**C# Basics**

* Is C# asynchronous or synchronous programming?
* **Difference between readonly and const in C#**
* Explain value types and reference types in C#.
* Describe the difference between DataSet.Copy and DataSet.Clone.
* What are the is and as operators in C#? Provide examples.
* What are the different error handling methods in C#?
* What is the difference between Response.Redirect and Server.Transfer in ASP.NET?
* How can you use nullable values in C#?

**.NET Framework**

* What is the .NET framework?
* What does CLR stand for in .NET, and what is its role?
* What does JIT stand for in .NET, and how does it work?
* What is the Common Type System (CTS) in .NET?
* What is the Common Language Specification (CLS) in .NET?
* What is the difference between managed code and unmanaged code in .NET?
* How does managed code execute in the .NET framework?
* What is the role of the garbage collector in .NET?
* What is caching, and how is it used in .NET applications?
* What is a delegate in .NET, and how is it used?
* Explain boxing and unboxing in .NET with examples.
* Explain localization and globalization in .NET applications.
* Describe role-based security in .NET.
* What is the appSettings section in the web.config file used for?

**ASP.NET & MVC**

* Explain .NET Core and its advantages.
* Describe the MVC architecture and its components.
* Explain the MVC life cycle in detail.
* What are action filters and exception filters in ASP.NET MVC?
* **Explain the concept of routing in ASP.NET MVC.**
* What is view state in ASP.NET, and how does it work?
* Describe the page life cycle in ASP.NET.
* What is the order of events in an ASP.NET page life cycle?
* Explain the different types of cookies available in ASP.NET.
* What is middleware in .NET Core, and how is it used?
* Describe dependency injection and its benefits in .NET Core.

**ADO.NET & Data Handling**

* What is ADO.NET, and how is it used for data access?
* **What is the difference between a DataTable and a DataSet in ADO.NET?**
* What is the difference between Windows application development and web-based development in .NET?
* Why is it important to use await for SQL responses in asynchronous programming?

**Configuration & Security**

* What is the purpose of the app.config file in .NET applications?
* Explain exception handling in .NET with examples.
* How do you create and use custom exceptions in .NET?
* Describe the difference between authentication and authorization in .NET.

**Front-End vs Back-End**

* What is the difference between front-end validation and server-side validation in web applications?

**Advanced .NET Topics**

* Explain LINQ (Language Integrated Query) with coding examples.
* What are microservices, and how are they implemented in .NET?
* Describe the SOLID principles in object-oriented programming.
* What are design patterns, and give examples of commonly used design patterns in .NET.
* Explain the Singleton design pattern with examples.
* How are delegates used in .NET, and what are delegate design patterns?

**Methodologies**

* Compare the Agile methodology with the Waterfall methodology.

**Web Development**

* Explain the various HTTP status codes (e.g., 400, 500) and their meanings.
* **What is a Web API, and how is it used in .NET?**
* Describe the different ways to manage sessions in ASP.NET.

**Miscellaneous**

* What are collections in C#, and how are they used?

**Additional Questions (to reflect current trends and potential MNC interview questions)**

* What are the new features in the latest version of .NET?
* Explain async and await in C# with examples.
* What are generics in C#, and how do they work?
* Describe the difference between Task and Thread in .NET.
* What are extension methods in C#, and how are they used?
* How do you implement logging in .NET Core applications?
* Explain the difference between .NET Core and .NET Framework.
* What is Docker, and how can it be used with .NET applications?
* What is Kubernetes, and how can it be used with .NET applications?
* Explain the SOLID principles in the context of .NET applications.

------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

**Important OOP Questions**

* What is Object-Oriented Programming (OOP)?
* What is the difference between a class and an object?
* What is encapsulation, and how is it implemented in OOP?
* What is abstraction in OOP, and why is it important?
* What is inheritance in OOP, and what are its benefits?
* What is polymorphism in OOP, and what are its types?
* Explain the difference between compile-time polymorphism and run-time polymorphism.
* **What is the difference between method overloading and method overriding?**
* What are access modifiers in OOP, and why are they important?
* What is the role of constructors in class design?
* What is a destructor in OOP?
* Explain the difference between public, private, protected, and internal access modifiers.
* **What is an abstract class, and how is it different from a concrete class?**
* What is an interface, and how is it used in OOP?
* What is the difference between an abstract class and an interface?
* How do you achieve abstraction in OOP?
* How do you implement inheritance in C# (or any other OOP language)?
* What is method overriding, and how is it related to inheritance?
* Explain the Singleton design pattern with examples.
* Explain the Factory design pattern, and how is it implemented?
* How do you implement SOLID principles in OOP?
* What are design patterns, and how are they related to OOP?
* What are generics in OOP, and how are they used?
* Explain the concept of delegates and events in C# (or any other OOP language).
* How do you implement logging in an object-oriented application?
* What is the role of dependency injection in OOP?
* How do you handle exceptions in an OOP language?
* What are some best practices for designing scalable and maintainable object-oriented applications?
* How do you use design patterns to solve common problems in OOP?

