CUET-UG Information Practices Sample Paper-17

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. Which of the following SQL expressions will return the string 'ormat' from the word 'Information'?

(A) SELECT SUBSTR('Information', 4, 5);

(B) SELECT MID('Information', 5, 5);

(C) SELECT INSTR('Information', 'ormat');

(D) SELECT SUBSTRING('Information', -7, 5);

Q2. What is the result of SELECT ROUND(15.765, 1) + TRUNCATE(15.765, 1)?

(A) 31.5

(B) 31.3

(C) 31.8

(D) 31.0

Q3. If a column 'DOB' contains '2005-12-31', which function returns the name of the day (e.g., Saturday)?

(A) DAYNAME(DOB)

(B) DAYOFWEEK(DOB)

(C) WEEKDAY(DOB)

(D) DATEPART(DOB)

Q4. The SQL function used to remove only trailing spaces from a string ' Data Science ' is:

(A) TRIM()

(B) LTRIM()

(C) RTRIM()

(D) STRIP()

Q5. To find the position of the first occurrence of 'e' in 'Database', which command is correct?

(A) SELECT POSITION('Database', 'e');

(B) SELECT INSTR('Database', 'e');

(C) SELECT SEARCH('e', 'Database');

(D) SELECT CHARINDEX('e', 'Database');

Q6. Which function will return the current system date and time simultaneously?

(A) CURDATE()

(B) DATE()

(C) NOW()

(D) SYSDATE_ONLY()

Q7. What is the output of SELECT MOD(11, -3)?

(A) 2

(B) -2

(C) 3

(D) -3

Q8. Which function is used to convert all characters in a string to uppercase?

(A) UPPERCASE()

(B) TOUPPER()

(C) UPPER()

(D) CAPS()

Q9. In Relational Algebra, the σ (sigma) operator is used for:

(A) Unary operation that selects specific columns.

(B) Unary operation that selects specific rows based on a condition.

(C) Binary operation for joining tables.

(D) Projecting unique values from a relation.

Q10. A candidate key that is not chosen as the primary key is known as a:

(A) Foreign Key

(B) Surrogate Key

(C) Alternate Key

(D) Secondary Key

Q11. Which Relational Algebra operation is equivalent to the Cartesian Product followed by a Selection?

(A) Projection

(B) Theta Join

(C) Intersection

(D) Union

Q12. A Primary Key must satisfy which two constraints?

(A) Unique and Null

(B) Not Null and Unique

(C) Check and Default

(D) Foreign Key and Not Null

Q13. Which network topology requires a multipoint connection and a central cable acting as a backbone?

(A) Star Topology

(B) Mesh Topology

(C) Bus Topology

(D) Ring Topology

Q14. A MAC address is a _____ bit hardware address, whereas an IPv4 address is a _____ bit logical address.

(A) 32, 48

(B) 48, 32

(C) 64, 32

(D) 128, 48

Q15. Which device operates at the Physical layer of the OSI model and regenerates signals to extend network distance?

(A) Switch

(B) Repeater

(C) Router

(D) Gateway

Q16. If 'df' is a DataFrame, what does df.iloc[1:4, 0:2] return?

(A) Rows 1 to 4 and Columns 0 to 2

(B) Rows 1 to 3 and Columns 0 to 1

(C) Rows 2 to 4 and Columns 1 to 2

(D) Rows 1 to 4 and all columns

Q17. Which attribute of a DataFrame returns a tuple representing the dimensionality?

(A) df.size

(B) df.ndim

(C) df.shape

(D) df.index

Q18. How can you change the index of an existing DataFrame 'df' to a column named 'EmpID'?

(A) df.reindex('EmpID')

(B) df.set_index('EmpID', inplace=True)

(C) df.index = 'EmpID'

(D) df.add_index('EmpID')

Q19. To delete a column 'Salary' from a DataFrame, which command is correct?

(A) `df.drop('Salary', axis=0)`

(B) `df.drop('Salary', axis=1)`

(C) `df.remove('Salary')`

(D) `df.delete('Salary')`

Q20. What is the output of a Series created as: `pd.Series([10, 20], index=['a', 'b']) * 2`?

(A) `a 10, b 20, a 10, b 20`

(B) `a 20, b 40`

(C) Index Error

(D) `[20, 40, 20, 40]`

Q21. Which function is used to read data from a CSV file where the separator is a semicolon (;)?

(A) pd.read_csv('file.csv', sep=';')

(B) pd.read_csv('file.csv', delim=';')

(C) pd.load_csv('file.csv', sep=';')

(D) pd.open_csv('file.csv', mode=';')

Q22. The method df.head() without any arguments returns how many rows?

(A) 1

(B) 10

(C) 5

(D) All rows

Q23. To check for missing values (NaN) in a Series 'S', we use:

(A) S.empty()

(B) S.isnull()

(C) S.nan()

(D) S.void()

Q24. Which Pandas property returns the number of elements in the underlying data?

(A) count

(B) size

(C) length

(D) items

Q25. What does df.T perform?

(A) Transform data types

(B) Transpose rows and columns

(C) Terminate the DataFrame

(D) Truncate the data

Q26. In df.loc['A':'C'], if the index is alphabetical, which rows are included?

(A) A and B only

(B) A, B, and C

(C) B and C only

(D) Only A

Q27. Which parameter in read_csv() is used to specify which column should be the index?

(A) index_col

(B) usecols

(C) header

(D) names

Q28. How do you rename a column 'Old' to 'New' in Pandas?

(A) df.rename(columns={'Old':'New'})

(B) df.rename({'Old'='New'})

(C) df.change('Old', 'New')

(D) df.columns['Old'] = 'New'

Q29. The result of df.count() on a DataFrame excludes:

(A) Zeros

(B) Empty strings

(C) NaN values

(D) Negative numbers

Q30. To save a DataFrame to a file without the index, the command is:

(A) df.to_csv('f.csv', index=False)

(B) df.to_csv('f.csv', index=None)

(C) df.write_csv('f.csv', no_index=True)

(D) Both A and B

Q31. Which method is used to fill NaN values with a specific value?

(A) df.replace_nan()

(B) df.fillna()

(C) df.fixnull()

(D) df.dropna()

Q32. A Series in Pandas is _____ dimensional and _____ in size.

(A) 1D, Mutable

(B) 1D, Immutable

(C) 2D, Mutable

(D) 2D, Immutable

Q33. Which clause is used to filter results after the GROUP BY clause has been applied?

(A) WHERE

(B) HAVING

(C) LIKE

(D) DISTINCT

Q34. What is the default sort order of the ORDER BY clause?

(A) DESC

(B) ASC

(C) Random

(D) Insertion order

Q35. Which type of Join returns all records from the left table and matched records from the right table?

(A) Inner Join

(B) Full Outer Join

(C) Left Join

(D) Right Join

Q36. To count the number of unique departments in an 'Employee' table, the query is:

(A) SELECT COUNT(Dept) FROM Employee;

(B) SELECT COUNT(DISTINCT Dept) FROM Employee;

(C) SELECT DISTINCT(COUNT Dept) FROM Employee;

(D) SELECT SUM(Dept) FROM Employee;

Q37. An Equi-join is a specific type of join that uses _____ in the join condition.

