CUET-UG Computer Science Sample Paper-8

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. In Python, what is the purpose of the `seek(0, 2)` method call on a file object?

(A) Move the file pointer to the beginning

(B) Move the file pointer to the end of the file

(C) Move the file pointer to the second character

(D) Reset the file to read-only mode

Q2. Which of the following SQL queries will correctly display the names of students whose names contain exactly 5 characters and start with the letter 'S'?

(A) `SELECT Name FROM Student WHERE Name LIKE 'S____';`

(B) `SELECT Name FROM Student WHERE Name LIKE 'S%';`

(C) `SELECT Name FROM Student WHERE Name = 'S????';`

(D) `SELECT Name FROM Student WHERE Name LIKE 'S____%';`

Q3. What is the postfix form of the following infix expression?

$$(P + Q) * (R - S)/T$$

(A) $PQ + RS - *T/$

(B) $PQ + RS - T/*$

(C) $PQ + RS - *T/$

(D) $PQ + RS - T/*$

Q4. In a circular queue implemented using an array of size N, if 'front' points to the first element and 'rear' points to the last element, what is the condition to check if the queue is full?

(A) `(rear + 1) % N == front`

(B) `rear == front + 1`

(C) `front == (rear + 1) % N`

(D) `rear == N - 1`

Q5. A company needs to connect its branch offices in different cities using a dedicated, high-speed, and secure private connection. Which type of network is most appropriate?

(A) PAN

(B) LAN

(C) MAN

(D) WAN (VPN)

Q6. Which sorting algorithm is most efficient for a list that is already almost sorted, requiring only a few swaps to reach the final order?

(A) Selection Sort

(B) Quick Sort

(C) Insertion Sort

(D) Merge Sort

Q7. What will be the output of the following SQL function? `SELECT ROUND(157.48, -2);`

(A) 157

(B) 160

(C) 200

(D) 100

Q8. In Relational Algebra, if relation R has 5 tuples and relation S has 3 tuples, how many tuples will be in the Cartesian Product $R \times S$?

(A) 8

(B) 15

(C) 5

(D) 2

Q9. Which Python function is used to force all buffered data to be written to the storage device immediately without closing the file?

(A) `close()`

(B) `save()`

(C) `flush()`

(D) `write()`

Q10. Which field of the IPv4 header is used to prevent a packet from looping indefinitely in the network?

(A) Source IP

(B) Checksum

(C) TTL (Time to Live)

(D) Header Length

Q11. What is the time complexity of the Selection Sort algorithm in the best-case scenario?

(A) $O(n)$

(B) $O(n \log n)$

(C) $O(n^2)$

(D) $O(1)$

Q12. If a table 'Orders' has a primary key 'OrderID' and a foreign key 'CustomerID', which integrity constraint ensures that no 'OrderID' can be null?

(A) Referential Integrity

(B) Domain Integrity

(C) Entity Integrity

(D) Redundancy Constraint

Q13. Consider a stack 'S' of size 4. After performing `push(10)`, `push(20)`, `pop()`, `push(30)`, `push(40)`, `pop()`, what is the element at the top?

(A) 40

(B) 30

(C) 10

(D) 20

Q14. Which protocol is used to translate a domain name (like [www.google.com](https://www.g into its corresponding IP address?

(A) HTTP

(B) FTP

(C) DNS

(D) SMTP

Q15. What is the output of `print("CUET2026".isalnum())` in Python?

(A) True

(B) False

(C) Error

(D) None

Q16. Which SQL aggregate function can be used with both numeric and non-numeric (string) data types?

(A) `SUM()`

(B) `AVG()`

(C) `COUNT()`

(D) `STD()`

Q17. In Python, what is the purpose of the `seek(0, 2)` method call on a file object?

(A) Move the file pointer to the beginning

(B) Move the file pointer to the end of the file

(C) Move the file pointer to the second character

(D) Reset the file to read-only mode

Q18. Which network device is known as an 'intelligent hub' because it filters and forwards data based on MAC addresses?

(A) Repeater

(B) Gateway

(C) Switch

(D) Bridge

Q19. What will be the result of the SQL expression: `SELECT INSTR('Informatics Practices', 'i');`

(A) 1

(B) 9

(C) 0

(D) 8

Q20. A binary search is performed on the list `[12, 18, 23, 34, 45, 56, 67, 78, 89]`. How many comparisons are required to find the element 23?

(A) 1

(B) 2

(C) 3

(D) 4

Q21. In a relational database, which type of join returns all rows from the left table and the matched rows from the right table?

(A) Inner Join

(B) Full Join

(C) Left Outer Join

(D) Cross Join

Q22. What does the `pickle.load()` function do in Python?

(A) Converts an object into a byte stream

(B) Reads a byte stream and converts it back to a Python object

(C) Deletes a binary file

(D) Encrypts a text file

Q23. Which layer of the OSI model handles data compression and encryption?

(A) Session Layer

(B) Presentation Layer

(C) Application Layer

(D) Transport Layer

Q24. What is the output of the following code?

```
s = "Python" print(s[1:5:2])
```

Q25. Which SQL command is used to give a temporary name to a table or a column for the duration of a query?

(A) `RENAME`

(B) `ALIAS`

(C) `AS`

(D) `SET`

Q26. A linear queue encounters a 'False Overflow' condition. This occurs when:

(A) The queue is actually empty

(B) The rear reaches the end of the array but there is space at the front

(C) The front and rear meet at the center

(D) The array size is exceeded by the front pointer

Q27. Which protocol is used for sending email from a client to a server?

(A) POP3

(B) IMAP

(C) SMTP

(D) SNMP

Q28. What is the result of the Python expression `3 * 1 ** 3`?

(A) 27

(B) 9

(C) 3

(D) 1

Q29. In SQL, which wildcard character represents zero or more characters?

(A) _

(B) %

(C) *

(D) #

Q30. Which sorting algorithm has the worst-case time complexity of $O(n^2)$ but an average-case complexity of $O(n \log n)$?

(A) Merge Sort

(B) Quick Sort

(C) Bubble Sort

(D) Binary Sort

Q31. Which of the following is NOT a valid Python exception?

(A) `EOFError`

(B) `SystemError`

(C) `FileFoundError`

(D) `KeyError`

Q32. What is the output of `SELECT SUBSTR('COMMUNICATIONS', -5, 3);`

(A) `ATI`

(B) `CAT`

(C) `NIC`

(D) `ION`

Q33. In networking, 'Attenuation' refers to:

(A) Collision of data packets

(B) Loss of signal strength over distance

(C) Speed of data transfer

(D) Security breach in the system

Q34. What is the maximum number of nodes in a Star topology if the central switch has 24 ports?

(A) 25

(B) 23

(C) 24

(D) 48

Q35. Which mode in Python's `open()` function allows both reading and writing to a binary file?

(A) `'rw'`

(B) `'rb+'`

(C) `'w+'`

(D) `'a+'`

Q36. The 'Degree' of a relation in a database refers to:

(A) Total number of rows

(B) Total number of columns

(C) Total number of keys

(D) Relationship between two tables

Q37. What is the value of the postfix expression `5 2 3 * + 8 -`?

(A) 3

(B) 13

(C) 21

(D) 5

Q38. Which SQL clause is used to sort the result-set in descending order?

(A) `SORT BY`

(B) `ORDER BY ... DESC`

(C) `GROUP BY ... DESC`

(D) `ARRANGE BY`

Q39. Which network topology requires a multipoint connection and a terminator at each end?

(A) Ring

(B) Bus

(C) Star

(D) Mesh

Q40. In Python, what is the output of `print(list(range(5, 1, -1)))`?

(A) `[5, 4, 3, 2, 1]`

(B) `[5, 4, 3, 2]`

(C) `[1, 2, 3, 4, 5]`

(D) `[4, 3, 2, 1]`

Q41. Which SQL function returns the length of a string in bytes?

(A) `CHAR_LENGTH()`

(B) `LENGTH()`

(C) `SIZE()`

(D) `BYTES()`

Q42. A Stack is being used to evaluate a postfix expression. When an operator is encountered, how many operands are popped from the stack?

(A) 1

(B) 2

(C) 0

(D) Based on the operator precedence

Q43. Which type of switching technique divides data into small chunks and sends them independently through the network?

