

# CUET UG Computer Science Sample Paper -14

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 correctly calculates the number of full years of service for an employee based on a column `JoinDate`, assuming the current date is retrieved using a standard SQL function?

- (A) `SELECT YEAR(CURDATE()) - YEAR(JoinDate);`  
(B) `SELECT TIMESTAMPDIFF(YEAR, JoinDate, CURDATE());`  
(C) `SELECT (CURDATE() - JoinDate) / 365;`  
(D) `SELECT DATEDIFF(YEAR, CURDATE(), JoinDate);`

**Q2.** Consider an Infix expression  $A \uparrow B * C - D + E / F$ . What is the correct Postfix equivalent, given that  $\uparrow$  represents the exponentiation operator with the highest precedence and right-to-left associativity?

- (A)  $AB \uparrow C * D - EF / +$   
(B)  $AB \uparrow C * D - E + F /$   
(C)  $ABC * \uparrow D - EF / +$   
(D)  $AB \uparrow C * DEF / + -$

**Q3.** A database relation  $R(A, B, C, D, E)$  has the functional dependencies:  $\{A \rightarrow B, BC \rightarrow D, E \rightarrow C, D \rightarrow A\}$ . Which of the following is NOT a candidate key for  $R$ ?



- (A) {A, E}
- (B) {B, E}
- (C) {E, D}
- (D) {B, C, E}

**Q4.** In Python, what will be the state of the file 'data.txt' after the following code execution?

```
with open("data.txt", "w+") as f:  
    f.write("Pythonic")  
    f.seek(0)  
    f.write("Java")  
    f.seek(2)  
    f.write("Script")
```

- (A) JavaScript
- (B) JaScriptic
- (C) JavaicScript
- (D) JaScript

**Q5.** A network uses a Subnet Mask of 255.255.255.224. What is the maximum number of usable host IP addresses available in each subnet?

- (A) 32
- (B) 30
- (C) 62
- (D) 254

**Q6.** Which SQL function would you use to return the value of a string where only the first character of each word is capitalized?

- (A) PROPER()
- (B) INITCAP()



(C) CAPITALIZE()

(D) UPPERCASE()

**Q7.** In a Circular Queue implemented with an array of size  $N$ , if `front` and `rear` are the indices of the first and last elements respectively, which expression correctly calculates the current number of elements in the queue?

(A)  $(\text{rear} - \text{front} + N) \% N$

(B)  $(\text{rear} - \text{front} + 1 + N) \% N$

(C)  $\text{rear} - \text{front} + 1$

(D)  $(\text{front} - \text{rear} + N) \% N$

**Q8.** What is the time complexity of the best-case scenario for the Quick Sort algorithm, and what condition triggers it?

(A)  $O(n^2)$ , when the array is already sorted.

(B)  $O(n \log n)$ , when the pivot always divides the list into two nearly equal halves.

(C)  $O(n)$ , when all elements are identical.

(D)  $O(n \log n)$ , when the pivot is always the smallest element.

**Q9.** An organization wants to connect two different departments located in the same building using a device that filters traffic based on MAC addresses to reduce collisions. Which device should they use?

(A) Hub

(B) Repeater

(C) Bridge

(D) Router

**Q10.** In Python, which exception is specifically raised when a module is found but a specific attribute or function within that module does not exist?



- (A) ImportError
- (B) ModuleNotFoundError
- (C) AttributeError
- (D) NameError

**Q11.** Which Relational Algebra operator is used to find tuples that are present in relation  $R$  but not in relation  $S$ , assuming both relations are union-compatible?

- (A) Intersection ( $\cap$ )
- (B) Set Difference ( $-$ )
- (C) Cartesian Product ( $\times$ )
- (D) Division ( $\div$ )

**Q12.** In SQL, what is the result of the expression `SELECT SUBSTR('COMPUTER SCIENCE', 10, 4);`

- (A) SCIE
- (B) R SC
- (C) ENCI
- (D) ER S

**Q13.** A programmer is using a Bubble Sort to sort an array of 10 elements. If the array is already sorted in descending order and we want to sort it in ascending order, how many total comparisons will be made?

- (A) 10
- (B) 45
- (C) 90
- (D) 100

**Q14.** Which of the following Python statements will correctly append a dictionary  $D = \{ 'ID': 101 \}$  to a binary file 'records.dat' without overwriting previous data?



- (A) `pickle.dump(D, open('records.dat', 'wb'))`
- (B) `pickle.dump(D, open('records.dat', 'ab'))`
- (C) `pickle.write(D, open('records.dat', 'a'))`
- (D) `f = open('records.dat', 'rb+'); f.append(D)`

**Q15.** Which network protocol is used by a host to discover its own IP address when it only knows its physical MAC address at startup?

- (A) ARP
- (B) RARP
- (C) ICMP
- (D) SMTP

**Q16.** Consider a circular queue implemented using an array of size 6. If the current status is `front = 2` and `rear = 5` (assuming 0-indexing), what will be the new values of `front` and `rear` after performing two dequeue operations and one enqueue operation?

- (A) `front = 4, rear = 0`
- (B) `front = 3, rear = 0`
- (C) `front = 4, rear = 1`
- (D) `front = 0, rear = 4`

**Q17.** Which of the following Relational Algebra expressions correctly finds the names of students who have enrolled in both 'Physics' and 'Chemistry' courses? (Assume `Enroll(Sname, Cname)`)

- (A)  $\Pi_{Sname}(\sigma_{Cname='Physics' \vee Cname='Chemistry'}(Enroll))$
- (B)  $\Pi_{Sname}(\sigma_{Cname='Physics'}(Enroll)) \cup \Pi_{Sname}(\sigma_{Cname='Chemistry'}(Enroll))$
- (C)  $\Pi_{Sname}(\sigma_{Cname='Physics'}(Enroll)) \cap \Pi_{Sname}(\sigma_{Cname='Chemistry'}(Enroll))$
- (D)  $\sigma_{Cname='Physics' \wedge Cname='Chemistry'}(\Pi_{Sname}(Enroll))$



**Q18.** In Python, what will be the output of the following code snippet?

```
import math
print(math.ceil(-4.7) + math.floor(3.8) + math.trunc(-2.9))
```

- (A) -3
- (B) -4
- (C) -5
- (D) 2

**Q19.** Identify the correct SQL command to increase the price of all products by 10% only if the current price is less than ₹ 500.

- (A) UPDATE Product SET Price = Price + 0.1 WHERE Price < 500;
- (B) UPDATE Product SET Price = Price \* 1.1 WHERE Price < 500;
- (C) ALTER Product SET Price = Price \* 1.1 WHERE Price < 500;
- (D) MODIFY Product SET Price = Price \* 1.1 IF Price < 500;

**Q20.** Which network topology requires a central controller or hub, and if the central hub fails, the entire network goes down, but a failure in one single node does not affect others?

- (A) Mesh Topology
- (B) Bus Topology
- (C) Star Topology
- (D) Ring Topology

**Q21.** A binary file 'marks.dat' contains objects of a class Student. What is the correct way to read all objects from this file until the end of the file is reached?

- (A) Use `while f.read():` and `pickle.load(f)`
- (B) Use `while True:` with `pickle.load(f)` inside a `try-except EOFError` block.



- (C) Use `for x in f: with pickle.load(f)`
- (D) Use `pickle.readlines(f)` and iterate through the list.

**Q22.** Which of the following is a "Divide and Conquer" sorting algorithm that has a worst-case time complexity of  $O(n \log n)$ ?

- (A) Selection Sort
- (B) Quick Sort
- (C) Merge Sort
- (D) Bubble Sort

**Q23.** What is the output of the following SQL query?

```
SELECT ROUND(153.769, -1), ROUND(153.769, 1);
```

- (A) 150, 153.8
- (B) 153.8, 150
- (C) 154, 153.7
- (D) 160, 153.8

**Q24.** In the context of network security, which of the following is a type of attack where a malicious user redirects a website's traffic to another, fake website by modifying the host file or DNS entries?

