### ****C# Basics and Fundamentals****

1. What is C#?
2. Explain the main features of C#.
3. What is the difference between value types and reference types in C#?
4. What is a nullable type in C#?
5. Explain the concept of boxing and unboxing.
6. What is the purpose of the var keyword in C#?
7. What is the difference between const and readonly in C#?
8. What are properties in C#?
9. Explain the concept of indexers in C#.
10. What is the difference between continue and break statements?
11. What is the purpose of the using statement in C#?
12. Explain the concept of extension methods.
13. What is the dynamic keyword in C#?
14. Explain the difference between == and Equals() in C#.
15. What are delegates in C#?
16. What is an anonymous method in C#?
17. Explain the concept of lambda expressions.
18. What is LINQ?
19. How does exception handling work in C#?
20. Explain the concept of async and await in C#.
21. What is the purpose of the lock statement in C#?
22. What is a Task in C#?
23. Explain the concept of yield in C#.
24. What are events in C#?
25. What is the difference between throw and throw ex?

### ****Object-Oriented Programming (OOP)****

1. Explain the four pillars of OOP.
2. What is inheritance in C#?
3. What is polymorphism in C#?
4. Explain the concept of abstraction in C#.
5. What is encapsulation in C#?
6. What is the difference between a class and an object?
7. What is a constructor in C#?
8. What are destructors in C#?
9. What is method overloading?
10. What is method overriding?
11. What is the difference between virtual, override, and abstract keywords?
12. Explain the concept of interfaces in C#.
13. What is an abstract class?
14. What is a sealed class in C#?
15. What is a partial class in C#?
16. What are generics in C#?
17. Explain covariance and contra variance.
18. What is a singleton pattern?
19. What is a factory pattern?
20. What is a dependency injection?

### ****Advanced C# Concepts****

1. What is reflection in C#?
2. Explain the concept of attributes in C#.
3. What is the difference between early binding and late binding?
4. What is a static class?
5. What is a sealed method?
6. What is the difference between interface and abstract class?
7. Explain the concept of a delegate in C#.
8. What is multithreading in C#?
9. What is a ThreadPool in C#?
10. Explain thread synchronization.
11. What is a Mutex in C#?
12. What is a Semaphore?
13. Explain volatile keyword.
14. What is interlocked in C#?
15. What is ConcurrentDictionary?

### ****.NET Framework****

1. What is the .NET Framework?
2. Explain the Common Language Runtime (CLR).
3. What is the Common Type System (CTS)?
4. What is the Common Language Specification (CLS)?
5. What is the .NET Class Library?
6. Explain the concept of assemblies in .NET.
7. What is the Global Assembly Cache (GAC)?
8. What are strongly-named assemblies?
9. What is MSIL?
10. What is JIT compilation?
11. Explain the concept of Garbage Collection in .NET.
12. What is the difference between managed and unmanaged code?
13. What are the different types of JIT compilers?
14. Explain the concept of AppDomain.
15. What is Code Access Security (CAS)?
16. What is the purpose of Reflection?
17. What are Attributes?
18. What is serialization and deserialization?
19. What is XML serialization?
20. What is JSON serialization?
21. What is SOAP?
22. What is WSDL?
23. Explain the concept of web services.
24. What is WCF?
25. What is ASP.NET?
26. What is MVC?
27. What is Web API?
28. What is Entity Framework?
29. What is LINQ to SQL?
30. Explain ADO.NET.

### ****Design Patterns****

1. What are design patterns?
2. Explain the Singleton pattern.
3. What is the Factory pattern?
4. What is the Abstract Factory pattern?
5. Explain the Builder pattern.
6. What is the Prototype pattern?
7. What is the Adapter pattern?
8. Explain the Bridge pattern.
9. What is the Composite pattern?
10. What is the Decorator pattern?
11. Explain the Facade pattern.
12. What is the Flyweight pattern?
13. What is the Proxy pattern?
14. Explain the Chain of Responsibility pattern.
15. What is the Command pattern?
16. What is the Interpreter pattern?
17. What is the Iterator pattern?
18. Explain the Mediator pattern.
19. What is the Memento pattern?
20. What is the Observer pattern?
21. What is the State pattern?
22. What is the Strategy pattern?
23. Explain the Template Method pattern.
24. What is the Visitor pattern?