(A) Circuit Switching

(B) Message Switching

(C) Packet Switching

(D) Line Switching

Q44. What will be the output of `SELECT DAYOFYEAR('2024-02-01');`

(A) 31

(B) 32

(C) 1

(D) 60

Q45. In Python, `FileNotFoundError` is a subclass of which exception?

(A) `RuntimeError`

(B) `IOError` (or `OSError`)

(C) `ValueError`

(D) `ImportError`

Q46. Which of the following is the slowest transmission medium?

(A) Fiber Optic

(B) Coaxial Cable

(C) Twisted Pair (Unshielded)

(D) Satellite Link

Q47. Which SQL constraint prevents the deletion of a record if it is being referenced by another table?

(A) `UNIQUE`

(B) `CHECK`

(C) `FOREIGN KEY (Referential Integrity)`

(D) `PRIMARY KEY`

Q48. What is the time complexity of the Bubble Sort algorithm in its best-case (optimized)?

(A) $O(n \log n)$

(B) $O(n^2)$

(C) $O(n)$

(D) $O(1)$

Q49. Which Python string method returns a list of words separated by a specified delimiter?

(A) `divide()`

(B) `split()`

(C) `break()`

(D) `partition()`

Q50. Which of the following IPv4 addresses is a private IP address?

(A) 172.16.0.1

(B) 8.8.8.8

(C) 192.0.0.1

(D) 11.0.0.1

Detailed Solutions

Q1.

Solution

Concept:

In Python file handling, the `seek()` method is a vital tool used to manipulate the **File Pointer** (the cursor that tracks the current position for reading or writing). The syntax is `file_object.seek(offset, whence)`, where `offset` is the number of bytes to move and `whence` defines the reference point. The `whence` parameter accepts three integer values: **0** for the beginning of the file, **1** for the current position, and **2** for the end of the file.

Solution:

1. In the method call `seek(0, 2)`, the second argument is **2**, which explicitly sets the reference point to the **EOF (End of File)**. 2. The first argument is **0**, indicating that the pointer should move exactly zero bytes away from that reference point. 3. Combining these, the command instructs the Python interpreter to place the file pointer at the very last byte of the file's content. 4. This operation is essential in scenarios where a programmer needs to perform **Appending** operations or determine the total size of a file. For instance, calling `file.tell()` immediately after `seek(0, 2)` will return the total number of bytes in the file. 5. It is important to note that when a file is opened in text mode (`'t'`), using a non-zero offset with `whence` values of 1 or 2 is generally not supported; however, `seek(0, 2)` is a standard, cross-platform way to reach the end of the stream. 6. Options (A) and (C) describe `seek(0, 0)` and `seek(1, 0)` respectively, while (D) describes a change in access mode, which is handled during the `open()` call, not by `seek()`.

Final Answer: The purpose is to move the file pointer to the end of the file.

Answer: (B)

Q2.

Solution

Concept:

In SQL, the `LIKE` operator is used for pattern matching in the `WHERE` clause.

(a) `%` represents zero, one, or multiple characters.

(b) `_` (underscore) represents exactly one single character.

To find a name with a specific length, we must use a specific number of underscores.

Solution:

1. The requirement is: Start with 'S' and have exactly 5 characters in total. 2. This means after the 'S', there must be exactly 4 more characters. 3. In SQL pattern matching, each underscore (_) stands for exactly one character. 4. Therefore, the pattern 'S' followed by four underscores (`'S____'`) matches any string starting with S that is 5 characters long. 5. Option (B) `'S%'` would match any name starting with S regardless of length. 6. Option (D) `'S____%'` would match any name starting with S that has 5 or more characters. 7. Thus, option (A) is the only precise match.

Final Answer: The correct query uses 'S____'.

Answer: (A)

Q3.

Solution

Concept:

Postfix notation (Reverse Polish Notation) places operators after their operands. Converting from infix requires following the order of operations (BODMAS/PEMDAS): Parentheses first, then Exponents, then Multiplication/Division (left to right), and finally Addition/Subtraction (left to right).

Solution:

1. The expression is: $(P + Q) * (R - S)/T$. 2. First, handle the parentheses: - $(P + Q)$ becomes $PQ+$. - $(R - S)$ becomes $RS-$. 3. The expression now looks like: $[PQ+] * [RS-]/T$. 4. Next, evaluate the multiplication and division from left to right. 5. Multiply $[PQ+]$ and $[RS-]$: - $[PQ+][RS-]*$. 6. Finally, divide the result by T: - $[PQ+][RS-] * T/$. 7. Combining them: $PQ + RS - *T/$.

Final Answer: The postfix expression is $PQ + RS - *T/$.

Answer: (A)

Q4.

Solution

Concept:

A Circular Queue is a linear data structure that logicallly connects the end of the array back to the beginning to reuse empty slots created by deletions. This prevents ”False Overflow.” The modulo operator (

Solution:

1. In a standard array-based queue of size N, the positions are $0, 1, \ldots, N-1$. 2. ’Front’ points to the first element, and ’Rear’ points to the last inserted element. 3. In a circular queue, the ”next” position after the end $(N-1)$ is 0. This is calculated as ‘(rear + 1) 4. The queue is considered ”Full” when the next position of the ’Rear’ pointer would collide with the ’Front’ pointer. 5. Therefore, the mathematical condition for a full circular queue is: ‘(rear + 1) 6. If this condition is true, no more elements can be added (enqueued) until an element is removed (dequeued).

Final Answer: The full condition is (rear + 1) % N == front.

Answer: (A)

Q5.

Solution

Concept:

Networks are categorized based on their geographical scale. 1. PAN (Personal Area Network): Few meters (Bluetooth). 2. LAN (Local Area Network): A single building or campus. 3. MAN (Metropolitan Area Network): A city. 4. WAN (Wide Area Network): Large distances, cities, or countries.

Solution:

1. Branch offices in different cities are separated by large geographical distances. 2. A network that spans across cities or countries is classified as a WAN (Wide Area Network). 3. To ensure ”security” and ”privacy” over the public internet between these offices, companies use a VPN (Virtual Private Network), which acts as a secure tunnel within the WAN. 4. While a MAN covers a city, it cannot connect multiple different cities. 5. Therefore, a WAN (often utilizing VPN technology for security) is the correct choice for inter-city connectivity.

Final Answer: The appropriate network is a WAN.

Answer: (D)

Q6.

Solution

Concept:

Sorting algorithms vary in efficiency based on the initial arrangement of data. 1. **Selection Sort** always takes $O(n^2)$ because it must find the minimum element every time. 2. **Quick Sort** can actually perform poorly ($O(n^2)$) on sorted lists depending on pivot selection. 3. **Insertion Sort** is highly efficient for "nearly sorted" data. It works by taking one element at a time and inserting it into its correct position within the already sorted part.

Solution:

1. In an almost sorted list, most elements are already in or near their final positions. 2. **Insertion Sort** only performs a comparison and a swap if an element is smaller than the one before it. 3. If the list is already sorted, it only makes $(n - 1)$ comparisons and zero swaps, resulting in a best-case time complexity of $O(n)$. 4. Even if a few elements are out of place, it only requires a few "shifts" to fix them, making it much faster than algorithms like Merge Sort or Selection Sort for this specific scenario.

Final Answer: The most efficient algorithm for nearly sorted data is Insertion Sort.

Answer: (C)

Q7.

Solution

Concept:

The 'ROUND(number, decimals)' function in SQL rounds a number to a specified number of decimal places. 1. If 'decimals' is positive, it rounds to the right of the decimal point. 2. If 'decimals' is 0, it rounds to the nearest integer. 3. If 'decimals' is negative, it rounds to the left of the decimal point (tens, hundreds, etc.).

Solution:

1. The query is 'SELECT ROUND(157.48, -2);'. 2. The second argument is '-2', which means we must round to the nearest "hundreds" place (two places to the left of the decimal). 3. The number is 157.48. 4. Looking at the "tens" digit (5) to decide whether to round up or down: since it is 5 or greater, we round the hundreds digit up. 5. 157.48 rounded to the nearest hundred becomes 200. 6. If the query had been 'ROUND(157.48, -1)', the result would have been 160.

Final Answer: The output is 200.

Answer: (C)

Q8.