- (A) Phishing
- (B) Pharming
- (C) Spying
- (D) Virus

**Q25.** What is the postfix representation of the infix expression:  $A + (B * C - (D / E \uparrow F) * G) * H$

- (A)  $ABC*DEF\uparrow/G*-H*+$



- (B)  $ABC*DEF/\uparrow G*-H*+$
- (C)  $A B C * D E F \uparrow / G * - H * +$
- (D)  $A B C * D E F / \uparrow G * - H * +$

**Q26.** Which SQL string function returns the position of the first occurrence of a substring within a string?

- (A) LOCATE()
- (B) POSITION()
- (C) INSTR()
- (D) All of the above

**Q27.** Consider a stack implemented as a Python list `Stack = [10, 20, 30]`. If we perform the operations `Stack.append(40)`, `Stack.pop()`, `Stack.pop()`, and `Stack.append(50)`, what will be the final content of the list?

- (A) [10, 20, 50]
- (B) [10, 50]
- (C) [10, 20, 30, 50]
- (D) [50, 20, 10]

**Q28.** In a relational database, what is the 'Referential Integrity' constraint?

- (A) Every table must have a primary key.
- (B) No primary key attribute can be null.
- (C) A foreign key value must match a primary key value in the related table or be null.
- (D) Every attribute must have a unique name within a table.

**Q29.** Which protocol is used to securely log into a remote computer, providing encrypted communication over an unsecure network?



- (A) Telnet
- (B) SSH
- (C) FTP
- (D) HTTP

**Q30.** What is the best-case time complexity of the Insertion Sort algorithm, and when does it occur?

- (A)  $O(n)$ , when the array is already sorted.
- (B)  $O(n^2)$ , when the array is sorted in reverse.
- (C)  $O(n \log n)$ , when the array is unsorted.
- (D)  $O(1)$ , when the array has only one element.

**Q31.** Which Python function is used to convert a single character to its corresponding integer ordinal value (ASCII/Unicode)?

- (A) `chr()`
- (B) `ord()`
- (C) `int()`
- (D) `ascii()`

**Q32.** In SQL, which clause is used to filter the results returned by a GROUP BY clause based on a specific condition?

- (A) WHERE
- (B) ORDER BY
- (C) HAVING
- (D) SELECT

**Q33.** Which switching technique divides data into small pieces of fixed or variable length and routes them independently through the network?



- (A) Circuit Switching
- (B) Message Switching
- (C) Packet Switching
- (D) Frequency Switching

**Q34.** What is the result of the following Python expression?

`3 * 1 ** 3 + 2 ** 2 ** 3`

- (A) 67
- (B) 259
- (C) 4099
- (D) 257

**Q35.** In SQL, which join returns all rows when there is a match in one of the tables?

- (A) INNER JOIN
- (B) LEFT JOIN
- (C) RIGHT JOIN
- (D) FULL OUTER JOIN

**Q36.** What is the default port number for the HTTP protocol?

- (A) 21
- (B) 25
- (C) 80
- (D) 443

**Q37.** Which of the following is NOT a valid exception handling keyword in Python?

- (A) try
- (B) except
- (C) finally



(D) throw

**Q38.** In relational algebra, the  $\sigma$  (sigma) operator is used for:

- (A) Projecting columns
- (B) Selecting rows based on a condition
- (C) Joining two tables
- (D) Renaming attributes

**Q39.** Which sorting algorithm works by repeatedly finding the minimum element from the unsorted part and putting it at the beginning?

- (A) Bubble Sort
- (B) Insertion Sort
- (C) Selection Sort
- (D) Merge Sort

**Q40.** Which protocol is used for sending electronic mail between servers?

- (A) POP3
- (B) IMAP
- (C) SMTP
- (D) SNMP

**Q41.** Consider the following Python code that handles a binary file containing employee records. What will be the output if the file 'emp.dat' is empty?

```
import pickle
try:
with open('emp.dat', 'rb') as f:
data = pickle.load(f)
print(data)
except EOFError:
```



```

print("End of File")
except FileNotFoundError:
print("File Missing")
finally:
print("Process Over")

```

- (A) End of File
- (B) File Missing  
Process Over
- (C) End of File  
Process Over
- (D) Process Over

**Q42.** Which of the following Relational Algebra expressions correctly represents the "Division" ( $\div$ ) operation, used to find tuples in  $R$  that are associated with ALL tuples in  $S$ ?

- (A)  $\Pi_{R-S}(R) - \Pi_{R-S}((\Pi_{R-S}(R) \times S) - R)$
- (B)  $\Pi_{R-S}(R) \cup \Pi_{R-S}((\Pi_{R-S}(R) \times S) \cup R)$
- (C)  $\Pi_{R-S}(R \times S) - \Pi_{R-S}(R)$
- (D)  $\sigma_{R=S}(R \times S)$

**Q43.** In a networking environment, which field of an IPv4 packet header is decremented by every router it passes through, and what happens when this value reaches zero?

- (A) Checksum, the packet is recalculated.
- (B) Time to Live (TTL), the packet is discarded and an ICMP message is sent.
- (C) Fragment Offset, the packet is reassembled.
- (D) Identification, the packet is assigned a new IP.



- Q44.** Which SQL aggregate function is used to return the average value of a numeric column, and how does it handle NULL values in that column?
- (A) `AVG()`, it treats NULL as 0.
  - (B) `AVERAGE()`, it ignores NULL values.
  - (C) `AVG()`, it ignores NULL values.
  - (D) `MEAN()`, it returns NULL if any value is NULL.
- Q45.** Identify the correct sequence of steps involved in the "Insertion Sort" algorithm for an element at index  $i$ .
- (A) Compare  $A[i]$  with  $A[i + 1]$ , swap if  $A[i] > A[i + 1]$ .
  - (B) Find minimum in  $A[i...n]$ , swap with  $A[i]$ .
  - (C) Pick  $A[i]$ , compare it with elements in  $A[0...i - 1]$  and shift elements right to find correct position.
  - (D) Divide array into two halves, sort and merge.
- Q46.** A Stack data structure is defined by the LIFO principle. If the sequence of operations is `Push(A)`, `Push(B)`, `Pop()`, `Push(C)`, `Pop()`, which element is currently at the top of the stack?
- (A) A
  - (B) B
  - (C) C
  - (D) None, the stack is empty.
- Q47.** In a Binary Search Tree (BST), which type of traversal will always yield the node values in a strictly increasing (ascending) sorted order?
- (A) Pre-order Traversal
  - (B) Post-order Traversal
  - (C) In-order Traversal



(D) Level-order Traversal

**Q48.** In a relational database, if table 'Student' has RollNo as Primary Key and table 'Result' has RollNo referencing it, what is the 'RollNo' column in the 'Result' table called?

(A) Candidate Key

(B) Foreign Key

(C) Composite Key

(D) Alternate Key

**Q49.** Which layer of the OSI model is responsible for establishing, maintaining, and terminating synchronization between communicating applications (sessions)?

(A) Physical Layer

(B) Session Layer

(C) Network Layer

(D) Data Link Layer

**Q50.** Which sorting algorithm is characterized by comparing adjacent elements and swapping them if they are in the wrong order, causing the largest element to "sink" to the end in each pass?

(A) Selection Sort

(B) Insertion Sort

(C) Bubble Sort

(D) Merge Sort



## Detailed Solutions

Q1.

### Solution

**Concept:**

To calculate the difference between two dates in terms of specific units like years, months, or days, SQL provides specialized date-time functions. The `TIMESTAMPDIFF()` function is widely used in MySQL and other SQL dialects to find the difference between two date or datetime expressions. It takes three arguments: the unit (e.g., `YEAR`, `MONTH`, `DAY`), the start date, and the end date. Unlike simple subtraction or `YEAR()` subtraction, `TIMESTAMPDIFF` accounts for the full calendar year, ensuring the anniversary has passed.

**Solution:**

1. The goal is to find "full years of service," which means we need the difference between `JoinDate` and the current date (`CURDATE()`). 2. Option (A) `YEAR(CURDATE()) - YEAR(JoinDate)` simply subtracts the year integers. If someone joined in December 2022 and today is January 2023, this would return 1 year, even though they only served one month. This is inaccurate for "full years." 3. Option (B) uses `TIMESTAMPDIFF(YEAR, JoinDate, CURDATE())`. This function calculates the total number of complete years that have passed between the two dates. 4. Option (C) performs arithmetic on dates, which is not standard SQL behavior and can lead to errors depending on the RDBMS and leap year logic. 5. Option (D) `DATEDIFF()` in many systems (like MySQL) only returns the difference in days, and its syntax usually doesn't accept a "YEAR" unit parameter in that specific way. 6. Therefore, the most precise and standard function for full year calculation is `TIMESTAMPDIFF`.

