

# CUET-UG Information Practices Sample Paper-11

Duration: 1 Hour

Maximum Marks: 250

## Instructions

- This paper contains a total of 50 Multiple Choice Questions.
- Each correct answer carries **+5 marks**.
- Each incorrect answer carries **-1 mark**.
- No negative marking for unattempted questions.

**Q1.** A database administrator wants to ensure that no two records in the 'Project' table have the same combination of 'ProjID' and 'ClientCode'. Which type of constraint should be applied if 'ProjID' alone is not unique across the table?

- (A) Primary Key on ProjID  
 (B) Composite Primary Key on (ProjID, ClientCode)  
 (C) Foreign Key on ClientCode  
 (D) Check Constraint on ProjID

**Q2.** Match the SQL String functions in List I with their descriptions in List II:

List I	Function	List II	Description
(i)	LTRIM()	(1)	Returns the position of first occurrence of substring
(ii)	INSTR()	(2)	Removes leading spaces
(iii)	SUBSTR()	(3)	Returns number of characters
(iv)	LENGTH()	(4)	Extracts a portion of string

- (A) i-2, ii-1, iii-4, iv-3  
 (B) i-1, ii-2, iii-3, iv-4  
 (C) i-3, ii-4, iii-1, iv-2  
 (D) i-2, ii-3, iii-4, iv-1



- Q3.** Which of the following SQL queries will correctly display the day name for the date '2023-12-25'?
- (A) `SELECT DAY('2023-12-25');`
  - (B) `SELECT DAYNAME('2023-12-25');`
  - (C) `SELECT DATE('2023-12-25');`
  - (D) `SELECT DAYOFWEEK('2023-12-25');`
- Q4.** Rearrange the following statements to correctly describe the process of a Projection ( $\Pi$ ) operation in Relational Algebra:
1. The operation returns a new relation.
  2. Duplicate rows are automatically removed from the result.
  3. The user specifies a subset of attributes (columns) to be retained.
  4. The operation is applied to an existing relation R.
- (A) 4-3-1-2
  - (B) 3-4-2-1
  - (C) 1-2-3-4
  - (D) 4-1-3-2
- Q5.** Case Study: A data analyst is working with a Pandas DataFrame 'df' containing columns: 'Region', 'Year', and 'Sales'.
- Statement 1: The command '`df.loc[0:5, 'Sales']`' will include the row at index label 5.
- Statement 2: The command '`df.iloc[0:5, 2]`' will include the row at integer position 5.
- (A) Both Statement 1 and Statement 2 are correct.
  - (B) Both Statement 1 and Statement 2 are incorrect.
  - (C) Statement 1 is correct but Statement 2 is incorrect.
  - (D) Statement 1 is incorrect but Statement 2 is correct.
- Q6.** Which of the following identifies a device on a network using a 48-bit hexadecimal address that is hard-coded into the Network Interface Card (NIC)?



- (A) IP Address
- (B) MAC Address
- (C) SSID
- (D) Port Address

**Q7.** Consider the string 'Database'. Match the SQL functions in List I with their respective outputs in List II:

List I	Function	List II	Output
(i)	LENGTH	(1)	'ta'
(ii)	SUBSTR(S,3,2)	(2)	8
(iii)	INSTR(S,'a')	(3)	'database'
(iv)	LCASE	(4)	2

- (A) i-2, ii-1, iii-4, iv-3
- (B) i-1, ii-2, iii-3, iv-4
- (C) i-2, ii-4, iii-1, iv-3
- (D) i-3, ii-2, iii-1, iv-4

**Q8.** Match the SQL Date Functions in List I with their Return Types in List II:

List I	Date Function	List II	Return Type
(i)	NOW()	(1)	Current Date only
(ii)	CURDATE()	(2)	Current Date and Time
(iii)	SYSDATE()	(3)	Name of the month
(iv)	MONTHNAME()	(4)	Current Date and Time

- (A) i-2, ii-1, iii-4, iv-3
- (B) i-1, ii-2, iii-3, iv-4
- (C) i-4, ii-3, iii-2, iv-1



(D) i-3, ii-2, iii-1, iv-4

**Q9.** Rearrange the following statements to correctly describe the property of a Primary Key:

1. It must contain unique values.
2. A table can have multiple primary keys.
3. It cannot contain NULL values.
4. It is used to uniquely identify a tuple.

(A) 1, 3, and 4 are correct

(B) 1, 2, and 4 are correct

(C) 2, 3, and 4 are correct

(D) All are correct

**Q10.** Rearrange the following Relational Algebra operations based on whether they are Unary or Binary:

1. Selection ( $\sigma$ )
2. Projection ( $\pi$ )
3. Union ( $\cup$ )
4. Cartesian Product ( $\times$ )

(A) 1, 2 are Unary; 3, 4 are Binary

(B) 1, 3 are Unary; 2, 4 are Binary

(C) 3, 4 are Unary; 1, 2 are Binary

(D) All are Binary

**Q11.** Which of the following statements about Candidate Keys are correct? 1. It is a minimal super key. 2. All primary keys are candidate keys. 3. A table can have only one candidate key. 4. It uniquely identifies a row.

(A) 1, 2, and 4

(B) 1, 3, and 4



- (C) 2 and 3 only
- (D) 1 and 4 only

**Q12.** Consider the following statements about the 'Selection' operator in Relational Algebra: 1. It is denoted by symbol Sigma ( $\sigma$ ). 2. It yields a vertical subset of a relation. 3. It yields a horizontal subset of a relation. 4. It filters rows based on a given condition.

- (A) 1, 3, and 4 are correct
- (B) 1, 2, and 4 are correct
- (C) 2, 3, and 4 are correct
- (D) 1 and 4 only

**Q13.** In networking, which of the following describes a Star Topology? 1. All nodes are connected to a central hub. 2. Failure of one node affects the entire network. 3. Hub failure brings down the whole network. 4. It requires more cabling than Bus topology.

- (A) 1, 3, and 4
- (B) 1, 2, and 3
- (C) 2, 3, and 4
- (D) 1 and 3 only

**Q14.** Identify the correct statements regarding network devices: 1. A Hub is an intelligent device that sends data to specific ports. 2. A Switch is a unicast device. 3. A Router connects different networks. 4. A Modem performs modulation and demodulation.

- (A) 2, 3, and 4
- (B) 1, 2, and 3
- (C) 1, 3, and 4
- (D) 1 and 4 only



- Q15.** Which of the following are true regarding MAC and IP addresses? 1. MAC address is 48-bit. 2. IP address is assigned by the manufacturer. 3. IPv4 is 32-bit. 4. MAC address works at the Data Link Layer.
- (A) 1, 3, and 4  
(B) 1, 2, and 3  
(C) 2, 3, and 4  
(D) 1 and 4 only
- Q16.** A network administrator needs to connect two different offices located in different cities. Which type of network would be most suitable for this purpose?
- (A) LAN (Local Area Network)  
(B) MAN (Metropolitan Area Network)  
(C) WAN (Wide Area Network)  
(D) PAN (Personal Area Network)
- Q17.** Given are Two Statements, Choose Correct:  
Statement 1: A Series in Pandas is a one-dimensional labeled array capable of holding data of any type.  
Statement 2: A Series is size-mutable but value-immutable. Select the correct option:
- (A) Both Statement 1 and Statement 2 are correct.  
(B) Both Statement 1 and Statement 2 are incorrect.  
(C) Statement 1 is correct but Statement 2 is incorrect.  
(D) Statement 1 is incorrect but Statement 2 is correct.
- Q18.** Which of the following attributes is used to check the dimensions of a Pandas DataFrame?
- (A) df.size  
(B) df.shape  
(C) df.ndim



(D) `df.values`

**Q19.** Consider a DataFrame 'df'. Which of the following code snippets will correctly display the first 3 rows and the first 2 columns using the 'iloc' method?

(A) `df.iloc[0:3, 0:2]`

(B) `df.iloc[1:3, 1:2]`

(C) `df.iloc[:2, :3]`

(D) `df.iloc[3, 2]`

**Q20.** Given are Two Statements, Choose Correct:

Statement 1: The 'loc' method in Pandas is label-based data selecting method.

Statement 2: When slicing with 'loc', the endpoint is excluded. Select the correct option:

(A) Both Statement 1 and Statement 2 are correct.