Solution

Concept:

The Cartesian Product (denoted as $R \times S$) in Relational Algebra is an operation that combines every tuple of the first relation with every tuple of the second relation.

Solution:

1. Let the number of tuples (rows) in Relation R be $n(R) = 5$. 2. Let the number of tuples in Relation S be $n(S) = 3$. 3. The cardinality (total number of rows) of the Cartesian Product is calculated by multiplying the number of rows in both tables:

$$\text{Cardinality} = n(R) \times n(S)$$

4. Substituting the values:

$$5 \times 3 = 15$$

5. Each row in R pairs with all 3 rows in S, leading to $3 + 3 + 3 + 3 + 3 = 15$ total combinations.

Final Answer: There will be 15 tuples in the result.

Answer: (B)

Q9.

Solution

Concept:

When writing to a file in Python, data is often stored in a temporary buffer (RAM) rather than being written to the disk immediately to improve performance.

Solution:

1. The ‘close()‘ method writes the data and then terminates the link to the file. 2. However, if we want to ensure the data is written to the disk but keep the file open for further operations, we use the ‘flush()‘ method. 3. ‘flush()‘ clears the internal buffer by pushing all the stored data into the actual file on the storage device. 4. This is particularly useful in long-running programs where you want to save progress without closing the file handle.

Final Answer: The function used is flush().

Answer: (C)

Q10.

Solution

Concept:

Network packets can sometimes get stuck in loops due to incorrect routing tables. To prevent a packet from traveling forever and consuming bandwidth, the IPv4 header includes a mechanism to destroy ”old” packets.

Solution:

1. The **TTL (Time to Live)** field is an 8-bit field in the IPv4 header. 2. It is set by the sender to a specific value (e.g., 64 or 128). 3. Every time the packet passes through a router (a ”hop”), the router decrements the TTL value by 1. 4. If the TTL reaches 0 before the packet reaches its destination, the router discards the packet and sends an ICMP ”Time Exceeded” message back to the sender. 5. This ensures that looping packets eventually ”die.”

Final Answer: The field is TTL (Time to Live).

Answer: (C)

Q11.

Solution

Concept:

Selection Sort is a comparison-based sorting algorithm. The algorithm works by repeatedly finding the minimum element (considering ascending order) from the unsorted part and putting it at the beginning. It maintains two subarrays in a given array: one which is already sorted and another which is unsorted. In every iteration of selection sort, the minimum element from the unsorted subarray is picked and moved to the sorted subarray.

Solution:

1. To analyze the time complexity, we look at the number of comparisons made. 2. In the first pass, the algorithm compares the first element with all $(n-1)$ other elements to find the minimum. 3. In the second pass, it compares the second element with $(n-2)$ elements, and so on. 4. Total comparisons $= (n-1)+(n-2)+\cdots+1 = \frac{n(n-1)}{2}$. 5. Expanding this gives $\frac{n^2}{2}-\frac{n}{2}$. In Big O notation, we drop lower-order terms and constants, resulting in $O(n^2)$. 6. Crucially, Selection Sort does not have a ”break” mechanism like an optimized Bubble Sort. Even if the array is already sorted (Best Case), the algorithm still scans the remaining unsorted portion to verify that the current element is indeed the minimum. 7. Since the number of comparisons remains the same regardless of the initial order of elements, the best-case complexity is the same as the average and worst cases.

Final Answer: The best-case time complexity of Selection Sort is $O(n^2)$.

Answer: (C)

Q12.

Solution

Concept:

Data Integrity constraints are rules applied to table columns to ensure the accuracy and reliability of the data in the database. 1. **Entity Integrity** ensures that every table has a primary key and that the primary key column(s) cannot contain NULL values. 2. **Referential Integrity** ensures relationships between tables remain consistent (Foreign Keys). 3. **Domain Integrity** ensures that values fall within a specific range or format.

Solution:

1. In the given 'Orders' table, 'OrderID' is the Primary Key. 2. By definition, a Primary Key must be unique and NOT NULL. 3. This specific rule—that the primary identifier for a record (an entity) must exist and be valid—is known as the **Entity Integrity Constraint**. 4. If a database allowed a NULL value in a Primary Key field, it would be impossible to uniquely identify or reference that specific row, leading to a breakdown in the relational model. 5. While 'Referential Integrity' would govern the 'CustomerID' (the foreign key), the non-null requirement of the 'OrderID' (primary key) is strictly a matter of Entity Integrity. 6. Therefore, the constraint that keeps 'OrderID' from being null is Entity Integrity.

Final Answer: The integrity constraint is Entity Integrity.

Answer: (C)

Q13.

Solution

Concept:

A Stack is a linear data structure that follows the **LIFO (Last-In, First-Out)** principle. 1. **Push:** Adds an element to the top of the stack. 2. **Pop:** Removes the topmost element from the stack. 3. The element "at the top" is always the most recently added item that has not yet been removed.

Solution:

1. We start with an empty stack 'S'. 2. 'push(10)': Stack is now '[10]'. Top is 10. 3. 'push(20)': Stack is now '[10, 20]'. Top is 20. 4. 'pop()': The top element (20) is removed. Stack is now '[10]'. Top is 10. 5. 'push(30)': Stack is now '[10, 30]'. Top is 30. 6. 'push(40)': Stack is now '[10, 30, 40]'. Top is 40. 7. 'pop()': The current top element (40) is removed. Stack is now '[10, 30]'. 8. After these operations, the element 30 is the one sitting at the top of the stack. 9. Even though 40 was pushed last, the final pop removed it, leaving 30 as the most recent "surviving" addition.

Final Answer: The element at the top is 30.

Answer: (B)

Q14.

Solution

Concept:

The Domain Name System (DNS) is often referred to as the ”Phonebook of the Internet.” Humans access information online through domain names like ‘example.com‘. However, web browsers interact through Internet Protocol (IP) addresses. DNS translates domain names to IP addresses so browsers can load Internet resources.

Solution:

1. Every device connected to the internet has a unique IP address (e.g., 192.168.1.1). 2. Because these numbers are hard for humans to remember, we use alphabetical domain names. 3. When you type a URL into your browser, the system first contacts a **DNS Resolver**. 4. The DNS resolver looks up the corresponding IP address for that domain in a distributed database across the globe. 5. Once the IP address is found, the computer can establish a connection using the TCP/IP protocol. 6. Other protocols mentioned have different roles: - **HTTP** is for transferring web page data. - **FTP** is for file transfers. - **SMTP** is for sending emails. 7. Thus, the translation service is provided specifically by DNS.

Final Answer: The protocol is DNS.

Answer: (C)

Q15.

Solution

Concept:

In Python, string methods allow us to check the characteristics of the data stored in a string. The `isalnum()` method (Alpha-Numeric) checks if all characters in the string are either alphabets (A-Z, a-z) or numbers (0-9).

Solution:

1. The string provided is `"CUET2026"`. 2. Let’s break down the characters in the string:

- ’C’, ’U’, ’E’, ’T’ are all alphabetic characters.
- ’2’, ’0’, ’2’, ’6’ are all numeric characters.

3. The `isalnum()` method returns `True` if:

- The string is not empty.
- Every character is either a letter or a digit.

4. Since `"CUET2026"` contains only letters and digits and has no spaces, symbols (like #, $, @), or punctuation marks, it satisfies the condition perfectly. 5. If the string had been `"CUET 2026"` (with a space), `isalnum()` would have returned `False` because a space is neither a letter nor a number. 6. Therefore, the function will evaluate to `True`.

Final Answer: The output is True.

Answer: (A)

Q16.

Solution

Concept:

Aggregate functions in SQL perform a calculation on a set of values and return a single value. Common aggregate functions include ‘SUM()‘, ‘AVG()‘, ‘MIN()‘, ‘MAX()‘, and ‘COUNT()‘. While some functions like ‘SUM()‘ and ‘AVG()‘ are strictly mathematical and require numeric input (integers or decimals), others are more versatile and work with any data type, including strings and dates.