(A) Inequality operators

(B) Equality operator (=)

(C) Pattern matching

(D) Range comparison

Q38. In a GROUP BY query, every column in the SELECT list must either be an aggregate function or appear in:

(A) The WHERE clause

(B) The ORDER BY clause

(C) The GROUP BY clause

(D) The HAVING clause

Q39. Which aggregate function ignores NULL values?

(A) COUNT(*)

(B) SUM()

(C) Both A and B

(D) Neither A nor B

Q40. The Natural Join is based on columns in both tables that have:

(A) Same name and same data type

(B) Different names but same data type

(C) Same name but different data types

(D) Primary Key and Foreign Key constraints only

Q41. What happens if you use ORDER BY 2 in a SQL query?

(A) It sorts the result by the second column in the SELECT list.

(B) It sorts the result twice.

(C) It results in a Syntax Error.

(D) It sorts by the primary key.

Q42. Which function is used to create a vertical bar chart in Matplotlib?

(A) plt.barh()

(B) plt.bar()

(C) plt.plot_bar()

(D) plt.vbar()

Q43. To add a label to the X-axis, which method is used?

(A) plt.xname()

(B) plt.xlabel()

(C) plt.set_x()

(D) plt.labelx()

Q44. Which chart type is best suited for showing the distribution of a continuous dataset?

(A) Bar Chart

(B) Pie Chart

(C) Histogram

(D) Line Chart

Q45. The plt.legend() function is used to:

(A) Show the title of the plot

(B) Identify different data series in the plot

(C) Change the color of the bars

(D) Save the plot to a file

Q46. In a Pie chart, which parameter is used to make a wedge stand out?

(A) separate

(B) explode

(C) detach

(D) highlight

Q47. Which of the following is an example of an Intellectual Property Right (IPR) violation?

(A) Phishing

(B) Plagiarism

(C) Cyberstalking

(D) E-waste dumping

Q48. What does 'Digital Footprint' refer to?

(A) The weight of electronic devices.

(B) The trail of data left by users on the internet.

(C) The speed of an internet connection.

(D) A secure biometric login.

Q49. The process of recycling or disposing of broken electronic equipment safely is related to:

(A) Green Computing

(B) Cyber Ethics

(C) E-waste Management

(D) Digital Rights

Q50. Identity theft and credit card fraud fall under the category of:

(A) Cybercrime

(B) Software Piracy

(C) Open Source Initiative

(D) Netiquettes

Detailed Solutions

Q1.

Solution

Concept: SQL string functions like `SUBSTR()`, `SUBSTRING()`, and `MID()` are used to extract parts of a string based on position and length.

Given string: `'Information'` We need to extract: `'ormat'`

Index breakdown (1-based in SQL):

I n f o r m a t i o n

Extracted: "ormat"

Step Analysis:

- Starting from position 4 gives: o r m a t i o n
- Taking 5 characters yields: `ormat`

Option Evaluation:

- (A) `SUBSTR('Information', 4, 5)` → starts at 4, length 5 → `ormat`
- (B) `MID('Information', 5, 5)` → returns `mation`
- (C) `INSTR('Information', 'ormat')` → returns position (not substring)
- (D) `SUBSTRING('Information', -7, 5)` → depends on DB, may not consistently return `ormat`

Final Answer: `SELECT SUBSTR('Information', 4, 5);`

Answer: (A)

Q2.

Solution

Concept: SQL provides different numeric functions for rounding and truncation.

- `ROUND(x, 1)` rounds the number to 1 decimal place.
- `TRUNCATE(x, 1)` removes digits beyond 1 decimal place without rounding.

Step Calculation:

$$15.765 \xrightarrow{\texttt{ROUND(,1)}} 15.8$$

$$15.765 \xrightarrow{\texttt{TRUNCATE(,1)}} 15.7$$

$$15.8 + 15.7 = 31.5$$

Visual Representation:

15.765 → 15.8 (ROUND)

15.765 → 15.7 (TRUNCATE)

Sum = 31.5

Final Answer: 31.5

Answer: (A)

Q3.

Solution

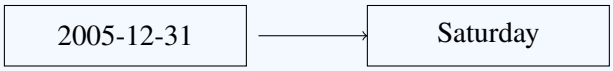
Concept: SQL provides date functions to extract meaningful parts from a date, such as the day name.

- `DAYNAME()` returns the name of the weekday (e.g., Monday, Saturday).
- Other functions return numeric representations or partial date information.

Example:

`2005-12-31` → Saturday

Visual Representation:



Option Analysis:

- (A) `DAYNAME(DOB)` → returns weekday name
- (B) `DAYOFWEEK(DOB)` → returns numeric value (1–7)
- (C) `WEEKDAY(DOB)` → returns numeric index (0–6)
- (D) `DATEPART(DOB)` → not standard SQL function for this output

Final Answer: `DAYNAME(DOB)`

Answer: (A)

Q4.

Solution

Concept: SQL provides functions to remove spaces from strings:

- `LTRIM()` removes leading spaces (from the left).
- `RTRIM()` removes trailing spaces (from the right).
- `TRIM()` removes both leading and trailing spaces.

Given: Remove only trailing spaces from `' Data Science '`
Explanation:

- We only want to remove spaces on the right side.
- `RTRIM()` removes trailing spaces without affecting leading spaces.

Visual Representation:

```
Input:  '  Data Science '
RTRIM(Input):  '  Data Science'
```

Final Answer: `RTRIM()`

Answer: (C)

Q5.

Solution

Concept: SQL provides string functions to find the position of a character within a string.

- `INSTR()` returns the position of a substring in many SQL systems (e.g., MySQL, Oracle).
- `POSITION()` syntax is `POSITION('e' IN 'Database')`, not in the given form.
- `SEARCH()` is not a valid SQL function.
- `CHARINDEX()` is used in SQL Server, but argument order differs.

String Analysis:

$$\texttt{Database} = D\,a\,t\,a\,b\,a\,s\,e$$

First occurrence of 'e' is at position 8

Visual Representation:

D a t a b a s **e**

Position = 8

Final Answer: `INSTR('Database', 'e')`

Answer: (B)

Q6.

Solution

Concept: SQL provides functions to retrieve current date and time values from the system.

- `CURDATE()` returns only the current date.
- `NOW()` returns both current date and time.

Visual Representation:

System Clock ——— `2026-05-07 10:30:00`

Explanation:

- `NOW()` returns both date and time.
- `CURDATE()` returns only date.
- `DATE()` is not used to return current datetime.
- `SYSDATE_ONLY()` is not a valid SQL function.

Final Answer: `NOW()`

Answer: (C)

Q7.

Solution

Concept: In SQL, the `MOD(a, b)` function returns the remainder after division of `a` by `b`. The sign of the result typically follows the dividend (first number).