(B) Both Statement 1 and Statement 2 are incorrect.

(C) Statement 1 is correct but Statement 2 is incorrect.

(D) Statement 1 is incorrect but Statement 2 is correct.

**Q21.** Which Pandas function is used to load data from a Comma Separated Values (CSV) file into a DataFrame?

(A) `pandas.load_csv()`

(B) `pandas.open_csv()`

(C) `pandas.read_csv()`

(D) `pandas.get_csv()`

**Q22.** To change the name of an existing column in a DataFrame 'df', which method is used?

(A) `df.rename()`

(B) `df.column_name()`

(C) `df.change()`



(D) `df.update()`

**Q23.** Which of the following is NOT an attribute of a Pandas DataFrame?

- (A) T
- (B) empty
- (C) columns
- (D) length

**Q24.** In a Pandas DataFrame, how can you delete a column named 'Salary'?

- (A) `df.drop('Salary', axis=0)`
- (B) `del df['Salary']`
- (C) `df.remove('Salary')`
- (D) `df.delete('Salary')`

**Q25.** What is the output of the following code?

```
import pandas as pd
s = pd.Series([10, 20, 30], index=['a', 'b', 'c'])
print(s['b'])
```

- (A) 10
- (B) 20
- (C) 30
- (D) Error

**Q26.** Match the DataFrame operations in List I with their descriptions in List II:

List I	Operation	List II	Description
(i)	<code>df.info()</code>	(1)	Summary of statistics
(ii)	<code>df.describe()</code>	(2)	First few rows
(iii)	<code>df.head()</code>	(3)	General information and Dtypes
(iv)	<code>df.tail()</code>	(4)	Last few rows



- (A) i-3, ii-1, iii-2, iv-4
- (B) i-1, ii-3, iii-4, iv-2
- (C) i-2, ii-4, iii-1, iv-3
- (D) i-4, ii-2, iii-3, iv-1

**Q27.** To add a new row to a DataFrame 'df', which of the following is commonly used?

- (A) `df.add_row()`
- (B) `df.append()(ordf.loc[index])`
- (C) `df.insert()`
- (D) `df.new()`

**Q28.** Which attribute of a DataFrame returns a tuple representing the number of rows and columns?

- (A) `df.size`
- (B) `df.shape`
- (C) `df.ndim`
- (D) `df.count()`

**Q29.** What does the 'inplace=True' parameter do in many Pandas methods like `drop()` or `rename()`?

- (A) It creates a copy of the DataFrame.
- (B) It modifies the original DataFrame without returning a new one.
- (C) It ignores the operation.
- (D) It displays the result.

**Q30.** Which of the following creates a Series from a dictionary?

- (A) `pd.Series('a':1, 'b':2)`
- (B) `pd.Series(['a', 'b'], [1, 2])`



- (C) pd.Series(1, 2)
- (D) pd.Series(dict=True)

**Q31.** Consider the following SQL query:

```
SELECT Department, AVG(Salary) FROM Employees GROUP BY Department  
HAVING AVG(Salary) > 50000;
```

Which of the following is true about the 'HAVING' clause in this query?

- (A) It filters individual rows before grouping.
- (B) It filters groups based on an aggregate condition.
- (C) It is interchangeable with the 'WHERE' clause.
- (D) It sorts the output in ascending order.

**Q32.** In SQL, which clause is used to arrange the result-set in descending order?

- (A) SORT BY DESC
- (B) ORDER BY DESC
- (C) GROUP BY DESC
- (D) ARRANGE BY DESC

**Q33.** Consider a SQL table 'Orders' with columns (OrderID, CustomerID, Amount).

What is the result of:

```
SELECT CustomerID, SUM(Amount) FROM Orders WHERE Amount > 500  
GROUP BY CustomerID HAVING SUM(Amount) > 2000;
```

- (A) It calculates the total amount for all customers whose individual orders are all > 500.
- (B) It calculates the total of orders > 500 for each customer, then filters for total sums > 2000.
- (C) It filters for customers with total sales > 2000, then checks which orders were > 500.
- (D) It returns an error because WHERE and HAVING cannot be used in the same query.



- Q34.** Identify the correct sequence of clauses in a SELECT statement:
- (A) SELECT, FROM, GROUP BY, WHERE, HAVING, ORDER BY  
 (B) SELECT, FROM, WHERE, GROUP BY, HAVING, ORDER BY  
 (C) SELECT, WHERE, FROM, GROUP BY, ORDER BY, HAVING  
 (D) SELECT, FROM, WHERE, HAVING, GROUP BY, ORDER BY
- Q35.** Fill in the blank: To find the total number of records in a table excluding NULL values in a specific column 'Comm', we use \_\_\_\_\_.
- (A) COUNT(\*)  
 (B) SUM(Comm)  
 (C) COUNT(Comm)  
 (D) TOTAL(Comm)
- Q36.** Which aggregate function in SQL is used to find the highest value in a column?
- (A) HIGH()  
 (B) MAXIMUM()  
 (C) TOP()  
 (D) MAX()
- Q37.** Match the SQL Clauses in List I with their functionalities in List II:

List I	SQL Clause	List II	Functionality
(i)	DISTINCT	(1)	Pattern matching
(ii)	AS	(2)	Aliasing
(iii)	LIKE	(3)	Unique values
(iv)	IN	(4)	Check multiple values

- (A) i-3, ii-2, iii-1, iv-4  
 (B) i-1, ii-2, iii-3, iv-4



(C) i-2, ii-3, iii-4, iv-1

(D) i-3, ii-1, iii-2, iv-4

**Q38.** Which type of Join returns only the rows that have matching values in both tables?

(A) Left Join

(B) Full Join

(C) Inner Join

(D) Right Join

**Q39.** What is the output of `SELECT COUNT(*) FROM Table;` if the table has 5 rows and one row contains a NULL value?

(A) 4

(B) 5

(C) 0

(D) Error

**Q40.** Which Matplotlib function is used to create a bar chart?

(A) `plt.plot()`

(B) `plt.pie()`

(C) `plt.bar()`

(D) `plt.hist()`

**Q41.** In Matplotlib, which function is used to add a label to the X-axis?

(A) `plt.xname()`

(B) `plt.xlabel()`

(C) `plt.labelx()`

(D) `plt.axes(x)`



- Q42.** Which chart type is best suited for showing the distribution of a continuous dataset by dividing it into bins?
- (A) Pie Chart
  - (B) Line Chart
  - (C) Bar Chart
  - (D) Histogram
- Q43.** To display the legend in a plot, which method must be called?
- (A) `plt.showilegend()`
  - (A) `plt.legend()`
  - (B) `plt.display()`
  - (C) `plt.label()`
- Q44.** Which parameter in `plt.bar()` is used to change the color of the bars?
- (A) `col`
  - (B) `colors`
  - (C) `color`
  - (D) `bcolor`
- Q45.** Intellectual Property Rights (IPR) include which of the following? 1. Patents, 2. Trademarks, 3. Copyrights, 4. Plagiarism
- (A) 1, 2, and 3 only
  - (B) 1, 2, and 4 only
  - (C) 2, 3, and 4 only
  - (D) All of the above
- Q46.** Which term describes the trail of data you leave behind when using the internet?
- (A) Cyber Trail
  - (B) Digital Footprint



- (C) Internet Shadow
- (D) Data Cookie

**Q47.** Stealing someone else's work or ideas and passing them off as your own is known as:

- (A) Phishing
- (B) Hacking
- (C) Plagiarism
- (D) Spamming

**Q48.** Which of the following is an example of E-waste?

- (A) Used plastic bottles
- (B) Old newspapers
- (C) Discarded mobile phones
- (D) Expired medicines

**Q49.** Which license allows anyone to use, modify, and distribute the software for free?

- (A) Proprietary
- (B) Open Source
- (C) Copyrighted
- (D) Shared

**Q50.** The process of converting readable text into an unreadable format to protect data is:

- (A) Decryption
- (B) Encryption
- (C) Formatting
- (D) Masking



**Detailed Solutions****Q1.****Solution****Concept:**

In relational database management systems, a Primary Key is used to uniquely identify each record in a table. While a Primary Key often consists of a single column, there are scenarios where a single attribute is insufficient to guarantee uniqueness. In such cases, two or more columns can be combined to form a single Primary Key, which is then referred to as a Composite Primary Key. This ensures that while individual columns may contain duplicate values, the combination of those values across the specified columns remains unique for every row.