### ****ASP.NET and ASP.NET Core****

1. What is ASP.NET?
2. What is the difference between ASP.NET Web Forms and ASP.NET MVC?
3. What is ASP.NET Core?
4. What are Razor Pages?
5. What is middleware in ASP.NET Core?
6. Explain the concept of dependency injection in ASP.NET Core.
7. What is the difference between IServiceCollection and IServiceProvider?
8. What are Tag Helpers?
9. Explain the concept of View Components.
10. What is routing in ASP.NET Core?
11. What are Filters in ASP.NET Core?
12. What is Model Binding?
13. What is Model Validation?
14. What is the difference between IActionResult and ActionResult?
15. What is the purpose of the ConfigureServices method?
16. What is the purpose of the Configure method?
17. Explain the concept of Kestrel.
18. What is the purpose of appsettings.json?
19. What is IConfiguration in ASP.NET Core?
20. What is ILogger in ASP.NET Core?
21. What is Health Checks in ASP.NET Core?
22. What is the difference between Build and Run in Program.cs?
23. Explain the concept of endpoint routing.
24. What is CORS?
25. What is SignalR?
26. What is the purpose of ViewData and ViewBag?
27. Explain the concept of TempData.
28. What is the difference between TempData and Session?
29. What is the difference between AddScoped, AddTransient, and AddSingleton?

### ****Entity Framework and Entity Framework Core****

1. What is Entity Framework?
2. What is Entity Framework Core?
3. What is the difference between Entity Framework 6 and Entity Framework Core?
4. What is Code First approach?
5. What is Database First approach?
6. Explain the concept of DbContext.
7. What is a DbSet?
8. Explain the concept of migrations in Entity Framework Core.
9. What is lazy loading?
10. What is eager loading?
11. What is explicit loading?
12. Explain the concept of change tracking.
13. What is the purpose of the SaveChanges method?
14. What is the difference between AsNoTracking and Tracking?
15. What are Data Annotations?
16. What are Fluent API configurations?
17. What is the purpose of the OnModelCreating method?
18. Explain the concept of concurrency handling in Entity Framework.
19. What is the purpose of the Include method?
20. What is the difference between First and FirstOrDefault?
21. What is the difference between Single and SingleOrDefault?
22. What is the difference between Find and SingleOrDefault?

### ****Advanced .NET Concepts****

1. What is .NET Core?
2. What is the difference between .NET Framework and .NET Core?
3. What is .NET 5/.NET 6?
4. What is the purpose of .NET Standard?
5. Explain the concept of cross-platform development with .NET Core.
6. What is a CLI (Command Line Interface)?
7. What is a runtime identifier (RID)?
8. What is a target framework?
9. What is a NuGet package?
10. How do you create a NuGet package?
11. What is Dependency Injection?
12. What is Inversion of Control (IoC)?
13. What is the purpose of Service Collection?
14. What is the purpose of Service Provider?
15. Explain the concept of scoped services.
16. What is the difference between scoped, transient, and singleton services?
17. What is Configuration in .NET Core?
18. What is the purpose of appsettings.json?
19. What is the purpose of launchSettings.json?
20. Explain the concept of Kestrel web server.
21. What is IHostingEnvironment?
22. What is IApplicationBuilder?
23. What is IWebHostBuilder?
24. Explain the concept of middleware in .NET Core.
25. What is IServiceScopeFactory?
26. What is IServiceProvider?
27. What is ILogger in .NET Core?
28. What is the purpose of CreateDefaultBuilder method?
29. What is the difference between Build and Run methods in Program.cs?
30. What is Dependency Injection and why is it important?
31. What is Microservices architecture?
32. What is gRPC?
33. What is Blazor?
34. What is Docker and how is it used with .NET Core?
35. What is Kubernetes and how is it used with .NET Core?