Calculation:

$$11 \div (-3) = -3 \text{ remainder } 2$$

So,

`MOD(11, -3)` = 2

Visual Representation:

11 ÷ (-3) ——— remainder = 2

→ quotient = -3

Final Answer: 2

Answer: (A)

Q8.

Solution

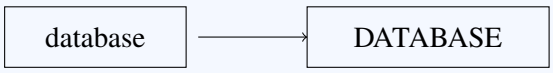
Concept: In SQL, string case conversion functions are used to change the letter casing of text data.

- `UPPER()` converts all characters to uppercase.

Example:

`database` $\longrightarrow$ `DATABASE`

Visual Representation:



Explanation:

- `UPPER()` is the standard SQL function for uppercase conversion.
- `UPPERCASE()`, `TOUPPER()`, and `CAPS()` are not valid SQL functions.

Final Answer: `UPPER()`

Answer: (C)

Q9.

Solution

Concept: In relational algebra, the sigma () operator is a fundamental unary operation used to filter tuples from a relation. It selects rows that satisfy a given predicate or condition. It does not modify columns; instead it reduces the number of tuples based on logical conditions involving attributes, constants, and comparison operators. It is analogous to the WHERE clause in SQL and is essential for query processing and data retrieval efficiency.

Solution: The sigma () operator in relational algebra is used to perform selection over a relation by filtering rows based on a specified condition. It is a unary operation, meaning it works on a single relation and returns a subset of tuples that satisfy the predicate. This is equivalent to the WHERE clause in SQL databases, where conditions are applied to extract relevant records. It does not affect the structure of attributes (columns), unlike projection, which is used for column selection. Sigma plays a key role in query optimization and data retrieval as it reduces the dataset size early in processing. It can involve logical operators such as AND, OR, and NOT, along with comparison operators like =, <, >, and BETWEEN. Thus, sigma is essential for filtering meaningful data before further relational operations are applied. It is widely used in database queries to improve efficiency by minimizing intermediate result sizes. In formal relational algebra expressions, it is denoted as $\sigma_{condition}(Relation)$. Complex predicates can be nested and combined for advanced filtering tasks. It is foundational in database theory and query execution planning. It ensures precise data retrieval from large datasets efficiently.

Final Answer: Unary operation that selects specific rows based on a condition

Answer: (B)

Q10.

Solution

Concept: A candidate key is a minimal set of attributes that uniquely identifies a tuple in a relation. When multiple candidate keys exist, one is chosen as the primary key. The remaining candidate keys that are not selected as the primary key are called alternate keys. They still uniquely identify records but are not used as the main identifier in the database schema.

Solution: A candidate key is an attribute or a set of attributes that can uniquely identify each record in a relation. Among multiple candidate keys, only one is selected as the primary key based on design choice. The remaining candidate keys that are not chosen as primary are still capable of uniquely identifying tuples. These unselected candidate keys are called alternate keys in relational database terminology. Alternate keys help maintain data integrity and provide additional unique identifiers for querying and indexing. They ensure that redundancy is minimized while still preserving uniqueness constraints. In practice, they are enforced using unique constraints in SQL databases. Unlike primary keys, multiple alternate keys can exist within the same table. They are useful when different attributes can serve as identifiers in different contexts. However, they are not referenced by foreign keys as primary keys are. Database designers choose primary keys carefully while keeping alternate keys as backup identifiers. This concept is crucial for normalization and relational schema design. It ensures data consistency and avoids duplication of meaningful identifiers. It is widely implemented in modern database systems to enforce uniqueness constraints effectively across tables for integrity checks.

Final Answer: Alternate Key

Answer: (C)

Q11.

Solution

Concept: The Cartesian product of two relations generates all possible combinations of tuples. When a selection condition is applied on this result, it produces a theta join. A theta join is a binary operation that combines tuples based on a general condition using comparison operators such as =, <, >, <=, and >= in relational algebra.

Solution: The Cartesian product between two relations produces all possible combinations of tuples from both relations. It is a fundamental operation in relational algebra but often generates large intermediate results. When a selection condition is applied to this result, only those tuples satisfying the condition are retained. This combination of Cartesian product followed by selection is known as theta join. The theta join allows joining relations using conditions other than equality. It is more flexible than equi join because it supports inequality operators as well. In relational algebra notation, it is expressed using the join symbol with a condition. Example conditions include A > B, A <= B, or A != B. It is widely used in query processing and database optimization because it enables flexible relational matching. However, it is computationally expensive due to large intermediate results. Database systems often optimize it into more efficient join algorithms such as hash join or merge join. It forms the theoretical basis of join operations used in SQL query execution. It is essential in database systems for combining related data from multiple tables efficiently and accurately under constraints of conditions in practice always.

Final Answer: Theta Join

Answer: (B)

Q12.

Solution

Concept: A primary key uniquely identifies each record in a database table. It must satisfy two essential constraints: uniqueness and not null. These properties ensure that no duplicate or missing values exist for the primary key attribute, maintaining entity integrity and consistency in relational database design.

Solution: A primary key uniquely identifies each record in a database table and ensures entity integrity. It is one of the most important constraints in relational database design. The primary key constraint enforces two main rules: uniqueness and not null values. Uniqueness ensures that no duplicate values exist in the column or set of columns. Not null ensures that every record must contain a valid value in the primary key field. Together, these constraints guarantee that each row in a table is uniquely identifiable. This is essential for maintaining data integrity in relational databases. Without a primary key, duplicate records and inconsistencies can occur. It also helps in establishing relationships between tables using foreign keys. Primary keys are used for indexing and faster data retrieval. They play a critical role in database normalization. They prevent redundancy and ensure consistency across datasets. In composite primary keys, multiple attributes together form a unique identifier. SQL systems automatically enforce primary key constraints. They are defined during table creation or alteration and cannot be left empty, ensuring strict data rules for database integrity and reliability across systems in practice widely used in SQL databases always.

Final Answer: Not Null and Unique

Answer: (B)

Q13.

Solution

Concept: A bus topology is a network configuration in which all devices are connected to a single central cable called the backbone. It is a multipoint connection where multiple devices share the same transmission medium. This structure is simple, cost-effective, but prone to collisions and single-point failure issues.

Solution: A bus topology is a network architecture where all devices are connected to a single communication line called the backbone. It is one of the simplest and most cost-effective network topologies. In this structure, all nodes share a common transmission medium. Data transmitted by any device travels along the bus in both directions. Each device checks whether the data is intended for it. If not, it ignores the data and passes it along. The bus topology does not require a central switching device making it inexpensive to implement in small networks. However, it suffers from a single point of failure. If the backbone cable fails, the entire network goes down. It is also prone to data collisions when multiple devices transmit simultaneously. Network performance decreases as more devices are added. It is rarely used in modern large-scale networks but still useful in simple or temporary setups due to its simplicity and low cost. It uses terminators at both ends of the cable to prevent signal reflection which ensures proper transmission. It is a basic form of LAN topology commonly studied in networking for beginners and exams conceptual understanding.

Final Answer: Bus Topology

Answer: (C)

Q14.