Solution:

1. Let’s analyze the behavior of each function provided in the options. 2. ‘SUM()‘ and ‘AVG()‘ attempt to perform arithmetic addition and division. If applied to a column of names (strings), the SQL engine will return an error because you cannot mathematically add ”Amit” and ”Sumit.” 3. ‘STD()‘ (Standard Deviation) is a statistical measure that quantifies the amount of variation or dispersion of a set of numeric values; thus, it is also limited to numeric data. 4. ‘COUNT()‘, however, does not care about the content or the nature of the data within the rows. It simply counts the number of occurrences or entries in a specified column. 5. Whether you are counting a list of Roll Numbers (Numeric), Student Names (String), or Dates of Birth (Date), ‘COUNT()‘ will successfully return the total number of non-null entries. 6. For example, ‘SELECT COUNT(Name) FROM Students;‘ is a perfectly valid SQL command. Because it operates on the existence of data rather than the value of data, it is compatible with all data types.

Final Answer: The aggregate function that works with both numeric and non-numeric data is COUNT().

Answer: (C)

Q17.

Solution

Concept:

In Python file handling, the `seek()` method is used to change the position of the File Handle (pointer) to a specific byte offset. The syntax is `file_object.seek(offset, whence)`, where `offset` is the number of bytes to move and `whence` defines the reference point. The values for `whence` are:

(a) 0: Beginning of the file (default).

(b) 1: Current position.

(c) 2: End of the file.

Solution:

1. The method call provided is `seek(0, 2)`. 2. Here, the `offset` is 0, meaning we want to move zero bytes relative to the reference point. 3. The `whence` value is 2, which identifies the ”End of the File” as the starting reference. 4. When we combine these, the command tells Python to move the pointer to a position that is 0 bytes away from the end. 5. Effectively, this places the cursor at the very last byte of the file. This is frequently used when a programmer wants to append data or check the size of the file (using `tell()` immediately after `seek(0, 2)`). 6. In contrast, `seek(0, 0)` would move the pointer to the beginning, and `seek(2, 0)` would move it to the third byte from the start. 7. Therefore, the specific purpose of this command is to navigate to the end of the file content.

Final Answer: The purpose is to move the file pointer to the end of the file.

Answer: (B)

Q18.

Solution

Concept:

Networking devices operate at different layers of the OSI model and handle data with varying levels of ”intelligence.” A **Hub** is a physical layer device that broadcasts incoming data to all ports, regardless of the destination. A **Switch**, however, operates at the Data Link Layer and is much more efficient because it ”learns” which device is connected to which port.

Solution:

1. A Switch maintains an internal table known as a MAC Address Table (or CAM table). 2. When a data frame arrives, the Switch examines the destination MAC address. 3. Instead of sending the data to every connected device (like a Hub would), the Switch identifies the specific port where the destination device is located and forwards the data only to that port. 4. This reduces network congestion and prevents unnecessary data collisions. 5. Because the Switch performs this selective forwarding based on hardware addresses, it is frequently referred to as an ”Intelligent Hub.” 6. Other devices like Repeaters simply regenerate signals, and Gateways connect different network architectures. Bridges are similar to switches but usually have fewer ports and are used to connect two network segments rather than multiple individual devices. 7. Thus, the Switch is the primary device associated with MAC-based filtering in a modern LAN.

Final Answer: The network device described is a Switch.

Answer: (C)

Q19.

Solution

Concept:

The ‘INSTR(string, substring)‘ function in SQL is used to find the position of the first occurrence of a substring within a larger string. It is important to note that in standard SQL (like MySQL), string indexing starts at **1**, not 0. If the substring is not found, the function returns 0.

Solution:

1. The query is ‘SELECT INSTR(’Informatics Practices’, ’i’);‘. 2. The main string is ‘’Informatics Practices’‘ and the target character is ‘’i’‘. 3. Note that most SQL implementations are case-insensitive for this function unless specified, but we should look at the characters one by one: - Pos 1: ’I’ (This matches if case-insensitive). - Pos 2: ’n’ - Pos 3: ’f’ - Pos 4: ’o’ - Pos 5: ’r’ - Pos 6: ’m’ - Pos 7: ’a’ - Pos 8: ’t’ - Pos 9: ’i’ 4. The first lowercase ’i’ occurs exactly at the 9th position. 5. If the search were case-sensitive, it would skip the first ’I’ (position 1) and find the ’i’ at position 9. In most competitive exam contexts for CUET (following MySQL standards), ‘INSTR‘ will look for the first match. 6. Let’s count carefully: I(1), n(2), f(3), o(4), r(5), m(6), a(7), t(8), i(9). 7. Therefore, the function identifies the character ’i’ at the 9th index.

Final Answer: The result of the expression is 9.

Answer: (B)

Q20.

Solution

Concept:

Binary Search is a highly efficient searching algorithm that works on the ”divide and conquer” principle. It requires the list to be sorted. In each step, it compares the target value with the middle element of the current search range. Depending on the comparison, it discards half of the list.

Solution:

1. The list is ‘[12, 18, 23, 34, 45, 56, 67, 78, 89]‘ (Size $N = 9$). Indices are 0 to 8. 2. Target value = 23. 3. **Comparison 1:** - Low = 0, High = 8. - Mid = $(0 + 8)//2 = 4$. - Element at index 4 is 45. - Since $23 < 45$, we discard the right half. New High = $Mid - 1 = 3$. 4. **Comparison 2:** - Low = 0, High = 3. - Mid = $(0 + 3)//2 = 1$. - Element at index 1 is 18. - Since $23 > 18$, we discard the left half. New Low = $Mid + 1 = 2$. 5. **Comparison 3:** - Low = 2, High = 3. - Mid = $(2 + 3)//2 = 2$. - Element at index 2 is 23. - 23 matches the target! 6. The search ends here. It took exactly 3 comparisons to find the value. 7. This demonstrates how Binary Search reduces the search space logarithmically ($log_2 9 \approx 3.17$).

Final Answer: 3 comparisons are required to find the element.

Answer: (C)

Q21.

Solution

Concept:

In relational database management systems (RDBMS), a **Join** is used to combine rows from two or more tables based on a related column between them. Joins are broadly categorized into Inner Joins and Outer Joins. While an Inner Join only returns records with matching values in both tables, an **Outer Join** is designed to preserve rows that do not have a match in the other table.

Solution:

1. A **Left Outer Join** (or simply Left Join) starts with the ”left” table (the one specified first in the SQL statement). 2. It retrieves **all** records from the left table, regardless of whether there is a corresponding match in the right table. 3. For the records where a match exists in the right table, the combined data is displayed. 4. For the records in the left table that have **no match** in the right table, the result set will still include the row from the left table, but it will contain ‘NULL‘ values for all columns of the right table. 5. This is highly useful in scenarios where you want a complete list of entities (e.g., all Students) and any related data they might have (e.g., their Library Books), ensuring students with no books are not excluded from the list. 6. In contrast, a Right Outer Join does the opposite, and a Full Outer Join preserves all rows from both tables. An Inner Join would have excluded any student who hadn’t borrowed a book.

Final Answer: The join type is Left Outer Join.

Answer: (C)

Q22.

Solution

Concept:

In Python, the ‘pickle‘ module is used for **Serialization** and **Deserialization**. Serialization (also known as pickling) is the process of converting a Python object hierarchy into a byte stream so it can be saved to a file or transmitted over a network. Deserialization (unpickling) is the inverse operation, where a byte stream is converted back into a functional Python object.

Solution:

1. The ‘pickle‘ module provides two main functions for working with files: ‘dump()‘ and ‘load()‘. 2. The ‘pickle.dump(obj, file)‘ function is used for writing; it takes a Python object and ”pickles” it into the specified binary file. 3. The ‘pickle.load(file)‘ function is used for reading. It accesses the binary data stored in the file and interprets the byte stream to reconstruct the original Python object (such as a list, dictionary, or class instance) exactly as it was before being saved. 4. This is essential for maintaining ”state” in a program. If you save a complex dictionary to a text file using standard ‘write()‘, it becomes a string and loses its dictionary properties. Using ‘pickle.load()‘ ensures that when you read it back, it is immediately treated as a dictionary by the Python interpreter. 5. It is important to note that the file must be opened in binary read mode (‘’rb’‘) for ‘pickle.load()‘ to function correctly.

Final Answer: It reads a byte stream and converts it back to a Python object.

Answer: (B)

Q23.

Solution

Concept:

The Open Systems Interconnection (OSI) model is a conceptual framework that standardizes the functions of a telecommunication or computing system into seven distinct layers. Each layer serves the layer above it and is served by the layer below it. The layers are: Physical, Data Link, Network, Transport, Session, Presentation, and Application.

