top of the plot describing what the plot shows.
X-label / Y-label	Labels that describe what the x-axis and y-axis represent.
Legend	A key that explains the meaning of different colours/symbols used in the plot.
Tick Marks	Small marks on the axes showing specific values.
Grid Lines	Horizontal and vertical lines in the background to help read values.
Data/Plot Area	The actual drawn data (lines, bars, points, etc.)

Exercise Question 3

Name the function which is used to save the plot.

```
plt.savefig('filename.png')
```

 **Note:** You can save as various formats: .png, .jpg, .pdf, .svg, etc. Always call `savefig()` *BEFORE* `plt.show()`, otherwise the saved file will be blank.

```
# Example:
plt.plot([1, 2, 3], [4, 5, 6])
plt.savefig('my_plot.png', dpi=300, bbox_inches='tight')
plt.show()
```

Exercise Question 4

Short notes on different customisation options available with any plot:

Matplotlib offers many customisation options:

(a) Color: Change the colour of lines, bars, markers.

```
plt.plot(x, y, color='red') # or color='#FF0000' or color='r'
```

(b) Line Style: Change the style of lines.

```
plt.plot(x, y, linestyle='--') # dashed line
# Options: '-' (solid), '--' (dashed), ':' (dotted), '-.' (dash-dot)
```

(c) Marker: Add markers at data points.

```
plt.plot(x, y, marker='o') # circle markers
# Options: 'o', 's' (square), '^' (triangle), 'D' (diamond), '*' (star)
```

(d) Line Width: Control the thickness of the line.

```
plt.plot(x, y, linewidth=2)
```

(e) Font Size: Control text size.

```
plt.title('My Plot', fontsize=16)
```

(f) Axis Limits: Set specific ranges for axes.

```
plt.xlim(0, 10)
plt.ylim(0, 100)
```

(g) Ticks and Labels: Customise tick positions and labels.

```
plt.xticks([1,2,3,4,5,6,7], ['Mon', 'Tue', 'Wed', 'Thu', 'Fri', 'Sat', 'Sun'])
```

(h) Grid: Add grid lines for readability.

```
plt.grid(True)
```

 **Note:** Combining these options makes your plot more informative and visually appealing.

Exercise Question 5

What is the purpose of a legend?

A legend is a small box on the plot that explains what each data series (line, bar, etc.) represents, using the same colours and/or markers that appear in the plot.

- Without a legend, it is difficult to distinguish between multiple datasets plotted on the same graph.
- The legend is especially important when comparing two or more datasets.

```
plt.plot(x, y1, color='blue', label='Boys')
plt.plot(x, y2, color='pink', label='Girls')
plt.legend() # Displays the legend box
```

 **Note:** The label parameter in `plt.plot()` defines what text appears in the legend for each line.

Exercise Question 6

Define Pandas Visualisation:

Pandas Visualisation refers to the built-in plotting functionality in the Pandas library that allows you to create charts directly from a DataFrame or Series without writing detailed Matplotlib code.

```
df.plot(kind='line') # Line chart
df.plot(kind='bar') # Bar chart
df.plot(kind='hist') # Histogram
df.plot(kind='scatter', x='col1', y='col2') # Scatter plot
df.plot(kind='box') # Box plot
```

 **Note:** Pandas visualisation is built on top of Matplotlib. It makes plotting very easy as you don't need to extract data manually. For example, `df.plot()` creates a line graph of all numerical columns in one line of code.

Exercise Question 7

What is open data? Name any two websites from which we can download open data.

Open data refers to data that is freely available to everyone to access, use, and share without restrictions. Governments, research institutions, and organisations publish open datasets to promote transparency and enable research.

Two popular websites for downloading open data:

1. data.gov.in – India's open government data platform. Contains datasets on agriculture, education, health, economy, etc.
2. kaggle.com – A popular platform for data science with thousands of datasets and competitions.

 **Note:** Other notable sites: archive.ics.uci.edu (UCI Machine Learning Repository), data.worldbank.org, census.gov

Exercise Question 8

Give an example of data comparison where we can use the scatter plot:

A scatter plot is ideal for showing the relationship (correlation) between two numerical variables.

Example: Comparing a student's study hours per day vs their exam score.

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

```
study_hours = [1, 2, 3, 4, 5, 6, 7, 8]
exam_scores = [40, 50, 55, 65, 70, 80, 85, 95]
plt.scatter(study_hours, exam_scores, color='blue')
plt.xlabel('Study Hours per Day')
plt.ylabel('Exam Score')
plt.title('Study Hours vs Exam Score')
plt.show()
```

 **Note:** The scatter plot would show a positive correlation – more study hours generally means higher scores. Each dot represents one student.