**Solution:**

- (a) The problem states that 'ProjID' alone is not unique. This means multiple rows can share the same Project ID, possibly because a project is split across different clients or phases.
- (b) The goal is to ensure the "combination" of 'ProjID' and 'ClientCode' is unique.
- (c) Option (A) is incorrect because 'ProjID' contains duplicates and thus violates the Primary Key constraint if applied alone.
- (d) Option (C) is used to establish a relationship between tables, not to enforce uniqueness within the current table.
- (e) Option (D) is used to limit the range of values in a column (e.g., Salary > 0) but does not handle multi-column uniqueness.
- (f) Option (B), the Composite Primary Key, is specifically designed for this scenario. By defining the key as (ProjID, ClientCode), the database will allow a ProjID to repeat and a ClientCode to repeat, but it will prevent the exact same pair from being entered twice.

**Final Answer:** The correct constraint is a Composite Primary Key on (ProjID, ClientCode).

**Answer: (B)**



Q2.

**Solution****Concept:**

SQL provides a variety of scalar functions to manipulate string data. These functions allow developers to clean data (removing whitespace), search within text (finding positions), extract specific segments (substrings), and measure the physical size of the data. Mastering these functions is essential for data preprocessing and effective querying in Informatics Practices.

**Solution:**

- (a) (i) LTRIM(): The "L" stands for Left. This function is used to trim or remove any leading spaces from the left side of a string. Matching with (2).
- (b) (ii) INSTR(): Short for "In String," this function searches for a substring within a main string and returns the integer position of its first occurrence. Matching with (1).
- (c) (iii) SUBSTR(): This is the "Substring" function. It is used to extract a specific portion of a string starting from a given index for a specified length. Matching with (4).
- (d) (iv) LENGTH(): This function calculates and returns the total number of characters present in a string, including spaces and special characters. Matching with (3).
- (e) Following this logic: i-2, ii-1, iii-4, iv-3.

**Final Answer:** The correct mapping is i-2, ii-1, iii-4, iv-3.

**Answer: (A)**



Q3.

**Solution****Concept:**

SQL date functions are categorized based on what they return: some return numeric values, some return date objects, and others return descriptive strings. For reporting and dashboarding, it is often necessary to convert a date into a human-readable format, such as the name of the day (e.g., 'Monday') rather than just the date or the day number.

**Solution:**

- (a) The user wants to display the "day name" for '2023-12-25'.
- (b) DAY() returns the day of the month as a number (1-31). For this date, it would return 25.
- (c) DATE() simply extracts the date part from a datetime expression. It would return '2023-12-25'.
- (d) DAYOFWEEK() returns the index of the day (e.g., 1 for Sunday, 2 for Monday in some systems).
- (e) DAYNAME() is the specific SQL function designed to return the full name of the day for a given date. Since Dec 25, 2023, was a Monday, this function would return 'Monday'.

**Final Answer:** The correct query is `SELECT DAYNAME('2023-12-25');`

**Answer: (B)**



Q4.

**Solution****Concept:**

Projection ( $\Pi$ ) is a unary operation in relational algebra used to select a subset of columns from a relation. It performs a vertical slicing of the table. A key characteristic of the Projection operator in theoretical relational algebra is that it automatically eliminates duplicate rows from the resulting relation to maintain the mathematical definition of a "set," where all elements must be unique.

**Solution:**

- (a) To perform a projection, we first identify the source of the data. Thus, the operation must be applied to an existing relation R (Statement 4).
- (b) Next, the specific columns or attributes that the user wants to keep must be identified (Statement 3).
- (c) Once the operation is executed, the process yields a result, which is a new relation containing only those columns (Statement 1).
- (d) Finally, as a fundamental rule of relational algebra, if the removal of columns results in rows that now look identical, duplicate rows are removed (Statement 2).
- (e) The logical flow follows the sequence 4 (Source)  $\rightarrow$  3 (Selection of attributes)  $\rightarrow$  1 (Result creation)  $\rightarrow$  2 (Duplicate handling).

**Final Answer:** The correct sequence is 4-3-1-2.

**Answer:** (A)



Q5.

**Solution****Concept:**

Pandas offers two primary methods for slicing data: `loc` and `iloc`. The `loc` indexer is label-based, meaning it uses the names of the rows and columns. Crucially, when slicing with `loc`, the stop index is **\*\*inclusive\*\***. On the other hand, `iloc` is integer-location based, referring to the zero-based position of the data. In standard Python fashion, the stop index in `iloc` is **\*\*exclusive\*\***.

**Solution:**

- (a) Statement 1: `df.loc[0:5, 'Sales']` uses label-based slicing. In Pandas, `loc` includes the endpoint. Therefore, the row with the label '5' will be included in the output. Statement 1 is correct.
- (b) Statement 2: `df.iloc[0:5, 2]` uses integer-based slicing. For `iloc`, the range follows the `[start:stop)` rule, where the stop index is excluded. This means it will return rows at positions 0, 1, 2, 3, and 4. Position 5 is excluded.
- (c) Therefore, Statement 2 is incorrect because it claims to include the row at integer position 5.

**Final Answer:** Statement 1 is correct, but Statement 2 is incorrect.

**Answer: (C)**



Q6.

**Solution****Concept:**

In computer networking, every device requires a unique identifier to communicate. There are two primary types of addresses: Logical (IP Address) and Physical (MAC Address). The Physical address, or Media Access Control address, is a unique identifier assigned to a Network Interface Card (NIC) by the manufacturer. It is considered "hard-coded" or "burned-in" because it is a permanent hardware address that does not change regardless of which network the device joins.

**Solution:**

- (a) The question specifies three key characteristics: 48-bit length, hexadecimal format, and hard-coded into the NIC.
- (b) **IP Address:** This is a logical address (32-bit for IPv4 or 128-bit for IPv6) assigned by software or a network administrator. It changes based on the network location.
- (c) **SSID:** This stands for Service Set Identifier, which is simply the technical name for a Wi-Fi network. It is not a device address.
- (d) **Port Address:** This identifies a specific process or service on a device (like port 80 for HTTP) and is a 16-bit number.
- (e) **MAC Address:** This matches all criteria. It is 48 bits long, usually displayed as six pairs of hexadecimal digits (e.g., 00:1A:2B:3C:4D:5E), and is permanently assigned to the hardware.

**Final Answer:** The device is identified by its MAC Address.

**Answer: (B)**



Q7.

**Solution****Concept:**

String functions in SQL are used to analyze and manipulate text. These functions operate on "strings" of characters to return specific data points like length, location of a character, or a modified version of the text itself. In SQL, indexing typically starts at 1, which is a common point of confusion for students coming from Python, where indexing starts at 0.

**Solution:**

- (a) **(i) LENGTH:** This counts all characters in 'Database'. D-a-t-a-b-a-s-e equals 8 characters. Matching with (2).
- (b) **(ii) SUBSTR(S,3,2):** This starts at the 3rd character ('t') and takes 2 characters forward. This results in 'ta'. Matching with (1).
- (c) **(iii) INSTR(S,'a'):** This searches for the first occurrence of 'a'. The first 'a' is at the second position in 'Database'. Matching with (4).
- (d) **(iv) LCASE:** This converts the entire string to lowercase. 'Database' becomes 'database'. Matching with (3).
- (e) Following this logic, the mapping is i-2, ii-1, iii-4, iv-3.

**Final Answer:** The correct mapping is i-2, ii-1, iii-4, iv-3.

**Answer:** (A)



Q8.

**Solution****Concept:**

SQL Date and Time functions are essential for managing temporal data. A critical distinction is made between functions that return the date alone and those that include time stamps. Functions like NOW() and SYSDATE() are dynamic and reflect the current system clock, whereas CURDATE() is used when the specific time of day is irrelevant to the query.