Solution

Concept: In computer networking, MAC and IP addresses differ in size and function:

- A **MAC address** is a hardware address used at the Data Link layer and is 48 bits long.
- An **IPv4 address** is a logical address used at the Network layer and is 32 bits long.

Visual Representation:

MAC Address = 48 bits $\longrightarrow$ IPv4 Address = 32 bits

Explanation:

- MAC address is 48-bit hardware address.
- IPv4 address is 32-bit logical address.

Final Answer: 48, 32

Answer: (B)

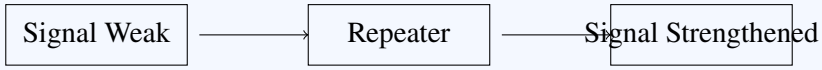
Q15.

Solution

Concept: In the OSI model, each device operates at a specific layer based on its function.

- The **Physical layer (Layer 1)** deals with transmission of raw bits over a physical medium.
- A **Repeater** operates at this layer by regenerating signals to extend network distance.

Visual Representation:



Explanation:

- A **Repeater** regenerates and amplifies signals at the Physical layer.
- Switch works at Data Link layer.
- Router works at Network layer.
- Gateway operates at multiple layers.

Final Answer: Repeater

Answer: (B)

Q16.

Solution

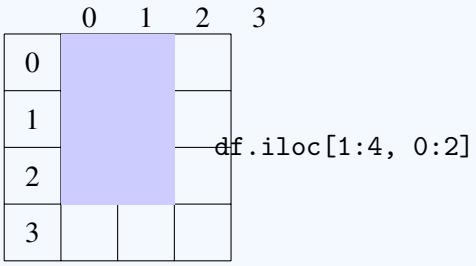
Concept: In Pandas, `iloc[]` is used for integer-based indexing, and slicing follows Python rules:

- Start index is included.
- End index is excluded.

Given: `df.iloc[1:4, 0:2]`

- Rows: 1 to 3 (since 4 is excluded)
- Columns: 0 to 1 (since 2 is excluded)

Visual Representation:



Explanation:

- Rows selected: 1, 2, 3
- Columns selected: 0, 1

Final Answer: Rows 1 to 3 and Columns 0 to 1

Answer: (B)

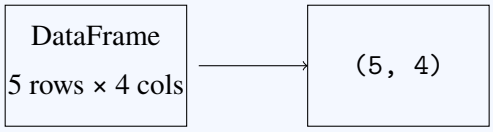
Q17.

Solution

Concept: In Pandas, different attributes provide structural information about a DataFrame.

- `df.shape` returns a tuple representing (rows, columns).
- `df.size` returns total number of elements.
- `df.ndim` returns number of dimensions.
- `df.index` returns row labels.

Visual Representation:



Explanation:

- `df.shape` returns a tuple (rows, columns).
- It is the standard way to get DataFrame dimensions.

Final Answer: `df.shape`

Answer: (C)

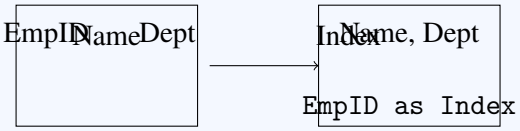
Q18.

Solution

Concept: In Pandas, we can set a column as the index of a DataFrame using the `set_index()` method.

- `set_index()` assigns a column as the new index.
- `inplace=True` modifies the original DataFrame.

Visual Representation:



Explanation:

- `df.set_index('EmpID', inplace=True)` correctly sets 'EmpID' as index.
- `reindex()` is used for aligning indexes, not setting a column as index.
- `df.index = 'EmpID'` is invalid because index must be array-like.
- `add_index()` is not a valid Pandas function.

Final Answer: `df.set_index('EmpID', inplace=True)`

Answer: (B)

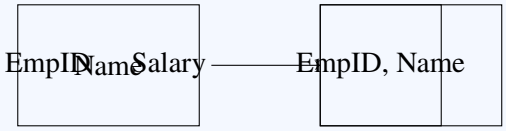
Q19.

Solution

Concept: In Pandas, columns can be removed using the `drop()` method.

- `axis=1` refers to columns.
- `axis=0` refers to rows.

Visual Representation:



Explanation:

- `df.drop('Salary', axis=1)` removes the column 'Salary'.
- `axis=0` would remove a row, not a column.
- `remove()` and `delete()` are not valid Pandas DataFrame methods.

Final Answer: `df.drop('Salary', axis=1)`

Answer: (B)

Q20.

Solution

Concept: In Pandas, arithmetic operations on a Series are applied element-wise.

- Each value in the Series is multiplied by 2 independently.
- Index labels remain unchanged.

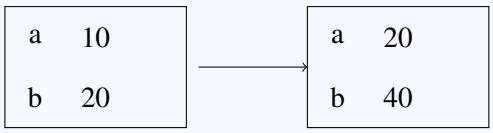
Given Series:

```
a:  10,  b:  20
```

Operation:

$$\times 2 \Rightarrow a : 20, \quad b : 40$$

Visual Representation:



Explanation:

- The multiplication is applied element-wise.
- Indices are preserved.

Final Answer: `a 20, b 40`

Answer: (B)

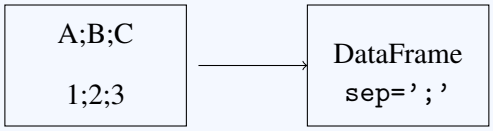
Q21.

Solution

Concept: In Pandas, CSV files with different separators can be read using the `sep` parameter in `read_csv()`.

- The `sep` parameter defines the delimiter used in the file.
- A semicolon-separated file requires `sep=';'`.

Visual Representation:



Explanation:

- `pd.read_csv('file.csv', sep=';')` correctly reads semicolon-separated data.
- `delim=';'` is not a valid parameter name.
- `pd.load_csv()` and `pd.open_csv()` are not valid Pandas functions.

Final Answer: `pd.read_csv('file.csv', sep=';')`

Answer: (A)

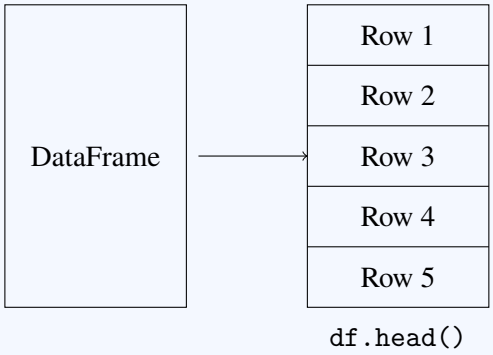
Q22.

Solution

Concept: In Pandas, the `head()` method is used to preview the top rows of a DataFrame.

- By default, `df.head()` returns the first 5 rows.
- The parameter `n` can be used to change the number of rows.

Visual Representation:



Explanation:

- `df.head()` returns the first 5 rows by default.
- This is useful for quick inspection of data.

Final Answer: 5

Answer: (C)

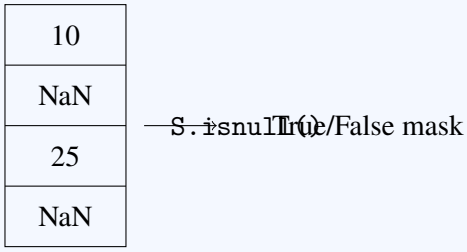
Q23.