Exercise Question 9

Name the plot which displays the statistical summary:

The Box Plot (also called Box-and-Whisker Plot) displays a statistical summary of data including:

- Minimum value (lower whisker)
- First Quartile / Q1 (25th percentile) – bottom of the box
- Median / Q2 (50th percentile) – line inside the box
- Third Quartile / Q3 (75th percentile) – top of the box
- Maximum value (upper whisker)
- Outliers (dots beyond the whiskers)

```
df.boxplot()    # Pandas method
# OR: plt.boxplot(data)
```

Exercise Question 10

Plot ticket sales using a line plot:

Day	Tickets Sold
Monday	2000
Tuesday	2800
Wednesday	3000
Thursday	2500
Friday	2300
Saturday	2500
Sunday	1000

```
import matplotlib.pyplot as plt

days = [1, 2, 3, 4, 5, 6, 7]
day_names = ['Monday', 'Tuesday', 'Wednesday', 'Thursday', 'Friday',
'Saturday', 'Sunday']
```

```

tickets = [2000, 2800, 3000, 2500, 2300, 2500, 1000]

plt.plot(days, tickets, color='magenta', marker='o')
plt.xticks(days, day_names, rotation=45)
plt.xlabel('Day of the Week')
plt.ylabel('Tickets Sold')
plt.title('Daily Ticket Sales for the Week')
plt.tight_layout() # Adjusts layout to prevent label cutoff
plt.show()

```

 **Note:** `plt.xticks(days, day_names)` replaces the numeric values 1-7 on the x-axis with the actual day names. `rotation=45` tilts the labels for readability. `color='magenta'` changes the line colour as required.

Exercise Question 11

Bar plot for colleges and their courses:

This is a practical/project-based question requiring you to:

3. Collect real data about colleges in your city/university.
4. Store the data in a CSV file with columns like: CollegeName, ScienceCourses, CommerceCourses, HumanitiesCourses
5. Plot a grouped bar chart.

```

import pandas as pd
import matplotlib.pyplot as plt

# Read from CSV
df = pd.read_csv('colleges.csv')

# Set college name as index
df.set_index('CollegeName', inplace=True)

# Plot grouped bar chart
df.plot(kind='bar', figsize=(10, 6))
plt.title('Courses Offered by Colleges')
plt.xlabel('College Name')
plt.ylabel('Number of Courses')
plt.legend(title='Stream')
plt.xticks(rotation=45)
plt.tight_layout()
plt.show()

```

Exercise Question 12

Box plot for screen time of students:

```
import pandas as pd
import matplotlib.pyplot as plt
import numpy as np

# Sample data (replace with actual collected data)
boys_screen_time = [4, 5, 3, 6, 7, 4, 5, 8, 3, 6] # hours per day
girls_screen_time = [3, 4, 2, 5, 4, 3, 4, 6, 2, 5]

data = [boys_screen_time, girls_screen_time]

plt.boxplot(data, labels=['Boys', 'Girls'])
plt.title('Screen Time Distribution: Boys vs Girls')
plt.xlabel('Gender')
plt.ylabel('Screen Time (hours/day)')
plt.show()
```

 **Note:** The box plot will show you: median screen time, spread (variability), and any outliers (students with unusually high or low screen time) for each gender.

Exercise Question 13

Explain the findings of the boxplot of Figure 4.18:

 **Note:** Figure 4.18 in the textbook shows a boxplot for 5 subjects. Since we cannot view the actual figure, we provide a method to interpret boxplots generally. Fill in the blanks based on what you see in your textbook's Figure 4.18.

How to read the boxplot and fill the blanks:

- Median: The line inside the box is the median (Q2) for each subject.
- Highest value: The upper end of the upper whisker represents the highest (non-outlier) value.
- Lowest value: The lower end of the lower whisker is the lowest (non-outlier) value.
- Outliers: Points plotted as individual dots beyond the whiskers are outliers.
- Minimum variation: The subject whose box is the smallest (shortest box height) shows minimum variation (data is clustered close together).

Fill in your answers based on reading Figure 4.18 in the textbook.

Exercise Question 14

Histogram for minimum and maximum temperatures:

```
import pandas as pd
import matplotlib.pyplot as plt
import numpy as np
```

```
# Sample data - collect actual temperature data for your city
min_temps = [12, 14, 13, 11, 10, 12, 15, 14, 13, 12,
             11, 10, 9, 11, 13, 14, 15, 16, 14, 12,
             11, 10, 12, 13, 14, 15, 13, 12, 11, 10]
max_temps = [22, 24, 23, 21, 20, 22, 25, 24, 23, 22,
             21, 20, 19, 21, 23, 24, 25, 26, 24, 22,
             21, 20, 22, 23, 24, 25, 23, 22, 21, 20]

plt.figure(figsize=(10, 5))
plt.hist(min_temps, bins=8, alpha=0.7, color='blue', label='Min Temp')
plt.hist(max_temps, bins=8, alpha=0.7, color='red', label='Max Temp')
plt.xlabel('Temperature (°C)')
plt.ylabel('Frequency (Number of Days)')
plt.title('Temperature Distribution for the Month')
plt.legend()
plt.show()
```

 **Note:** *alpha=0.7 makes the bars slightly transparent so overlapping bars are visible. bins=8 divides data into 8 groups/intervals.*

Exercise Question 15

Class census project with questionnaire:

This is a project activity. Steps to follow:

6. Design a questionnaire with at least 5 questions. Example questions:
 - How many family members do you have? (2, 3, 4, 5+)
 - How many hours do you study per day? (0-2, 2-4, 4-6, 6+)
 - What is your attendance percentage? (below 75%, 75-90%, above 90%)
 - Do you participate in sports? (Yes/No)
 - How many hours of sleep do you get? (Less than 6, 6-8, More than 8)
7. Collect data from all students and compile into numerical/percentage form.
8. Create a CSV file and load into Pandas.
9. Create bar and scatter diagrams.

```
import pandas as pd
import matplotlib.pyplot as plt

# Load data
df = pd.read_csv('class_census.csv')

# Bar chart - example for study hours
study_counts = df['Study_Hours'].value_counts()
study_counts.plot(kind='bar', color='teal')
plt.title('Study Hours Distribution')
```

```
plt.xlabel('Study Hours per Day')
plt.ylabel('Number of Students')
plt.show()
```