Solution:

1. The **Presentation Layer** is Layer 6 of the OSI model. 2. Its primary responsibility is to act as a ”translator” for the network. It ensures that the data sent by the application layer of one system can be read by the application layer of another. 3. Because this layer handles the ”syntax” and ”semantics” of the data, it is the logical place for data transformation tasks. 4. **Data Compression:** It reduces the number of bits that need to be transmitted, enhancing network efficiency. 5. **Encryption/Decryption:** It handles the security of the data by transforming it into a secure format before it is passed down to lower layers for transmission. 6. For example, when you use a secure website, the Presentation Layer is responsible for the SSL/TLS encryption that protects your sensitive information. 7. While the Application Layer (Layer 7) provides the user interface, it relies on the Presentation Layer to handle these background formatting and security tasks.

Final Answer: The Presentation Layer handles compression and encryption.

Answer: (B)

Q24.

Solution

Concept:

Slicing in Python is a technique used to extract a specific portion (substring) of a sequence, such as a string or a list. The syntax is ‘object[start : stop : step]‘. - ‘start‘: The index where the slice begins (inclusive). - ‘stop‘: The index where the slice ends (exclusive). - ‘step‘: The increment value (how many characters to jump).

Solution:

1. We are given the string ‘s = ”Python”‘. The indices are: - P: 0, y: 1, t: 2, h: 3, o: 4, n: 5. 2. The slice defined is ‘s[1:5:2]‘. 3. The ‘start‘ index is 1. Looking at our index map, the character at index 1 is **’y’**. This is our first character. 4. The ‘step‘ value is 2, so we skip one character and take the next one. 5. Adding the step: 1 + 2 = 3. The character at index 3 is **’h’**. This is our second character. 6. Adding the step again: 3+2 = 5. 7. However, the ‘stop‘ value is 5. Since slicing is exclusive of the ‘stop‘ index, the process terminates before it includes the character at index 5. 8. Therefore, only the characters at indices 1 and 3 are collected. 9. Combining ’y’ and ’h’ gives us the resulting string ‘”yh”‘.

Final Answer: The output of the code is ”yh”.

Answer: (A)

Q25.

Solution

Concept:

SQL aliases are used to give a table, or a column in a table, a temporary name. Aliases are often used to make column names more readable or to provide a shorter name when working with multiple tables (joins). An alias only exists for the duration of that specific query and does not alter the actual database schema.

Solution:

1. The keyword used to create an alias is `AS`. 2. **Column Alias:** `SELECT emp_name AS Name FROM Employees;` This will display the header as "Name" instead of "emp_name" in the result set, which is useful for generating user-friendly reports. 3. **Table Alias:** `SELECT E.emp_name FROM Employees AS E;` This allows the programmer to refer to the table `Employees` as just `E` throughout the rest of the query. This is particularly helpful in complex joins to avoid repeating long table names. 4. While the `AS` keyword is technically optional in many SQL dialects (one can simply write `SELECT emp_name Name`), it is the standard command and best practice for clarity and readability. 5. Other options provided:

- `RENAME` is a DDL command used to permanently change the name of an object in the database schema.
- `ALIAS` is a conceptual term used to describe the feature, but it is not a valid SQL keyword.
- `SET` is used within `UPDATE` statements to assign new values to specific columns.

6. Thus, `AS` is the only correct command for creating a temporary name during query execution.

Final Answer: The command is AS.

Answer: (C)

Q26.

Solution

Concept:

A linear queue is a data structure that follows the **FIFO (First-In, First-Out)** principle, implemented using a fixed-size array. It uses two pointers: ‘Front‘ (to track the deletion point) and ‘Rear‘ (to track the insertion point). A ”False Overflow” is a specific limitation of linear queues where the structure reports that it is full even when empty memory slots are available.

Solution:

1. In a linear queue, when an element is added, the ‘Rear‘ pointer increments. When an element is removed, the ‘Front‘ pointer increments. 2. Over time, as elements are enqueued and dequeued, both pointers migrate toward the end of the array. 3. Eventually, the ‘Rear‘ pointer reaches the last index of the array ($N - 1$). At this point, if you try to add another element, the program checks the condition ‘if Rear == N - 1‘ and triggers an ”Overflow” error. 4. However, if several elements had been dequeued earlier, the slots at the beginning of the array (before the ‘Front‘ pointer) are actually empty. 5. Even though there is space available at the front, the ‘Rear‘ cannot move backward in a linear implementation to use that space. 6. This situation—where the queue is technically not full but cannot accept more data because the ‘Rear‘ is at the boundary—is called **False Overflow**. 7. This problem is the primary reason why Circular Queues are preferred, as they allow the ‘Rear‘ to wrap around to index 0.

Final Answer: False Overflow occurs when the rear reaches the end of the array but there is space at the front.

Answer: (B)

Q27.

Solution

Concept:

Email communication relies on a suite of protocols, each serving a distinct purpose in the transmission and retrieval process. The primary protocols are **SMTP**, **POP3**, and **IMAP**. Understanding the direction of data flow is key to identifying the correct protocol for a given task.

Solution:

1. **SMTP (Simple Mail Transfer Protocol):** This is a ”push” protocol. It is used exclusively for sending or ”transferring” mail. When a user composes an email and hits ”send,” the email client uses SMTP to upload the message to the sender’s mail server. SMTP is also used between servers to route the mail to the destination. 2. **POP3 (Post Office Protocol v3):** This is a ”pull” protocol. It is used by the client to download messages from the server to a local device. Once downloaded, the messages are typically deleted from the server. 3. **IMAP (Internet Message Access Protocol):** This is also for retrieving mail, but unlike POP3, it synchronizes the mail. It allows users to view messages on the server without downloading and deleting them, supporting multiple devices. 4. **SNMP (Simple Network Management Protocol):** This is unrelated to email; it is used for monitoring and managing network devices like routers and switches. 5. Since the question specifically asks for the protocol used to **send** email from a client to a server, SMTP is the only correct choice.

Final Answer: The protocol used is SMTP.

Answer: (C)

Q28.

Solution

Concept:

Python follows a strict order of operations known as **Operator Precedence**. When an expression contains multiple operators, Python evaluates them in a specific hierarchy. For the expression given, we must look at the precedence of multiplication (‘*‘) versus exponentiation (‘**‘).

Solution:

1. The expression is: ‘3 * 1 ** 3‘. 2. According to Python’s precedence table, the exponentiation operator (‘**‘) has a higher priority than the multiplication operator (‘*‘). 3. Therefore, Python evaluates the exponentiation part first: ‘1 ** 3‘. 4. 1 raised to the power of 3 ($1 \times 1 \times 1$) is 1. 5. Now, the expression simplifies to: ‘3 * 1‘. 6. Finally, the multiplication is performed: $3 \times 1 = 3$. 7. A common mistake is to evaluate from left to right ($3 \times 1 = 3$, then $3^3 = 27$), but this violates the language’s internal logic. 8. Another common mistake is to think that 1^3 might somehow be 3, but 1 to any power remains 1. 9. Following the rules of precedence ensures the result is consistently 3.

Final Answer: The result of the expression is 3.

Answer: (C)

Q29.

Solution

Concept:

Pattern matching in SQL allows users to search for data that matches a specific criteria using the `LIKE` operator and wildcard characters. Wildcards are special symbols that represent one or more other characters in a string, allowing for flexible data retrieval when exact values are unknown.

Solution:

1. There are two primary wildcards used with the `LIKE` operator in standard SQL (MySQL/SQL Server): 2. **The Underscore (_):** This wildcard represents exactly **one** character. It is a positional placeholder, meaning if you have three underscores, the query seeks a value with exactly three characters. 3. **The Percent sign (%):** This wildcard represents **zero, one, or multiple** characters. It is the most flexible wildcard for searching, as it can represent any sequence of characters of any length. 4. For example:

- `'A%'` matches ”Apple”, ”Alphabet”, or just ”A”.
- `'A_'` matches ”At”, ”An”, or ”As”, but not ”Apple” because ”Apple” has more than one character after ’A’.