Solution

Concept: In Pandas, missing values (NaN) can be detected using specific functions.

- `isnull()` or `isna()` returns True where values are missing.
- It is commonly used to check for NaN values in a Series or DataFrame.

Visual Representation:



Explanation:

- `S.isnull()` correctly identifies missing values.
- `S.empty()` checks if Series has no elements.
- `S.nan()` and `S.void()` are not valid Pandas functions.

Final Answer: `S.isnull()`

Answer: (B)

Q24.

Solution

Concept: In Pandas, different attributes and methods are used to measure or inspect the structure of data objects like Series and DataFrames. The `size` property returns the total number of elements in the underlying data structure, including all rows and columns. It differs from `count`, which only counts non-null values. Understanding these properties is essential for data analysis, memory estimation, and dataset inspection in Python-based data manipulation tasks.

Solution: In Pandas, the size property is used to determine the total number of elements present in a Series or DataFrame. It counts every value in the dataset, including null and non-null entries, giving a complete measure of data volume. This is different from the count method, which only considers non-null values and ignores missing data. The length function, on the other hand, is not a direct Pandas property but rather a general Python function used to get the number of rows in a structure. The items option is incorrect because it is used for iteration over key-value pairs in a Series or DataFrame columns, not for counting elements. The size attribute is especially useful in data analysis when estimating memory usage or understanding dataset dimensions. It returns a single integer value representing total elements. For a DataFrame, size equals rows multiplied by columns. This makes it useful for quick structural inspection of datasets before performing operations like cleaning, filtering, or transformation. It helps analysts understand dataset scale and complexity efficiently and is widely used in exploratory data analysis workflows.

Final Answer: size

Answer: (B)

Q25.

Solution

Concept: In Pandas, DataFrame attributes and methods are used to manipulate and transform data structures efficiently. The `df.T` property is used to transpose a DataFrame, which means converting rows into columns and columns into rows. This operation is commonly used in data restructuring, mathematical operations, and data visualization preparation, especially when orientation of data needs to be changed for analysis or reporting purposes.

Solution: In Pandas, df.T is a property that performs the transpose operation on a DataFrame. Transposing means switching rows into columns and columns into rows, effectively rotating the structure of the dataset. This is particularly useful when the dataset needs to be reoriented for analysis, visualization, or compatibility with machine learning models. The original index becomes column headers, and column headers become the new index. It does not modify data values but only changes their arrangement. This operation is similar to matrix transposition in linear algebra. The transform option is incorrect because it refers to element-wise or column-wise function application rather than structural change. Terminate and truncate options are not valid Pandas operations for DataFrame manipulation. df.T is widely used in exploratory data analysis when comparing variables across observations more conveniently. It can also help in simplifying computations where column-wise operations are required after restructuring. Transposition is especially useful in time series data, pivot-like transformations, and reporting formats. It improves readability in certain analytical scenarios by changing the orientation of data presentation. Overall, df.T is a fundamental tool for restructuring datasets efficiently in Pandas workflows and data preprocessing tasks.

Final Answer: Transpose rows and columns

Answer: (B)

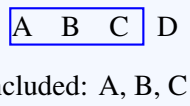
Q26.

Solution

Concept: In Pandas, `loc[]` is label-based indexing. When slicing with labels:

- The start label is included.
- The end label is also included.

Given: `df.loc['A':'C']` with alphabetical index
Visual Representation:



Explanation:

- `'A':'C'` includes both endpoints in `loc[]`.
- So rows A, B, and C are included.

Final Answer: A, B, and C

Answer: (B)

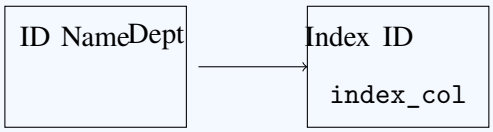
Q27.

Solution

Concept: In Pandas, `read_csv()` has a parameter to set a column as the index of the DataFrame.

- `index_col` is used to specify which column becomes the index.

Visual Representation:



Explanation:

- `index_col` sets a specific column as the index.
- `usecols` selects specific columns to load.
- `header` defines row used as column names.
- `names` assigns custom column names.

Final Answer: `index_col`

Answer: (A)

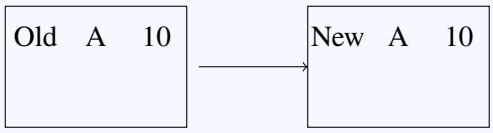
Q28.

Solution

Concept: In Pandas, column names can be changed using the `rename()` method by providing a dictionary inside the `columns` parameter.

- `df.rename(columns={'Old':'New'})` correctly maps the old column name to the new one.

Visual Representation:



Explanation:

- Option (A) is correct Pandas syntax.
- Option (B) is invalid syntax (dictionary format is incorrect).
- Option (C) is not a valid Pandas method.
- Option (D) is incorrect because column names cannot be reassigned like dictionary keys.

Final Answer: `df.rename(columns={'Old':'New'})`

Answer: (A)

Q29.

Solution

Concept: In Pandas, the `df.count()` function is used to count non-null values in each column or row depending on the axis. It automatically excludes missing values represented as NaN. It does not ignore zeros, empty strings, or negative numbers because they are considered valid data entries. Understanding this distinction is important in data preprocessing and data quality analysis when handling missing or incomplete datasets.

Solution: The df.count() function in Pandas is used to return the number of non-null values in each column or row of a DataFrame. It plays an important role in data cleaning and exploratory data analysis by helping identify how many valid entries exist in the dataset. Importantly, df.count() excludes only NaN (Not a Number) values while counting data. It does not exclude zeros because zero is a valid numerical value. Similarly, empty strings are also considered valid string values unless explicitly converted to NaN. Negative numbers are treated as normal numeric values and are included in the count. Therefore, among the given options, only NaN values are excluded by df.count(). This behavior is crucial when analyzing missing data patterns and understanding data completeness. It allows analysts to compare total expected values with actual available values in each column. This function is frequently used before applying imputation or data cleaning techniques. It also helps in identifying columns with high missing data percentage. Overall, df.count() is a fundamental tool in Pandas for assessing data availability and ensuring dataset quality before performing further analysis or modeling tasks.

Final Answer: NaN values

Answer: (C)

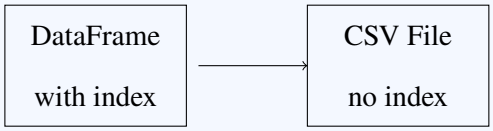
Q30.

Solution

Concept: In Pandas, the `to_csv()` method is used to export a DataFrame to a CSV file. The parameter `index=False` prevents writing row indices to the file.

- `index=False` removes index from output file.
- `index=None` is not the correct way to control this behavior.

Visual Representation:



Explanation:

- `df.to_csv('f.csv', index=False)` correctly saves without index.
- Option B is not a valid way to disable index.
- Option C is not a Pandas function.
- Therefore, only option A is correct.

