GOLDEN OPPORTUNITIES

CS FUNDAMENTALS

Object-Oriented Programming (OOP) Interview Questions

Object-Oriented Programming (OOP) is a programming paradigm that organizes software design around objects rather than functions and logic. It is based on the concept of "objects," which can contain data in the form of fields (also known as attributes or properties) and code in the form of procedures (also known as methods). OOP aims to increase code reusability, scalability, and maintainability.

Key Features of OOP

Encapsulation : The concept of bundling data and methods that operate on the data within a single unit (class). This helps protect the integrity of the data by restricting direct access to it.
Abstraction : Hides complex implementation details and only exposes the necessary functionalities. It helps simplify code and reduce complexity.
Inheritance : Allows a class to inherit properties and behaviors from another class, promoting code reuse and hierarchical relationships.
Polymorphism : Enables a single function, method, or operator to take on multiple forms. It allows different classes to be treated as instances of the same class through a common interface.

Why is OOP Important?

Code Reusability: Inheritance allows the reuse of existing code, reducing redundancy.
Maintainability: Encapsulation ensures better organization and easier maintenance of the codebase.
Scalability: OOP facilitates modular design, making it easier to add new features.

Data Security: Access specifiers (public, private, protected) help in securing data.

Real-World Examples of OOP

Banking System: Different types of accounts (savings, current) can inherit common features from a base account class.
E-commerce Applications: Products, users, and orders can be represented as objects with their attributes and behaviors.
Gaming: Characters in a game can be created using OOP principles, where each character has its own attributes and actions.

1. Basic OOP Concepts

- What is the need for OOPs?
- What are some major Object Oriented Programming languages?
- What are some other programming paradigms other than OOPs?
- What is meant by Structured Programming?
- What are the main features of OOPs?
- What are some advantages of using OOPs?
- Why is OOPs so popular?
- What is meant by the term OOPs?
- What are the four main principles of OOP?
- Explain the difference between a class and an object.
- What is encapsulation? Provide an example.
- What is abstraction? How is it different from encapsulation?
- What is polymorphism? Explain with an example.
- What is inheritance? How does it promote code reusability?
- What are access specifiers in OOP?
- What do you understand by OOP?
- Name any seven widely used OOP languages.
- What is the purpose of using OOPs concepts?
- What are the four main features of OOPs?
- What are the advantages and disadvantages of OOP?
- What are the limitations of OOPs?

- What are the differences between object-oriented programming and structural programming?
- What do you understand by pure object-oriented language? Why is Java not a pure object-oriented programming language?

2. Classes and Objects

- What is a class?
- What is an object?
- How do you define a class in [Your Preferred Language]?
- How do you create an object from a class?
- What is the difference between a constructor and a method?
- What is a destructor? When is it called?
- Can a constructor be private? Why or why not?
- How much memory does a class occupy?
- Is it always necessary to create objects from class?
- What do you understand by class and object? Also, give an example.
- What are the differences between class and object?
- What are the key differences between class and structure?

3. Encapsulation

- What is encapsulation?
- How does encapsulation improve code maintainability?
- What are getters and setters?
- What is the difference between public, private, and protected access modifiers?
- What is the concept of access specifiers, and when should we use these?

4. Abstraction

- What is Abstraction?
- How do abstract classes differ from interfaces?
- Can we create an instance of an abstract class?
- What happens if a subclass does not implement all abstract methods of a parent class?

- How is abstraction implemented in different programming languages?
- How is data abstraction accomplished?
- What is an abstract class?
- How is an abstract class different from an interface?
- What are the characteristics of an abstract class?

5. Inheritance

- What is meant by Inheritance?
- What is the difference between single and multiple inheritance?
- What are the various types of inheritance?
- What is a subclass?
- Define a superclass?
- What is an interface?
- How does method overriding work in inheritance?
- What is the role of the 'super' keyword in inheritance?
- Can constructors be inherited?
- How does the 'final' keyword affect inheritance?
- Are there any limitations of Inheritance?
- Explain Inheritance with an example?
- Is it possible for a class to inherit the constructor of its base class?
- What are the limitations of inheritance?
- What is the difference between Inheritance and Polymorphism?

6. Polymorphism

- What is Polymorphism?
- What is the difference between method overloading and method overriding?
- What is meant by static polymorphism?
- What is meant by dynamic polymorphism?
- Can we override a static method? Why or why not?
- What is dynamic method dispatch?
- What is operator overloading?
- How does C++ support Polymorphism?