### ****Software Architecture and Design****

1. What is software architecture?
2. Explain the difference between monolithic and microservices architecture.
3. What are the benefits and challenges of microservices architecture?
4. Describe the Model-View-Controller (MVC) pattern.
5. What is the Model-View-ViewModel (MVVM) pattern?
6. Explain the concept of RESTful APIs.
7. What is the purpose of an API Gateway in microservices architecture?
8. What is the Repository pattern? How does it help with data access?
9. Describe the Unit of Work pattern.
10. What is the CQRS (Command Query Responsibility Segregation) pattern?
11. Explain the Event Sourcing pattern.
12. What is Domain-Driven Design (DDD)?
13. What are the main components of DDD?
14. Describe the concept of Bounded Context in DDD.
15. What is the difference between orchestration and choreography in microservices?
16. What are the 12-Factor Apps?
17. How do you ensure scalability in a distributed system?
18. What is the CAP theorem? How does it apply to distributed systems?
19. Explain the concept of eventual consistency.
20. What are idempotent operations and why are they important?

### ****Miscellaneous Software Engineering Concepts****

1. What is the difference between synchronous and asynchronous programming?
2. Explain the concept of multithreading and parallelism.
3. What are the different ways to handle concurrency in .NET?
4. Describe the Task Parallel Library (TPL) in .NET.
5. What is async/await in C#? How does it help with asynchronous programming?
6. What are the benefits of using dependency injection?
7. What are some common design anti-patterns?
8. How do you perform performance tuning in a .NET application?
9. What are the best practices for exception handling in C#?
10. Explain the concept of garbage collection in .NET.
11. What are value types and reference types in C#?
12. How do you manage memory in .NET applications?
13. What is the purpose of IDisposable interface in C#?
14. Describe the concept of immutability in programming.
15. What is the difference between imperative and declarative programming?

----xx----

## ****MVC (Model-View-Controller) Interview Questions****

### ****Basic MVC Questions:****

1. What is MVC?
2. How does the MVC pattern work?
3. What are the components of MVC?
4. How is MVC different from WebForms?
5. What is the role of the Model in MVC?
6. What is the role of the View in MVC?
7. What is the role of the Controller in MVC?
8. What are the benefits of using MVC?
9. What is Razor in MVC?
10. What is the difference between Razor and ASPX view engine?
11. What is the difference between ViewData, ViewBag, and TempData?
12. What are strongly typed views in MVC?
13. What is the purpose of Partial Views?
14. What are Layouts in MVC?
15. What is an Action Method in MVC?
16. What is an ActionResult in MVC?
17. What is the difference between ActionResult and JsonResult?
18. What is the use of the RedirectToAction() method?
19. What is the use of View() in MVC?
20. What are HTML Helpers in MVC?

### ****Intermediate MVC Questions:****

1. What is Model Binding in MVC?
2. What are Action Filters in MVC?
3. What are the different types of Action Filters in MVC?
4. What is the difference between Authorize and AllowAnonymous attributes?
5. What is the purpose of the ValidateAntiForgeryToken attribute?
6. How does routing work in MVC?
7. What is the difference between Conventional Routing and Attribute Routing?
8. How can you implement custom routing in MVC?
9. What are Areas in MVC?
10. How do you implement authentication in MVC?
11. What are Bundling and Minification in MVC?
12. What is Dependency Injection in MVC?
13. How do you implement logging in MVC?
14. How do you implement exception handling in MVC?
15. What is CORS in MVC, and how do you enable it?
16. What is AJAX in MVC? How do you use it?
17. What is the purpose of the JsonResult in MVC?
18. How do you perform file upload in MVC?
19. What is Web API, and how does it relate to MVC?
20. How do you call a Web API in an MVC application?

### ****Advanced MVC Questions:****

1. What are the different types of result types in MVC?
2. What is the difference between ChildActionOnly and NonAction attributes?
3. How do you implement role-based authentication in MVC?
4. How do you handle multiple models in a single view?
5. What is TempData Keep() and Peek()?
6. What is a ViewComponent?
7. What are TagHelpers in MVC?
8. How do you optimize the performance of an MVC application?
9. What is OutputCache in MVC?
10. How do you implement localization and globalization in MVC?