Exercise Question 16

Scatter plot with population and literacy rate data:

This is a data project. Steps to follow:

10. Visit data.gov.in and search for 'Final population Totals' and 'State wise literacy rate'.
11. Download and create a combined CSV with columns: State, Population, LiteracyRate, Region.
12. Run the following code:

```
import pandas as pd
import matplotlib.pyplot as plt
import numpy as np

df = pd.read_csv('population_literacy.csv')

# Scatter plot for each region
regions = df['Region'].unique()
colors = {'East': 'blue', 'West': 'green', 'North': 'red', 'South':
'purple'}

plt.figure(figsize=(12, 8))
for region in regions:
    subset = df[df['Region'] == region]
    plt.scatter(subset['Population'], subset['LiteracyRate'],
                marker='D', # Diamond marker
                s=subset['LiteracyRate']**0.5, # Size = sqrt(literacy rate)
                label=region, color=colors.get(region, 'black'))

plt.xlabel('Population')
plt.ylabel('Literacy Rate (%)')
plt.title('Population vs Literacy Rate by Region')
plt.legend()
plt.show()

# Bar chart: average literacy rate per region
avg_literacy = df.groupby('Region')['LiteracyRate'].mean()
avg_literacy.plot(kind='bar', color='orange')
plt.title('Average Literacy Rate by Region')
plt.xlabel('Region')
plt.ylabel('Average Literacy Rate (%)')
plt.show()
```

 **Note:** *marker='D' sets the marker shape to Diamond as required. $s=\text{subset}[\text{'LiteracyRate'}]**0.5$ sets the size of each point as the square root of the literacy rate.*

CHAPTER 5: Internet and Web

This chapter covers the fundamental concepts of computer networks, the Internet, and the World Wide Web.

Exercise Question 1

Fill in the blanks:

(a) To transmit data for sharing on a network, it has to be divided into smaller chunks called _____.

Answer: Packets

 **Note:** Data is broken into smaller units called packets for efficient transmission over a network. Each packet travels independently and is reassembled at the destination.

(b) The set of rules that decide the functioning of a network is called _____.

Answer: Protocol

 **Note:** A protocol is like a common language or set of rules that all devices on a network must follow to communicate. Example: TCP/IP, HTTP, FTP.

(c) A LAN can be extended up to a distance of _____ km.

Answer: 1 km (approximately)

(d) The _____ connects a local area network to the internet.

Answer: Gateway (or Router)

(e) The _____ topology is of hierarchical nature.

Answer: Tree topology

 **Note:** Tree topology is a hierarchical arrangement where nodes are connected in a parent-child structure, like branches of a tree.

(f) _____ is a standard naming convention used for accessing resources over the Internet.

Answer: URL (Uniform Resource Locator)

(g) _____ is a collection of related web pages.

Answer: Website

(h) A _____ is a computer that provides services to other programs or computers.

Answer: Server

Exercise Question 2

Expand the following:

(a) ARPANET: Advanced Research Projects Agency NETWORK

 **Note:** ARPANET was the first operational packet-switched network, developed in the late 1960s by the US Department of Defense. It was the predecessor of the modern Internet.

(b) ISP: Internet Service Provider

 **Note:** ISP is a company that provides internet connectivity to users. Examples: BSNL, Airtel, Jio, ACT in India.

(c) URL: Uniform Resource Locator

 **Note:** URL is the complete web address used to access a specific resource on the Internet.
Example: <https://www.ncert.nic.in/textbook.php>

Exercise Question 3

Name the device for the following:

(a) It stands for Modulator Demodulator:

Answer: Modem

 **Note:** A modem converts digital data from a computer into analog signals for transmission over telephone lines, and vice versa (from analog back to digital). Modulates = converts digital to analog. Demodulates = converts analog back to digital.

(b) It regenerates the signals:

Answer: Repeater

 **Note:** Signals weaken as they travel long distances. A Repeater amplifies/regenerates the signal to extend the range of the network. Unlike a hub, it does not interpret the data.

Exercise Question 4

Differentiate between:

(a) MAN and WAN:

MAN (Metropolitan Area Network)	WAN (Wide Area Network)
Covers a city or metropolitan area.	Covers a large geographic area – states, countries, or continents.
Range: approximately 10–100 km.	Range: Unlimited (can span the globe).
Example: Cable TV network within a city.	Example: The Internet is the largest WAN.
Faster than WAN, slower than LAN.	Generally slower due to long distances.

(b) Website and Web Page:

Website	Web Page
A collection of related web pages hosted on a web server under a common domain name.	A single document/page on the web, usually written in HTML.
Identified by its domain name. Example: www.ncert.nic.in	Has its own specific URL. Example: www.ncert.nic.in/textbook.php
A website contains many web pages.	A web page is a single unit within a website.

(c) Router and Gateway:

Router	Gateway
--------	---------

Connects multiple networks and directs data packets between them.	A gateway connects networks using different protocols (serves as a 'door' between two different types of networks).
Works at the Network layer (Layer 3).	Works at all layers of the OSI model.
Used to connect networks of the same type.	Used to connect networks of different types (e.g., connecting a LAN to the Internet).