**Solution:**

- (a) **(i) NOW():** This returns the current date and time in 'YYYY-MM-DD HH:MM:SS' format. Matching with (2).
- (b) **(ii) CURDATE():** Short for Current Date, it returns only the date in 'YYYY-MM-DD' format. Matching with (1).
- (c) **(iii) SYSDATE():** Like NOW(), it returns the current system date and time. Matching with (4).
- (d) **(iv) MONTHNAME():** This takes a date value and extracts the name of the month as a string (e.g., 'December'). Matching with (3).
- (e) The resulting sequence is i-2, ii-1, iii-4, iv-3.

**Final Answer:** The correct mapping is i-2, ii-1, iii-4, iv-3.

**Answer:** (A)



Q9.

**Solution****Concept:**

A Primary Key is a constraint that uniquely identifies each record in a database table. It has two strict requirements: uniqueness and non-nullability.

**Solution:**

- (a) Statement 1: "It must contain unique values." This is correct as it prevents duplicate records.
- (b) Statement 2: "A table can have multiple primary keys." This is incorrect. A table can have only ONE primary key (though it can be a composite key made of multiple columns).
- (c) Statement 3: "It cannot contain NULL values." This is correct; every record must have a valid, known primary key.
- (d) Statement 4: "It is used to uniquely identify a tuple." This is the fundamental definition of a primary key.

**Final Answer:** Statements 1, 3, and 4 are correct.

**Answer: (A)**

Q10.

**Solution****Concept:**

Relational Algebra operators are classified by the number of relations (tables) they operate on. Unary operators act on a single relation, while Binary operators require two relations.

**Solution:**

- (a) Selection ( $\sigma$ ): This filters rows from a single table based on a condition. It is a Unary operator.
- (b) Projection ( $\pi$ ): This chooses specific columns from a single table. It is a Unary operator.
- (c) Union ( $\cup$ ): This combines rows from two different tables. It is a Binary operator.
- (d) Cartesian Product ( $\times$ ): This combines every row of one table with every row of another. It is a Binary operator.

**Final Answer:** 1 and 2 are Unary; 3 and 4 are Binary.

**Answer: (A)**



Q11.

**Solution****Concept:**

Relational Algebra is a procedural query language. The 'Selection' operator ( $\sigma$ ) and 'Projection' operator ( $\pi$ ) are the most fundamental unary operations used to filter data from a single relation.

**Solution:**

- (a) Statement 1: "It is denoted by symbol Sigma ( $\sigma$ )." This is correct;  $\sigma$  is the standard notation for selection.
- (b) Statement 2: "It yields a vertical subset of a relation." This is incorrect. Projection ( $\pi$ ) yields a vertical subset (columns), while Selection yields a horizontal subset (rows).
- (c) Statement 3: "It yields a horizontal subset of a relation." This is correct as it extracts specific rows.
- (d) Statement 4: "It filters rows based on a given condition." This is the primary purpose of the selection operator.

**Final Answer:** Statements 1, 3, and 4 are correct.

**Answer: (A)**

Q12.

**Solution****Concept:**

Candidate keys and Primary keys are part of relational database integrity constraints. A Candidate Key is a minimal set of attributes that can uniquely identify a tuple.

**Solution:**

- (a) Statement 1: "It is a minimal super key." This is the formal definition of a candidate key.
- (b) Statement 2: "All primary keys are candidate keys." This is correct; the primary key is simply the candidate key chosen by the database designer.
- (c) Statement 3: "A table can have only one candidate key." This is incorrect. A table can have multiple candidate keys (e.g., Roll No and Enrollment No).
- (d) Statement 4: "It uniquely identifies a row." This is correct.

**Final Answer:** Statements 1, 2, and 4 are correct.

**Answer: (A)**



Q13.

**Solution****Concept:**

Network topology refers to the layout of a network. Star topology is one of the most common layouts where every node is connected to a central device like a hub or switch.

**Solution:**

- (a) Statement 1: "All nodes are connected to a central hub." This is the defining characteristic of a Star topology.
- (b) Statement 2: "Failure of one node affects the entire network." This is incorrect. In a star network, if one node (workstation) fails, the rest of the network continues to function.
- (c) Statement 3: "Hub failure brings down the whole network." This is correct; the central hub is a single point of failure.
- (d) Statement 4: "It requires more cabling than Bus topology." This is correct because every node needs its own dedicated cable to the hub.

**Final Answer:** Statements 1, 3, and 4 are correct.

**Answer: (A)**

Q14.

**Solution****Concept:**

Connecting devices facilitate data flow in a network. They operate at different layers of the OSI model and vary in "intelligence" regarding how they handle data packets.

**Solution:**

- (a) Statement 1: "A Hub is an intelligent device..." This is incorrect. A Hub is a "dumb" device that broadcasts data to all ports regardless of the destination.
- (b) Statement 2: "A Switch is a unicast device." This is correct; it uses MAC addresses to send data only to the specific intended port.
- (c) Statement 3: "A Router connects different networks." This is correct; routers manage traffic between separate networks (like a LAN and the Internet).
- (d) Statement 4: "A Modem performs modulation and demodulation." This is correct; it converts digital signals to analog and vice versa.

**Final Answer:** Statements 2, 3, and 4 are correct.

**Answer: (A)**



Q15.

**Solution****Concept:**

Addressing in networking involves physical (MAC) and logical (IP) addresses. MAC addresses are hardcoded into the hardware, while IP addresses are assigned software-wise to identify a device on a network.

**Solution:**

- (a) Statement 1: "MAC address is 48-bit." This is correct; it is usually represented as 6 pairs of hexadecimal digits.
- (b) Statement 2: "IP address is assigned by the manufacturer." This is incorrect. The MAC address is assigned by the manufacturer (burned into the NIC), while the IP address is assigned by the network administrator or ISP.
- (c) Statement 3: "IPv4 is 32-bit." This is correct; it consists of four octets (e.g., 192.168.1.1).
- (d) Statement 4: "MAC address works at the Data Link Layer." This is correct (Layer 2 of the OSI model).

**Final Answer:** Statements 1, 3, and 4 are correct.

**Answer:** (A)



Q16.

**Solution****Concept:**

Computer networks are categorized based on their geographical spread and scale. A Local Area Network (LAN) covers a small area like a room or building. A Metropolitan Area Network (MAN) covers a city-wide area. A Wide Area Network (WAN) spans across cities, countries, or even continents, usually utilizing public or leased telecommunication lines.

**Solution:**

- (a) The scenario describes connecting two different offices located in "different cities."
- (b) A LAN is restricted to a very limited range, typically within a single building or a campus, so it cannot span across cities.
- (c) A MAN (Metropolitan Area Network) connects locations within the same city but usually does not extend to different cities.
- (d) A PAN (Personal Area Network) is for personal devices within a range of a few meters (like Bluetooth).
- (e) A WAN (Wide Area Network) is specifically designed to handle long-distance communication and connect networks across vast geographical distances, such as between different cities. Therefore, it is the only appropriate choice for this professional networking requirement.

**Final Answer:** The most suitable network for connecting offices in different cities is a WAN.

**Answer:** (C)



Q17.

**Solution****Concept:**

In Python's Pandas library, a Series is a fundamental data structure. It is essentially a one-dimensional array that holds data and has associated labels called an index. Understanding mutability is crucial: "value-mutable" means the data inside can change, while "size-mutable" means the length of the structure can change.

**Solution:**

- (a) Statement 1 states that a Series is a one-dimensional labeled array. This is the textbook definition of a Pandas Series. It can hold integers, strings, floating-point numbers, or Python objects. Thus, Statement 1 is correct.
- (b) Statement 2 discusses mutability. In Pandas, a Series is **value-mutable** (you can change an element at a specific index) but **size-immutable** (you cannot generally change the number of elements in a Series after it is created without creating a new object).
- (c) A DataFrame, by contrast, is both value-mutable and size-mutable.
- (d) Since Statement 2 claims a Series is size-mutable but value-immutable (the exact opposite of reality), Statement 2 is incorrect.

**Final Answer:** Statement 1 is correct, but Statement 2 is incorrect.

**Answer:** (C)



Q18.

**Solution****Concept:**

Pandas DataFrames have several attributes that provide metadata about the object. 'shape' returns a tuple representing dimensions (rows, columns). 'size' returns the total number of elements (rows  $\times$  columns). 'ndim' returns the number of axes/dimensions (which is always 2 for a DataFrame).