------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

**SQL Basics**

* What is a stored procedure, and how is it used?
* What is a Function, and how is it used?
* **Explain the concept of indexing and how it improves query performance**.
* What is a composite index, and when would you use it?
* **What are SQL triggers, and how do they work?**
* How do you create and execute views in MySQL?
* Describe the different types of JOINs (INNER JOIN, LEFT JOIN, RIGHT JOIN, FULL JOIN) and provide examples.
* What is the purpose of a view in SQL, and how do you create one?
* **Explain the difference between DELETE, TRUNCATE, and DROP commands in SQL.**
* **Difference between clustered and non-clustered index**
* **Difference between char and varchar**
* **Difference between a stored procedure and a function in SQL**
* **Difference between DELETE and TRUNCATE commands**
* What is the difference between UNION and UNION ALL in SQL?
* How can you enhance the performance of a stored procedure?
* How can you duplicate a table structure and data to another table in a single query?
* How can you write a SQL query to find the top 10 records of something?
* What database are you using in your project and what is its version?
* Explain SQL queries like JOIN and indexing.
* **Create a table structure for doctor and patient in the healthcare sector (Schema Design)**

**Indexing**

* What are the differences between a clustered index and a non-clustered index?
* How do you decide which columns to index in a table?
* Explain the impact of indexing on database performance.
* How do you remove an index from a table in SQL?

**SQL Performance Tuning**

* What are some common methods for optimizing SQL queries?
* How do you analyze the execution plan of a SQL query?
* Explain the use of indexing in query optimization.
* What are some best practices for writing efficient SQL queries?

**Schema Design**

* **Create a table structure for doctor and patient in the healthcare sector (Schema Design)**
* What are the key considerations when designing a database schema?
* How do you normalize a database schema to eliminate redundancy?
* Explain the concept of foreign keys and how they are used in schema design.

**Additional Questions (to reflect current trends and potential MNC interview questions)**

* What are the new features in the latest version of your SQL database?
* **Explain the concept of ACID properties in SQL databases.**
* What is the role of a transaction in SQL, and how do you manage transactions?
* How do you implement error handling in SQL?
* Describe the difference between OLTP and OLAP systems.
* What is a data warehouse, and how is it different from a traditional database?
* Explain the concept of Big Data and how SQL databases are used in Big Data applications.
* How do you integrate SQL databases with other programming languages?
* What are some security best practices for SQL databases?
* How do you perform database backup and recovery in SQL?

------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

### Top 20 HR Interview Questions for 3 Years of Experience

#### Introduction

1. **Can you tell me about yourself?**
2. **What are your key skills and strengths?**
3. **What do you consider your biggest accomplishment so far?**

#### Project Related Questions

1. **Can you describe a recent project you worked on and your role in it?**
2. **What challenges did you face in your last project, and how did you overcome them?**
3. **How do you ensure the quality and timeliness of project deliverables?**

#### Questions Related to Project Experience

1. **What technologies and tools did you use in your recent projects?**
2. **How do you prioritize tasks and manage your time during a project?**
3. **Can you describe a situation where you had to collaborate with a difficult team member? How did you handle it?**

#### Transition and Motivation

1. **Why are you leaving your current organization?**
2. **Why did you choose Infosys?**
3. **How do you handle high-pressure situations and tight deadlines?**
4. **What motivates you to perform well at work?**
5. **How do you set priorities for your team and yourself at work?**

#### Past Experience and Challenges

1. **Can you discuss a significant technical issue you faced in a past project and how you resolved it?**
2. **How do you approach problem-solving when faced with a complex issue?**
3. **Describe a time when you received constructive criticism and how you handled it.**
4. **What have you learned from your past job experiences that you can apply to this role?**
5. **What was a major achievement in your previous job and how did you accomplish it?**
6. **Can you talk about a time when you had to adapt to a significant change at work? How did you manage it?**