(d) Bus and Star topology:

Bus Topology	Star Topology
All nodes share a single common cable called a bus/backbone.	All nodes are connected to a central device (hub or switch) through separate cables.
Failure of the main cable brings down the entire network.	Failure of one cable affects only that node; other nodes remain connected.
Less expensive but harder to troubleshoot.	Easier to troubleshoot but requires more cable.

 **Note:** Diagram suggestion: Refer to the bus and star topology diagrams in Chapter 5 of the textbook (pages 140–142).

(e) Static and Dynamic web pages:

Static Web Pages	Dynamic Web Pages
Content remains fixed and does not change unless manually updated.	Content changes dynamically based on user input, time, or data from a database.
Written in HTML/CSS only.	Uses server-side scripting like PHP, Python, JavaScript (Node.js), etc.
Same content is shown to all users.	Different users may see different content.
Example: A simple HTML portfolio page.	Example: Facebook, Amazon (content differs per user).

Exercise Question 5

Define a network. What is the need of forming a network?

A computer network is a group of two or more computers (or devices) connected together to share resources (data, printers, internet, etc.) and to communicate with each other.

Need for a network:

- Resource Sharing: Share hardware (printers, scanners) and software across multiple computers.
- Data Communication: Send and receive emails, messages, files instantly.
- Internet Access: Shared internet connection is more economical.
- Data Centralisation: Store data on a central server for easy access and backup.
- Collaboration: Multiple users can work on the same project simultaneously.
- Cost Reduction: Fewer devices needed (e.g., one printer shared by all).

Exercise Question 6

Give any two examples of networks:

- LAN (Local Area Network): A network of computers within a school lab or office building.
 - Internet: A global network of millions of networks, connecting computers worldwide.
-

Exercise Question 7

Give any three applications on the Internet:

13. Email (Electronic Mail): Sending and receiving digital messages (e.g., Gmail, Yahoo Mail).
 14. World Wide Web (WWW): Accessing websites and browsing information using a browser.
 15. VoIP (Voice over Internet Protocol): Making voice/video calls over the internet (e.g., Zoom, WhatsApp calls).
-

Exercise Question 8

Name any two mail service providers:

- Gmail (by Google) – www.gmail.com
 - Yahoo Mail – www.mail.yahoo.com
-

Exercise Question 9

Explain VoIP:

VoIP stands for Voice over Internet Protocol. It is a technology that allows voice calls to be made over the Internet instead of traditional telephone lines.

- The voice is digitised, compressed, and sent as data packets over the Internet.
- VoIP is cheaper than traditional phone calls, especially for international calls.
- Examples: Skype, WhatsApp Calls, Google Meet, Zoom.

 **Note:** *Traditional phone calls use circuit switching (dedicated connection). VoIP uses packet switching (same as internet data), making it more efficient and cheaper.*

Exercise Question 10

What is DNS?

DNS stands for Domain Name System. It is often called the 'phone book of the Internet'.

DNS translates human-readable domain names (like www.google.com) into machine-readable IP addresses (like 142.250.190.46) so that computers can locate and connect to websites.

 **Note:** Without DNS, you would have to memorise the IP address of every website you want to visit. DNS makes the web user-friendly by letting us use easy names instead.

How it works:

16. You type www.google.com in your browser.
17. Your browser asks the DNS server for the IP address.
18. DNS returns the IP address (e.g., 142.250.190.46).
19. Your browser connects to that IP address and loads the website.

Exercise Question 11

Identify the type of topology:

(a) Each node is connected with the help of a single cable:

Answer: Bus Topology

 **Note:** In a bus topology, all devices share one long cable (the 'bus'). Data travels in both directions along this cable.

(b) Each node is connected with central switching through independent cables:

Answer: Star Topology

 **Note:** In a star topology, each device has its own dedicated cable connecting it to a central hub or switch.

Exercise Question 12

Difference between World Wide Web and Internet (for Sahil):

Sahil, here is the clear difference:

Internet	World Wide Web (WWW)
A global network of interconnected computers sharing data.	A service that runs ON the Internet – it is a collection of web pages and websites.
The 'infrastructure' or 'highway'.	The 'content' or 'vehicles' that travel on the highway.
Existed before the WWW (since 1969 as ARPANET).	Invented by Tim Berners-Lee in 1991.
Includes email, FTP, VoIP, and WWW.	Accessed using a web browser (Chrome, Firefox, Safari).
Example: Sending an email via Gmail uses the Internet.	Example: Opening www.ncert.nic.in uses the WWW.

 **Note:** Simple analogy: The Internet is like the road network (infrastructure), while the WWW is like the cars/vehicles that use those roads.

Exercise Question 13

Murugan wants to send a report with images and videos to his mentor:

Murugan can use the following Internet services to send his report:

20. Email with attachments: He can use Gmail or Yahoo Mail to send the report as an email attachment. For large files (images/videos), he should compress them first using zip/rar.
21. Cloud Storage with Link Sharing: If the files are too large for email (most email services limit attachments to 25 MB), he can upload to Google Drive, Dropbox, or OneDrive, and share a link with his mentor.
22. File Transfer Protocol (FTP): If his organisation has an FTP server, he can upload files there.