Final Answer: `df.to_csv('f.csv', index=False)`

Answer: (A)

Q31.

Solution

Concept: In Pandas, missing values (NaN) can be handled using specific built-in functions.

- `fillna()` replaces NaN values with a specified value.
- `dropna()` removes rows or columns containing NaN values.

Visual Representation:

10
NaN
25
NaN

`fillna(value)` →

10
0
25
0

Explanation:

- `df.fillna()` replaces missing values with a given value.
- `replace_nan()`, `fixnull()` are not valid Pandas methods.
- `dropna()` removes missing values instead of filling them.

Final Answer: `df.fillna()`

Answer: (B)

Q32.

Solution

Concept: In Pandas, a Series is a fundamental one-dimensional labeled data structure. It can store data of any type and is mutable, meaning its values can be changed after creation. Unlike DataFrames, which are two-dimensional, Series represent a single column of data with an associated index, making them flexible for data manipulation and analysis tasks.

Solution: A Pandas Series is a one-dimensional labeled array capable of holding data of any type such as integers, strings, or floating-point numbers. It is one of the core data structures in Pandas and is often used to represent a single column of data. A Series is one-dimensional, meaning it has only one axis, unlike a DataFrame which has two dimensions (rows and columns). It is also mutable, meaning its values can be modified after creation. The index associated with a Series allows efficient data access and alignment. The option stating 1D, Immutable is incorrect because Series can be modified after creation. The options suggesting 2D structure are incorrect because Series does not have two dimensions. Therefore, the correct description is that a Series is 1D and mutable. This property makes Series highly useful in data analysis, as it allows easy manipulation, filtering, and transformation of data. It is commonly used as the building block for DataFrames, where multiple Series are combined into a tabular structure. Understanding Series behavior is essential for working effectively with Pandas in data science and analytics workflows, especially during data preprocessing and exploratory analysis tasks.

Final Answer: 1D, Mutable

Answer: (A)

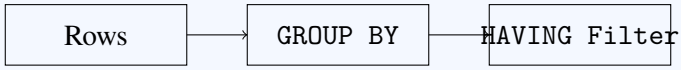
Q33.

Solution

Concept: In SQL, filtering after aggregation (after `GROUP BY`) is done using the `HAVING` clause.

- `WHERE` filters rows before grouping.
- `HAVING` filters groups after aggregation.

Visual Representation:



Explanation:

- `HAVING` is used to filter grouped results.
- `WHERE` cannot be used after aggregation.
- `LIKE` is for pattern matching.
- `DISTINCT` removes duplicates.

Final Answer: `HAVING`

Answer: (B)

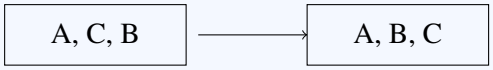
Q34.

Solution

Concept: The `ORDER BY` clause in SQL is used to sort query results.

- The default sorting order is ascending (`ASC`).
- If not specified, SQL automatically sorts in ascending order.

Visual Representation:



`ORDER BY (default ASC)`

Explanation:

- `ASC` sorts data in increasing order (default).
- `DESC` must be explicitly specified for descending order.
- Random and insertion order are not default SQL behaviors.

Final Answer: `ASC`

Answer: (B)

Q35.

Solution

Concept: In SQL joins, different types of joins are used to combine rows from two tables based on related columns. A left join is used when all records from the left table are required, along with matching records from the right table. If no match is found, NULL values are returned for the right table columns. It is commonly used to preserve complete data from the primary table.

Solution: A Left Join in SQL is used to return all records from the left table and only the matching records from the right table based on a join condition. If there is no matching record in the right table, the result still includes the left table row, but with NULL values for the right table columns. This makes the left join particularly useful when the left table contains primary or mandatory data that must be preserved in the output. Inner join, in contrast, returns only matching rows from both tables, while full outer join returns all rows from both tables regardless of matches. Right join behaves similarly to left join but prioritizes the right table instead. The left join is widely used in database queries where maintaining complete information from one dataset is important, such as customer lists with optional order details. It helps in identifying unmatched records as well. The syntax typically uses LEFT JOIN along with an ON condition specifying the relationship between tables. Overall, left join is a fundamental relational operation that ensures completeness of the primary dataset while integrating related information from another table in a controlled manner.

Final Answer: Left Join

Answer: (C)

Q36.

Solution

Concept: SQL provides aggregate functions and distinct operations to analyze data efficiently. To count unique values in a column, the DISTINCT keyword is used inside the COUNT function. This ensures that only non-duplicate values are considered, making it essential for summarizing categorical data like departments, categories, or groups in a dataset.

Solution: In SQL, when we need to count the number of unique values in a column, we use the COUNT(DISTINCT column) function. This ensures that duplicate values are ignored and only unique entries are counted. In the case of an Employee table, to find the number of unique departments, the correct query is SELECT COUNT(DISTINCT Dept) FROM Employee. The simple COUNT(Dept) function would count all non-null rows, including duplicates, which does not give the number of unique departments. The option SELECT DISTINCT(COUNT Dept) FROM Employee is syntactically incorrect in SQL. The SUM(Dept) function is also incorrect because SUM is used for numeric aggregation, not categorical counting. Using DISTINCT inside COUNT is the standard approach for identifying unique values in relational databases. This is widely used in reporting and analytics to understand diversity in data fields such as departments, job roles, or categories. It helps in summarizing data efficiently without manual filtering. Therefore, COUNT(DISTINCT Dept) is the correct and optimized way to retrieve the number of unique departments from a dataset in SQL queries used in real-world database systems.

Final Answer: SELECT COUNT(DISTINCT Dept) FROM Employee;

Answer: (B)

Q37.

Solution

Concept: In SQL, an Equi-join is a type of join operation where the joining condition is based on equality between columns of two tables. It is the most commonly used join type and forms the basis for inner joins. It uses the equality operator to match rows with identical values in related columns.

Solution: An Equi-join in SQL is a specific type of join that combines rows from two tables based on equality between specified columns. The join condition uses the equality operator (=) to match records where values in the related columns are identical. This makes equi-join a fundamental and widely used join operation in relational databases. It is essentially a special case of a theta join where only the equality condition is allowed. Inequality operators such as <, >, or != are not used in equi-joins. Pattern matching using LIKE is also not part of equi-join conditions, nor are range comparisons like BETWEEN. Equi-joins are commonly used to combine primary key and foreign key relationships between tables, ensuring data consistency and meaningful associations. They are also the basis for inner joins in SQL implementation. By enforcing equality conditions, equi-joins ensure accurate matching of related records across different tables. This operation is essential for retrieving structured and relational data efficiently in database systems used in real-world applications.

Final Answer: Equality operator (=)

Answer: (B)

Q38.

Solution

Concept: In SQL, GROUP BY is used to aggregate data based on one or more columns. When using GROUP BY, all selected columns must either be included in the grouping clause or used inside aggregate functions like SUM, COUNT, AVG, etc. This ensures logical consistency in grouped query results and avoids ambiguity in aggregated outputs.

