Multi Choice Type Question What is the primary purpose of unit testing in ASP.NET Core Web API development? Marks : 1 Negative Marks : 0 image Answer here To check the overall functionality of the application To test the interaction between different components of the system To ensure the accuracy of the database queries To verify the correctness of individual code units (methods or functions)

The correct answer is:

**To verify the correctness of individual code units (methods or functions)**

Unit testing focuses on testing small, isolated pieces of code (like methods or functions) to ensure they behave as expected. It does **not** test the overall application flow, database, or interactions between components — those are covered by integration or system testing.

**You said:**

Multi Choice Type Question What is MSBuild in the context of ASP.NET Core Web API? Marks : 1 Negative Marks : 0 image Answer here A testing framework A build automation tool A database management system A web server technology

The correct answer is:

**A build automation tool**

**Explanation:** MSBuild is Microsoft’s build platform used to compile, package, and deploy .NET applications, including ASP.NET Core Web APIs. It automates tasks like compiling code, resolving dependencies, and creating executable files.

**You said:**

Multi Choice Type Question Which file contains the instructions for the MSBuild process in an ASP.NET Core Web API project? Marks : 1 Negative Marks : 0 image Answer here appsettings.json web.config Startup.cs .csproj

The correct answer is:

**.csproj**

**Explanation:** The .csproj file (C# project file) contains XML-based instructions for MSBuild, such as which files to compile, project dependencies, target frameworks, and build configurations. It directs the build automation process for ASP.NET Core projects.

**You said:**

Multi Choice Type Question In an ASP.NET Core Web API project, where is the common location to store unit test classes? Marks : 1 Negative Marks : 0 image Answer here Controllers folder Views folder Models folder Tests folder

The correct answer is:

**Tests folder**

**Explanation:** Unit test classes are typically stored in a separate **Tests** or **UnitTests** project/folder within the solution. This keeps testing code isolated from the main application code (Controllers, Models, Views) and promotes better organization and maintainability.

**You said:**

Multi Choice Type Question In NUnit, what is the purpose of using the [TestCase] attribute in ASP.NET Core Web API unit tests? Marks : 1 Negative Marks : 0 image Answer here To categorize test methods based on their characteristics. To mark a method as a test case. To specify the test cases to execute with different input parameters. To define a setup method that runs before each test case.

The correct answer is:

**To specify the test cases to execute with different input parameters.**

**Explanation:** In NUnit, the [TestCase] attribute allows you to run the same test method multiple times with different sets of input values. This helps in testing various scenarios without writing separate test methods for each input combination.

**You said:**

Multi Choice Type Question Which attribute is used in NUnit to define a setup method that runs before each test case in a test fixture? Marks : 1 Negative Marks : 0 image Answer here [Before] [SetUp] [TestInitialize] [BeforeEach]

The correct answer is:

**[SetUp]**

**Explanation:** In NUnit, the [SetUp] attribute is used to mark a method that should run **before each test case** in a test fixture. It is commonly used to initialize objects or set up the test environment needed for the tests.

**You said:**

Multi Choice Type Question Which attribute is used in NUnit to mark a method as an ignored or skipped test case? Marks : 1 Negative Marks : 0 image Answer here [Ignore] [Exclude] [Skip] [IgnoreTest]

The correct answer is:

**[Ignore]**

**Explanation:** In NUnit, the [Ignore] attribute is used to mark a test method that should be **skipped** during test execution. You can optionally provide a reason as a parameter, e.g., [Ignore("Not implemented yet")].

**You said:**

Multi Choice Type Question What is a test fixture in NUnit used for? Marks : 1 Negative Marks : 0 image Answer here It groups multiple test classes together. It contains common setup and teardown methods for test classes. It allows running tests in a random order for each fixture. Test fixtures are not supported in NUnit.

The correct answer is:

**It contains common setup and teardown methods for test classes.**

**Explanation:** In NUnit, a **test fixture** is a class decorated with [TestFixture] that contains one or more test methods. It allows you to define **common setup ([SetUp]) and teardown ([TearDown]) methods** that run before and after each test within the fixture, ensuring consistent test initialization and cleanup.

**You said:**

Multi Choice Type Question What is the purpose of MSBuild in the context of ASP.NET Core Web API? Marks : 1 Negative Marks : 0 image Answer here To deploy the application to a web server To run unit tests automatically To manage NuGet packages and dependencies To build and compile the application into an executable form

The correct answer is:

**To build and compile the application into an executable form**

**Explanation:** MSBuild is a **build automation tool** used in ASP.NET Core to compile source code, resolve dependencies, and produce executable assemblies or DLLs. It does **not** handle deployment, running tests, or directly managing NuGet packages.

**You said:**

Multi Choice Type Question Which attribute is used in NUnit to mark a method to be executed after each test method? Marks : 1 Negative Marks : 0 image Answer here [After] [TearDown] [AfterTest] [OneTimeTearDown]

The correct answer is:

**[TearDown]**

**Explanation:** In NUnit, the [TearDown] attribute is used to mark a method that runs **after each test method** in a test fixture. It is commonly used to clean up resources or reset the environment after a test completes.

**You said:**

Multi Choice Type Question What is the purpose of dependency injection (DI) in ASP.NET Core Web API? Marks : 1 Negative Marks : 0 image Answer here To optimize database queries To manage client-side dependencies To simplify unit testing and improve application maintainability To secure API endpoints

The correct answer is:

**To simplify unit testing and improve application maintainability**

**Explanation:** Dependency Injection (DI) in ASP.NET Core Web API allows you to **inject dependencies (like services or repositories) into classes** rather than hard-coding them. This promotes loose coupling, makes the code easier to maintain, and facilitates unit testing by allowing mock implementations to be injected.

**You said:**

Multi Choice Type Question What does the term "middleware" refer to in ASP.NET Core Web API? Marks : 1 Negative Marks : 0 image Answer here Software that enables communication between different network protocols Components that handle HTTP requests and responses in the request pipeline A database management system used for storing application data Libraries that provide authentication and authorization mechanisms

The correct answer is:

**Components that handle HTTP requests and responses in the request pipeline**

**Explanation:** In ASP.NET Core, **middleware** refers to components that process **HTTP requests and responses** as they travel through the request pipeline. Middleware can perform tasks like logging, authentication, error handling, routing, and response modification.

**You said:**

Multi Choice Type Question What is MSBuild primarily used for in .NET development? Marks : 1 Negative Marks : 0 image Answer here Database schema management Compilation and build automation User interface design Cloud deployment

The correct answer is:

**Compilation and build automation**

**Explanation:** MSBuild is the **build automation tool** in .NET development. It compiles source code, resolves dependencies, and generates executable files or libraries. It does **not** handle database schema management, UI design, or cloud deployment directly.

**You said:**

Multi Choice Type Question What does the <Target> element represent in an MSBuild project file? Marks : 1 Negative Marks : 0 image Answer here A platform-specific build task A unit testing framework A task to clean up temporary files A unit of work that MSBuild performs

The correct answer is:

**A unit of work that MSBuild performs**