- What is Compile time Polymorphism and how is it different from Runtime Polymorphism?
- Explain overloading and overriding with examples.
- Give a real-world example of polymorphism.

7. Advanced OOP Concepts

- What are access specifiers and what is their significance?
- What is a static method? How is it different from an instance method?
- What is a singleton class? How do you implement it?
- What are design patterns in OOP?
- What is dependency injection?
- What is the difference between composition and aggregation?
- How do mixins work in OOP?
- What is composition?
- What is the difference between Composition and Inheritance?
- What is constructor chaining?
- What is Coupling in OOP, and why is it helpful?
- Name the operators that cannot be overloaded.
- What is the difference between new and override?
- What is Cohesion in OOP?
- What is the difference between a base class and a superclass?

8. Exception Handling in OOP

- What is an exception?
- What is meant by exception handling?
- How do you handle exceptions in OOP?
- What is the difference between checked and unchecked exceptions?
- Can a constructor throw an exception?
- What is the 'finally' block used for?
- What are the differences between error and exception?

9. OOP in Practice

- How does OOP compare to procedural programming?
- What are the advantages of using OOP in software development?

- Can you give an example of an OOP-based real-world application?
- What are some common pitfalls when using OOP?
- How does OOP support code scalability and maintainability?
- Can we run a Java application without implementing the OOPs concept?
- Identify which OOPs concept should be used in the following scenario: A group of 5 friends, one boy never gives any contribution when the group goes for the outing. Suddenly a beautiful girl joins the same group. The boy who never contributes is now spending a lot of money for the group.

10. Miscellaneous OOP Questions

- What is the difference between shallow copy and deep copy?
- How does garbage collection work in OOP languages?
- What is meant by Garbage Collection in OOPs world?
- What is method chaining?
- What is the difference between mutable and immutable objects?
- What is reflection in OOP?
- Are class and structure the same? If not, what's the difference between a class and a structure?
- What are the various types of constructors in C++?
- What is a copy constructor?
- What is a destructor?
- What are the manipulators in OOP, and how do they work?
- What are the rules for creating a constructor?
- What are the differences between the constructor and the method in Java?
- What are the types of variables in OOP?
- Is it possible to overload a constructor?
- Can we overload the main() method in Java? Also, give an example.

DBMS Interview Questions

Check the list of DBMS interview questions for different topics like definitions, basic SQL queries, transactions, performance tuning, and advanced topics, which will help you prepare well for interviews. This will be useful for beginners and seasoned people.

What is DBMS?

A **Database Management System (DBMS)** is a software application that interacts with users, applications, and the database itself to capture and analyze data. It allows for the creation, retrieval, updating, and management of data efficiently.

Why DBMS is Important from a Placements Perspective

Having a strong grip on **DBMS** is one of the keys to cracking technical interviews and securing dream placements in the software development, data, and IT sectors. Here's why it should be a priority for you:

Great Demand: The majority of firms require professionals who can effectively manage and analyze data, making DBMS knowledge a significant advantage.
Fundamental Concept: DBMS is the foundation of data management, and many technical interviews for various roles include questions related to data management concepts.
Real-World Application: Proficiency in DBMS tools and concepts such as SQL, normalization, and transactions demonstrates your ability to solve real-world problems.
Versatility: DBMS knowledge is applicable across a variety of industries, including database administration, backend development, and data engineering.
Data Structure: Understanding DBMS enhances your grasp on data structures and their application, which is crucial for building efficient and scalable systems.