 **Note:** Best approach: Upload the report to Google Drive, share the folder with the mentor's email address, and email them the shareable link. This avoids size limitations.

Exercise Question 14

Mampi wants to advertise rural handicrafts on a social platform:

Mampi should use Social Media Networking services available on the Internet, specifically:

- Instagram – highly visual platform, perfect for showcasing handcraft images/videos. Products can be tagged and linked to an online store.
- Facebook (Meta) – allows creating a business page, running paid advertisements targeted to specific audiences.
- Pinterest – ideal for visual products like handicrafts; millions of users search for unique products here.

Why social media?

- Large user base means wide reach.
- Low-cost or free advertising options.
- Targeted ads can reach potential buyers interested in handicrafts.
- Direct interaction with customers through comments and messages.

Exercise Question 15

Ruhani wants to edit privacy settings of her browser:

Steps to access and edit browser privacy settings (using Google Chrome as an example):

23. Click on the three-dot menu (⋮) in the top-right corner of the browser.
24. Click on Settings.
25. Scroll down and click on Privacy and Security.
26. Here you can manage cookies, site permissions, content settings, and clear browsing data.

For Firefox:

27. Click the hamburger menu (☰) > Settings > Privacy & Security.

 **Note:** Privacy settings let you control what data websites can collect, block third-party cookies, manage permissions for camera/microphone/location, and clear browsing history.

Exercise Question 16

Shubham wants to install Adobe Flash Player plugin:

Steps to add a plugin/extension in a browser (Chrome example):

28. Click on the three-dot menu (:) > More Tools > Extensions.
29. Go to the Chrome Web Store (or click 'Get more extensions').
30. Search for 'Adobe Flash Player'.
31. Click 'Add to Chrome' and confirm the installation.

 **Note:** Important: Adobe Flash Player was officially discontinued on December 31, 2020. Modern browsers no longer support Flash. HTML5 has replaced Flash for video and interactive content. Shubham should tell the website owner to upgrade their content from Flash to HTML5.

Exercise Question 17

Joe typed a URL and got Error 404. Why?

Error 404 means 'Page Not Found'. This error occurs when the web server cannot find the requested web page.

Reasons why this happened:

- The URL was typed incorrectly (typo in the web address).
- The web page has been deleted or moved from its original location.
- The website has been reorganised and the page has a new URL.
- The link Joe clicked is broken/outdated.

How to avoid Error 404:

- Double-check the URL for spelling mistakes.
- Try visiting the website's homepage and navigate from there.
- Search for the content using a search engine (Google) rather than directly typing the URL.
- Check if the website has moved using the Wayback Machine (web.archive.org).

 **Note:** HTTP Status Codes to know: 404 = Not Found, 200 = OK (success), 403 = Forbidden, 500 = Internal Server Error.

CHAPTER 6: Societal Impacts

This chapter covers ethical use of technology, cyber crimes, digital footprint, intellectual property rights, and safe internet practices.

Exercise Question 1

Revaan's activity is an example of which cyber crime?

Revaan's activity is an example of: (b) Identity Theft

Justification:

Identity theft occurs when someone uses another person's identity (credentials, account, personal information) without their consent, to perform actions. Revaan used Atharv's logged-in email account to send emails, thereby impersonating Atharv. The recipients believed the emails came from Atharv, not Revaan.

 **Note:** Why not the others? Hacking involves breaking into systems, which Revaan didn't do. Cyber bullying involves harassing someone online. Plagiarism is copying someone else's work. The act of sending messages from someone else's account is identity theft/impersonation.

Exercise Question 2

What is ethically correct for Rishika to do?

Answer: (a) Inform Garvit so that he may change his password.

Justification:

The ethically correct action is to inform Garvit privately that she found a paper containing his credentials, so he can immediately change his password and secure his account. This respects his privacy and helps him protect himself.

- Option (b) is wrong: Sharing someone's password with others is a serious breach of privacy and trust.
 - Option (c) is wrong: Accessing someone's account without their permission is identity theft and a cyber crime.
-

Exercise Question 3

Shaurya's act of modifying and claiming a downloaded video is an example of:

Answer: (c) Copyright Infringement

Justification:

Shaurya downloaded a video from the internet that belongs to someone else (the original creator). He then modified it by adding text 'Prepared by Shaurya' – this falsely credits him as the creator. This is copyright infringement because:

- He did not have permission from the original creator to use, modify, or distribute the video.
- He attributed the work to himself, which is also plagiarism.

 **Note:** Fair use (option a) allows limited use of copyrighted material for educational purposes *WITHOUT* modification and without claiming authorship. Since Shaurya claimed authorship with 'Prepared by Shaurya', it's copyright infringement, not fair use.

Exercise Question 4

Which activity is NOT an example of cyber bullying?

Answer: (a) You sent an email to your friend saying 'I am sorry'.

Justification:

Sending an apology message is a positive, constructive use of digital communication – NOT cyber bullying.

- Option (b) IS cyber bullying: Sending threatening messages ('Do not try to call or talk to me') is online harassment.
- Option (c) IS cyber bullying: Creating and uploading embarrassing pictures of someone to social media is cyber bullying – it is meant to humiliate and harm.

 **Note:** Cyber bullying is the use of digital technology to repeatedly harass, threaten, or harm someone. Positive communication (like apologising) is the opposite of bullying.