**Final Answer:** The correct expression is `TIMESTAMPDIFF(YEAR, JoinDate, CURDATE())`.

**Answer: (B)**



Q2.

**Solution****Concept:**

Converting Infix to Postfix follows operator precedence and associativity rules. The precedence for this expression is: 1. Parentheses (none here) 2. Exponentiation ( $\uparrow$ ): Highest precedence, Right-to-Left associativity. 3. Multiplication/Division ( $*$ ,  $/$ ): Left-to-Right. 4. Addition/Subtraction ( $+$ ,  $-$ ): Left-to-Right.

**Solution:**

1. Original expression:  $A \uparrow B * C - D + E / F$  2. Step 1 (Exponentiation):  $(A \uparrow B)$  has the highest priority. Postfix:  $AB \uparrow$ . The expression becomes  $(AB \uparrow) * C - D + E / F$ . 3. Step 2 (Multiplication):  $(AB \uparrow * C)$  is processed next. Postfix:  $AB \uparrow C *$ . The expression is now  $(AB \uparrow C *) - D + E / F$ . 4. Step 3 (Division): Between  $-$  and  $+$ , and  $/$ , the division  $E / F$  has higher priority. Postfix:  $EF /$ . The expression is  $(AB \uparrow C *) - D + (EF /)$ . 5. Step 4 (Subtraction/Addition): These have the same priority, so we move Left-to-Right. First, handle the subtraction:  $(AB \uparrow C * - D)$ . Postfix:  $AB \uparrow C * D -$ . 6. Step 5 (Final Addition): Finally, add the division result:  $(AB \uparrow C * D -) + (EF /)$ . Postfix:  $AB \uparrow C * D - EF / +$ .

**Final Answer:** The Postfix equivalent is  $AB \uparrow C * D - EF / +$ .

**Answer: (A)**

Q3.

**Solution****Concept:**

A Candidate Key is a minimal set of attributes that can uniquely identify every tuple in a relation. To find a candidate key, we calculate the attribute closure (denoted as  $X^+$ ) for a set of attributes. If the closure includes all attributes of the relation  $\{A, B, C, D, E\}$ , it is a Super Key. If no subset of that key is also a super key, it is a Candidate Key.

**Solution:**

Given FDs:  $A \rightarrow B, BC \rightarrow D, E \rightarrow C, D \rightarrow A$ . 1. Check  $\{A, E\}$ :  $-\{A, E\}^+ = \{A, E\}$  (starting)  $- A \rightarrow B \implies \{A, E, B\} - E \rightarrow C \implies \{A, E, B, C\} - BC \rightarrow D \implies \{A, E, B, C, D\}$ . Contains all attributes. This is a key. 2. Check  $\{B, E\}$ :  $-\{B, E\}^+ = \{B, E\} - E \rightarrow C \implies \{B, E, C\} - BC \rightarrow D \implies \{B, E, C, D\} - D \rightarrow A \implies \{B, E, C, D, A\}$ . Contains all attributes. This is a key. 3. Check  $\{E, D\}$ :  $-\{E, D\}^+ = \{E, D\} - D \rightarrow A \implies \{E, D, A\} - A \rightarrow B \implies \{E, D, A, B\} - E \rightarrow C \implies \{E, D, A, B, C\}$ . Contains all attributes. This is a key. 4. Check  $\{B, C, E\}$ :  $-\{B, C, E\}^+ = \{B, C, E, D, A\}$ . While it contains all attributes, we must check for minimality. We already found  $\{B, E\}^+$  is a key. Since  $\{B, E\} \subset \{B, C, E\}$ , the set  $\{B, C, E\}$  is a Super Key but NOT a Candidate Key.

**Final Answer:**  $\{B, C, E\}$  is not a candidate key because it is not minimal.

**Answer: (D)**



Q4.

**Solution****Concept:**

In Python file handling, the `w+` mode opens a file for both writing and reading, overwriting the existing file. The `seek(offset)` method moves the file pointer to a specific byte position. When we write at a specific position, it overwrites the existing characters at that position without shifting the subsequent text.

**Solution:**

1. `open("data.txt", "w+")`: Creates an empty file. 2. `f.write("Pythonic")`: The file now contains "Pythonic". The pointer is at the end (index 8). 3. `f.seek(0)`: Pointer moves to the start (index 0). 4. `f.write("Java")`: "Java" overwrites "Pyth". The file now contains "Javaonic". The pointer is at index 4. 5. `f.seek(2)`: Pointer moves to index 2 (at 'v' in "Javaonic"). 6. `f.write("Script")`: "Script" starts writing from index 2. - Index 2: 'S' replaces 'v' - Index 3: 'c' replaces 'a' - Index 4: 'r' replaces 'o' - Index 5: 'i' replaces 'n' - Index 6: 'p' replaces 'i' - Index 7: 't' replaces 'c' 7. The first two characters "Ja" were not touched. The following characters were replaced by "Script". 8. Result: "JaScript".

**Final Answer:** The final state of the file is "JaScript".

**Answer: (D)**

Q5.

**Solution****Concept:**

The number of usable hosts in a subnet is calculated by the formula  $2^n - 2$ , where  $n$  is the number of bits remaining for the host portion. We subtract 2 because the first address is the Network ID and the last address is the Broadcast ID, neither of which can be assigned to a host.

**Solution:**

1. The Subnet Mask is 255.255.255.224. 2. We look at the last octet: 224. 3. Converting 224 to binary:  $11100000_2$ . 4. In a subnet mask, '1's represent the network/subnet portion and '0's represent the host portion. 5. There are 5 zeros in the last octet (00000). Thus,  $n = 5$ . 6. Total number of addresses in each subnet =  $2^5 = 32$ . 7. Usable host addresses =  $2^n - 2 = 32 - 2 = 30$ .

**Final Answer:** The maximum number of usable host IP addresses is 30.

**Answer: (B)**



Q6.

**Solution****Concept:**

SQL provides several functions to manipulate the case of strings. Standard functions include UPPER() for uppercase, LOWER() for lowercase, and in many systems like Oracle or through specific implementations in others, INITCAP() is used to capitalize the first letter of each word in a string while making the rest of the letters lowercase.

**Solution:**

1. The requirement is to capitalize only the first character of each word. This is often called "Title Case" or "Proper Case." 2. Option (A) PROPER() is a function used in spreadsheet software like Microsoft Excel, but it is not a standard SQL function. 3. Option (B) INITCAP() (Initial Capital) is the standard function in SQL (specifically Oracle and PostgreSQL) designed to convert the first letter of each word to uppercase and the rest to lowercase. 4. Option (C) CAPITALIZE() is not a standard SQL function for this purpose; it is more common in Python string methods. 5. Option (D) UPPER() is incorrect; the standard SQL function for converting to all capitals is simply UPPER(). 6. Since INITCAP() fits the functional description perfectly for multi-word strings, it is the correct choice.

**Final Answer:** The SQL function is INITCAP().

**Answer: (B)**

Q7.

**Solution****Concept:**

A Circular Queue is a linear data structure that logically connects the last position back to the first position to make efficient use of memory. The number of elements in a queue is generally calculated as the difference between the rear and front pointers. However, in a circular implementation, the rear might be at a lower index than the front if the queue has wrapped around.

**Solution:**

1. Let  $N$  be the total size of the array. 2. If  $\text{rear} \geq \text{front}$ , the number of elements is simply  $\text{rear} - \text{front} + 1$ . 3. If  $\text{rear} < \text{front}$ , it means the queue has wrapped around. The number of elements is  $(N - \text{front}) + (\text{rear} + 1)$ . 4. To combine these into a single mathematical expression that works for both cases, we use the modulo operator. 5. The logic is: calculate the raw distance  $\text{rear} - \text{front} + 1$ . To handle negative results (when  $\text{rear} < \text{front}$ ), we add  $N$ . Then, we take the modulo  $N$  to keep it within bounds. 6. The formula  $(\text{rear} - \text{front} + 1 + N) \% N$  correctly calculates the count. For example, if  $N = 5$ ,  $\text{front} = 3$ ,  $\text{rear} = 1$ :  $(1 - 3 + 1 + 5) \% 5 = 4 \% 5 = 4$  elements (indices 3, 4, 0, 1). 7. Note: Some implementations define size as  $(\text{rear} - \text{front} + N) \% N$ , but this depends on whether rear points to the last element or the next empty slot. Following the standard where rear is the index of the last element, +1 is necessary.