**Solution:**

- (a) The question asks for the attribute used to check the dimensions (specifically how many rows and columns exist).
- (b) `df.size` returns an integer representing the total count of cells (e.g., if a DF is 5x2, size is 10).
- (c) `df.ndim` simply returns the number 2 because a DataFrame is by definition two-dimensional.
- (d) `df.values` returns the data as a Numpy ndarray.
- (e) `df.shape` is the specific attribute used to retrieve a tuple like (5, 2), indicating 5 rows and 2 columns. This is the standard way to check the "shape" or dimensions of the table.

**Final Answer:** The correct attribute is `df.shape`.

**Answer: (B)**



Q19.

**Solution****Concept:**

The `.iloc` indexer is used for integer-location based indexing/selection by position. It follows Python's standard slicing rules: `start:stop`, where the 'start' index is inclusive and the 'stop' index is exclusive.

**Solution:**

- (a) The goal is to select the first 3 rows (indices 0, 1, 2) and the first 2 columns (indices 0, 1).
- (b) For rows: To get 0, 1, and 2, the slice must be `0:3` because 3 is excluded.
- (c) For columns: To get 0 and 1, the slice must be `0:2` because 2 is excluded.
- (d) Syntax for `iloc` is `df.iloc[rowslice, columnslice]`.
- (d) `df.iloc[0:3, 0:2]` perfectly fulfills this requirement.
- (e) Option (C) `df.iloc[:2, :3]` would select 2 rows and 3 columns, which is the inverse of the requirement.

**Final Answer:** The correct snippet is `df.iloc[0:3, 0:2]`.

**Answer:** (A)

Q20.

**Solution****Concept:**

While `iloc` uses integer positions, `loc` is a label-based indexer. This means you use the actual names of the rows and columns. A very important distinction between the two is how they handle the end point of a slice.

**Solution:**

- (a) Statement 1: "The 'loc' method... is label-based." This is correct. If your row index has labels like 'Row1', 'Row2', you use those strings within `loc`.
- (b) Statement 2: "When slicing with 'loc', the endpoint is excluded." This is **incorrect**.
- (c) Unlike standard Python slicing and `iloc`, the `loc` indexer **includes** both the start and the end labels in the result. For example, `df.loc['A': 'C']` will include row 'A', row 'B', and row 'C'.
- (d) This design choice in Pandas was made because, with labels, it is often more intuitive to include the final item specified.

**Final Answer:** Statement 1 is correct, but Statement 2 is incorrect.

**Answer:** (C)



Q21.

**Solution****Concept:**

The process of data ingestion is a critical step in data analysis using Pandas. Comma Separated Values (CSV) files are the most common format for storing tabular data in a plain-text format. Pandas provides a robust, highly customizable function to parse these files and convert them into a structured DataFrame object, handling headers, delimiters, and data types automatically.

**Solution:**

- (a) The task is to identify the specific Pandas function used to import CSV data.
- (b) In the Pandas library, naming conventions usually follow a `read_[format]` pattern for importing data and `to_[format]` for exporting data.
- (c) `pandas.load_csv()` and `pandas.open_csv()` are not valid functions within the Pandas namespace; they are common misnomers derived from other programming languages or general file handling logic.
- (d) `pandas.read_csv()` is the official and correct function. It allows for numerous parameters like `sep` (delimiter), `header` (row number for column names), and `index_col` (column to use as the row labels).
- (e) `pandas.get_csv()` is likewise non-existent in the standard library.

**Final Answer:** The correct function is `read_csv()`.

**Answer: (C)**



Q22.

**Solution****Concept:**

Metadata management in DataFrames often requires modifying column labels to make them more descriptive or to fix formatting issues. Pandas provides the `rename()` method, which is highly flexible because it can rename both row indices and column headers simultaneously using a dictionary-based mapping.

**Solution:**

- (a) Changing a column name requires a method that maps the 'old' name to the 'new' name.
- (b) The `df.rename()` method is the standard tool for this. It typically takes a dictionary in the format `columns={'OldName': 'NewName'}`.
- (c) It is important to note that `rename()` returns a new DataFrame by default unless the `inplace=True` parameter is used to modify the existing object.
- (d) Options like `df.column_name()`, `df.change()`, or `df.update()` do not exist for the purpose of renaming headers in the Pandas API.
- (e) Another way to rename all columns at once is to assign a new list to the `df.columns` attribute, but for specific column changes, `rename()` is the preferred expert approach.

**Final Answer:** The method used to change column names is `df.rename()`.

**Answer:** (A)



Q23.

**Solution****Concept:**

DataFrame attributes provide information about the structure, metadata, and state of the data without actually performing a calculation or transformation. Attributes are accessed without parentheses, unlike methods which are functions that perform actions.

**Solution:**

- (a) `df.T` (Transpose) is a valid attribute that flips the DataFrame over its main diagonal, switching rows and columns.
- (b) `df.empty` is a boolean attribute that returns `True` if the DataFrame has no data (zero elements), which is useful for error checking in automation.
- (c) `df.columns` is a valid attribute that returns the index object containing all column labels.
- (d) `df.length` is NOT a valid attribute of a DataFrame. To find the "length" or number of rows in a DataFrame, Python's built-in `len(df)` function is used, or the `df.shape[0]` attribute is accessed.
- (e) Since the question asks for what is NOT an attribute, `length` is the correct choice.

**Final Answer:** `length` is not a valid attribute of a Pandas DataFrame.

**Answer: (D)**



Q24.

**Solution****Concept:**

In data cleaning, removing unnecessary or redundant features (columns) is a frequent task. In Python and Pandas, there are multiple ways to achieve this, including the `del` keyword for in-place deletion and the `drop()` method for more controlled removal.

**Solution:**

- (a) The question asks how to delete a column named 'Salary'.
- (b) Option (A) `df.drop('Salary', axis=0)` is incorrect because `axis=0` refers to rows. To delete a column using `drop()`, you must specify `axis=1`.
- (c) Option (B) `del df['Salary']` is a standard Python keyword that performs an in-place deletion of the column from the DataFrame. This is a very common and efficient way to remove a single column.
- (d) Options (C) and (D) using `.remove()` or `.delete()` are not valid methods in the Pandas DataFrame API; `.remove()` is typically used for Python lists.
- (e) Therefore, `del` is the correct syntax among the choices provided.

**Final Answer:** The correct way to delete the column is `del df['Salary']`.

**Answer: (B)**

Q25.

**Solution****Concept:**

Accessing data in a Series involves using the index labels. When a Series is created with a custom index (like 'a', 'b', 'c'), those labels become the keys to access the corresponding data values, much like a dictionary.

**Solution:**

- (a) The code defines a Series `s` with values `[10, 20, 30]` and labels `['a', 'b', 'c']`.
- (b) This creates the following mapping: 'a' → 10, 'b' → 20, 'c' → 30.
- (c) The command `print(s['b'])` specifically requests the value associated with the label 'b'.
- (d) Looking at our mapping, the value corresponding to 'b' is 20.
- (e) Since the label exists, no error is thrown, and the integer value 20 is returned as the output.

**Final Answer:** The output of the code is 20.

**Answer: (B)**



Q26.

**Solution****Concept:**

In the Python Pandas library, exploratory data analysis is facilitated by several built-in methods designed to provide a snapshot of a DataFrame's structure, content, and statistical distribution. Understanding the distinction between structural metadata (information about the container) and statistical summaries (information about the data values) is fundamental for any data scientist or analyst.

**Solution:**

- (a) **(i) df.info():** This method is used to print a concise summary of a DataFrame. It provides essential metadata, including the index dtypes and columns, non-null values, and memory usage. It tells you "what the table is made of" rather than "what the data says." This matches with **(3) General information and Dtypes.**
- (b) **(ii) df.describe():** This method generates descriptive statistics that summarize the central tendency, dispersion, and shape of a dataset's distribution, excluding NaN values. For numeric data, it provides the mean, standard deviation, min, max, and quartiles. This matches with **(1) Summary of statistics.**
- (c) **(iii) df.head():** This function returns the first  $n$  rows for the object based on position. It is useful for quickly testing if your object has the right type of data in it after a loading or transformation step. By default, it returns the first 5 rows. This matches with **(2) First few rows.**
- (d) **(iv) df.tail():** This function returns the last  $n$  rows for the object based on position. It is useful for verifying data at the end of a dataset, especially after sorting or appending operations. By default, it returns the last 5 rows. This matches with **(4) Last few rows.**

**Logical Mapping:** \* (i)  $\rightarrow$  (3) \* (ii)  $\rightarrow$  (1) \* (iii)  $\rightarrow$  (2) \* (iv)  $\rightarrow$  (4)

Comparing this to the given options, Option (A) correctly represents this mapping.