Exercise Question 5

Which of Sourabh's steps is plagiarism or copyright infringement?

(a) He rephrased the paragraph from webpage 1 in his own words:

Answer: This is NOT plagiarism. Rephrasing/paraphrasing content and attributing the idea to the source is acceptable academic practice.

(b) He downloaded 3 images and made a collage from webpage 2:

Answer: This IS copyright infringement. Images on the web are protected by copyright. Downloading and using them (even in a collage) without permission violates copyright.

(c) He downloaded the 'Digital India Initiative' icon from webpage 3:

Answer: This IS likely copyright infringement. Government logos and icons are typically protected. However, if it's officially released for public use, it may not be infringement – this depends on the terms of use.

 **Note:** Key principle: Always check the licensing terms of images/content. Look for Creative Commons licenses. Government of India images are often freely usable, but always verify.

Exercise Question 6

Match the following:

Column A	Column B (Matched)
Plagiarism	Copy and paste information from the Internet into your report and then organise it.

Hacking	Breaking into computers to read private emails and other files.
Credit Card Fraud	Fakers, by offering special rewards or money prize, ask for personal information such as bank account details.
Digital Foot Print	The trail that is created when a person uses the Internet.

Exercise Question 7

You received an SMS from your bank questioning a recent transaction:

(a) Will you SMS your PIN number to the given contact number?

Answer: NO! Absolutely not.

This is a classic phishing/vishing (voice phishing) scam. Your bank NEVER asks for your PIN, OTP, or password via SMS, email, or phone call. Never share your PIN with anyone, including people claiming to be bank officials.

(b) Will you call the bank helpline number to recheck the validity of the SMS?

Answer: YES. Call the bank's official helpline number (found on their official website or the back of your debit card) to verify if the SMS is genuine.

 **Note:** Red flags for bank fraud SMS: Urgency ('immediate action required'), request for PIN/OTP, unknown phone numbers, spelling mistakes, links to suspicious URLs.

Exercise Question 8

Preeti's birthday photos situation:

(a) How could Himanshu access an image which Preeti had already deleted?

Once Preeti posted the images, Himanshu (and other followers) could have downloaded or saved them to their own devices before they were deleted. Deleting from the social networking site removes the image from the server but does NOT delete copies already saved by others.

(b) Can anybody else also access these deleted images?

Yes. Anyone who saved, screenshot, or downloaded the images before deletion can still access and share them. Once something is posted online and seen by others, you lose full control over it.

(c) Had these images not been deleted from her digital footprint?

No. The images remain part of her digital footprint. Digital footprint consists of all traces of your online activity. Even after deletion from the original site, copies may exist in screenshots, downloads, cached versions, and other people's devices.

 **Note:** Lesson: Think before you post. Once information is shared online, you cannot fully control its spread. Privacy settings help but don't guarantee 100% control.

Exercise Question 9

Email about password expiry for school WiFi:

(a) Do you find any discrepancy in this email?

Yes! Several red flags suggest this is a phishing email:

- The registered URL is 'schoolwifi.edu' but the email provides a different URL 'A+B' (suspicious/unfamiliar URL).
- Urgency: '24 hours to renew' creates panic to make you act without thinking.
- Mass distribution to all students – legitimate IT staff would handle this differently.
- Legitimate IT departments rarely ask you to submit passwords via email/link.

(b) What will happen if the student clicks the given URL?

The URL likely leads to a fake (spoofed) login page designed to look like the real school WiFi page. When the student enters their credentials, the attackers capture the username and password. This is called credential harvesting.

(c) Is the email an example of cyber crime? Which type?

Yes. This is Phishing – a type of cyber crime where attackers send fraudulent emails or messages posing as legitimate organisations to trick people into revealing sensitive information (passwords, personal data).

 **Note:** How to identify phishing: Check the sender's email address carefully, hover over links before clicking, look for spelling errors, be suspicious of urgency, and never enter passwords on pages you reached via email links.

Exercise Question 10

Which vacation research activities created a digital footprint?

- (a) Weather conditions – YES, creates a digital footprint (search history, location data accessed).
- (b) Availability of air tickets and fares – YES, creates a significant digital footprint (browsing tickets creates a profile; airlines/advertisers track this and may show targeted ads).
- (c) Places to visit – YES, creates a digital footprint (browsing history, location searches).
- (d) Best hotel deals – YES, creates a digital footprint (hotel websites track your searches).

 **Note:** All four activities create digital footprints. Your search history, clicked links, cookies placed by websites, and location data all form part of your digital footprint. This is why you often see ads for hotels/flights after searching for them!

Exercise Question 11

How to recognise if a friend is being cyber bullied:

(a) Online activities to detect cyber bullying:

- Suddenly stops using devices or seems upset/anxious after using them.
- Closes browser/switches off phone when others approach.
- Becomes withdrawn, sad, or depressed without obvious reason.
- Refuses to discuss their online activities.
- Receives unusually large number of messages that disturb them.
- Avoids attending school or social events.

(b) Provisions in IT Act 2000 (amended 2008) to combat cyber bullying:

- Section 66A (struck down, but historically dealt with offensive messages): Sending offensive or threatening electronic messages.
- Section 66C: Identity theft – stealing someone's password or credentials.
- Section 66E: Violating privacy by capturing/publishing private images.
- Section 67: Publishing obscene material in electronic form.
- Section 507 of IPC: Criminal intimidation via anonymous communication.