Complete DBMS & SQL Free resources

Basic DBMS Interview Questions

1. What is DBMS?

2. What is a database?

3. What is a database system?

4. What are the advantages of DBMS over traditional file-based systems?

5. Explain the difference between DBMS and RDBMS.

6. What is a database schema?

7. What is normalization, and why is it used?

8. What are the different types of keys in DBMS?

9. Explain primary key, foreign key, and unique key.

10. What is a composite key?

11. What is a surrogate key?

12. What is a candidate key?

13. What is the difference between a super key and a candidate key?

14. What is a relationship in DBMS, and what are the types of relationships?

15. What is an entity in DBMS?

16. What is an attribute in DBMS?

17. What is a tuple in DBMS?

18. What is the difference between primary key and unique constraints?

19. What is the purpose of normalization in DBMS?

20. What is the difference between a database schema and a database state?

21. What is the concept of sub-query in terms of SQL?

22. What is the use of the DROP command, and what are the differences between DROP, TRUNCATE, and DELETE commands?

23. What is the main difference between UNION and UNION ALL?

24. What is Correlated Subquery in DBMS?

25. Explain Entity, Entity Type, and Entity Set in DBMS.

26. What are the different levels of abstraction in the DBMS?

27. What integrity rules exist in the DBMS?

28. What is the E-R model in DBMS?

29. What is a functional dependency in DBMS?

30. What is 1NF in DBMS?

31. What is 2NF in DBMS?

32. What is 3NF in DBMS?

33. What is BCNF in DBMS?

34. What is a CLAUSE in SQL?

35. What is the difference between a database and a schema?

36. What is the purpose of SQL?

37. What are the main differences between Primary Key and Unique Key?

38. What is the difference between a database and a filesystem?

39. What is the purpose of the CASE statement in SQL?

40. What is the difference between a database and a blockchain?

SQL Queries

41. What is SQL?

42. Explain the difference between DDL, DML, and DCL.

43. What is the difference between DELETE and TRUNCATE?

44. What is the difference between WHERE and HAVING?

45. What is a JOIN? Explain different types of JOINs.

46. What is a subquery?

47. What is a view in SQL? How to create a view?

48. What are the uses of a view?

49. What is an index? Explain different types of indexes.

50. What is a stored procedure?

51. What is a trigger?

52. What is the difference between UNION and UNION ALL?

53. What is the difference between INNER JOIN and OUTER JOIN?

54. What is a self-join?

55. What is a correlated subquery?

56. What is the difference between GROUP BY and ORDER BY?

57. What is the purpose of the DISTINCT keyword?

58. What is the difference between CHAR and VARCHAR data types?

59. What is the purpose of the LIMIT clause?

60. What is the difference between a database and a table?

61. What is the purpose of the EXISTS clause?

62. How to print duplicate rows in a table?

63. What is Identity?

64. What is the difference between Trigger and Stored Procedure?

65. What is the purpose of the WITH clause in SQL?

66. How can you get the alternate records from the table in SQL?

67. How is pattern matching done in SQL?

68. What are the different types of joins in SQL?

69. What is the difference between HAVING and WHERE clauses?

70. What is the purpose of the DROP command?

Transactions and Concurrency

71. What is a transaction?

72. Explain ACID properties.

73. What is a deadlock?

74. What is concurrency control?

75. Explain the difference between optimistic and pessimistic locking.

76. What is a rollback?

77. What is a savepoint?

78. What is the difference between a transaction and a batch?

79. What is a dirty read in DBMS?

80. What is a phantom read in DBMS?

81. What is a non-repeatable read in DBMS?

82. What is two-phase locking (2PL)?

83. What is the difference between serializable and repeatable read isolation levels?

84. What is a log file in DBMS?

85. What is the purpose of the COMMIT statement?

86. What is a Live Lock?

87. What is the difference between a shared lock and an exclusive lock?

Advanced DBMS Concepts

88. What is database partitioning?

89. What is database sharding?

90. Explain CAP theorem.

91. What is NoSQL?

92. Explain the difference between SQL and NoSQL.

93. What is a distributed database?

94. What is database replication?

95. What is a materialized view?

96. What is database indexing, and how does it work?

97. What is a B-tree index?

98. What is a hash index?

99. What is the difference between horizontal and vertical partitioning?

100. What is a database cluster?

101. What is a database trigger, and how is it different from a stored procedure?

102. What is a cursor in DBMS?

103. What is Denormalization?

104. What is QBE (Query By Example)?

105. Why are cursors necessary in embedded SQL?

106. What is the main goal of RAID technology?

107. What is a checkpoint in DBMS?

108. When does a checkpoint occur in DBMS?

109. What do you mean by transparent DBMS?

110. What are the unary operations in Relational Algebra?

111. What is Relational Algebra?

112. What is Relational Calculus?

113. What do you understand by query optimization?

114. What do you mean by durability in DBMS?

115. What is System R? How many of its two major subsystems?

116. What is Data Independence?

117. What are the three levels of data abstraction?

Performance Tuning

118. How do you optimize a SQL query?

119. What is a query execution plan?

120. What are the common performance issues in databases?

121. How do you handle slow-running queries?

122. What is database caching?

123. What is the difference between a clustered and a non-clustered index?

124. What is query optimization?