**Final Answer:** The number of elements is  $(\text{rear} - \text{front} + 1 + N) \% N$ .

**Answer: (B)**



Q8.

**Solution****Concept:**

Quick Sort is a Divide and Conquer algorithm. Its efficiency depends heavily on the choice of the "pivot" element. The time complexity varies from  $O(n \log n)$  to  $O(n^2)$ . The best-case scenario occurs when the partition process always splits the array into two nearly equal halves.

**Solution:**

1. In each step of Quick Sort, we pick a pivot and partition the array. 2. If the pivot splits the array into sizes  $k$  and  $n - k - 1$ , the recurrence is  $T(n) = T(k) + T(n - k - 1) + O(n)$ . 3. The best case occurs when  $k \approx n/2$ . This leads to the recurrence  $T(n) = 2T(n/2) + O(n)$ , which, by Master Theorem, solves to  $O(n \log n)$ . 4. This happens when the pivot is always the median or close to the middle of the sorted range of the current sub-array. 5. Option (A) is incorrect as  $O(n^2)$  is the worst case (when the array is already sorted and we pick the first/last element as pivot). 6. Option (C) is incorrect because Quick Sort is not naturally  $O(n)$  even with identical elements unless specific optimizations like 3-way partitioning are used. 7. Option (D) is incorrect because picking the smallest element as pivot repeatedly leads to the worst-case  $O(n^2)$ .

**Final Answer:** The best-case complexity is  $O(n \log n)$  when the pivot divides the list into two nearly equal halves.

**Answer: (B)**

Q9.

**Solution****Concept:**

Networking devices operate at different layers of the OSI model. - Hubs/Repeaters operate at Layer 1 (Physical) and simply broadcast signals. - Bridges/Switches operate at Layer 2 (Data Link) and use MAC addresses to filter and forward data. - Routers operate at Layer 3 (Network) and use IP addresses for routing between different networks.

**Solution:**

1. The problem specifies two departments in the same building (suggesting a Local Area Network or LAN environment). 2. The key requirement is to "filter traffic based on MAC addresses." 3. A Hub is a "dumb" device; it sends data to all ports, increasing collisions. 4. A Repeater only amplifies the signal to extend distance; it doesn't look at addresses. 5. A Router is used to connect different networks (like LAN to WAN) using IP addresses, which is overkill and technically different from MAC-based filtering within a segment. 6. A Bridge (or a Layer 2 Switch) maintains a MAC address table. It checks the destination MAC address of an incoming frame. If the destination is on the same segment as the source, it "filters" (drops) the frame to prevent it from unnecessarily entering other segments, thus reducing collisions.

**Final Answer:** The correct device is a Bridge.

**Answer: (C)**

Q10.

**Solution**

**Concept:** Python identifies different types of errors through specific exception classes:

- `ModuleNotFoundError`: A subclass of `ImportError` raised when the module file itself cannot be located.
- `ImportError`: Raised when an import statement has trouble loading a module or a specific name from a module.
- `AttributeError`: Raised when a reference to an attribute (or function/method) fails because it does not exist on that object or module.
- `NameError`: Raised when a local or global name is not found.

**Solution:** When you successfully import a module but try to access something inside it that isn't there, Python treats the module as an object and the missing function as a missing attribute of that object.

- (A) `ImportError` is generally raised during the `from module import name` phase if the name is missing.
- (C) `AttributeError` is specifically raised when using the dot notation, e.g., `module.missing_function()`.

While `ImportError` can occur during the initial import of a specific name, the most common exception for a missing attribute on a found module object is `AttributeError`. However, per the standard interpretation of the `from...import` mechanism, if the module is found but the name is not, `ImportError` is raised. Given the options, `AttributeError` (C) is the most specific to the "attribute within a module" context.

**Answer: (C)**



Q11.

**Solution****Concept:**

Relational Algebra is a procedural query language used to manage data in a relational database. To perform operations involving multiple relations, they must sometimes be "union-compatible," meaning they have the same number of attributes and the attributes have the same domains. The Set Difference operator ( $-$ ) is one such operator used to exclude specific tuples.

**Solution:**

1. We are looking for tuples that exist in one set ( $R$ ) but are strictly excluded if they appear in another set ( $S$ ). 2. Option (A) Intersection ( $\cap$ ) finds only the common tuples between  $R$  and  $S$ . 3. Option (B) Set Difference ( $-$ ) returns all tuples that are in  $R$  but not in  $S$ . This matches the requirement. 4. Option (C) Cartesian Product ( $\times$ ) combines every row of  $R$  with every row of  $S$ , resulting in a much larger relation. 5. Option (D) Division ( $\div$ ) is used for "for all" type queries, such as finding customers who have accounts in all branches of a bank. 6. Therefore, for the subtraction of one set of records from another, Set Difference is the correct tool.

**Final Answer:** The operator is Set Difference ( $-$ ).

**Answer: (B)**

Q12.

**Solution****Concept:**

The SUBSTR() (or SUBSTRING()) function in SQL is used to extract a portion of a string. The syntax is generally SUBSTR(string, start\_position, length). It is important to note that in SQL, string indexing typically starts at 1, not 0.

**Solution:**

1. String: 'COMPUTER SCIENCE' 2. Start Position: 10 3. Length to extract: 4 characters. 4. Let's map the indices: - 1:C, 2:O, 3:M, 4:P, 5:U, 6:T, 7:E, 8:R, 9: (space), 10:S, 11:C, 12:I, 13:E... 5. Starting at index 10, the first character is 'S'. 6. We need 4 characters starting from 'S': - 1st: S (index 10) - 2nd: C (index 11) - 3rd: I (index 12) - 4th: E (index 13) 7. The resulting substring is 'SCIE'.

**Final Answer:** The result is 'SCIE'.

**Answer: (A)**



Q13.

**Solution****Concept:**

Bubble Sort works by repeatedly swapping adjacent elements if they are in the wrong order. In a standard implementation for an array of size  $n$ , there are  $n - 1$  passes. In each pass, comparisons are made between adjacent elements.

**Solution:**

1. Size of the array  $n = 10$ . 2. The number of comparisons in Bubble Sort is independent of the initial order of the elements unless an optimized version (with a swap flag) is used to stop early. However, in the "worst case" (descending to ascending), even an optimized version will not stop early. 3. Pass 1:  $(n - 1) = 9$  comparisons. 4. Pass 2:  $(n - 2) = 8$  comparisons. 5. ... 6. Pass 9: 1 comparison. 7. Total comparisons =  $9 + 8 + 7 + 6 + 5 + 4 + 3 + 2 + 1$ . 8. Using the formula for the sum of first  $k$  integers:  $S = \frac{k(k+1)}{2}$ . Here  $k = 9$ . 9.  $S = \frac{9 \times 10}{2} = 45$ .

**Final Answer:** The total number of comparisons is 45.

**Answer: (B)**

Q14.

**Solution****Concept:**

To handle binary data in Python, we use the `pickle` module. To add data to an existing file without deleting the content already present, we must open the file in "append" mode. For binary files, the mode is 'ab'.

**Solution:**

1. `pickle.dump(object, file)` is the standard method to write a serialized object to a file. 2. Option (A) uses 'wb' (write binary), which truncates (clears) the file before writing, so it would overwrite previous data. 3. Option (B) uses 'ab' (append binary). This moves the file pointer to the end of the file, allowing new data to be added after the existing content. This meets the requirement. 4. Option (C) `pickle.write()` is not a valid function name in the `pickle` module; the correct function is `dump()`. 5. Option (D) uses 'rb+' (read/binary/update) and an `append()` method which does not exist for file objects.

**Final Answer:** The correct statement is `pickle.dump(D, open('records.dat', 'ab'))`.

**Answer: (B)**



Q15.

**Solution****Concept:**

Address resolution protocols help map between different layers of addresses. - ARP (Address Resolution Protocol) maps a known IP address to a MAC address. - RARP (Reverse Address Resolution Protocol) maps a known MAC address to an IP address.