Solution: In SQL, the GROUP BY clause is used to group rows that have the same values in specified columns into summary rows. When using GROUP BY, SQL enforces a rule that every column in the SELECT statement must either be part of the GROUP BY clause or used within an aggregate function such as SUM, COUNT, AVG, MIN, or MAX. This rule ensures that the query output is logically consistent and unambiguous. If a column is not grouped or aggregated, SQL cannot determine how to represent multiple values in a single result row. Therefore, including non-aggregated and non-grouped columns leads to errors. The WHERE clause is used for filtering rows before grouping, while ORDER BY is used for sorting results after grouping. The HAVING clause is used to filter grouped results, not to define column rules. Hence, the correct requirement is that columns must appear in the GROUP BY clause. This is a fundamental rule in relational database query processing and is essential for producing valid grouped outputs. It ensures structured aggregation and correct summarization of data in analytical queries.

Final Answer: The GROUP BY clause

Answer: (C)

Q39.

Solution

Concept: In SQL, aggregate functions are used to perform calculations on sets of values. Most aggregate functions automatically ignore NULL values because NULL represents missing or unknown data. Understanding how different functions handle NULLs is important for accurate data analysis and reporting in relational databases.

Solution: In SQL, aggregate functions are used to compute a single result from multiple rows of data. Most aggregate functions, such as SUM() and AVG(), automatically ignore NULL values during computation because NULL represents missing or undefined data. COUNT(*) is an exception because it counts all rows, including those with NULL values, whereas COUNT(column) ignores NULLs in that specific column. SUM() ignores NULL values and only adds non-null numeric entries, making it reliable for numerical aggregation. Therefore, both SUM() and COUNT(*) behave differently with respect to NULL handling. However, the key observation is that SUM() inherently ignores NULL values, while COUNT(*) does not ignore rows but also does not treat NULL as a value to sum. Given the options, both functions are considered in terms of NULL handling behavior in aggregate operations. This distinction is important in data analysis because NULL handling can significantly affect results. Proper understanding ensures accurate reporting and avoids misleading summaries in database queries. Hence, knowledge of aggregate function behavior is essential in SQL data processing and analytics.

Final Answer: Neither A nor B

Answer: (D)

Q40.

Solution

Concept: A Natural Join in SQL automatically joins two tables based on columns that have the same name and compatible data types. It eliminates duplicate columns in the result and assumes equality between matching column names. It is a simplified form of equi-join and is widely used when schema design follows consistent naming conventions across related tables.

Solution: A Natural Join is a type of join operation in SQL that automatically combines two tables based on columns that share the same name and compatible data types. The database system identifies columns with identical names in both tables and performs an equality-based join on them without requiring the user to explicitly specify the join condition. This makes Natural Join convenient but also potentially risky if unintended column matches exist. The condition requires both the column name and data type to match so that meaningful comparisons can be performed without type conflicts. It removes duplicate columns from the final output, presenting only one instance of the matching attribute. Options involving different column names are incorrect because Natural Join does not work on differently named attributes. The option mentioning primary key and foreign key constraints only is also incorrect because Natural Join does not require explicit key relationships, only matching column names. It is essentially an automated equi-join based on schema similarity. Therefore, Natural Join relies on columns having the same name and same data type for correct execution and meaningful relational output in SQL systems.

Final Answer: Same name and same data type

Answer: (A)

Q41.

Solution

Concept: In SQL, the ORDER BY clause is used to sort query results based on one or more columns. Instead of column names, positional indexing can also be used, where numbers represent column positions in the SELECT list. This allows flexible sorting based on output structure rather than explicit column names.

Solution: In SQL, the ORDER BY clause is used to sort the result set of a query either in ascending or descending order. When a numeric value is used in ORDER BY, it refers to the position of the column in the SELECT statement rather than the column name itself. Therefore, ORDER BY 2 means that the result will be sorted based on the second column listed in the SELECT clause. This is a positional reference technique and is useful when working with complex queries or when column names are long or derived from expressions. It does not sort the data twice, nor does it generate a syntax error, as it is valid SQL syntax in most database systems. It also does not automatically sort by primary key unless the primary key happens to be the second selected column. The sorting behavior depends entirely on the structure of the SELECT list. This feature provides flexibility but can reduce readability, so it is often recommended to use column names in production queries. Overall, ORDER BY 2 is a shorthand method for sorting results based on the second selected column in the query output.

Final Answer: It sorts the result by the second column in the SELECT list.

Answer: (A)

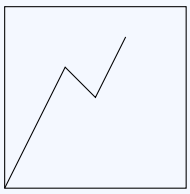
Q42.

Solution

Concept: In Matplotlib, bar charts are used to represent categorical data:

- `plt.bar()` creates a vertical bar chart.
- `plt.barh()` creates a horizontal bar chart.

Visual Representation:



Vertical Bars

Explanation:

- `plt.bar()` is used for vertical bar charts.
- `plt.barh()` is used for horizontal bar charts.
- Other options are not valid Matplotlib functions.

Final Answer: `plt.bar()`

Answer: (B)

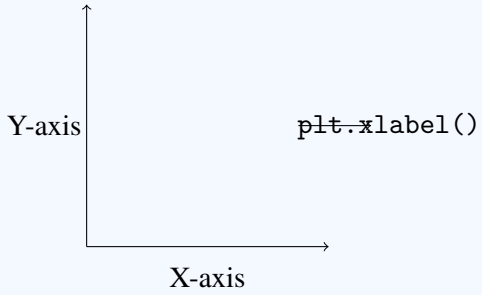
Q43.

Solution

Concept: In Matplotlib, axis labels are added using specific labeling functions.

- `plt.xlabel()` sets the label for the X-axis.
- `plt.ylabel()` sets the label for the Y-axis.

Visual Representation:



Explanation:

- `plt.xlabel()` is used to assign a label to the X-axis.
- Other options are not valid Matplotlib functions.

Final Answer: `plt.xlabel()`

Answer: (B)

Q44.

Solution

Concept: In data visualization, different chart types are used based on the nature of the dataset. A histogram is specifically designed to represent the distribution of continuous numerical data by dividing it into intervals (bins). It helps in understanding frequency distribution, spread, and shape of data, making it essential in statistical analysis.

Solution: A histogram is the most appropriate chart type for showing the distribution of a continuous dataset. It groups continuous numerical data into intervals known as bins and displays the frequency of data points within each bin. This helps in visualizing how data is distributed across a range, identifying patterns such as skewness, normal distribution, or outliers. Unlike bar charts, which are used for categorical data, histograms deal with continuous data and have no gaps between bars. Pie charts are used to represent proportions of a whole, not distributions. Line charts are primarily used for showing trends over time rather than frequency distribution. Therefore, histogram is the correct choice for analyzing continuous data distribution. It is widely used in statistics, data analysis, and machine learning for understanding data behavior before applying models. It provides a clear visual summary of data density and variability. Histograms are also useful in detecting anomalies and comparing distributions across datasets. Overall, they are a fundamental tool in exploratory data analysis for continuous variables.

Final Answer: Histogram

Answer: (C)

Q45.

