

Sample Question Paper-3

SECTION-A

1. False

Why? `loc[]` can access both rows and columns using label-based indexing. Example: `df.loc[2, 'column_name']` accesses a specific cell.

Why not? The statement says "only rows" — that's incorrect because `loc` can handle both.

2. (a) 11

Why? 'Informatics' has 11 characters (I-n-f-o-r-m-a-t-i-c-s).

Why not?

(b) 10 → would mean miscounting letters.

(c) 9 → too short.

(d) 12 → too long.

3. (c) Intellectual Property Rights

Why? Distributing paid software without permission infringes the copyright and violates IPR.

Why not?

(a) Cyber Security Policy → relates to securing systems, not copyright theft.

(b) Data Protection → about safeguarding personal data, not piracy.

(d) Privacy Rights → about personal information, not software copying.

4. (b) `index=False`

Why? `df.to_csv('file.csv', index=False)` prevents index from being written.

Why not? b,c,d → Syntax error.

5. (c) Switch

Why? Switch works at Data Link layer (Layer 2) and forwards frames to the MAC address of the intended device.

Why not?

(a) Hub → broadcasts to all devices.

(b) Modem → works at Physical/Data Link for internet access, not switching.

(d) Repeater → regenerates signals.

6. (a) `ROUND()`

Why? `ROUND(7.8)` directly returns 8 by rounding off.

Why not?

(b) `FLOOR()` → returns 7.

(c) `POWER()` AND `MOD()` → modulus function, irrelevant.

7. (c) A poem written by a student

Why? Original literary works are copyright-protected.

Why not?

(a) Slogan → usually protected under trademark, not copyright.

(b) Algorithm → not copyrightable (can be patented).

(d) Company name → trademark.

8. (c) `series.index`

Why? `index` returns index labels.

Why not?

(a) `items()` → returns (index, value) pairs iterator.

(b) `labels` → no such attribute.

(d) `head()` → returns first few rows.

9. (b) Any attribute or combination that uniquely identifies a row

Why? Candidate key = any possible unique identifier (before primary key is chosen).

Why not?

(a) Only one column → not necessarily true.

(c) Only primary key → candidate keys are chosen before deciding PK.

(d) Only foreign key → FK is not unique.

10. (b) VoIP

Why? VoIP = Voice over Internet Protocol.

Why not?

(a) FTP → file transfer.

(c) SMTP → email sending.

(d) HTTP → web communication.

11. (c) Average of non-NULL salaries

Why? AVG ignores NULL values.

Why not?

(a) Max salary → use `MAX()`.

(b) Sum → use `SUM()`.

(d) Count → use `COUNT()`.

12. (c) `fillna()`

Why? `fillna(value)` replaces NaN with given value.

Why not?

(a) `fill_value()` → not a Series method.

(b) `replace()` → works but not specifically for NaN.

(d) `dropna()` → removes NaN values.

13. (b) Unauthorised access to someone's computer
Why? Unauthorised access is explicitly punishable under IT Act, 2000.

Why not?

- (a) Installing licensed software → legal.
 (c) Opening email account → legal.
 (d) Browsing educational sites → legal.

14. (c) ORDER BY column name DESC
Why? Standard SQL syntax.

Why not?

- (a) ORDER DESC → invalid.
 (b) SORT DOWN → invalid in SQL.
 (d) GROUP DESC → invalid.

15. (c) First 3 rows of df
Why? `iloc[:3]` selects rows at positions 0,1,2.

Why not?

- (a) Last 3 rows → need `iloc[-3:]`.
 (b) From index 3 to end → need `iloc[3:]`.
 (d) Only row with index 3 → `iloc[3]`.

16. (d) Star
Why? In Star topology, central hub failure disconnects all devices.

Why not?

- (a) Mesh → multiple connections, no single failure point.
 (b) Tree → has hierarchy but redundancy possible.
 (c) Ring → failure breaks loop, but not central node.

17. (c) TRIM()
Why? TRIM removes both leading and trailing spaces.

Why not? REMOVE(), STRIP(), CUT() → not standard SQL functions.

18. (b) `df.describe()`
Why? `describe()` gives mean, std, min, max, etc.

Why not?

- (c) `df.info()` → structure & data types.
 (d) `df.stats()`, `df.summary()` → not Pandas methods.

19. (c) MIN()
Why? MIN returns smallest value.
Why not?
 (a) LOW() → no such function.
 (b) SMALL() → in Excel, not SQL.
 (d) LEAST() → returns smallest among expressions, not column values.

20. (a) Both A and R are True, and R correctly explains A.

Why? Assertion: `df.loc[2]` accesses row with label 2 — True.

Reason: `loc` is label-based, `iloc` is integer-position based — True, and explains A.

21. (c) A is True, R is False

Why? Assertion: UPDATE modifies table structure — False (ALTER does that).

Reason: UPDATE is DML for updating records — True.

SECTION-B

22. (a) **Method 1:** From a List

```
import pandas as pd
data = [10, 20, 30]
s = pd.Series(data)
print(s)
```

Method 2: From a Dictionary

```
import pandas as pd
data = {'Maths': 90, 'Science': 85}
s = pd.Series(data)
print(s)
```

OR

(b)

Library	Use
Pandas	For data manipulation and analysis using DataFrames.
NumPy	For numerical computations and handling arrays.
Matplotlib	For data visualisation using graphs and charts.
Seaborn	For statistical data visualisation built on Matplotlib.

23. Plagiarism is the act of using someone else's work, ideas or content without giving proper credit, presenting it as your own.

To avoid plagiarism:

Always give credit to original authors using citations.

Use plagiarism detection tools.

Paraphrase and write in your own words.

Include references or bibliography.