**Final Answer:** The correct mapping is i-3, ii-1, iii-2, iv-4.

**Answer: (A)**



Q27.

**Solution****Concept:**

Adding data to an existing structure is a fundamental operation in Pandas. While there are multiple ways to increase the size of a DataFrame, the approach depends on whether you are adding a single row, multiple rows, or combining two different DataFrame objects. Understanding the difference between label-based assignment and method-based expansion is key to efficient coding.

**Solution:**

- (a) The question explores the syntax for adding a new row to an existing DataFrame 'df'.
- (b) In modern Pandas, the most common and direct way to add a row is using the label-based indexer `df.loc[index]`. For example, `df.loc['new_row'] = [val1, val2]` adds a row at the specified label.
- (c) The `df.append()` method was historically used to add rows, though it has been deprecated in recent Pandas versions in favor of `pd.concat()`. However, in many academic contexts and older versions, it is still cited as a primary way to add rows.
- (d) `df.insert()` is specifically used to add a new **column** at a specific location, not a row.
- (e) `df.add_row()` and `df.new()` are not valid methods in the Pandas library.
- (f) Given the options provided in competitive exams, `df.loc` or the historical `append` are the recognized answers.

**Final Answer:** To add a new row, `df.loc[index]` or `df.append()` is used.

**Answer: (B)**



Q28.

**Solution****Concept:**

In the context of the Pandas library, attributes provide immediate metadata about the object's geometry. The dimensionality of a table—the count of rows and the count of columns—is the most basic piece of information required for loops, condition checking, and ensuring data alignment. This is represented by a Python tuple structure.

**Solution:**

- (a) The objective is to identify the attribute that specifically returns a tuple containing the row count and column count.
- (b) `df.size` is an attribute that returns the total number of elements in the DataFrame (Rows  $\times$  Columns). For a 10x3 table, size would return the integer 30.
- (c) `df.ndim` returns an integer representing the number of axes. Since a DataFrame is a 2D structure, this always returns 2.
- (d) `df.count()` is a method (not an attribute) that returns the number of non-NA/null observations for each column or row.
- (e) `df.shape` is the correct attribute. It returns a tuple in the format (rows, columns). For instance, a DataFrame with 100 rows and 4 columns would have a shape of (100, 4).

**Final Answer:** The attribute is `df.shape`.

**Answer: (B)**



Q29.

**Solution****Concept:**

By default, most Pandas operations (like `drop`, `rename`, or `fillna`) return a new object and leave the original DataFrame untouched. This "functional" approach prevents accidental data loss. However, for memory efficiency or code brevity, the `inplace` parameter allows the user to decide whether to create a copy or modify the current object directly.

**Solution:**

- (a) The `inplace` parameter is a boolean flag (True or False) found in many DataFrame methods.
- (b) When `inplace=False` (the default), the method performs the action and returns a **copy** of the modified DataFrame. You must assign it back to a variable (e.g., `df = df.drop(...)`) to save changes.
- (c) When `inplace=True`, the method performs the operation directly on the original object 'df' and returns None. This means the original data is permanently changed without needing an assignment statement.
- (d) This is particularly useful when working with very large datasets where creating multiple copies of the data would exhaust the system's RAM.

**Final Answer:** It modifies the original DataFrame without returning a new one.

**Answer: (B)**



Q30.

**Solution****Concept:**

Pandas is designed to integrate seamlessly with standard Python data structures. A dictionary, which consists of key-value pairs, is a natural fit for creating a Series. When a dictionary is passed to the Series constructor, the dictionary keys automatically become the Series index labels, and the dictionary values become the data elements.

**Solution:**

- (a) To create a Series from a dictionary, you pass the dictionary object directly into the `pd.Series()` constructor.
- (b) In the example `pd.Series({'a':1, 'b':2})`, 'a' and 'b' serve as the index, while 1 and 2 serve as the values. This is valid syntax.
- (c) Option (B) `pd.Series(['a', 'b'], [1, 2])` would create a Series using two lists (data and index), but it is not "from a dictionary."
- (d) Options (C) and (D) are syntactically incorrect for the Pandas API.
- (e) Creating Series from dictionaries is highly useful because it maintains the logical mapping of labels to data during the instantiation process.

**Final Answer:** `pd.Series({'a':1, 'b':2})` creates a Series from a dictionary.

**Answer:** (A)



Q31.

**Solution****Concept:**

In SQL, data filtration can happen at different stages of the query execution. While the WHERE clause is used to filter individual records before any grouping occurs, the HAVING clause is designed to filter the results after the GROUP BY clause has aggregated the data into buckets. This is particularly useful when you need to apply conditions based on aggregate functions like SUM(), AVG(), or COUNT().

**Solution:**

- (a) The query provided is: `SELECT Department, AVG(Salary) FROM Employees GROUP BY Department HAVING AVG(Salary) > 50000;`
- (b) Step 1: The database identifies the Employees table.
- (c) Step 2: The database groups the employees based on their Department.
- (d) Step 3: For each department group, the average salary is calculated.
- (e) Step 4: The HAVING clause acts on these calculated averages. It checks which groups (departments) have an average salary greater than 50,000.
- (f) If a department's average is 45,000, that entire group is excluded from the final output.
- (g) Therefore, HAVING does not look at individual employees; it looks at the summary data of the groups.

**Final Answer:** The 'HAVING' clause filters groups based on an aggregate condition.

**Answer: (B)**



Q32.

**Solution****Concept:**

Sorting is the process of arranging data in a specific logical order—either numerical or alphabetical. In SQL, the `ORDER BY` clause is used for this purpose. By default, `ORDER BY` sorts in ascending order (ASC). To reverse this and sort from highest to lowest or Z to A, a specific keyword must be appended to the column name.

**Solution:**

- (a) The question asks for the clause and keyword used to arrange results in descending order.
- (b) `SORT BY` is used in some other languages (like HiveQL or specific reporting tools), but it is not standard SQL for this purpose.
- (c) `GROUP BY` is used for aggregation, not for sorting the final result-set.
- (d) `ORDER BY` is the standard SQL clause for sorting.
- (e) To specify the direction, we use `ASC` for ascending and `DESC` for descending.
- (f) Thus, the complete syntax is `ORDER BY [ColumnName] DESC`.

**Final Answer:** The correct clause is `ORDER BY DESC`.

**Answer: (B)**



Q33.

**Solution****Concept:**

To understand this query, one must master the **SQL Order of Execution**. Although we write queries starting with **SELECT**, the database engine processes them in a specific logical sequence. The two most critical filtering clauses are **WHERE** and **HAVING**. The **WHERE** clause acts as a "pre-filter" (filtering individual rows before they are grouped), while the **HAVING** clause acts as a "post-filter" (filtering the summarized groups after aggregation).

**Solution:**

The database engine executes the query in the following logical steps:

- (a) **FROM Orders:** The engine identifies the source table.
- (b) **WHERE Amount > 500:** This is the first filter. The engine scans the table and discards any individual order where the Amount is 500 or less. Only "large" orders remain in the temporary result set.
- (c) **GROUP BY CustomerID:** The remaining "large" orders are sorted into buckets based on the CustomerID.
- (d) **SUM(Amount):** The engine calculates the total sum of the "large" orders for each specific customer bucket.
- (e) **HAVING SUM(Amount) > 2000:** This is the second filter. It looks at the total sums calculated in the previous step. If a customer's total sum of large orders does not exceed 2000, that entire customer group is removed from the results.
- (f) **SELECT:** Finally, the CustomerID and their respective SUM(Amount) are displayed for the qualifying customers.

**Analysis of Options:** \* Option (A) is incorrect because it implies all orders must be > 500, whereas the query simply ignores those ≤ 500. \* Option (C) is incorrect because it reverses the logical flow of filtering. \* Option (D) is incorrect because **WHERE** and **HAVING** are frequently used together to filter rows and groups respectively.

