Adnan Maqbool Khan • 2nd
50k+ LinkedIn | Engineering Lead | .NET | Microservices | Azure | Vue | React | Angular | Typescr...
1d •

Follow

Func, Action And Predicate Delegates In C# 💡

Delegates serve as the foundation for callbacks in .NET. They are reference types that hold references to methods with compatible signatures. In simple terms, delegates allow us to pass methods as arguments to other methods, thereby enabling us to achieve dynamic invocation.

In .net 3.5 some new generic delegates -Func<T>, Action<T>, and Predicate<T> were introduced. Using generic delegates, it is possible to concise delegate type which means you don't have to define the delegate statement. These delegates are the Func<T>, Action<T>, and Predicate<T> delegates and are defined in the System namespace.

1- The **Func** delegate takes up to 16 input parameters and returns a value.
2-The **Action** takes one or more input parameters but does not return anything.
3- The **Predicate** delegate represents a method that takes an input parameter and returns a Boolean value indicating whether the input satisfies a certain condition.

If you find this useful, repost ♻ and spread the knowledge. Follow Adnan Maqbool Khan **for more such content.**