24. (i) Extract 'Intel'

```
SELECT SUBSTRING('Artificial Intelligence', 12, 5);
```

- (ii) Convert to uppercase

```
SELECT UPPER('Artificial Intelligence');
```

25. (a) A web browser is a software application used to access and display content from the World Wide Web (e.g., Chrome, Firefox). A search engine,

on the other hand, is a website that helps users find web pages by indexing content across the internet (e.g., Google, Bing).

Browsers display the results; search engines provide links to relevant sites.

The two work together: users enter queries into search engines, which run inside browsers.

OR

- (b) A firewall is a security system—either hardware, software or both—that monitors and controls incoming and outgoing network traffic based on predetermined rules.

It acts as a barrier between trusted internal networks and untrusted external networks (like the internet).

Firewalls help prevent unauthorised access to or from private networks, making systems safer from malware or hacking attempts.

For example, corporate networks use firewalls to filter harmful web traffic.

26. A Foreign Key is a column in one table that refers to the Primary Key of another table. It is used to establish a relationship between two tables.

It helps maintain referential integrity by ensuring that the value in the foreign key column must exist in the referenced primary key column.

For example, in an Orders table, CustomerID can be a foreign key referencing the CustomerID in the Customers table.

If a customer is deleted, this constraint prevents orphan records in the Orders table.

27. Two good cyber hygiene practices are:

Use Strong and Unique Passwords: Use a mix of characters and avoid reusing passwords across platforms.

Regular Software Updates: Keep antivirus and OS updated to patch security vulnerabilities.

Practicing these ensures protection from malware, phishing attacks and unauthorised access to accounts.

25. (a) import pandas as pd

```
data = { 'Name': ['Amit', 'Riya', 'Kunal'], 'Age': [15, 14, 18]} #
Error 1: unequal list lengths
df = pd.DataFrame(data) # Error
2: wrong function name (should be
DataFrame with capital D)
print(df)
```

OR

- (b) s = pd.Series(data, index=cities)

SECTION-C

29. (i) Empathy and civic responsibility
(ii) Practical environmental education
(iii) NGOs like Saahas, Toxics Link or Ecoreco
30. (a) import pandas as pd

```
data = [
    {"City": "Delhi", "Population": 19800000},
    {"City": "Mumbai", "Population": 20400000},
    {"City": "Bangalore", "Population": 12300000},
    {"City": "Kolkata", "Population": 14600000}]
df = pd.DataFrame(data)
print(df)
```

OR

- (b) import pandas as pd

```
data = {
    "Apple": "Red",
    "Banana": "Yellow",
    "Kiwi": "Green"
}
s = pd.Series(data)
print(s)
```

31. CREATE TABLE COURSES (
CourseID INTEGER,
CourseName VARCHAR(40),
Duration INTEGER
);
ALTER TABLE COURSES ADD Fees FLOAT(8,2);
INSERT INTO COURSES VALUES (201, 'Python
Programming', 60, 15000.00);
32. (i) SELECT Subject, AVG(Marks) AS Avg_Marks
FROM MARKS GROUP BY Subject;
(ii) SELECT S.Name, S.City FROM STUDENTS S
JOIN MARKS M ON S.RollNo = M.RollNo
WHERE M.Subject = 'IP' AND M.Marks > 85;
(iii) SELECT City, COUNT(*) AS No_of_Students
FROM STUDENTS GROUP BY City;

SECTION-D

33. Statement-1: matplotlib.pyplot
Statement-2: students, hours
Statement-3: xlabel
Statement-4: 'Study Hours per Week'
34. (a) (i) SELECT COUNT(*) FROM SCHOOLBUS
GROUP BY TRANSPORTER;
(ii) SELECT AVG(CHARGES) FROM
SCHOOLBUS WHERE CHARGES> 60000
GROUP BY TRANSPORTER;
(iii) SELECT SUM(NOOFSTUD) FROM
SCHOOLBUS GROUP BY TRANSPORTER;
(iv) SELECT MAX(CAPACITY) FROM
SCHOOLBUS GROUP BY TRANSPORTER;

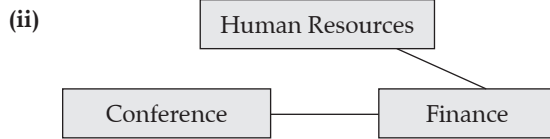
OR

(b) (i)	COUNT (Product)
	6

(ii)	SUM (Price * Qty)
	34,000
(iii)	LEFT (Product,4)
	FOUN
	NIGH
(iv)	MAX (PRICE)
	2,100

SECTION-E

35. (i) Finance block because it has the maximum number of computers.



- (iii) Satellite link
(iv) Switch
(v) LAN

36. (i) `print(df_books[df_books['Price'] > 300])`

- (ii) `df_books.drop('Author', axis=1, inplace=True)`
(iii) `print(df_books.head(3))`
(iv) `df_books.rename(columns={'Price': 'Cost'}, inplace=True)`
(v) `print(df_books['Title'])`

37. (a) (i) `SELECT RIGHT(Registration_Number, 3) FROM Vehicles;`
(ii) `SELECT UPPER(OwnerName) FROM Vehicles;`
(iii) `SELECT COUNT(*) FROM Vehicles;`
(iv) `SELECT LENGTH(Registration_Number) FROM Vehicles;`
(v) `SELECT TRIM(OwnerName) FROM Vehicles;`

OR

- (b) (i) `SELECT MAX(Salary) FROM Employees;`
(ii) `SELECT AVG(Salary) FROM Employees;`
(iii) `SELECT COUNT(*) FROM Employees WHERE Salary > 40000;`
(iv) `SELECT LOWER(Name) FROM Employees;`
(v) `SELECT COUNT(*) FROM Employees;`

□□□