**Final Answer:** The query calculates the total of orders > 500 for each customer, then filters for total sums > 2000.

**Answer: (B)**



Q34.

**Solution****Concept:**

SQL queries follow a very strict syntax order. Even though the database engine might execute parts of the query in a different logical order (like FROM being processed first), the user must write the clauses in the standard sequence defined by the SQL language specification.

**Solution:**

- (a) Every standard query begins with SELECT (to choose columns) and FROM (to choose the table).
- (b) Next comes WHERE, which filters the raw rows from the table.
- (c) After filtering, the rows are grouped using GROUP BY.
- (d) Once groups are formed, the HAVING clause is applied to filter those groups.
- (e) Finally, the resulting dataset is sorted using ORDER BY.
- (f) Any deviation from this order (e.g., putting WHERE after GROUP BY) will result in a syntax error.

**Final Answer:** The correct sequence is SELECT, FROM, WHERE, GROUP BY, HAVING, ORDER BY.

**Answer: (B)**



Q35.

**Solution****Concept:**

Aggregate functions like COUNT() are used to tally records. However, there is a distinct difference between COUNT(\*) and COUNT(column\_name). COUNT(\*) counts every single row in the table, regardless of whether the columns contain data or NULLs. In contrast, COUNT(column\_name) specifically ignores any row where that particular column has a NULL value.

**Solution:**

- (a) The question specifies that we need to count records but **\*\*exclude NULL values\*\*** in a specific column named 'Comm'.
- (b) COUNT(\*) is incorrect because it counts all rows, including those with NULLs in 'Comm'.
- (c) SUM(Comm) is incorrect because it adds up the values instead of counting the number of records.
- (d) TOTAL(Comm) is not a standard SQL aggregate function.
- (e) COUNT(Comm) will look at every row, check the 'Comm' column, and only increment the count if the value is not NULL.

**Final Answer:** To exclude NULLs in a specific column, we use COUNT(Comm).

**Answer:** (C)



Q36.

**Solution****Concept:**

Aggregate functions in SQL are used to perform a calculation on a set of values and return a single result. For numerical or string-based data, finding the extremum (the highest or lowest value) is a common requirement. SQL provides specific functions for this: `MAX()` for the maximum value and `MIN()` for the minimum value.

**Solution:**

- (a) The objective is to identify the standard SQL function that retrieves the largest value from a column.
- (b) `HIGH()` and `MAXIMUM()` are not valid keywords in the standard SQL aggregate function library; they are conceptually descriptive but syntactically incorrect.
- (c) `TOP()` is a clause used in some SQL dialects (like T-SQL/SQL Server) to limit the number of rows returned, but it is not an aggregate function that computes the highest value of a set.
- (d) `MAX()` is the universal SQL aggregate function. It operates on a column and returns the highest value, ignoring `NULL` values in its calculation.

**Final Answer:** The correct aggregate function is `MAX()`.

**Answer: (D)**



Q37.

**Solution****Concept:**

SQL utilizes various clauses and operators to refine data selection and improve the readability of the output. These range from eliminating duplicates and renaming columns to performing complex pattern matching within strings and checking for inclusion within a specific set of values.

**Solution:**

- (a) (A) **DISTINCT**: This keyword is used in a **SELECT** statement to filter out duplicate records, ensuring only unique values are returned. Matches (III).
- (b) (B) **AS**: This is the "Alias" keyword. It is used to give a table or a column a temporary name, making the output more readable or easier to reference in joins. Matches (II).
- (c) (C) **LIKE**: This operator is used in a **WHERE** clause to search for a specified pattern in a column using wildcards like % and \_. Matches (I).
- (d) (D) **IN**: This operator allows you to specify multiple values in a **WHERE** clause, acting as a shorthand for multiple **OR** conditions. Matches (IV).

**Final Answer:** The correct mapping is (A)-(III), (B)-(II), (C)-(I), (D)-(IV).

**Answer: (A)**

Q38.

**Solution****Concept:**

In relational databases, an 'Inner Join' is the most common type of join. It creates a new result-set by combining column values of two tables based on a join-predicate (usually a matching ID). Unlike outer joins, which may include rows that do not have a match, an inner join is exclusive to intersection.

**Solution:**

- (a) The question asks for the join that returns only rows with matches in both tables.
- (b) A **Left Join** returns everything from the left table regardless of a match.
- (c) A **Right Join** returns everything from the right table regardless of a match.
- (d) A **Full Join** returns all rows when there is a match in either the left or the right table.
- (e) An **Inner Join** only includes rows where the join condition is met in both the left and right tables. If a row in table A has no matching row in table B, it is omitted from the result.

**Final Answer:** The join that returns only matching values is the Inner Join.

**Answer: (C)**



Q39.

**Solution****Concept:**

Understanding the behavior of the COUNT() function is vital for accurate data reporting. There is a specific behavioral difference between COUNT(\*) and COUNT(column\_name) when it comes to handling rows that contain NULL values.

**Solution:**

- (a) The table in question has 5 rows in total.
- (b) One of those rows contains a NULL value.
- (c) The query is `SELECT COUNT(*) FROM Table;`
- (d) In SQL, the asterisk (\*) indicates that the function should count every row that exists in the table, irrespective of the content of the individual columns or the presence of NULLs.
- (e) Because the table has 5 physical rows, the count will be 5. If the query had been `COUNT(column_name)`, and that column contained the NULL, the result would have been 4.

**Final Answer:** The output of the query is 5.

**Answer: (B)**



Q40.

**Solution****Concept:**

Matplotlib is a comprehensive library for creating static, animated, and interactive visualizations in Python. Its `pyplot` module provides a MATLAB-like interface for plotting. Different functions are dedicated to different types of charts, such as line plots, scatter plots, histograms, and bar charts.

**Solution:**

- (a) The requirement is to identify the function used specifically for creating bar charts.
- (b) `plt.plot()` is the general-purpose function used primarily for creating line charts or scatter plots.
- (c) `plt.pie()` is specifically used to generate pie charts to show proportions.
- (d) `plt.hist()` is used for histograms to show frequency distributions of continuous data.
- (e) `plt.bar()` is the dedicated function for creating vertical bar charts, which are used to compare categorical data or show changes over time.

**Final Answer:** The correct function for a bar chart is `plt.bar()`.

**Answer: (C)**



Q41.

**Solution****Concept:**

Visualizing data effectively requires clearly labeled axes so that the viewer can understand what the data points represent. In the Matplotlib library, the `pyplot` module provides simple, intuitive functions to set the labels for both the horizontal (X) and vertical (Y) axes. These functions are essential for transforming a simple line or bar into a professional and informative chart.

**Solution:**

- (a) The question asks for the specific function used to add a label to the X-axis of a plot.
- (b) `plt.xname()` is not a valid Matplotlib function; it is likely a distractor based on general naming conventions.
- (c) `plt.labelx()` is also incorrect and does not exist in the Matplotlib API.
- (d) `plt.axes(x)` is part of a more complex command structure used for creating or managing axes objects, but it is not used directly for simple labeling.
- (e) `plt.xlabel()` is the correct and standard function. It takes a string as an argument (e.g., `plt.xlabel("Time")`) and places that text centered below the X-axis of the current figure.

**Final Answer:** The correct function is `plt.xlabel()`.

**Answer: (B)**



Q42.

**Solution****Concept:**

Choosing the right chart type is fundamental to data communication. While bar charts are used for categorical data (discrete categories), histograms are designed for continuous numerical data. A histogram groups data into "bins" (ranges of values) and shows the frequency or count of data points falling within each bin, allowing the observer to see the distribution (shape) of the data.

**Solution:**

- (a) The question focuses on a "continuous dataset" and the use of "bins" for distribution.
- (b) A Pie Chart shows parts of a whole (proportions) and is unsuitable for continuous distributions.
- (c) A Line Chart is best for showing trends over time or continuous intervals.
- (d) A Bar Chart is best for comparing discrete categories (e.g., Sales by Region).
- (e) A Histogram is specifically designed to handle continuous data. For example, if you have the ages of 1,000 people, a histogram will group them (e.g., 0-10, 11-20) and show how many people fall into each age range.