## ****ADO.NET Interview Questions****

### ****Basic ADO.NET Questions:****

1. What is ADO.NET?
2. What are the components of ADO.NET?
3. What is a Connection object in ADO.NET?
4. What is the difference between a DataReader and a DataSet?
5. What is the purpose of the Command object in ADO.NET?
6. What is Connection Pooling in ADO.NET?
7. What is the role of SqlDataAdapter?
8. What is the difference between ExecuteScalar(), ExecuteNonQuery(), and ExecuteReader()?
9. What is the use of Transactions in ADO.NET?
10. How do you handle exceptions in ADO.NET?
11. What is a DataTable in ADO.NET?
12. What is a DataRow in ADO.NET?
13. What is the use of a DataRelation in ADO.NET?
14. What is the difference between connected and disconnected architecture in ADO.NET?
15. What is a stored procedure, and how do you call it using ADO.NET?

### ****Intermediate ADO.NET Questions:****

1. What are the different data providers in ADO.NET?
2. What is the difference between SqlCommand and OleDbCommand?
3. What is the purpose of the SqlParameter class?
4. How do you handle transactions in ADO.NET?
5. What is the difference between optimistic and pessimistic concurrency control?
6. How do you execute multiple queries in a single database call in ADO.NET?
7. What is the difference between a DataView and a DataTable?
8. What is the purpose of the AcceptChanges() and RejectChanges() methods?
9. How do you implement paging in ADO.NET?
10. What is the difference between DataSet and DataTable?

### ****Advanced ADO.NET Questions:****

1. How do you use bulk copy operations in ADO.NET?
2. What is the role of the TableAdapter in ADO.NET?
3. How do you implement asynchronous operations in ADO.NET?
4. What are the advantages and disadvantages of using ADO.NET over an ORM like Entity Framework?
5. What are connection resiliency and retry policies in ADO.NET?

## ****Dapper Interview Questions****

### ****Basic Dapper Questions:****

1. What is Dapper?
2. Why is Dapper called a micro ORM?
3. How does Dapper compare to ADO.NET and Entity Framework?
4. What are the benefits of using Dapper?
5. How do you install and set up Dapper?
6. How do you perform CRUD operations using Dapper?
7. How do you execute parameterized queries in Dapper?
8. What is the Query() method in Dapper?
9. What is QueryFirstOrDefault() in Dapper?
10. What is the Execute() method in Dapper?

### ****Intermediate Dapper Questions:****

1. How do you execute stored procedures using Dapper?
2. What is Multi-Mapping in Dapper?
3. How do you handle transactions in Dapper?
4. What is the difference between Query() and QueryMultiple()?
5. How does Dapper handle joins?
6. How do you implement caching with Dapper?
7. What is the difference between Query and Execute methods in Dapper?
8. How do you manage connection lifetime in Dapper?
9. How do you return multiple result sets using Dapper?
10. How do you implement asynchronous methods in Dapper?

### ****Advanced Dapper Questions:****

1. How do you integrate Dapper with dependency injection?
2. What is Dapper.Contrib?
3. How do you improve performance when using Dapper?
4. What are the limitations of Dapper?
5. How do you map complex types in Dapper?
6. How do you use Dapper with multiple databases?
7. What are some best practices when using Dapper?
8. How do you log queries in Dapper?
9. What are custom type handlers in Dapper?
10. How do you handle stored procedures that return multiple result sets in Dapper?

## ****2. .NET Core Interview Questions****

### ****Basics of .NET Core****

1. What is .NET Core?
2. What are the key differences between .NET Core and .NET Framework?
3. What are the benefits of using .NET Core?
4. What platforms does .NET Core support?
5. What is .NET 5/.NET 6/.NET 7?
6. What is the purpose of .NET Standard?

### ****.NET Core Application Structure****

1. What is the Program.cs file in .NET Core?
2. What is the Main() method in .NET Core?
3. What is the purpose of Startup.cs in an ASP.NET Core application?
4. How does dependency injection work in .NET Core?
5. What is the difference between IServiceCollection and IServiceProvider?

### ****Middleware and Routing****

1. What is middleware in ASP.NET Core?
2. How does routing work in ASP.NET Core?
3. What is endpoint routing?
4. What are filters in ASP.NET Core?

### ****Configuration and Logging****

1. What is the purpose of appsettings.json?
2. What is IConfiguration in .NET Core?
3. What is the difference between IConfiguration and IOptions<T>?
4. What is ILogger in ASP.NET Core?
5. What are health checks in ASP.NET Core?