 **Note:** Section 66A was struck down by the Supreme Court of India in 2015 (*Shreya Singhal vs. Union of India*) as unconstitutional. However, other sections of IPC and IT Act still apply to cyber bullying.

Exercise Question 12

Differences between:

(a) Copyrights and Patents:

Copyright	Patent
Protects original creative works like books, music, films, software, and art.	Protects new inventions, processes, or products.
Automatically granted to the creator upon creation.	Must be formally applied for and registered.
Lasts for the creator's lifetime + 60 years (in India).	Usually valid for 20 years from the date of filing.
Examples: A novel, a song, a movie, a software program.	Examples: A new drug formula, a machine, a manufacturing process.

(b) Plagiarism and Copyright Infringement:

Plagiarism	Copyright Infringement
Taking someone else's ideas, words, or work and presenting them as your own without giving credit.	Using copyrighted material without permission, even if you give credit.
An ethical/academic issue – no permission or credit given.	A legal issue – using work without the owner's licence/permission.
Example: Copying an essay and submitting it as yours.	Example: Downloading and selling copies of a copyrighted movie.

(c) Non-ethical Hacking and Ethical Hacking:

Non-Ethical Hacking (Black Hat)	Ethical Hacking (White Hat)
Unauthorised access to systems for personal gain, malicious intent, or disruption.	Authorised testing of computer systems to identify vulnerabilities before malicious hackers do.
Illegal and punishable under IT Act.	Legal – done with the owner's permission and for security improvement.
Steals data, destroys systems, installs malware.	Finds weaknesses and reports them to the organisation to fix.

(d) Active and Passive digital footprints:

Active Footprint	Passive Footprint
Data you deliberately share online.	Data collected about you without your explicit knowledge.
Examples: Social media posts, emails sent, comments posted.	Examples: Websites tracking your location, cookies tracking browsing behaviour, IP address logging.
You are aware and in control of what is shared.	You may not even know it is being collected.

(e) Free software and Free and Open Source Software (FOSS):

Free Software	Free and Open Source Software (FOSS)
'Free' as in freedom (not necessarily free of cost). Users have the freedom to use, study, modify, and distribute.	Combines free software principles with open source: source code is publicly available to view, modify, and distribute.
Focuses on user freedom and ethical issues.	Focuses on practical benefits of accessible source code + user freedom.
Example: GNU/Linux software philosophy.	Example: Linux OS, Mozilla Firefox, LibreOffice.

Exercise Question 13

Steps to credit sources when using text from the web:

32. Identify the source: Note the author's name, title of the article, website name, and URL.
33. Use quotation marks: If you quote directly, put the text in quotation marks.
34. Paraphrase or summarise: Rewrite the idea in your own words (still needs citation).
35. Add a citation/reference: Include the source information in a reference list at the end of your work.
36. Use a citation format (APA, MLA, etc.):

```
Author Name. (Year). Title of article. Website Name. URL
Example: NCERT. (2020). Informatics Practices Class XII. NCERT.
https://ncert.nic.in
```

 **Note:** Simply copying text from the web without any of these steps is plagiarism. Even if you change a few words, the idea must still be attributed to the original author.

Exercise Question 14

How to find pictures in the free public domain:

Steps to find copyright-free images:

37. Use Creative Commons search: search.creativecommons.org
38. Use filter options in Google Images: Search images—Tools—Usage Rights—'Creative Commons licences'
39. Use public domain image sites: Pixabay, Unsplash, Pexels, Wikimedia Commons

How can public domain images be used without copyright violations?

- Check the specific licence. Creative Commons has different types: CC0 (no rights reserved – fully free), CC BY (attribution required), CC BY-SA (share alike), etc.
- For CC0 and public domain images: Use freely without any restrictions.
- For CC BY images: Use freely but give credit to the photographer/creator.
- Do NOT use images marked 'All Rights Reserved' without permission.

Exercise Question 15

Importance of securing wireless router at home:

Why secure your home router?

- Prevents unauthorised users from connecting to your network and stealing bandwidth.
- Protects your personal data from being intercepted by cybercriminals.
- Prevents your internet from being used for illegal activities (you would be held responsible).
- Protects your smart home devices (IoT) from being hacked.

Rules for a secure password (guidelines from security experts):

- At least 12 characters long.
- Mix of uppercase letters, lowercase letters, numbers, and special symbols (@, #, \$, etc.).
- Avoid common words, names, or predictable sequences (e.g., '123456', 'password').
- Use a passphrase (e.g., 'Blue!Elephant@Runs95').

Example imaginary password: R@1nB0w#Sky2024

Sharing the router password:

- (a) Parents: YES – they are trusted family members who need internet access.
- (b) Friends: Generally NO – friends visiting occasionally don't need the router password. Use a 'Guest Network' if your router supports it.
- (c) Neighbours: NO – no need to give neighbours access to your home network.
- (d) Home Tutors: NO (or only temporarily, using Guest Network) – tutors are not permanent residents.

 **Note:** Best practice: Create a separate Guest Network for visitors. This keeps your main network secure while allowing guests to access the internet.