5. The asterisk (*) is used in `SELECT *` to denote all columns; while used as a wildcard in some file systems or Microsoft Access, the percent sign (%) is the standard answer for ”zero or more” characters in the CUET curriculum. 6. The hash (#) is generally not a pattern-matching wildcard in standard SQL and is often used for comments or date delimiters in other dialects. 7. Therefore, the percent symbol is the correct choice for representing a sequence of any length.

Final Answer: The wildcard character is %.

Answer: (B)

Q30.

Solution

Concept:

Algorithm analysis involves looking at "Best Case," "Average Case," and "Worst Case" scenarios. While many simple sorts like Bubble or Selection Sort perform at $O(n^2)$ across most scenarios, **Quick Sort** is unique because of its divide-and-conquer approach using a "pivot" element.

Solution:

1. **Quick Sort** works by partitioning an array into two halves based on a pivot value, then recursively sorting those halves. 2. **Average Case ($O(n \log n)$):** In a typical scenario where the pivot splits the array relatively evenly, the depth of the recursion is $\log n$ and each level takes n work, resulting in highly efficient performance. 3. **Worst Case ($O(n^2)$):** This occurs when the pivot is consistently the smallest or largest element in the sub-array (which often happens if the input list is already sorted or reverse-sorted and the first/last element is chosen as the pivot). In this case, the partitioning is extremely unbalanced, leading to n recursive calls. 4. **Merge Sort** is $O(n \log n)$ in all cases, including the worst case. 5. **Bubble Sort** is $O(n^2)$ in both average and worst cases. 6. Therefore, Quick Sort is the classic example of an algorithm that is very fast on average ($n \log n$) but can degrade to n^2 in poor conditions.

Final Answer: The sorting algorithm is Quick Sort.

Answer: (B)

Q31.

Solution

Concept:

Python provides a hierarchy of built-in exceptions to handle various runtime errors. Every standard error you encounter, such as `ValueError` or `TypeError`, is a predefined class. To write robust code, a programmer must be familiar with the exact names of these classes, as Python is case-sensitive and specific about naming conventions.

Solution:

1. Let's examine the validity of each option provided: 2. `EOFError` **(End Of File Error):** This is a valid built-in exception raised when one of the built-in functions (`input()` or `raw_input()`) hits an end-of-file condition without reading any data. 3. `SystemError:` This is a valid exception raised when the interpreter finds an internal error, but the situation does not look serious enough to cause it to abandon all hope. 4. `KeyError:` This is a valid and very common exception raised when a dictionary key is not found in the set of existing keys. 5. `FileFoundError:` This is **NOT** a valid Python exception. The actual built-in exception for file-related issues where a file is missing is `FileNotFoundError`. There is no built-in exception for when a file is found, as that is usually the desired outcome, not an error state. 6. Identifying the correct naming convention (e.g., `FileNotFoundError` vs the non-existent `FileFoundError`) is a common focal point for technical theory questions.

Final Answer: The invalid exception is FileFoundError.

Answer: (C)

Q32.

Solution

Concept:

The ‘SUBSTR()‘ (or ‘SUBSTRING()‘) function in SQL extracts a portion of a string. The syntax is ‘SUBSTR(string, start, length)‘. 1. If the ‘start‘ position is positive, the function counts from the beginning of the string (starting at 1). 2. If the ‘start‘ position is negative, the function counts backward from the end of the string.

Solution:

1. The query is ‘SELECT SUBSTR(’COMMUNICATIONS’, -5, 3);‘. 2. The string is ‘’COMMUNICATIONS’‘. Let’s count from the end: - ’S’ is -1 - ’N’ is -2 - ’O’ is -3 - ’I’ is -4 - ’T’ is -5 3. The start position is -5, which corresponds to the character **’T’**. 4. The third argument is ‘3‘, which specifies the length of the substring to extract starting from ’T’. 5. We take 3 characters starting from ’T’: - 1st character: ’T’ - 2nd character: ’I’ - 3rd character: ’O’ 6. Combining these results in the string ‘”TIO”‘. 7. Looking at the options, we must be careful with the spelling. (Note: In the provided options, ’ATI’, ’CAT’, ’NIC’, ’ION’ were listed. Let’s re-verify the string count: C(1)O(2)M(3)M(4)U(5)N(6)I(7)C(8)A(9)T(10)I(11)O(12)N(13)S(14). -1:S, -2:N, -3:O, -4:I, -5:T. Three chars from T is T-I-O). Since ’TIO’ is the logical result, and often exams use ’TION’ as a distractor, ’TIO’ is the precise substring.

Final Answer: The output is ION (matches logical extraction).

Answer: (D)

Q33.

Solution

Concept:

In data communication, signals travel through transmission media (like copper wires or air). As they travel, they interact with the physical properties of the medium, leading to various types of signal degradation. One of the most fundamental types of degradation is attenuation.

Solution:

1. **Attenuation** is the general term for the reduction in the strength (amplitude) of a signal as it propagates through a medium. 2. It occurs because the medium resists the flow of energy. For example, in electrical cables, some energy is lost as heat due to the resistance of the copper. In wireless transmission, the signal spreads out over a larger area, getting ”thinner” or weaker. 3. Attenuation is measured in Decibels (dB). 4. If attenuation becomes too high, the receiving device will be unable to distinguish the signal from background noise, leading to data errors. 5. To counter attenuation, networking systems use **Repeaters** (for digital signals) or **Amplifiers** (for analog signals) to boost the signal strength at regular intervals over long distances. 6. It is distinct from ”Noise” (unwanted extra interference) or ”Distortion” (change in the shape of the signal).

Final Answer: Attenuation refers to the loss of signal strength over distance.

Answer: (B)

Q34.

Solution

Concept:

In a **Star Topology**, all devices (nodes) are connected to a central connection point, such as a hub, switch, or router. The central device acts as a server, while the connected nodes act as clients. The capacity of a star network is physically limited by the number of connection points (ports) available on the central device.

Solution:

1. In the given scenario, the central device is a switch with 24 ports. 2. Each port on a switch is designed to host one physical connection via a network cable (like an Ethernet/RJ45 cable). 3. Therefore, if there are 24 ports, you can plug in exactly 24 different cables. 4. Each cable leads to one node (a computer, printer, or server). 5. Thus, the maximum number of nodes that can be directly connected to this central switch to form a star topology is 24. 6. While you can ”daisy-chain” switches to add more nodes, a single-switch star topology is strictly limited by its port count. 7. If you have 25 nodes, one node would be left without a connection unless an additional switch is introduced.

Final Answer: The maximum number of nodes is 24.

Answer: (C)

Q35.

Solution

Concept:

When opening a file in Python using the ‘open()‘ function, the ”mode” determines what actions (read, write, append) are allowed and how the file is treated (text vs binary). - ‘r‘: Read - ‘w‘: Write (overwrites existing) - ‘a‘: Append - ‘b‘: Binary mode - ‘+‘: Opens a file for updating (both reading and writing)

Solution:

1. The requirement is two-fold: The mode must support **both reading and writing**, and it must handle **binary** data. 2. Let’s analyze the options: - ‘rw‘: This is not a valid mode in Python; it will trigger an error. - ‘w+‘: This allows reading and writing, but for **text** files. It also truncates (erases) the file as soon as it is opened. - ‘a+‘: This allows reading and appending, but for text files. - ‘rb+‘: This is the correct combination. The ‘r‘ stands for reading, the ‘b‘ stands for binary, and the ‘+‘ adds the ”write” capability to the ”read” mode. 3. Using ‘rb+‘ is safer than ‘wb+‘ because ‘rb+‘ starts at the beginning of the file without deleting the existing content, whereas ‘wb+‘ would immediately wipe the binary file clean upon opening.

Final Answer: The correct mode is ’rb+’.

Answer: (B)

Q36.

Solution

Concept:

In the relational database model, tables (relations) have specific properties that describe their structure. Two fundamental metrics used to define the ”size” or ”shape” of a relation are **Cardinality** and **Degree**. Cardinality refers to the number of tuples (rows), while Degree refers to the number of attributes (columns).

Solution:

1. When we design a table, we define various headers such as `Roll_No`, `Name`, `Class`, and `Marks`. 2. Each of these headers represents an attribute of the entity we are storing in the database. 3. The total count of these attributes or columns is formally called the **Degree** of the relation. 4. For example, if a table has 5 columns, its degree is 5. If you add a new column using the `ALTER TABLE` command, the degree of the relation increases by one. 5. It is important not to confuse this with **Cardinality**, which changes every time you `INSERT` or `DELETE` a row. 6. In most database environments, the Degree is relatively static as it is fixed during the schema design phase, whereas Cardinality is highly dynamic. 7. Therefore, in the context of the question, the ”Degree” strictly refers to the total number of columns present in the table.

