实体对实体的Transform之前在IPP3里已经看得太多了，没有技术含量却非常耗时，让大家很不爽。

       所以这里做了一个EntityConvertor<S, T>的类，来实现自动的对象属性赋值，以节省工作量，现在是定义在ECCentral.Portal.Common的Utility下的，主要考虑是在通过Entity给ViewModel赋值的地方使用。（因为我们Service端是使用一套Entity了，所以服务端暂时没有引入这个工具类）

       该工具类赋值的主要逻辑 是根据 T （也就是Target，目标类型）的所有可写属性，然后在S（也就是Source，提供数据的类型）里需找具有相同属性名并且可读得属性，然后将其值读出，通过简单类型转换（工具里会封装好nullable、枚举、基本数据类型等之间的转换）再赋值给T对象的对于属性，如果这里转换失败，会自动忽略；

       如果T类型的属性不是简单类型，该工具类会递归遍历该属性的属性，只到遇到值类型或String为止（所以这里大家要注意递归死循环，不要在一个类型里再通过自身引用自己）。另外如果2个对应的属性都是集合（要求S的属性集合类型实现了IEnumerator<>，而T的属性集合类型实现了ICollection<>），工具也会自动帮你填充集合（并在填充集合时做好集合内元素的转换）

所以方法有重载，可以直接支持将IEnumerable<S>转换为ICollection<T>（S 🡪 T的映射）

同时方法会提供重载，允许传入一个Action<S, T>的callback，从而来人工做属性的mapping赋值（这样针对一些特殊的属性，就可以人工处理下）