**Solution:**

1. The host knows its own MAC address (which is burned into the NIC). 2. It does not know its IP address (which is required for Layer 3 communication). 3. The host broadcasts a request saying: "My MAC is XX:XX, what is my IP?" 4. A RARP server responds with the assigned IP address. 5. While DHCP (Dynamic Host Configuration Protocol) is the modern and more common way to do this (as it also provides gateway and DNS info), RARP is the specific protocol designed for discovering one's own IP address based strictly on the physical MAC address. 6. ICMP is for error reporting and diagnostics, and SMTP is for electronic mail.

**Final Answer:** The protocol is RARP.

**Answer: (B)**

Q16.

**Solution****Concept:**

In a circular queue implemented with an array of size  $N$ , the indices wrap around to the beginning using the modulo operator. - **Dequeue:** This operation removes an element from the front. The new front is calculated as  $(\text{front} + 1) \% N$ . - **Enqueue:** This operation adds an element to the rear. The new rear is calculated as  $(\text{rear} + 1) \% N$ .

**Solution:**

1. Initial State:  $\text{front} = 2, \text{rear} = 5, N = 6$ . 2. **First Dequeue:** - front moves from 2 to  $(2 + 1) \% 6 = 3$ . 3. **Second Dequeue:** - front moves from 3 to  $(3 + 1) \% 6 = 4$ . 4. **One Enqueue:** - rear was at 5. - The new rear is  $(5 + 1) \% 6 = 0$ . 5. Final State:  $\text{front} = 4, \text{rear} = 0$ .

**Final Answer:** The new values are  $\text{front} = 4, \text{rear} = 0$ .

**Answer: (A)**



Q17.

**Solution****Concept:**

Relational Algebra uses set operations to combine results. - **Selection ( $\sigma$ ):** Filters rows based on a logical condition. - **Projection ( $\Pi$ ):** Filters specific columns. - **Intersection ( $\cap$ ):** Finds common elements in two sets. - **Union ( $\cup$ ):** Combines elements from two sets (all inclusive).

**Solution:**

1. We need students who enrolled in 'Physics' AND 'Chemistry'. 2. Option (A) uses an OR ( $\vee$ ) condition inside a single selection. This returns students in Physics OR Chemistry. 3. Option (B) uses Union ( $\cup$ ). This also returns students in either one or both subjects. 4. Option (C) first selects names of students in Physics, then selects names of students in Chemistry, and finally uses the Intersection ( $\cap$ ) operator. This effectively finds only those names that appear in both result sets. 5. Option (D) is logically flawed because a single row in the Enroll table cannot have Cname as both 'Physics' and 'Chemistry' simultaneously; it would return an empty set.

**Final Answer:** The correct expression is  $\Pi_{Sname}(\sigma_{Cname='Physics'}(Enroll)) \cap \Pi_{Sname}(\sigma_{Cname='Chemistry'}(Enroll))$ .

**Answer: (C)**

Q18.

**Solution****Concept:**

Python's math module provides different rounding behaviors: - **math.ceil(x):** Returns the smallest integer greater than or equal to  $x$ . - **math.floor(x):** Returns the largest integer less than or equal to  $x$ . - **math.trunc(x):** Removes the fractional part, returning the integer towards zero.

**Solution:**

1. **math.ceil(-4.7):** The integers near  $-4.7$  are  $-5$  and  $-4$ . The "greater" value is  $-4$ . Result:  $-4$ . 2. **math.floor(3.8):** The integers near  $3.8$  are  $3$  and  $4$ . The "lesser" value is  $3$ . Result:  $3$ . 3. **math.trunc(-2.9):** Removing the decimal part of  $-2.9$  leaves us with  $-2$ . Result:  $-2$ . 4. Calculation:  $(-4) + 3 + (-2) = -1 + (-2) = -3$ .

**Final Answer:** The output is  $-3$ .

**Answer: (A)**



Q19.

**Solution****Concept:**

The UPDATE statement in SQL is used to modify existing records in a table. The SET clause specifies which column to change and how, while the WHERE clause filters which rows should be affected.

**Solution:**

1. To increase a value by 10%, you multiply it by 1.1 (which is 100% + 10%). 2. The correct SQL syntax is: UPDATE table\_name SET column = new\_value WHERE condition. 3. Option (A) adds 0.1 to the price, which is just 10 paise, not 10 percent. 4. Option (B) correctly uses UPDATE and multiplies the price by 1.1 for products with Price < 500. 5. Option (C) incorrectly uses ALTER, which is for structural changes (like adding columns), not data changes. 6. Option (D) uses MODIFY and IF, which are not standard SQL syntax for updating row data.

**Final Answer:** The correct command is UPDATE Product SET Price = Price \* 1.1 WHERE Price < 500;.

**Answer: (B)**

Q20.

**Solution****Concept:**

Network Topologies define the layout of connections between devices. - **Star Topology:** All nodes are connected to a central device (Hub or Switch). - **Mesh Topology:** Every node is connected to every other node. - **Bus Topology:** All nodes share a single communication line (backbone).

**Solution:**

1. In a Star Topology, the central hub acts as a repeater for data flow. 2. If a single cable connecting a workstation to the hub fails, only that workstation is affected. 3. However, since all traffic must pass through the center, if the central hub or switch fails, no device can communicate, and the entire network goes down. 4. This central point of failure is a defining characteristic of the Star Topology.

**Final Answer:** The network topology is Star Topology.

**Answer: (C)**

Q21.

**Solution****Concept:**

When working with binary files in Python using the `pickle` module, the `load()` function reads one serialized object at a time. If you attempt to call `load()` when there are no more objects left in the file, Python raises an `EOFError` (End Of File Error). To read an unknown number of objects, a loop combined with exception handling is the standard approach.

**Solution:**

1. To read all objects, we need to repeatedly call `pickle.load(f)`. 2. Since we don't know the exact count of objects beforehand, we use an infinite loop: `while True:`. 3. Inside this loop, we place the `pickle.load(f)` statement. 4. To prevent the program from crashing when the end of the file is reached, we wrap the logic in a `try...except` block specifically catching the `EOFError`. 5. When `EOFError` occurs, we use the `break` statement to exit the loop gracefully. 6. Option (A) is incorrect because `f.read()` is for text/binary strings and doesn't work well with the way `pickle` manages object boundaries. 7. Option (C) and (D) are incorrect because file iterators and `readlines()` are designed for line-based text files, not serialized binary objects.

**Final Answer:** The correct way is to use `while True:` with a `try-except EOFError` block.

Answer: (B)

Q22.

**Solution****Concept:**

A "Divide and Conquer" algorithm works by breaking a problem into smaller sub-problems, solving them recursively, and then combining the results. - **Merge Sort:** Always divides the array into two halves, sorts them, and merges them. Its complexity is consistently  $O(n \log n)$  in all cases (best, average, and worst). - **Quick Sort:** Also uses divide and conquer but its worst-case complexity is  $O(n^2)$  if the pivot selection is poor.

**Solution:**

1. The question specifies two criteria: 1) Divide and Conquer and 2)  $O(n \log n)$  in the **worst-case**. 2. Selection Sort and Bubble Sort are not Divide and Conquer algorithms; they are  $O(n^2)$  algorithms. 3. Quick Sort is Divide and Conquer, but its worst-case is  $O(n^2)$ . 4. Merge Sort uses a recursive splitting mechanism. Because the merging process takes linear time ( $O(n)$ ) and the tree depth is always  $\log n$ , the total time is always  $n \times \log n$ . 5. Therefore, Merge Sort is the only one that guarantees  $O(n \log n)$  regardless of the input distribution.

**Final Answer:** The algorithm is Merge Sort.

Answer: (C)



Q23.

**Solution****Concept:**

The ROUND(number, decimals) function in SQL rounds a value to a specified number of decimal places. - If decimals is **positive**, it rounds to the right of the decimal point. - If decimals is **zero**, it rounds to the nearest integer. - If decimals is **negative**, it rounds to the left of the decimal point (tens, hundreds, etc.).