125. What is the purpose of indexing in databases?

126. How do you identify and resolve deadlocks?

127. What is the role of the database buffer pool?

Security

128. What is database security?

129. What is SQL injection?

130. How do you prevent SQL injection?

131. What is database encryption?

132. What is role-based access control?

133. What is the difference between authentication and authorization?

134. What is database auditing?

135. What is data masking?

136. What is a firewall in the context of database security?

137. What is the purpose of SSL/TLS in database connections?

Miscellaneous

138. What is a data warehouse?

139. What is OLAP and OLTP?

140. Explain the difference between a data lake and a data warehouse.

141. What is ETL?

142. What is a database trigger?

143. What is the difference between a database and a data warehouse?

144. What is the purpose of a data dictionary?

145. What is the difference between a fact table and a dimension table?

146. What is a star schema?

147. What is a snowflake schema?

148. What is the purpose of the WITH clause in SQL?

149. What is the difference between a database and a blockchain?

150. What is the difference between logical database design and physical database design?

151. What are temporary tables? When are they useful?

152. Explain different types of failures that occur in the Oracle database.

Operating System Interview Questions

Check the list of Operating System interview questions covering various topics like definitions, memory management, processes, file systems, scheduling algorithms, and advanced concepts, which will help you prepare well for interviews. This will be useful for both beginners and experienced professionals.

What is an Operating System?

An **Operating System (OS)** is system software that manages computer hardware, software resources, and provides common services for computer programs. It acts as an intermediary between users and the computer hardware.

Why Operating System is Important from a Placements Perspective

Having a strong understanding of **Operating System** concepts is crucial for technical interviews, particularly for roles related to system-level programming, software engineering, and IT infrastructure. Here's why it should be a priority for you:

Core Knowledge: Operating systems are fundamental to understanding how computer systems function, and many technical interviews test your knowledge of OS concepts.
Real-World Relevance: OS knowledge is crucial in handling memory, processes, and scheduling in real-world applications and server environments.

Performance Tuning: An understanding of OS concepts helps in optimizing system performance, especially for backend or embedded systems roles.
Problem-Solving Skills: OS interview questions often involve problem-solving in areas like resource allocation, synchronization, and deadlock handling, showcasing your analytical skills.
Industry Demand: Professionals who are proficient in OS concepts, especially in areas like system design and multi-threading, are in high demand across various industries, including tech, embedded systems, and telecommunications.

1. Basic Concepts

- What is an operating system?
- What are the main functions of an operating system?
- Explain the difference between monolithic and microkernel architectures.
- What is a kernel?
- What is the difference between user mode and kernel mode?
- What is IPC? What are the different IPC mechanisms?
- What's the main purpose of an OS? What are the different types of OS?
- What are the benefits of a multiprocessor system?
- What is RAID structure in OS? What are the different levels of RAID configuration?
- What is GUI?
- What is a Pipe and when is it used?
- What are the different kinds of operations that are possible on semaphore?
- What is a bootstrap program in OS?
- Explain demand paging.
- What do you mean by RTOS?
- What do you mean by process synchronization?
- Why is the operating system important?
- What is the difference between main memory and secondary memory?
- What do you mean by overlays in OS?
- Write top 10 examples of OS.
- What is multitasking?
- What is caching?
- What is spooling?
- What is the functionality of an Assembler?
- What are interrupts?

- What is preemptive multitasking?
- What is a zombie process?
- What are orphan processes?
- What is starvation and aging in OS?
- What is a dispatcher?
- Define the term dispatch latency.
- What are the goals of CPU scheduling?
- What is a critical section?
- Write the name of synchronization techniques.
- What is Peterson's approach?
- Define the term Bounded waiting.
- What are the solutions to the critical section problem?
- What is a Banker's algorithm?
- What is concurrency?
- Write a drawback of concurrency.
- What are the necessary conditions which can lead to a deadlock in a system?
- What are the issues related to concurrency?
- Why do we use precedence graphs?
- Explain the resource allocation graph.
- What is a deadlock?
- What is the goal and functionality of memory management?
- Write the difference between physical address and logical address.
- Explain address binding.
- Write different types of address binding.
- Write an advantage of dynamic allocation algorithms.
- Define Compaction.
- Write about the advantages and disadvantages of a hashed-page table.
- Write the difference between paging and segmentation.
- Write a definition of Associative Memory and Cache Memory.
- What is "Locality of reference"?
- Write down the advantages of virtual memory.
- How to calculate performance in virtual memory?
- Write down the basic concept of the file system.
- Write the names of different operations on a file.
- Define the term Bit-Vector.
- What is a File allocation table?
- What is rotational latency?
- What is seek time?

