**What is delegate built in types in details**

**What Are Delegates?**

A delegate is a reference type that defines a method signature (return type and parameters). You can assign any method matching that signature to a delegate instance and invoke it. Built-in delegate types are predefined by the .NET Framework/Core to avoid the need for developers to repeatedly define custom delegates for common use cases.

**1. Action Delegates**

**Namespace**: System

**Purpose**: Represents a method that returns void and takes zero or more parameters.

**Variants**:

Action (no parameters)

Action<T> (1 parameter)

Action<T1, T2> (2 parameters)

Up to Action<T1, T2, ..., T16> (16 parameters in .NET)

public delegate void Action(); // No parameters

public delegate void Action<in T> (T arg); // One parameter

public delegate void Action<in T1, in T2>(T1 arg1, T2 arg2); // Two parameters

**2. Func Delegates**

**Namespace**: System

**Purpose**: Represents a method that returns a value and takes zero or more parameters.

**Variants:**

Func<TResult> (no parameters, returns TResult)

Func<T, TResult> (1 parameter, returns TResult)

Func<T1, T2, TResult> (2 parameters, returns TResult)

Up to Func<T1, T2, ..., T16, TResult> (16 parameters + return type)

public delegate TResult Func<out TResult>(); // No parameters

public delegate TResult Func<in T, out TResult>(T arg); // One parameter

public delegate TResult Func<in T1, in T2, out TResult>(T1 arg1, T2 arg2); // Two parameters

**3. Predicate<T> Delegate**

**Namespace**: System

**Purpose**: Represents a method that takes one parameter and returns a boolean to test a condition.

public delegate bool Predicate<in T>(T obj);

**4. EventHandler** and EventHandler<TEventArgs>

Namespace: System

Purpose: Used for event handling, defining a standard signature for event delegates.

public delegate void EventHandler(object sender, EventArgs e);

**5. Comparison<T> Delegate**

**Namespace**: System

Purpose: Represents a method that compares two objects of the same type and returns an integer indicating their relative order.

public delegate int Comparison<in T>(T x, T y);

**6. Converter<TInput, TOutput> Delegate (Less Common)**

**Namespace**: System

**Purpose**: Represents a method that converts an object of one type (TInput) to another type (TOutput).

public delegate TOutput Converter<in TInput, out TOutput>(TInput input);

**Example**

Converter<int, string> intToString = (number) => number.ToString();

string result = intToString(42); // result = "42"