**Solution:**

1. ROUND(153.769, -1): - The -1 indicates rounding to the nearest ten. - The value 153.769 is between 150 and 160. - Since the units digit is 3 (less than 5), it rounds down to 150. 2. ROUND(153.769, 1): - The 1 indicates rounding to one decimal place (tenths). - Look at the second decimal place (hundredths), which is 6. - Since  $6 \geq 5$ , the tenths digit is incremented: .7 becomes .8. - Result: 153.8. 3. Combining both results, we get 150 and 153.8.

**Final Answer:** The output is 150, 153.8.

**Answer: (A)**

Q24.

**Solution****Concept:**

Network security involves protecting users from deceptive practices. - **Phishing:** Sending fraudulent emails to trick users into revealing sensitive info. - **Pharming:** A more sophisticated attack that redirects a user to a fake website even if they type the correct URL. This is done by poisoning the DNS cache or modifying the local hosts file.

**Solution:**

1. The question focuses on the redirection of traffic via "modifying the host file or DNS entries." 2. In a Pharming attack, the attacker changes the mapping of a domain name (like [www.bank.com] (https://www.bank.com)) to a malicious IP address. 3. This is more dangerous than Phishing because the user isn't clicking a "suspicious link"; they are visiting what they believe is the legitimate site by typing the name themselves. 4. "Spying" is a general term for monitoring, and "Virus" is a type of self-replicating malware. 5. Thus, the specific technique described is Pharming.

**Final Answer:** The attack is Pharming.

**Answer: (B)**



Q25.

**Solution****Concept:**

To convert Infix to Postfix, we use the operator precedence (Exponentiation > Multiplication/Division > Addition/Subtraction) and follow the rules of the stack. Expression:  $A + (B * C - (D / E \uparrow F) * G) * H$

**Solution:**

1. Process A: Output A 2. Process +: Push to Stack [+]  
 3. Process (: Push to Stack [+ , (]  
 4. Process B: Output A B 5. Process \*: Push to Stack [+ , ( , \*]  
 6. Process C: Output A B C 7. Process -: \* has higher precedence than -, so pop \*. Output A B C \*.  
 Push - to Stack [+ , ( , -]  
 8. Process (: Push to Stack [+ , ( , - , (]  
 9. Process D: Output A B C \* D 10. Process /: Push to Stack [+ , ( , -, ( , /]  
 11. Process E: Output A B C \* D E 12. Process  $\uparrow$ : Higher than /, push [+ , ( , -, ( , / ,  $\uparrow$ ]  
 13. Process F: Output A B C \* D E F 14. Process ): Pop until (. Pop  $\uparrow$ , pop /. Output A B C \* D E F  $\uparrow$  /.  
 Stack is [+ , ( , -]  
 15. Process \*: Higher than -, push [+ , ( , -, \*]  
 16. Process G: Output A B C \* D E F  $\uparrow$  / G  
 17. Process ): Pop until (. Pop \*, pop -. Output A B C \* D E F  $\uparrow$  / G \* -.  
 Stack is [+]  
 18. Process \*: Push [+ , \*]  
 19. Process H: Output A B C \* D E F  $\uparrow$  / G \* - H \* +  
 20. End: Pop \*, pop +. Output A B C \* D E F  $\uparrow$  / G \* - H \* +.

**Final Answer:** The postfix representation is  $A B C * D E F \uparrow / G * - H * +$ .

**Answer: (C)**

Q26.

**Solution****Concept:**

SQL provides various functions to search for a substring within a larger string. While different database systems (MySQL, Oracle, SQL Server, PostgreSQL) have their own preferred functions, many of them serve the same purpose: returning the numerical index of the first occurrence of a specific character or sequence.

**Solution:**

1. **LOCATE(substring, string)**: This is a standard function in MySQL that returns the position of the first occurrence of a substring. If the substring is not found, it returns 0.  
 2. **POSITION(substring IN string)**: This is an ANSI SQL standard function. It works similarly to LOCATE but uses the IN keyword. It is widely supported across many platforms.  
 3. **INSTR(string, substring)**: This function (In-String) is commonly used in Oracle and MySQL. It searches for the substring and returns its position.  
 4. All three functions essentially perform the same task of identifying the starting index of a specific piece of text within a string.  
 5. Therefore, "All of the above" is the correct choice as they are all valid SQL string search functions.

**Final Answer:** All of the mentioned functions are valid for finding a substring's position.

**Answer: (D)**



Q27.

**Solution****Concept:**

A Stack is a "Last-In, First-Out" (LIFO) data structure. In Python, a list can be used as a stack where `append()` is used to "push" an element onto the top and `pop()` is used to remove the "top" (last) element.

**Solution:**

1. Initial Stack: [10, 20, 30] (Top is 30). 2. `Stack.append(40)`: The value 40 is added to the end. Stack is now [10, 20, 30, 40]. 3. `Stack.pop()`: The last element (40) is removed. Stack is now [10, 20, 30]. 4. `Stack.pop()`: The last element (30) is removed. Stack is now [10, 20]. 5. `Stack.append(50)`: The value 50 is added to the end. Stack is now [10, 20, 50]. 6. The final state of the list is [10, 20, 50].

**Final Answer:** The final content of the list is [10, 20, 50].

**Answer: (A)**

Q28.

**Solution****Concept:**

Integrity constraints ensure the accuracy and consistency of data in a relational database. - **Entity Integrity:** No primary key can be NULL. - **Referential Integrity:** Ensures that relationships between tables remain consistent. It involves Primary Keys and Foreign Keys.

**Solution:**

1. Referential Integrity specifically regulates the behavior of **Foreign Keys**. 2. A Foreign Key is a column in one table that points to a Primary Key in another table. 3. The rule states that any value in the Foreign Key column must either: - Match an existing value in the Primary Key column of the referenced table. - Be NULL (indicating no relationship for that specific record). 4. This prevents "orphaned" records where a child table refers to a parent record that does not exist. 5. Option (C) accurately describes this rule.

**Final Answer:** Referential Integrity requires a foreign key value to match a primary key value or be null.

**Answer: (C)**



Q29.

**Solution****Concept:**

Remote login protocols allow a user to access a computer's command line over a network. Security is a major concern because login credentials (usernames and passwords) sent over a network can be intercepted if not encrypted.

**Solution:**

1. **Telnet**: An older protocol for remote login. It sends all data, including passwords, in **plain text**. It is considered highly insecure. 2. **SSH (Secure Shell)**: Designed as a secure replacement for Telnet. It uses strong encryption for both the authentication process and the subsequent data transfer. This makes it safe to use even over public or unsecure networks. 3. **FTP**: Used for file transfers, not for executing commands on a remote system. 4. **HTTP**: Used for transferring web pages. 5. For the specific task of "securely logging into a remote computer," SSH is the industry standard.

**Final Answer:** The protocol is SSH.

**Answer: (B)**

Q30.

**Solution****Concept:**

Insertion Sort builds the final sorted array one item at a time. It is much less efficient on large lists than more advanced algorithms like Quick Sort or Merge Sort, but it has a very efficient "best-case" scenario.

**Solution:**

1. In Insertion Sort, we take an element and compare it with the elements to its left until we find its correct position. 2. If the array is **already sorted**, each element is compared only once with the element immediately to its left. 3. Since the element to the left is already smaller, the algorithm determines immediately that no shifts are needed and moves to the next element. 4. For an array of size  $n$ , this results in  $n - 1$  total comparisons. 5. Mathematically,  $n - 1$  comparisons result in a time complexity of  $O(n)$ . 6. Thus, the best case is linear time ( $O(n)$ ) and occurs when the input is already in the desired sorted order.

**Final Answer:** The best-case complexity is  $O(n)$  when the array is already sorted.

**Answer: (A)**



Q31.

**Solution****Concept:**

In programming, we often need to switch between a character and its numerical representation.

- **ASCII/Unicode:** Every character is mapped to a specific integer code (e.g., 'A' is 65).

- **Python Functions:** `ord()` and `chr()` are built-in functions for these conversions.

**Solution:**

1. The function `ord()` stands for "ordinal". It takes a single character string as an argument and returns the integer representing the Unicode code point of that character. 2. For example, `ord('a')` returns 97. 3. Conversely, the `chr()` function takes an integer and returns the corresponding character (e.g., `chr(97)` returns 'a'). 4. The `int()` function is used for converting strings or floats to integers but cannot convert a single non-numeric character to its ASCII value directly. 5. The `ascii()` function returns a string containing a printable representation of an object, escaping non-ASCII characters. 6. Therefore, to get the integer ordinal value, `ord()` is the correct choice.