- What is Belady’s Anomaly?
- What happens if a non-recursive mutex is locked more than once?
- What are the advantages of a multiprocessor system?
- What are real-time systems?
- How to recover from a deadlock?
- What factors determine whether a detection algorithm must be utilized in a deadlock avoidance system?

2. Process Management

- What is a process?
- What is a process and process table?
- What are the different states of a process?
- Explain the states of a process (e.g., new, ready, running, waiting, terminated).
- What is a Process Control Block (PCB)?
- What is context switching?
- What is the difference between a process and a thread?
- What is a Thread?
- What are the differences between process and thread?
- What are the benefits of multithreaded programming?
- What is multithreading?
- Explain the concept of process scheduling.
- What are the different types of schedulers (long-term, short-term, medium-term)?
- What is the difference between preemptive and non-preemptive scheduling?
- Explain common scheduling algorithms (FCFS, SJF, Round Robin, Priority Scheduling).
- What is FCFS?
- What is the RR scheduling algorithm?
- What is Reentrancy?
- What is a Scheduling Algorithm? Name different types of scheduling algorithms.
- What is cascading termination?

3. Memory Management

What is memory management?
Explain the concept of paging.
What is segmentation?
What is virtual memory?
Explain the difference between internal and external fragmentation.
What is a page fault?

What is a page replacement algorithm? Explain FIFO, LRU, and Optimal algorithms.
What is thrashing?
What is fragmentation?
What is the basic function of paging?
How does swapping result in better memory management?
What is the best page size when designing an operating system?

4. File Systems

What is a file system?
Explain the difference between FAT, NTFS, and ext4 file systems.
What is a directory structure?
Explain the concept of inodes in Unix-based systems.
What is RAID? Explain different RAID levels.
What is a journaling file system? How does it improve reliability?
What is the difference between hard links and symbolic links?
What is file fragmentation, and how does it affect performance?
What is a file descriptor? How is it used in file operations?
What is the purpose of the `open()`, `read()`, `write()`, and `close()` system calls?
What is a mount point in a file system?
What is the difference between a block device and a character device?
What is the purpose of the `fsck` utility in Unix-based systems?
What is a distributed file system? Give examples.
What is the role of the File Allocation Table (FAT) in the FAT file system?
What is the difference between a file system and a database system?

5. Deadlocks

- What is a deadlock?
- Explain the four necessary conditions for a deadlock (mutual exclusion, hold and wait, no preemption, circular wait).
- What are the strategies for handling deadlocks (prevention, avoidance, detection, and recovery)?
- Explain the Banker's Algorithm for deadlock avoidance.
- What is a resource allocation graph? How is it used to detect deadlocks?
- What is the difference between deadlock prevention and deadlock avoidance?
- What is the Ostrich Algorithm in deadlock handling?
- What is livelock, and how is it different from deadlock?
- What is the difference between deadlock and starvation?
- How can timeouts be used to handle deadlocks?
- What is the role of the wait-die and wound-wait schemes in deadlock prevention?

- What is a safe state in the context of deadlock avoidance?
- What is the difference between a single-instance and multi-instance resource system in deadlock handling?

6. Inter-Process Communication (IPC)

- What is IPC?
- Explain different IPC mechanisms (pipes, message queues, shared memory, sockets).
- What is a semaphore?
- Explain the difference between a mutex and a semaphore.
- What is a race condition?
- What are the advantages of semaphores?
- What are the drawbacks of semaphores?
- What is a named pipe (FIFO)? How is it different from an unnamed pipe?
- What is the difference between synchronous and asynchronous IPC?
- What is a message queue, and how does it work?
- What is shared memory, and how is it implemented?
- What is the difference between blocking and non-blocking IPC?
- What is a condition variable, and how is it used in IPC?
- What is the producer-consumer problem, and how can IPC mechanisms solve it?
- What is the difference between direct and indirect communication in IPC?
- What is the role of signals in IPC?
- What is a socket, and how is it used for network communication?