Exercise Question 16

Steps to ensure (a) computer in good working condition, (b) safe internet surfing:

(a) Computer in good working condition:

40. Regularly update the operating system and software to get security patches.
41. Install and update reliable antivirus/anti-malware software.
42. Regularly clean up disk space: delete temporary files, empty recycle bin.
43. Defragment the hard drive periodically (for HDDs, not SSDs).
44. Avoid downloading software from untrustworthy sources.
45. Take regular backups of important data.

46. Keep hardware clean – remove dust from vents to prevent overheating.

(b) Smart and safe internet surfing:

47. Only visit trusted and secure websites (look for 'https://' and the padlock icon).
48. Never click on suspicious links in emails, messages, or pop-ups.
49. Don't share personal information on unknown websites.
50. Use strong, unique passwords for different accounts.
51. Enable two-factor authentication (2FA) wherever possible.
52. Use a VPN when on public Wi-Fi networks.
53. Log out of accounts when using public computers.
54. Regularly clear browser cookies and cache.

Exercise Question 17

What is data privacy? What type of information do websites collect?

Data privacy refers to the right of individuals to control how their personal information is collected, stored, used, and shared by organisations.

Types of information websites collect:

- Browsing history: Pages visited, links clicked, time spent on each page.
- Location data: GPS coordinates, city/country (through IP address).
- Device information: Browser type, operating system, screen resolution, device model.
- Cookies data: Login status, preferences, shopping cart contents.
- Search queries: What you search for on the website.
- Personal data: Name, email, phone number, address (when you fill forms).
- Financial data: Credit/debit card details (on shopping sites).



Note: Most websites have a Privacy Policy page that explains what data they collect and how they use it. Always read it before sharing personal information.

Exercise Question 18

Difference between technologies used by Sunil and Jagdish:

Sunil uses: Google Chrome browser + Google Docs

Jagdish uses: Mozilla Firefox browser + LibreOffice Writer

Feature	Sunil's Technology	Jagdish's Technology
Browser	Google Chrome (by Google – proprietary/commercial)	Mozilla Firefox (Open Source by Mozilla Foundation)
Word Processor	Google Docs (cloud-based, requires internet, commercial)	LibreOffice Writer (offline, free and open source software)
Data Storage	Data stored on Google's cloud servers.	Data stored locally on the computer.

Internet Required?	Yes – Google Docs is cloud-based.	No – LibreOffice works offline.
Cost	Both are free to use, but Google's services may monetise data.	Completely free and open source.

Exercise Question 19

Examples of cyber crimes and IT Act provisions:

(a) Identity Theft:

Example: Someone hacks into your social media account and posts messages in your name, or uses your Aadhaar/PAN number to apply for a loan.

- IT Act Section 66C: Punishment for Identity Theft – using someone else's electronic signature, password, or unique identification feature dishonestly/fraudulently. Punishment: up to 3 years imprisonment and fine up to Rs. 1 lakh.

(b) Credit Card Account Theft:

Example: You receive a fake email from your bank asking you to verify your card details. You fill in your card number and OTP, and your account is drained.

- IT Act Section 66D: Cheating by impersonation using computer resources. Punishment: up to 3 years imprisonment and fine up to Rs. 1 lakh.
- IT Act Section 43: Unauthorised access to computer systems for financial gain.

Exercise Question 20

Missing email etiquettes in Neerja's email:

Based on the question, the following email etiquettes are missing:

(a) Subject of the mail: The email should clearly state its purpose in the subject line (e.g., 'Computer Science Project Submission – Neerja, Class XI').

(b) Formal greeting: The email should start with a respectful salutation like 'Respected Ma'am' or 'Dear Ms. [Teacher's Name]'.

(c) Self-explanatory terms: The body of the email should clearly mention what is attached and any necessary context, not just attach a file with no explanation.

(d) Identity of the sender: Neerja should sign the email with her full name, class, roll number, so the teacher knows who sent it.

(e) Regards: The email should end formally with 'Yours sincerely' or 'With regards' followed by her name.

 **Note:** A properly formatted email: Subject: Project Submission. Greeting: Dear Ma'am. Body: Please find attached my project on [topic]. Closing: Yours sincerely, Neerja (Class XI, Roll No. X)

Exercise Question 21

Health hazards of inappropriate and excessive laptop use for Sumit's father:

Points to discuss with Sumit:

55. Eye Strain (Computer Vision Syndrome): Staring at a screen for long hours causes dry eyes, headaches, and blurred vision. Use the 20-20-20 rule: every 20 minutes, look at something 20 feet away for 20 seconds.
56. Bad Posture and Back Pain: Using a laptop in poor positions (on bed, sofa) causes neck pain, back pain, and spinal problems. Use a proper desk and chair.
57. Repetitive Strain Injuries: Long hours of typing and touchpad use can cause carpal tunnel syndrome (wrist pain).
58. Sleep Disruption: Blue light from screens suppresses melatonin production, making it harder to sleep. Avoid screens 1 hour before bed.
59. Sedentary Lifestyle: Excessive laptop use means less physical activity, leading to weight gain, poor cardiovascular health.
60. Mental Health Issues: Excessive use (especially social media) can cause anxiety, depression, and social isolation.
61. Radiation and Heat: Keeping a laptop on your lap for long periods exposes you to heat and electromagnetic radiation.

Safe practices:

- Take breaks every 30-45 minutes.
 - Use a laptop stand and external keyboard/mouse.
 - Set screen time limits using parental control apps.
 - Balance laptop use with outdoor activities and exercise.
-