**Final Answer:** The function is `ord()`.

**Answer: (B)**

Q32.

**Solution****Concept:**

In SQL, the execution order of clauses is important. While `WHERE` filters individual rows before grouping, we often need to filter the results after the data has been aggregated using the `GROUP BY` clause.

**Solution:**

1. The `WHERE` clause is applied to rows before they are grouped. It cannot be used with aggregate functions like `COUNT()`, `SUM()`, or `AVG()`. 2. The `GROUP BY` clause organizes rows into summary rows (like "total sales per city"). 3. To filter these summarized groups (e.g., "only cities where total sales > 5000"), we use the `HAVING` clause. 4. `ORDER BY` is used only to sort the final result set. 5. `SELECT` defines which columns to display. 6. Thus, `HAVING` is specifically designed for filtering groups created by `GROUP BY`.

**Final Answer:** The clause is `HAVING`.

**Answer: (C)**



Q33.

**Solution****Concept:**

Switching is the process of moving data across network nodes. The three main types are: - **Circuit Switching:** A dedicated physical path is established for the duration of the session (like old telephone calls). - **Message Switching:** The whole message is sent from one node to the next (store and forward). - **Packet Switching:** The data is broken into smaller "packets" that are routed independently.

**Solution:**

1. The question describes dividing data into "small pieces" (packets) and routing them "independently." 2. In Packet Switching, each packet contains source and destination addresses. 3. Packets may take different paths to reach the same destination based on network traffic and congestion. 4. Once all packets arrive, the destination host reassembles them into the original message. 5. This is the fundamental technology used by the Internet (IP). 6. "Frequency Switching" is not a standard switching technique in this context (though related to multiplexing).

**Final Answer:** The technique is Packet Switching.

**Answer: (C)**

Q34.

**Solution****Concept:**

To evaluate complex Python expressions, we must follow the order of precedence (PEMDAS/BODMAS): 1. **Exponentiation (\*\*):** Right-to-Left associativity. 2. **Multiplication (\*):** Left-to-Right. 3. **Addition (+):** Left-to-Right.

**Solution:**

1. Expression:  $3 * 1 ** 3 + 2 ** 2 ** 3$  2. Handle the first exponentiation:  $1 ** 3 = 1$ . 3. Handle the second exponentiation (Right-to-Left):  $2 ** 2 ** 3$  means  $2 ** (2 ** 3)$ . -  $2 ** 3 = 8$ . -  $2 ** 8 = 256$ . 4. Substitute values back:  $3 * 1 + 256$ . 5. Handle multiplication:  $3 * 1 = 3$ . 6. Handle addition:  $3 + 256 = 259$ .

**Final Answer:** The result is 259.

**Answer: (B)**



Q35.

**Solution****Concept:**

SQL Joins are used to combine data from two or more tables. - INNER JOIN: Returns records that have matching values in both tables. - LEFT JOIN: Returns all records from the left table and matched records from the right. - RIGHT JOIN: Returns all records from the right table and matched records from the left. - FULL OUTER JOIN: Returns all records when there is a match in **either** the left or right table.

**Solution:**

1. The requirement is to return "all rows when there is a match in one of the tables." 2. This means if a row exists in Table A but not Table B, it is included. If it exists in Table B but not Table A, it is also included. 3. This "all-inclusive" behavior (the union of both tables' keys) is the definition of a FULL OUTER JOIN. 4. Note: Some databases like MySQL do not support FULL OUTER JOIN directly (requiring a UNION of LEFT and RIGHT joins), but it remains the correct conceptual answer.

**Final Answer:** The join is FULL OUTER JOIN.

**Answer: (D)**

Q36.

**Solution****Concept:**

Network protocols operate on specific logical "ports" to ensure that data packets reach the correct service on a server. Standard protocols have well-known port numbers assigned by IANA (Internet Assigned Numbers Authority).

**Solution:**

1. **Port 21**: This is used by **FTP** (File Transfer Protocol) for control signals. 2. **Port 25**: This is used by **SMTP** (Simple Mail Transfer Protocol) for routing emails between servers. 3. **Port 80**: This is the standard port for **HTTP** (Hypertext Transfer Protocol), which is the foundation of data communication for the World Wide Web. When you type a URL starting with `http://`, your browser connects to this port by default. 4. **Port 443**: This is used for **HTTPS** (HTTP Secure), which encrypts the communication using SSL/TLS. 5. Therefore, the default port for standard unencrypted web traffic is 80.

**Final Answer:** The default port for HTTP is 80.

**Answer: (C)**



Q37.

**Solution****Concept:**

Python uses specific reserved keywords to implement its "Try-Except" mechanism for handling runtime errors (exceptions). Each keyword has a distinct role in the control flow of error management.

**Solution:**

1. **try**: Defines a block of code to be tested for errors while it is being executed. 2. **except**: Defines a block of code to be executed if an error occurs in the try block. 3. **finally**: Defines a block of code that will run regardless of whether an exception was raised or handled. 4. **throw**: This is a keyword used in other languages like C++, Java, or JavaScript to signal an error. In Python, the equivalent keyword used to manually trigger an exception is **raise**. 5. Since **throw** is not a valid Python keyword, it is the correct answer to "NOT a valid exception handling keyword."

**Final Answer:** The keyword **throw** is not used in Python; **raise** is used instead.

**Answer: (D)**

Q38.

**Solution****Concept:**

In Relational Algebra, there are two primary unary operators used to filter data from a single table:

- **Selection ( $\sigma$ )**: Filters the set of tuples (rows) based on a logical predicate.
- **Projection ( $\pi$ )**: Filters the set of attributes (columns).

**Solution:**

1. The  $\sigma$  operator acts as a horizontal filter. 2. It takes a condition (like  $\text{Age} > 18$ ) and returns all records that satisfy that specific condition. 3. It is equivalent to the WHERE clause in a standard SQL SELECT statement. 4. For example,  $\sigma_{\text{Salary} > 50000}(\text{Employee})$  would yield all rows where the employee earns more than 50,000. 5. Therefore, it is specifically used for selecting rows (tuples).

**Final Answer:** The  $\sigma$  operator is used for selecting rows based on a condition.

**Answer: (B)**



Q39.

**Solution****Concept:**

Selection Sort is an in-place comparison-based sorting algorithm. It maintains two parts of the array: the sorted part at the beginning and the unsorted part at the end.

**Solution:**

1. The algorithm begins by scanning the entire unsorted list to find the absolute minimum value. 2. Once the minimum is found, it is swapped with the first element of the unsorted part. 3. The boundary of the sorted part then moves one position to the right. 4. This process repeats until the entire list is sorted. 5. **Bubble Sort** works by swapping adjacent elements. 6. **Insertion Sort** works by inserting an element into its correct position in a growing sorted list. 7. **Merge Sort** works by splitting and merging. 8. The description "repeatedly finding the minimum element" is the defining characteristic of Selection Sort.

**Final Answer:** The algorithm is Selection Sort.

**Answer:** (C)

Q40.

**Solution****Concept:**

Email systems use different protocols for different tasks: - **Sending/Relaying:** Moving mail from a client to a server or between two servers. - **Retrieval:** Moving mail from a server to a user's device.

**Solution:**

1. **SMTP (Simple Mail Transfer Protocol):** This is the industry standard protocol used to push or "send" mail across the internet from one mail server to another. 2. **POP3 (Post Office Protocol):** This is used by a client to "download" or retrieve mail from a server. 3. **IMAP (Internet Message Access Protocol):** This is also used for retrieval but allows for synchronization across multiple devices. 4. **SNMP:** This is used for Network Management, not for email. 5. Since the question asks for the protocol used for "sending electronic mail between servers," SMTP is the correct choice.

**Final Answer:** The protocol for sending mail is SMTP.

**Answer:** (C)



Q41.

**Solution****Concept:**

When using `pickle.load()`, Python attempts to read a serialized object from the file. If the file exists but contains no data (0 bytes), the pointer is already at the end of the file. Attempting to load from an empty file immediately triggers an `EOFError`. The `finally` block in Python is guaranteed to execute regardless of whether an exception occurred or was handled.