7. Storage Management

What is disk scheduling?
Explain common disk scheduling algorithms (FCFS, SSTF, SCAN, C-SCAN, LOOK, C-LOOK).
What is the difference between sequential and random access?
Explain the concept of disk partitioning.
What is the purpose of the Master Boot Record (MBR)?
What is the difference between primary and extended partitions?
What is the role of the disk controller in storage management?
What is the difference between logical and physical formatting of a disk?
What is the purpose of bad block management in storage systems?
What is the difference between RAID 0, RAID 1, RAID 5, and RAID 10?
What is the purpose of caching in storage systems?

What is the difference between hot-swappable and non-hot-swappable storage devices?
What is the role of the swap space in memory management?
What is the difference between a solid-state drive (SSD) and a hard disk drive (HDD)?

8. Security and Protection

What is the difference between authentication and authorization?
Explain the concept of access control lists (ACLs).
What is a buffer overflow?
What is a rootkit?
Explain the principle of least privilege.
What is the difference between symmetric and asymmetric encryption?
What is a firewall, and how does it enhance system security?
What is the role of a security kernel in an operating system?
What is the difference between a virus, worm, and Trojan horse?
What is the purpose of a sandbox in system security?
What is the difference between discretionary access control (DAC) and mandatory access control (MAC)?
What is the role of a security token in authentication?
What is the difference between intrusion detection and intrusion prevention systems?
What is the purpose of encryption in file systems?
What is the difference between a vulnerability and an exploit?

9. Advanced Topics

- What is a real-time operating system (RTOS)?
- Explain the concept of virtualization.
- What is a hypervisor?
- What is containerization? How is it different from virtualization?
- Explain the concept of distributed operating systems.
- What are different types of Kernel?
- What do you mean by Semaphore in OS? Why is it used?
- What is Kernel and write its main functions?
- Write the difference between microkernel and monolithic kernel.
- What is SMP (Symmetric Multiprocessing)?
- What is a time-sharing system?
- What is the difference between Kernel and OS?
- What are various sections of the process?
- What is Cycle Stealing?
- What are a Trap and Trapdoor?

- Write the difference between program and process.
- Write the difference between a user-level thread and a kernel-level thread.
- Write down the advantages of multithreading.
- Difference between Multithreading and Multitasking.

10. Practical Scenarios

How would you troubleshoot a system that is running slowly?
What steps would you take to recover from a system crash?
How would you handle a memory leak in a process?
Explain how you would debug a deadlock situation.

11. OS-Specific Questions

- Explain the architecture of Linux.
- What is the Windows Registry?
- How does macOS handle memory management differently from Windows?
- What is the role of the `init` process in Unix-based systems?
- Write about monolithic kernel.

12. Coding and Implementation

- Write a program to demonstrate the producer-consumer problem.
- Implement a simple shell in C.
- Write a program to simulate the Round Robin scheduling algorithm.
- Implement a basic file system in Python.

Computer Network Interview Questions

Welcome to our curated list of interview questions on **Computer Network**! Whether you're a fresh graduate stepping into the world of IT or an experienced professional looking to brush up on your networking knowledge, this blog is designed to help you ace your next interview.

What is a Computer Network?

A Computer Network is a collection of computers, servers, mainframes, network devices, and other devices connected to each other to share resources and communicate. Networks enable data transmission across various devices and facilitate services like the internet, email, file sharing, and more.

Why Computer Networks are Important from a Placements Perspective ?

Having a strong understanding of Computer Networks is essential for cracking technical interviews and securing dream placements in networking, system administration, and software development roles.Here's why it should be a priority for you:

High Demand: Tech firms, telecom, and cloud service providers need network professionals for efficient communication and data sharing.

Core Concept: Networking fundamentals, including protocols, OSI model, and security, are crucial for technical roles.
Real-World Application: Expertise in networking helps solve issues like troubleshooting, security breaches, and optimizing data transfer.
Versatility: Networking knowledge is valuable across industries like telecom, cybersecurity, cloud computing, and IT infrastructure.
System Performance: Understanding networks helps optimize performance and ensure smooth, reliable communication.

Virtual reality headsets

General Network Concepts

- What is a Link?
- What is a node?
- What is a gateway? Is there any difference between a gateway and router?
- What is a Network?
- What is a node and link?
- What are routers?
- What is the difference between domain and workgroup?
- What are nodes and links?
- What is point to point link?
- What is the main job of the ARP?