**Final Answer:** The chart type best suited for continuous distribution and bins is the Histogram.

**Answer: (D)**



Q43.

**Solution****Concept:**

When multiple data series are plotted on the same chart, a legend is necessary to distinguish between them. In Matplotlib, the `label` parameter is first defined within the plotting function (like `plt.plot(x, y, label="Sales")`), but the legend itself will not appear on the canvas until a specific command is called to render it.

**Solution:**

- (a) The task is to identify the command that triggers the display of the legend.
- (b) `plt.show_legend()` and `plt.display()` are non-existent functions in the standard Matplotlib `pyplot` interface.
- (c) `plt.label()` is generally used as a sub-component of other functions but is not a standalone command to show the legend box.
- (d) `plt.legend()` is the correct method. When called, Matplotlib looks for all plot elements that have a "label" assigned to them and creates a legend box in the most optimal (best) location on the plot, unless a specific location is specified by the user.

**Final Answer:** To display the legend, the `plt.legend()` method must be called.

**Answer: (B)**



Q44.

**Solution****Concept:**

Customizing the aesthetic appearance of a chart is a key feature of Matplotlib. Most plotting functions, including `plt.bar()`, accept a variety of keyword arguments that allow the user to change line widths, styles, and colors. Consistency in parameter naming is a hallmark of the library, making it easier for users to remember how to style different types of plots.

**Solution:**

- (a) The question asks for the parameter name used to specify the color of bars in a bar chart.
- (b) In Matplotlib, the parameter used to define color across almost all plotting functions (`plot`, `bar`, `scatter`, `hist`) is simply `color`.
- (c) `col` is an abbreviation often used in the R language (`ggplot2`) but not in Matplotlib.
- (d) `colors` is used in some specific functions like `plt.pie()`, but for a standard bar chart, the singular `color` is the primary parameter.
- (e) `bcolor` is not a valid parameter.
- (f) Example usage: `plt.bar(x, y, color='green')`.

**Final Answer:** The correct parameter is `color`.

**Answer: (C)**



Q45.

**Solution****Concept:**

Intellectual Property Rights (IPR) are legal rights that protect the creations of the mind. These laws are designed to encourage innovation by allowing creators to benefit from their work. Common forms of IPR protect different types of "property," such as inventions, brand identities, and artistic works. Conversely, unethical practices like plagiarism represent a violation of these standards rather than a type of right.

**Solution:**

- (a) The question lists four items and asks which are considered IPR.
- (b) Patents: These protect new inventions and technical processes. This is a core component of IPR.
- (c) Trademarks: These protect signs, logos, or expressions used to identify a brand. This is a core component of IPR.
- (d) Copyrights: These protect original literary, dramatic, musical, and artistic works. This is a core component of IPR.
- (e) Plagiarism: This is the act of using someone else's work without permission or credit. While it relates to IPR, it is an ethical/legal violation, not a "Right" itself.
- (f) Therefore, 1, 2, and 3 are the legal rights included in IPR.

**Final Answer:** Intellectual Property Rights include Patents, Trademarks, and Copyrights (1, 2, and 3).

**Answer: (A)**



Q46.

**Solution****Concept:**

Every time an individual interacts with the digital world—whether by sending an email, posting on social media, or simply browsing a website—they leave behind a unique trail of data. This cumulative record of a person’s online activity is a permanent part of the internet’s history and can be used to track, identify, or profile users for various commercial or security purposes.

**Solution:**

- (a) The question asks for the term that describes the trail of data left behind on the internet.
- (b) Cyber Trail is a descriptive term but is not the formal academic or technical term used in informatics.
- (c) Internet Shadow is a poetic metaphor but lacks technical standing in the curriculum.
- (d) Data Cookie refers specifically to small files stored on a user’s computer by a website to remember information, which is only a subset of the broader data trail.
- (e) Digital Footprint is the official term. It is further divided into two types: “Active” (data you intentionally share, like social media posts) and “Passive” (data collected without your active knowledge, like your IP address or browsing history).

**Final Answer:** The trail of data left behind is called a Digital Footprint.

**Answer: (B)**



Q47.

**Solution****Concept:**

Integrity in academic and professional work requires giving proper credit to original authors. When someone copies text, ideas, or results from another person and presents them as if they were their own creation, it constitutes a serious ethical breach. While it is related to copyright infringement, it specifically focuses on the false claim of authorship.

**Solution:**

- (a) The question defines "stealing work or ideas and passing them off as your own."
- (b) Phishing is a cybercrime where targets are contacted by email or phone by someone posing as a legitimate institution to lure individuals into providing sensitive data.
- (c) Hacking refers to unauthorized access to a computer system or network.
- (d) Spamming refers to the sending of bulk unsolicited messages.
- (e) Plagiarism is the correct term for academic or creative theft. It includes "copy-pasting" without citation as well as paraphrasing someone's unique ideas without acknowledging the source.

**Final Answer:** The act of passing off someone else's work as your own is Plagiarism.

**Answer:** (C)



Q48.

**Solution****Concept:**

Electronic waste, commonly known as E-waste, refers to electrical or electronic devices that are discarded, obsolete, or no longer functional. Unlike traditional waste, E-waste contains hazardous materials like lead, cadmium, and mercury, which require specialized recycling processes to prevent environmental contamination and health risks.

**Solution:**

- (a) We must identify which of the provided items qualifies as electronic waste.
- (b) Used plastic bottles are classified as plastic or recyclable waste, but not electronic.
- (c) Old newspapers fall under the category of paper waste.
- (d) Expired medicines are classified as biomedical or chemical waste.
- (e) Discarded mobile phones are electronic devices containing circuit boards, batteries, and screens. Therefore, they are a primary example of E-waste. Other examples include old computers, printers, and televisions.

**Final Answer:** Discarded mobile phones are an example of E-waste.

**Answer: (C)**



Q49.

**Solution****Concept:**

Software licensing determines how a user can interact with, modify, and share a computer program. Licenses generally fall into two broad categories: Proprietary (where the source code is hidden and usage is restricted) and Open Source (where the source code is public and collaborative improvement is encouraged).

**Solution:**

- (a) The question describes a license that allows for free use, modification, and distribution.
- (b) Proprietary software (like Microsoft Windows) is owned by a company that restricts modification and usually requires a paid license.
- (c) Copyrighted software is a broad term; even open-source software is copyrighted, but the owner uses that copyright to grant freedoms rather than restrict them.
- (d) Shared (or Shareware) is often a trial-based software that eventually requires payment.
- (e) Open Source software (like Linux or Python) provides the "Source Code" to the public, allowing anyone to inspect, change, and redistribute the software without paying a fee, provided they follow the license terms (like the GPL or MIT license).

**Final Answer:** Open Source licenses allow free use, modification, and distribution.

**Answer: (B)**



Q50.

**Solution****Concept:**

Data security relies on the ability to make information inaccessible to unauthorized parties. The core technique used for this is a mathematical transformation that turns "Plaintext" (readable data) into "Ciphertext" (scrambled data). Only individuals possessing the correct "Key" can reverse this process to view the original information.

**Solution:**

- (a) The question focuses on the "conversion to unreadable format" to protect data.
- (b) Decryption is the reverse process: converting unreadable ciphertext back into readable plaintext.
- (c) Formatting refers to the arrangement of data or preparing a storage medium for use, which has nothing to do with security.
- (d) Masking is a method of hiding specific data (like the middle digits of a credit card) but doesn't necessarily involve a full conversion of the entire dataset.
- (e) Encryption is the correct term for the proactive process of securing data by making it unreadable without the proper decryption key.

**Final Answer:** The process of protecting data by making it unreadable is Encryption.

**Answer: (B)**



**Answer Key**

Q	Ans	Q	Ans	Q	Ans	Q	Ans	Q	Ans
1	B	2	A	3	B	4	A	5	C
6	B	7	A	8	A	9	A	10	A
11	A	12	A	13	A	14	A	15	A
16	C	17	C	18	B	19	A	20	C
21	C	22	A	23	D	24	B	25	B
26	A	27	B	28	B	29	B	30	A
31	B	32	B	33	B	34	B	35	C
36	D	37	A	38	C	39	B	40	C
41	B	42	D	43	B	44	C	45	A
46	B	47	C	48	C	49	B	50	B