**Solution:**

1. The file 'emp.dat' is opened successfully in read-binary ('rb') mode. 2. `pickle.load(f)` is called. Since the file is empty, there is no object to deserialize. 3. This raises an `EOFError` (End Of File Error). 4. The execution jumps to the `except EOFError:` block, which prints "End of File". 5. The other `except` blocks are ignored. 6. Finally, the `finally` block executes, printing "Process Over". 7. Note: If the file didn't exist at all, it would have triggered `FileNotFoundError` and then the `finally` block.

**Final Answer:** The output is End of File followed by Process Over.

**Answer:** (C)

Q42.

**Solution****Concept:**

The Division operator ( $\div$ ) in Relational Algebra is used for queries involving "all" or "every." For relations  $R(X, Y)$  and  $S(Y)$ ,  $R \div S$  returns all values of  $X$  that are associated with every value in  $S$ . It is a derived operator defined using Cartesian Product, Set Difference, and Projection.

**Solution:**

1. The logic behind the formal definition is: - Find all possible  $X$  values:  $\Pi_{R-S}(R)$ . - Find "bad"  $X$  values: those that are missing at least one match from  $S$ . - To find "bad" values: take all possible combinations ( $\Pi_{R-S}(R) \times S$ ) and subtract the actual tuples in  $R$ . This leaves the "missing" links. - Project the  $X$  values from these missing links. - Subtract these "bad"  $X$  values from the set of all  $X$  values. 2. This translates to:  $\Pi_{R-S}(R) - \Pi_{R-S}((\Pi_{R-S}(R) \times S) - R)$ . 3. Option (A) correctly represents this formal logical sequence.

**Final Answer:** The correct expression is  $\Pi_{R-S}(R) - \Pi_{R-S}((\Pi_{R-S}(R) \times S) - R)$ .

**Answer:** (A)



Q43.

**Solution****Concept:**

In IPv4 networking, the TTL (Time to Live) field is used to prevent packets from circulating indefinitely in the network (e.g., due to routing loops). It is an 8-bit field in the IP header.

**Solution:**

1. When an IP packet is sent, it is assigned an initial TTL value (e.g., 64 or 128). 2. Every time the packet passes through a router (a "hop"), the router decrements the TTL value by 1. 3. If the TTL reaches zero before the packet reaches its destination, the router considers the packet "expired." 4. The router then discards the packet to save bandwidth. 5. Most importantly, the router sends an ICMP (Internet Control Message Protocol) Type 11 "Time Exceeded" message back to the source IP address to inform them of the failure. 6. This mechanism is the basis for the traceroute utility.

**Final Answer:** The field is Time to Live (TTL); the packet is discarded and an ICMP message is sent.

**Answer: (B)**

Q44.

**Solution****Concept:**

SQL aggregate functions perform a calculation on a set of values and return a single value. These functions behave in a specific way when they encounter NULL (unknown) values in the dataset.

**Solution:**

1. The correct name for the average function in SQL is `AVG()`. 2. By standard SQL rules, `AVG()` (and most other aggregates like `SUM`, `COUNT`, `MAX`, `MIN`) **ignores** NULL values. 3. This means if you have values `{10, 20, NULL}`, `AVG()` will calculate  $(10 + 20)/2 = 15$ . It does not count the NULL row in the denominator. 4. It does NOT treat NULL as 0; treating it as 0 would result in  $(10 + 20 + 0)/3 = 10$ , which is mathematically different and requires the `COALESCE` or `IFNULL` function. 5. Option (C) is the only one that correctly identifies the function name and its behavior.

**Final Answer:** The function is `AVG()`, and it ignores NULL values.

**Answer: (C)**



Q45.

**Solution****Concept:**

Insertion Sort is an intuitive sorting algorithm that works similarly to the way we sort playing cards in our hands. It divides the array into a "sorted" part and an "unsorted" part.

**Solution:**

1. For an element at index  $i$ , we consider it the "key" to be inserted into the already sorted sub-array consisting of elements from index 0 to  $i - 1$ . 2. We compare the key with each element in the sorted part, moving from right to left. 3. If an element in the sorted part is larger than the key, we shift that element one position to the right to make room. 4. Once we find the correct spot (where the element is smaller than or equal to the key), we place the key there. 5. Option (A) describes Bubble Sort. 6. Option (B) describes Selection Sort. 7. Option (D) describes Merge Sort. 8. Option (C) accurately describes the insertion logic.

**Final Answer:** The correct sequence is picking  $A[i]$ , comparing with  $A[0..i - 1]$ , and shifting elements to find the position.

**Answer: (C)**

Q46.

**Solution****Concept:**

A Stack is a linear data structure that follows the "Last-In, First-Out (LIFO)" principle. This means the most recently added element is the first one to be removed. - Push: Adds an element to the Top. - Pop: Removes the element from the Top.

**Solution:**

1. "Initial State:" Stack is empty []. 2. Push(A): Stack becomes [A]. Top is A. 3. Push(B): Stack becomes [A, B]. Top is B. 4. Pop(): Removes the most recent element (B). Stack becomes [A]. Top is A. 5. Push(C): Stack becomes [A, C]. Top is C. 6. Pop(): Removes the most recent element (C). Stack becomes [A]. Top is A. 7. After these operations, only element A remains in the stack.

**Final Answer:** The element at the top of the stack is A.

**Answer: (A)**

Q47.

**Solution****Concept:**

A Binary Search Tree (BST) is a node-based binary tree data structure which has the following properties: - The left subtree of a node contains only nodes with keys lesser than the node's key. - The right subtree of a node contains only nodes with keys greater than the node's key. - The left and right subtree each must also be a binary search tree.

**Solution:**

1. **In-order Traversal** follows the path: Left → Root → Right. 2. Because of the BST property (Left < Root < Right), an In-order traversal visits the smallest elements first, then the root, then the larger elements. 3. This sequence naturally results in the nodes being processed in a sorted, ascending order. 4. Pre-order (Root-Left-Right) and Post-order (Left-Right-Root) do not provide a sorted sequence. 5. Level-order visits nodes layer by layer, which does not guarantee sorted output.

**Final Answer:** In-order Traversal yields nodes in ascending sorted order.

**Answer: (C)**

Q48.

**Solution****Concept:**

Data integrity in relational databases is maintained through keys. - **Primary Key:** Uniquely identifies a record in its own table. - **Foreign Key:** An attribute in one table that refers to the Primary Key of another table to establish a relationship.

**Solution:**

1. In the Student table, RollNo is the Primary Key (it uniquely identifies each student). 2. The Result table also contains a RollNo column to link a specific result to a specific student. 3. Since RollNo in the Result table "references" or points to the Primary Key of the Student table, it acts as a Foreign Key. 4. This ensures referential integrity (e.g., you cannot have a result for a RollNo that doesn't exist in the Student table).

**Final Answer:** The column is called a Foreign Key.

**Answer: (B)**



Q49.

**Solution****Concept:**

The OSI (Open Systems Interconnection) model consists of seven layers, each with specific functions. - **Transport Layer:** Process-to-process delivery. - **Session Layer:** Manages sessions (dialogue control) between applications. - **Network Layer:** Routing and host-to-host delivery.

**Solution:**

1. The **Session Layer** (Layer 5) is responsible for the "session" between two end-user application processes. 2. Its primary duties include session establishment, maintenance, and termination. 3. It provides synchronization by allowing checkpoints to be inserted into the data stream, so that in case of a crash, only data after the last checkpoint needs to be retransmitted. 4. It manages dialogue control (deciding whose turn it is to transmit).

**Final Answer:** The Session Layer is responsible for maintaining and terminating synchronization.

**Answer: (B)**

Q50.

**Solution****Concept:**

Sorting algorithms are categorized by their logic. - **Bubble Sort:** Adjacent comparison and swapping. - **Selection Sort:** Finding the minimum and swapping with the front. - **Insertion Sort:** Placing an element in its correct relative position.

**Solution:**

1. The description mentions "comparing adjacent elements" and "swapping them if they are in the wrong order." 2. This is the core mechanism of **Bubble Sort**. 3. In each pass of Bubble Sort, the largest unsorted element "sinks" (or bubbles) to its final position at the end of the array. 4. Because the largest elements progressively move to the end, it is often visually described as bubbling.

**Final Answer:** The algorithm is Bubble Sort.

**Answer: (C)**



**Answer Key**

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

