**C# Topics:**

**Events:**

Mit C#-Ereignissen können Sie Klassen über Ereignisse in Ihrer Anwendung benachrichtigen und Daten zu bestimmten Zeiten übertragen.

Löst eine Klasse ein Event aus dann handelt es sich um den Publisher, empfängt sie ein Event dann handelt es sich um den Subscriber(mehrere möglich).  
Du kannst ein Event abonnieren, in dem du einem Event einen Eventhandler hinzufügst.

**Generics:**

Generics ermöglichen die Verwendung von Datentypen, die man zum Zeitpunkt der Entwicklung noch nicht festlegen kann oder will.

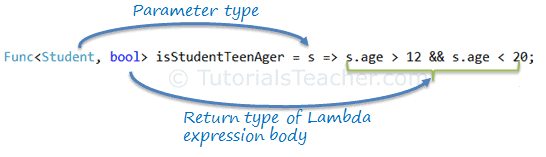
Dazu wird anstelle des konkreten Datentyps in der Klassendefinition ein Platzhalter angegeben.  
Am häufigsten trifft man auf das »T«, das für »Typ« steht.

In der generischen Klasse kann der Platzhalter wie ein regulärer Datentyp verwendet werden.

**Lambda Expressions:**

Auch als anonyme Methoden bezeichnet ermöglichen sie durch ihre Syntax eine verkürzte Schreibweise.

Sie werden oft in Verbindung mit Delegaten verwendet, oder wenn eine Methode nur einmal ausgeführt werden muss und somit eine eigene Methodendefintion umgangen werden kann.



**Collections:**

Ein wichtiges .NET Element zum Verwalten vom Datenbeständen im Speicher sind Collections (Auflistungen). Sie bieten die Möglichkeit während der Laufzeit Objekte hinzuzufügen, ohne vorher die Größe/die Anzahl der Objekte in der Collection zu kennen. Collections bieten Funktionen zum Hinzufügen, Entfernen und Suchen/Finden von Objekten. Ist ein Objekt in einer Auflistung erstmal gefunden, kann dieses über die Objektreferenz auch direkt in der Auflistung geändert werden.

Beispiele: List<T>, Array, ArrayList, Stack, Hashtable, Dictionary<TKey, TValue>

**Assemblys:**

Assemblys bilden die Grundeinheit für Bereitstellung, Versionierung, Wiederverwendung, Aktivierungsbereich und Sicherheitsberechtigungen für .NET-basierte Anwendungen.

Eine Assembly ist eine Sammlung von Typen und Ressourcen, die so konzipiert sind, dass sie zusammenarbeiten, um eine logische Funktionseinheit zu bilden. Assemblys haben die Form von ausführbaren ( .exe ) oder Dynamic Link Library ( .dll ) Dateien und sind die Bausteine ​​von .NET-Anwendungen. Sie stellen der Common Language Runtime die Informationen bereit, die sie benötigt, um die Implementierung eines Typs zu erkennen.

**Namespaces / Namensräume:**

Der Namensraum für .NET Framework Collections liegt unter System.Collections.\*, wobei weitere Sub-Namensräume existieren.

System.Collections.Concurrent (mehrere Threads / Tasks können parallel auf diesen Collections im Speicher operieren)

System.Collections.Immutable (man arbeitet hierbei nur auf Kopien / die ursprünglichen Daten werden nicht geändert – muss zusätzlich über NuGet-Paket installiert werden)

System.Collections.Generic

**Interface:**

Mit C#- Interfacekönnen Sie eine Art Blaupause für Ihre Klassen erstellen. Schnittstellen können mit dem Schlüsselwort interface definiert werden. Eine Interface kann Methoden-, Eigenschafts- und Ereignisdeklarationen enthalten. Felder und automatisch implementierte Eigenschaften sind jedoch nicht zulässig.