### ****Entity Framework Core (EF Core)****

1. What is Entity Framework Core?
2. What is the difference between EF Core and EF 6?
3. What is Code First and Database First approach in EF Core?
4. What are migrations in EF Core?
5. What is lazy loading, eager loading, and explicit loading?

### ****Asynchronous Programming****

1. What is asynchronous programming?
2. What is the purpose of async and await in C#?
3. What is the Task class in .NET Core?
4. What is the difference between Task and ValueTask?
5. What is a CancellationToken in .NET Core?

### ****API Development****

1. What is REST API?
2. How do you create a REST API in .NET Core?
3. What is the difference between IActionResult and ActionResult<T>?
4. What is Model Binding in ASP.NET Core?
5. What is Model Validation in ASP.NET Core?

### ****Security and Authentication****

1. What is authentication and authorization in ASP.NET Core?
2. What is the difference between JWT and OAuth?
3. How do you implement JWT authentication in .NET Core?
4. What is Identity Server?
5. What is CORS (Cross-Origin Resource Sharing)?

### ****Microservices and Cloud Development****

1. What is Microservices architecture?
2. What is Docker, and how is it used in .NET Core?
3. What is Kubernetes, and how does it work with .NET Core?
4. What is gRPC in .NET Core?
5. What is API Gateway?

### ****Performance and Optimization****

1. How do you optimize a .NET Core application?
2. What are best practices for improving .NET Core performance?
3. What are caching mechanisms in .NET Core?
4. How do you handle memory leaks in .NET Core?
5. How does .NET Core handle garbage collection differently from .NET Framework?

## ****1. .NET Framework Interview Questions****

### ****Basics of .NET Framework****

1. What is the .NET Framework?
2. What are the main components of the .NET Framework?
3. What is the difference between CLR (Common Language Runtime) and CTS (Common Type System)?
4. What is the Common Language Specification (CLS)?
5. What is the .NET Class Library?
6. What is the role of the Just-In-Time (JIT) compiler in .NET?
7. What are the different types of JIT compilers?
8. What is the difference between managed and unmanaged code?
9. What is the difference between .NET Framework and .NET Standard?
10. What is MSIL (Microsoft Intermediate Language)?

### ****Assemblies and Deployment****

1. What is an assembly in .NET?
2. What are the types of assemblies in .NET?
3. What is a strongly named assembly?
4. What is the Global Assembly Cache (GAC)?
5. What is the difference between a private and a shared assembly?
6. What is the purpose of the app.config and web.config files?
7. How does .NET support versioning in assemblies?

### ****Garbage Collection and Memory Management****

1. What is garbage collection in .NET?
2. How does the garbage collector work in .NET?
3. What are the different generations in .NET garbage collection?
4. What is the purpose of the IDisposable interface?
5. What is the difference between Finalize() and Dispose()?

### ****Exception Handling****

1. What are the different types of exceptions in .NET?
2. What is the difference between throw and throw ex?
3. What is the purpose of the finally block in exception handling?
4. What is a custom exception? How do you create one?

### ****Multithreading and Parallelism****

1. What is multithreading in .NET?
2. What is the difference between a thread and a process?
3. What is a ThreadPool in .NET?
4. What is thread synchronization?
5. What is a Mutex?
6. What is a Semaphore?
7. What is the difference between lock and Monitor?
8. What is the volatile keyword in C#?
9. What is the purpose of the Interlocked class?

### ****Reflection and Attributes****

1. What is reflection in .NET?
2. What are attributes in .NET?
3. What is the difference between early binding and late binding?
4. How can you create a custom attribute in C#?

### ****Serialization****

1. What is serialization in .NET?
2. What is the difference between binary, XML, and JSON serialization?
3. What is the purpose of the [Serializable] attribute?
4. What is deserialization in .NET?
5. What is SOAP serialization?

### ****ADO.NET and Data Access****

1. What is ADO.NET?
2. What are the different components of ADO.NET?
3. What is the difference between DataSet and DataReader?
4. What is connection pooling in ADO.NET?
5. What is the purpose of the SqlCommand and SqlDataAdapter classes?
6. What is LINQ to SQL?