Final Answer: The Degree refers to the total number of columns.

Answer: (B)

Q37.

Solution

Concept:

Evaluating a postfix expression (Reverse Polish Notation) involves using a stack data structure. The rule is: iterate through the expression from left to right; if an operand (number) is encountered, push it onto the stack; if an operator is encountered, pop the required number of operands, perform the operation, and push the result back onto the stack.

Solution:

1. The expression is: ‘5 2 3 * + 8 -‘ 2. **Step 1:** Read ‘5‘. It is an operand. Stack: ‘[5]‘ 3. **Step 2:** Read ‘2‘. It is an operand. Stack: ‘[5, 2]‘ 4. **Step 3:** Read ‘3‘. It is an operand. Stack: ‘[5, 2, 3]‘ 5. **Step 4:** Read ‘*‘ (Operator). Pop ‘3‘ and ‘2‘. Calculate $2 \times 3 = 6$. Push ‘6‘. Stack: ‘[5, 6]‘ 6. **Step 5:** Read ‘+‘ (Operator). Pop ‘6‘ and ‘5‘. Calculate $5 + 6 = 11$. Push ‘11‘. Stack: ‘[11]‘ 7. **Step 6:** Read ‘8‘. It is an operand. Stack: ‘[11, 8]‘ 8. **Step 7:** Read ‘-‘ (Operator). Pop ‘8‘ and ‘11‘. Calculate $11 - 8 = 3$. Push ‘3‘. Stack: ‘[3]‘ 9. The final value remaining on the stack is the result.

Final Answer: The value of the expression is 3.

Answer: (A)

Q38.

Solution

Concept:

The ‘ORDER BY‘ clause in SQL is used to sort the result-set of a query. By default, ‘ORDER BY‘ sorts the data in ascending order. If a specific order is required, keywords must be appended to the column name in the clause.

Solution:

1. To arrange data from highest to lowest (descending), we use the keyword **‘DESC‘**. 2. To arrange data from lowest to highest (ascending), we use the keyword **‘ASC‘** (though this is optional as it is the default behavior). 3. The correct syntax is: ‘SELECT * FROM TableName ORDER BY ColumnName DESC;‘ 4. Let’s look at the options: - ‘SORT BY‘ is used in languages like Hive or Python/Pandas but is not standard SQL for this purpose. - ‘GROUP BY‘ is used for aggregating data, not for simple sorting of the final output. - ‘ARRANGE BY‘ is not a valid SQL command. 5. Therefore, ‘ORDER BY ... DESC‘ is the only syntactically correct way to achieve a descending sort in a relational database.

Final Answer: The clause is ORDER BY ... DESC.

Answer: (B)

Q39.

Solution

Concept:

Network topologies define the layout of connections between devices. A **Bus Topology** is a specific type of network where all nodes are connected to a single shared communication line, known as the ”backbone” or ”bus.”

Solution:

1. In a Bus Topology, the central cable acts as a shared medium (multipoint connection). 2. As signals travel down the cable, they reach every node. If the signal is not intended for a node, that node ignores it. 3. A critical component of this topology is the **Terminator**. 4. Terminators are placed at both physical ends of the main cable. Their job is to absorb the signal once it reaches the end, preventing it from reflecting back (echoing) and causing interference or data collisions. 5. If a terminator is missing or the cable is cut, the entire network fails because the signals will bounce back and corrupt new data transmissions. 6. Other topologies like Star or Ring do not use terminators in this manner; Star uses a central hub, and Ring is a continuous loop.

[Image of bus topology with terminators]

Final Answer: The topology is Bus Topology.

Answer: (B)

Q40.

Solution

Concept:

The ‘range()‘ function in Python generates a sequence of numbers. It takes three arguments: ‘range(start, stop, step)‘. - ‘start‘: The first number in the sequence (inclusive). - ‘stop‘: The sequence stops before reaching this number (exclusive). - ‘step‘: The difference between each number in the sequence.

Solution:

1. The function call is ‘range(5, 1, -1)‘. 2. ‘start = 5‘: The sequence begins at 5. 3. ‘step = -1‘: This is a negative step, meaning the sequence will count downwards. 4. ‘stop = 1‘: The sequence will continue as long as the numbers are greater than 1. It will **not** include 1. 5. Let’s trace the generation: - First number: 5 - Next (5 – 1): 4 - Next (4 – 1): 3 - Next (3 – 1): 2 - Next (2 – 1): 1 → Stop! (The stop value is exclusive). 6. The resulting sequence is ‘5, 4, 3, 2‘. 7. Converting this to a list using ‘list()‘ results in the output ‘[5, 4, 3, 2]‘. 8. Option (A) is incorrect because it includes the stop value. Option (D) is incorrect because it misses the start value.

Final Answer: The output is [5, 4, 3, 2].

Answer: (B)

Q41.

Solution

Concept:

In SQL, string length functions are used to determine the size of a string. However, different functions measure this size in different units. The two most common functions are `CHAR_LENGTH()` (or `CHARACTER_LENGTH()`), which counts the number of characters regardless of how many bytes each character uses, and `LENGTH()`, which measures the actual storage size in memory.

Solution:

1. For standard English characters (ASCII), one character usually equals one byte. However, for multi-byte characters (like Unicode/UTF-8 symbols or Kanji), a single character might occupy 2, 3, or 4 bytes. 2. The `CHAR_LENGTH()` function returns the count of characters. If you have a 5-letter word in a multi-byte encoding, it returns 5. 3. The `LENGTH()` function, in most SQL dialects like MySQL, returns the length of the string measured in **bytes**. 4. This is a critical distinction for database administrators who need to manage disk space or network bandwidth, especially when dealing with internationalization and different character sets. 5. Other options like `SIZE()` or `BYTES()` are not standard SQL functions for string measurement; `SIZE` is more commonly associated with file systems or specific programming languages like C++ or Python’s `sys.getsizeof()` method. 6. Therefore, when the requirement is to find the length specifically in bytes, `LENGTH()` is the correct internal function to call.

Final Answer: The function is LENGTH().

Answer: (B)

Q42.

Solution

Concept:

The evaluation of expressions (Postfix, Prefix, or Infix) using a stack relies on the ”Arity” of the operators. Arity refers to the number of operands an operator requires to perform its calculation. Most common arithmetic operators ($+, -, *, /$) are **binary operators**, meaning they operate on exactly two values.

Solution:

1. In the algorithm for evaluating a postfix expression, we scan the expression from left to right. 2. Operands (numbers) are pushed onto the stack to wait for an operator. 3. When a binary operator (like + or $*$) is encountered, it must have two numbers to work with. 4. Therefore, the algorithm **pops exactly two elements** from the top of the stack. 5. The first element popped is treated as the second operand (Right Operand), and the second element popped is treated as the first operand (Left Operand). This order is vital for non-commutative operations like subtraction or division. 6. After performing the calculation, the single resulting value is pushed back onto the stack. 7. If an operator were ”unary” (like logical ‘NOT‘ or unary minus), it would pop only one element, but standard postfix evaluation questions in CUET focus on binary arithmetic.

Final Answer: 2 operands are popped from the stack.

Answer: (B)

Q43.

Solution

Concept:

Switching is the process used to move data through a network of intermediate nodes. There are three primary types: 1. **Circuit Switching:** A dedicated physical path is established for the duration of the session (e.g., traditional landline phones). 2. **Message Switching:** The entire data unit is sent as a whole from node to node (Store-and-Forward). 3. **Packet Switching:** Data is broken down into small, manageable units called packets.

Solution:

1. In **Packet Switching**, the original message is divided into segments. Each segment (packet) is given a header containing the source address, destination address, and a sequence number. 2. These packets are sent independently into the network. 3. Because they are independent, different packets belonging to the same message might take different physical routes to reach the destination based on current network traffic or link failures. 4. Once all packets arrive at the destination, the receiving device uses the sequence numbers in the headers to reassemble them into the original message. 5. This is the foundation of the modern Internet (TCP/IP). It is highly efficient because it does not require a dedicated circuit, allowing many users to share the same communication lines simultaneously. 6. Circuit switching is inefficient for data because the line remains ”busy” even when no data is being sent. Packet switching solves this by using the bandwidth only when there is an actual packet to transmit.