Solution

Concept: In Matplotlib, different functions are used to enhance plot readability and interpretation. The `plt.legend()` function is specifically used to display a legend that identifies different data series in a plot. It helps distinguish between multiple datasets or lines using labels, improving clarity in data visualization.

Solution: In Matplotlib, the plt.legend() function is used to add a legend to a plot. A legend is a visual guide that helps identify different data series represented in the same graph. When multiple lines, bars, or markers are plotted together, each dataset is usually assigned a label. The plt.legend() function displays these labels in a boxed area within the plot, making it easier for users to understand which line or element corresponds to which dataset. It does not change the title of the plot, modify colors of bars, or save the plot to a file. Instead, its primary purpose is to improve interpretability of visual data representations. Legends are especially important in comparative analysis where multiple datasets are shown in a single chart. Without legends, interpreting complex graphs becomes difficult and error-prone. Therefore, plt.legend() plays a crucial role in enhancing clarity and communication in data visualization. It is widely used in scientific plotting, statistical analysis, and reporting dashboards where multiple data series are present.

Final Answer: Identify different data series in the plot

Answer: (B)

Q46.

Solution

Concept: In Matplotlib, pie charts are used to represent proportions of a whole. To emphasize a specific slice or wedge, an explode parameter is used. It separates the wedge slightly from the center of the pie chart, making it visually stand out for better attention and interpretation.

Solution: In a pie chart created using Matplotlib, each wedge represents a proportion of the whole dataset. To highlight or emphasize a particular wedge, the explode parameter is used. This parameter allows one or more slices of the pie chart to be slightly separated from the center, making them visually stand out from the rest of the chart. This is especially useful when drawing attention to a specific category in data visualization, such as the largest or most important segment. The separate, detach, and highlight options are not valid parameters in Matplotlib for pie charts. Only explode is the correct and officially supported parameter for this purpose. The explode parameter takes a list of values that determine how far each wedge is offset from the center. A value of 0 means no separation, while higher values increase the distance. This feature improves visual emphasis and helps viewers quickly identify important data segments. Therefore, explode is the correct parameter used to make a wedge stand out in a pie chart.

Final Answer: explode

Answer: (B)

Q47.

Solution

Concept: Intellectual Property Rights (IPR) protect original works such as ideas, writings, software, and inventions from unauthorized use. Violations occur when someone uses or reproduces protected content without permission. Plagiarism is a common IPR violation where someone copies another person's work and presents it as their own.

Solution: Intellectual Property Rights (IPR) are legal protections given to creators for their original works such as literature, inventions, software, and artistic content. These rights ensure that creators receive credit and control over their work. A violation of IPR occurs when someone uses, copies, or distributes protected content without proper authorization or acknowledgment. Among the given options, plagiarism is a direct violation of IPR because it involves copying someone else's work or ideas and presenting them as one's own without giving credit. Phishing is a cybercrime related to stealing sensitive information, cyberstalking involves harassment through digital means, and e-waste dumping relates to environmental issues, not intellectual property. Therefore, plagiarism is the correct example of an IPR violation. It is commonly seen in academic and professional contexts where originality is required. Proper citation and referencing are used to avoid plagiarism. IPR laws help maintain fairness, creativity, and innovation by protecting creators' rights. Hence, understanding plagiarism is essential in maintaining ethical use of information and respecting intellectual ownership in digital and academic environments.

Final Answer: Plagiarism

Answer: (B)

Q48.

Solution

Concept: A digital footprint refers to the data trail created by users when they interact with the internet. This includes websites visited, searches made, social media activity, and online transactions. It can be passive or active and plays an important role in privacy, security, and online behavior analysis.

Solution: A digital footprint is the record of data that users leave behind while using the internet. Every online activity such as browsing websites, posting on social media, sending emails, or making online purchases contributes to this footprint. It can be active, where users intentionally share information, or passive, where data is collected without direct input. This footprint helps organizations understand user behavior but also raises concerns about privacy and security. It is not related to physical attributes like device weight or internet speed. It is also not a biometric login system, which involves authentication using physical traits. Instead, it represents the overall digital trace of user activity across platforms. Digital footprints can be monitored by websites, advertisers, and sometimes malicious entities, making awareness important for cybersecurity. Users are encouraged to manage their digital footprint carefully to protect personal information. Therefore, the correct definition is the trail of data left by users on the internet, which forms their digital identity in the online world.

Final Answer: The trail of data left by users on the internet

Answer: (B)

Q49.

Solution

Concept: E-waste management refers to the safe handling, recycling, and disposal of electronic waste such as computers, mobile phones, and other electronic devices. It is an important part of environmental sustainability and is closely associated with green computing practices that aim to reduce environmental impact of technology.

Solution: The safe recycling or disposal of broken or outdated electronic equipment is known as e-waste management. It involves processes such as collection, recycling, recovery, and safe disposal of electronic components to minimize environmental harm. Electronic waste contains hazardous materials like lead, mercury, and cadmium, which can be harmful if not handled properly. Therefore, proper management is essential to protect both human health and the environment. Green computing is a broader concept that focuses on environmentally sustainable use of computers and IT resources, including energy efficiency and reducing carbon footprint. Cyber ethics deals with moral behavior in digital environments, while digital rights refer to users' rights in the online world. Among the given options, e-waste management specifically addresses the safe handling of discarded electronic devices. It ensures that valuable materials are recycled and toxic substances are disposed of safely. This process is increasingly important due to the rapid growth of electronic consumption worldwide. Therefore, e-waste management is the correct concept related to recycling or disposing of broken electronic equipment safely.

Final Answer: E-waste Management

Answer: (C)

Q50.

Solution

Concept: Cybercrime refers to illegal activities carried out using computers, networks, or the internet. It includes offenses such as identity theft, credit card fraud, hacking, phishing, and data breaches. These crimes target individuals, organizations, or financial systems by exploiting digital technologies. Understanding cybercrime is essential in digital safety and cybersecurity, as it helps in identifying, preventing, and responding to online threats effectively in modern internet-based environments.

Solution: Identity theft and credit card fraud are both classified as cybercrimes because they involve the illegal use of digital systems and online platforms to steal personal or financial information. In identity theft, an attacker illegally obtains someone's personal data such as name, address, passwords, or identification numbers and uses it for fraudulent purposes. Credit card fraud involves unauthorized use of another person's card details to make transactions or withdraw money. Both activities are carried out using computers, networks, or online systems, making them part of cybercrime. These crimes exploit weaknesses in digital security systems and often cause financial loss, privacy violations, and emotional distress to victims. Software piracy, on the other hand, refers to illegal copying or distribution of software. Open source initiative relates to freely available software development models. Netiquettes are rules of polite behavior on the internet and are not related to criminal activity. Therefore, identity theft and credit card fraud clearly fall under cybercrime, which is a major concern in cybersecurity. Governments and organizations use laws, encryption, authentication systems, and awareness programs to prevent such crimes. Hence, cybercrime is the correct classification for these illegal digital activities.

Final Answer: Cybercrime

Answer: (A)

Answer Key

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