

# INTRODUCTION TO C# (CSHARP)

**Teacher:** Syed Muhammad Arsalan Shah

**Typist:** Yousuf Naveed Khan

## Overview of Programming Languages

Programming languages can be broadly categorized into two types: Low-level and High-level.

### Low-level Language

Low-level languages are based on number systems such as binary and octal. These languages delve into concepts that require a deeper understanding. Developing websites using low-level languages can be challenging.

While compilers understand low-level languages, they can be difficult for developers to work with.

### High-level Language

High-level languages are understood by both developers and compilers. They can be further divided into two types: Document-based and Programming-based languages.

**Document-based Languages:** Examples include HTML and CSS. These languages take input and provide output based on the input provided.

**Programming-based Languages:** These languages not only take input but also provide output based on the logic provided. They are divided into two categories: Client-side and Server-side languages.

### Client-side Languages

Client-side languages involve processes where both the request and response are handled on the client's side. For instance, an age calculator application that calculates the number of days or minutes since a person's birth without involving a database operates on the client side.

## Server-side Languages

Server-side languages handle processes where data is stored in a database, and the server manages all the information. For example, when a user submits incorrect login information on a website, the error response is generated by the server.

## Understanding C#

C# is a server-side scripting language. While it is possible to create console applications using C#, a deeper understanding is required when working with frameworks. Within frameworks, we encounter various possibilities such as:

- **Windows Applications:** These are desktop applications that can be installed as .exe files.

C# can be used to create various types of applications, including:

- **Windows Forms**
- **WinForms**
- **ASP Web Forms** (Although now considered outdated, replaced by newer versions such as ASP.NET MVC and ASP.NET Core)
- **ASP.NET Core MVC**

The primary goal of using these languages and frameworks is to ensure efficient website loading times and enhance user experience.