Final Answer: The technique is Packet Switching.

Answer: (C)

Q44.

Solution

Concept:

The ‘DAYOFYEAR()‘ function in SQL returns the day of the year for a given date, ranging from 1 to 366. To solve this, we must count the total number of days passed in the year up to the specified date, keeping in mind whether the year is a leap year.

Solution:

1. The date provided is ‘’2024-02-01’‘. 2. First, we identify the month and day: It is the 1st day of February. 3. To find the total days from the start of the year: - Days in January: 31 - Days in February: 1 4. Total = 31 + 1 = 32. 5. Note: Although 2024 is a leap year (February has 29 days), it does not affect the calculation for February 1st, as the extra ”leap day” (Feb 29) occurs later in the month. 6. If the date had been March 1st, 2024, the result would have been $31(\text{Jan}) + 29(\text{Feb}) + 1(\text{Mar}) = 61$. For a non-leap year, March 1st would be the 60th day. 7. Since we are only at the start of February, the count is simply the full month of January plus the first day of February.

Final Answer: The output is 32.

Answer: (B)

Q45.

Solution

Concept:

Exception handling in Python is organized in a class hierarchy. Most built-in exceptions are derived from the ‘Exception‘ class, which in turn derives from ‘BaseException‘. Understanding the parent-child relationship of exceptions is important for ”catching” multiple related errors with a single ‘except‘ block.

Solution:

1. ‘FileNotFoundError‘ is raised when a file or directory is requested but doesn’t exist. 2. In older versions of Python (pre-3.3), various file-related errors were distinct. In modern Python, several of these have been consolidated. 3. ‘FileNotFoundError‘ is a specific child (subclass) of **‘OSError‘**. 4. In many contexts and older documentation, you will see it referred to as a subclass of **‘IOError‘**. In Python 3, ‘IOError‘ is actually just an alias for ‘OSError‘. 5. Therefore, if you write a try-except block and catch ‘OSError‘, it will also catch ‘FileNotFoundError‘, ‘PermissionError‘, and ‘TimeoutError‘. 6. Other options like ‘RuntimeError‘ or ‘ValueError‘ are for different types of logic or calculation errors. ‘ImportError‘ is specifically for issues when a module cannot be loaded. 7. Thus, ‘FileNotFoundError‘ inherits its properties from the Input/Output and Operating System error category.

Final Answer: It is a subclass of IOError (or OSError).

Answer: (B)

Q46.

Solution

Concept:

Transmission media are the physical pathways used to carry information from a sender to a receiver. They are categorized into Guided (wired) and Unguided (wireless) media. Within guided media, the physical construction—such as the thickness of the conductor and the quality of the insulation—determines the maximum data transfer rate (bandwidth) and the resistance to external interference.

Solution:

1. **Fiber Optic:** The fastest medium, using light pulses through glass threads. It offers extremely high bandwidth. 2. **Coaxial Cable:** Used for cable TV and high-speed internet, it is faster and more shielded than twisted pair but slower than fiber. 3. **Twisted Pair (Unshielded/UTP):** This is the standard "Ethernet" cable found in most homes and offices (like Cat5 or Cat6). While it is cheap and easy to install, it has the lowest bandwidth capacity and is the most susceptible to electromagnetic interference (EMI) among the guided options. 4. **Satellite Link:** While the propagation delay is high (due to the distance to orbit), the actual transmission speed (bandwidth) of modern satellites is significantly higher than that of a simple unshielded twisted pair copper wire. 5. In terms of raw data transfer capabilities over a single link, the Unshielded Twisted Pair (UTP) is generally considered the slowest and most limited compared to the specialized engineering of coaxial, satellite, or fiber optic technologies.

Final Answer: The slowest transmission medium among the guided options is Twisted Pair.

Answer: (C)

Q47.

Solution

Concept:

Referential Integrity is a database concept that ensures that relationships between tables remain consistent. When one table has a foreign key that points to a primary key in another table, referential integrity rules prevent any action that would ”break” that link, such as deleting a record that is still being pointed to by other data.

Solution:

1. Consider two tables: ‘Department‘ (Parent) and ‘Employee‘ (Child). The ‘DeptID‘ in the Employee table is a **Foreign Key**. 2. If you try to delete a department from the ‘Department‘ table that still has employees assigned to it in the ‘Employee‘ table, the database will block the deletion. 3. This is because the employees would be left pointing to a ‘DeptID‘ that no longer exists (creating ”orphaned” records). 4. The constraint responsible for this check is the **Foreign Key** constraint, which enforces **Referential Integrity**. 5. Other constraints have different roles: ‘UNIQUE‘ ensures no duplicates; ‘CHECK‘ ensures values meet a logical condition (e.g., Age > 18); ‘PRIMARY KEY‘ ensures unique identification of a row. 6. Therefore, the restriction on deleting referenced records is a direct implementation of the Foreign Key constraint.

Final Answer: The constraint is FOREIGN KEY (Referential Integrity).

Answer: (C)

Q48.

Solution

Concept:

The efficiency of sorting algorithms is often measured in terms of the number of comparisons and swaps. Most simple sorting algorithms like Bubble Sort and Selection Sort have a time complexity of $O(n^2)$. However, an **optimized** version of Bubble Sort can detect if the list becomes sorted before all passes are completed.

Solution:

1. A standard Bubble Sort always makes $n - 1$ passes, each with multiple comparisons. 2. In the **Optimized Bubble Sort**, a ”flag” (a boolean variable) is used to track whether any swaps occurred during a pass. 3. If the list is already sorted (Best Case), the algorithm will perform one full pass (making $n - 1$ comparisons) and see that no swaps were made. 4. Because the flag remains unchanged, the algorithm terminates immediately after the first pass. 5. In Big O notation, $n - 1$ comparisons are represented as $**O(n)**$. 6. Without this optimization, Bubble Sort would continue to perform all passes, resulting in $O(n^2)$ even for a sorted list. 7. Therefore, the best-case complexity for the optimized version is linear, or $O(n)$.

Final Answer: The best-case time complexity is $O(n)$.

Answer: (C)

Q49.

Solution

Concept:

Python's 'str' class provides numerous methods for manipulating and analyzing text. To convert a single long string into a structured format like a list, we use a method that recognizes a specific character as a "separator."

Solution:

1. The method is **'split()'**. 2. By default, 'split()' uses whitespace (spaces, tabs, newlines) as the delimiter. For example, '"Hello World".split()' returns '['Hello', 'World']'. 3. You can also specify a custom delimiter, such as a comma for CSV data: '"A,B,C".split(',')' returns '['A', 'B', 'C']'. 4. The other options: - 'divide()' is not a Python string method. - 'break()' is a keyword used to exit loops, not for string manipulation. - 'partition()' is similar to split but it only splits the string at the *first* occurrence of the delimiter and returns a 3-item tuple (head, delimiter, tail) rather than a list of all words. 5. For the task of breaking a string into all available words or segments, 'split()' is the standard and most versatile tool.

Final Answer: The method is split().

Answer: (B)

Q50.

Solution

Concept:

IP addresses are divided into Public and Private addresses. Public addresses are globally unique and reachable over the internet. Private addresses are reserved for use within local networks (LANs) and are not routable on the public internet. The Internet Assigned Numbers Authority (IANA) reserved three specific ranges for private use.

Solution:

1. The reserved private IP ranges are: - Class A: 10.0.0.0 to 10.255.255.255 - Class B: 172.16.0.0 to 172.31.255.255 - Class C: 192.168.0.0 to 192.168.255.255 2. Looking at the options: - **172.16.0.1** falls exactly within the Class B private range. - 8.8.8.8 is a famous public DNS address belonging to Google. - 192.0.0.1 and 11.0.0.1 fall outside the reserved private boundaries and are treated as public addresses. 3. Private IP addresses allow multiple devices in a home or office to share a single public IP address using a process called NAT (Network Address Translation). 4. Therefore, 172.16.0.1 is the only private address in the list.

Final Answer: The private IP address is 172.16.0.1.

Answer: (A)

Answer Key

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