Network Layers & Models

- What are the layers of the OSI reference model?
- What is OSI and what role does it play in computer networks?
- What is the job of the Network Layer under the OSI reference model?
- How many layers are there under TCP/IP?
- What is the TCP/IP model? What are its layers?
- What is the difference between the TCP/IP model and the OSI model?
- Define the 7 different layers of the OSI Reference Model
- Describe the TCP/IP Reference Model
- Define the 4 different layers of the TCP/IP Reference Model
- What is the equivalent layer or layers of the TCP/IP Application layer in terms of the OSI reference model?
- What is the role of the Data Link Layer in the OSI model?
- Explain the concept of flow control in networking.

Network Topology & Architecture

- Describe Network Topology.
- What is network topology?
- What is VPN?
- Describe star topology.
- What is Hybrid Network?
- What is mesh topology?
- What is the advantage of mesh topology?
- What is the disadvantage of a star topology?
- What are the advantages of Distributed Processing?
- What are the disadvantages of implementing a ring topology?
- What is the backbone network?
- How does a network topology affect your decision in setting up a network?
- What is bus topology?
- What are the characteristics of a hybrid network?

Protocols & IP Addressing

- What is RIP?
- What is the main purpose of OSPF?
- What is DNS?
- What is DHCP?
- What is TCP/IP?
- What is the FTP protocol?
- What is the SMTP protocol?
- What is ICMP?
- What is Ping?
- What is ARP?
- What are the different types of networks?
- What are private IP addresses?
- What is a private IP address?
- What is the difference between MAC address and IP address?
- What is the MAC address and how is it related to NIC?
- How can you identify the IP class of a given IP address?
- What are some examples of private network addresses?
- What are the different network protocols that are supported by Windows RRAS services?
- What are Unicasting, Anycasting, Multicasting, and Broadcasting?

- What is the purpose of cables being shielded and having twisted pairs?
- What is the difference between IPv4 and IPv6?
- What is the purpose of NAT in IP addressing?

Security

- What are different ways of securing a computer network?
- What is the importance of implementing a Fault Tolerance System? Are there limitations?
- What is the importance of Encryption on a network?
- What are firewalls?
- What is DoS (Denial of Service)?
- What are proxy servers and how do they protect computer networks?
- What is the importance of authentication?
- One way of securing a network is through the use of passwords. What can be considered as good passwords?
- What are the different factors that affect the security of a network?
- What is the role of IEEE in computer networking?
- What is a DDoS attack?
- Explain the differences between symmetric and asymmetric encryption.

Devices & Hardware

What is NIC?
What is the maximum length allowed for a UTP cable?
What is the standard color sequence of a straight-through cable?
What is the difference between a hub and a switch?
What is the maximum segment length of a 100Base-FX network?
What is the proper termination rate for UTP cables?
What happens when you use cables longer than the prescribed length?
What is the number of network IDs in a Class C network?
What is multiplexing in networking?
What is the network ID of a Class A, B, and C network?
What is a router and how does it differ from a switch?
What is a firewall?**

Networking Configuration & Commands

- What is ipconfig?

- What is the difference between a straight-through and crossover cable?
- What is netstat?
- What is tracert?
- How can you manage a network using a router?
- What is the difference between ipconfig and ifconfig?
- How are IP addresses arranged and displayed?
- What is the function of the OSI Session Layer?
- What is the difference between TCP and UDP?
- What is the difference between a client/server model and a peer-to-peer model?

Advanced Networking

- What is the RSA algorithm?
- What is the difference between CSMA/CD and CSMA/CA?
- What are gateways?
- What is the difference between a gateway and router?
- What is the difference between CSMA/CD and CSMA/CA?
- What is IP addressing? What is an IPv4 address?
- What is the criteria to check the network reliability?
- What is subnetting and why is it important?
- How does dynamic host configuration protocol aid in network administration?
- What is the difference between the IP address and the MAC address?
- What is a subnet mask?
- What is NAT (Network Address Translation)?
- What is the function of the OSI Session Layer?
- What are proxy servers and how do they protect computer networks?
- What is the concept of a subnet mask and how is it used in IP addressing?
- What is BGP (Border Gateway Protocol)?

Miscellaneous

- What is the purpose of the OSI Physical Layer?
- What is the main job of ARP?
- What is DNS forwarder?
- What is the difference between the ipconfig and ifconfig?
- What is a congested switch?
- What is the importance of VLANs in networking?
- What is the role of IEEE in computer networking?

- What are the different factors that affect the performance of a network?