1. Difference Between Cast and OfType operators:

The LINQ methods Cast and OfType in C# are used for converting collections to a specified type, but they differ in handling elements that cannot be converted to the target type.

**The Cast operator in C# will try to cast all the elements in the source sequence into a specified type. If any of the elements in the source sequence cannot be cast to the specified type, it will throw an InvalidCastException.**

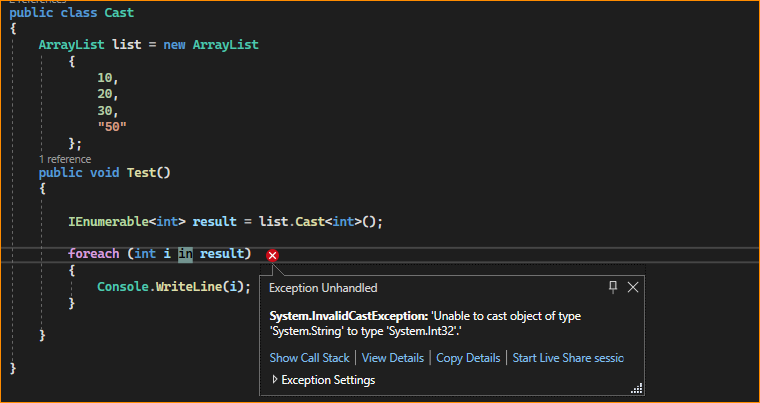
**On the other hand, the OfType operator in C# returns only the elements of the specified type, and the rest of the elements in the source sequence will be ignored and excluded from the result.**

**Key Differences:**

* Exception Handling: Cast throws an exception for elements of the wrong type, while OfType excludes them without throwing an exception.
* Intended Use: Cast is used when the entire collection is expected to be of a single type, whereas OfType filters elements of a specified type from a mixed collection.

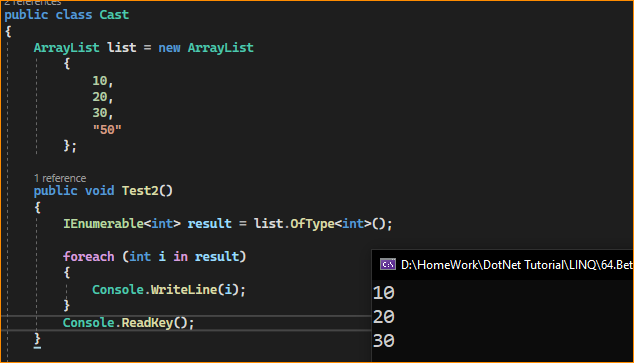
1. Cast Operator Example:

In the following example, the Cast operator throws InvalidCastException as it cannot convert the value “50” (in the form of a string) to an integer.



1. OfType Operator Example:

Let us rewrite the example using the OfType operator, as shown below.



When you run the application, you will not get any exceptions. The value “50” is ignored and excluded from the result.

1. When to use Cast over OfType and Vice Versa?

LINQ Cast Method

* Purpose: The Cast method converts all elements in a non-generic collection (like ArrayList or IEnumerable) to a specified generic type.
* Behavior: It attempts to cast every element in the collection to the specified type.
* Error Handling: If any element cannot be cast to the specified type, Cast throws an InvalidCastException at runtime.
* Use Case: Use Cast when you are certain that every element in the collection can be cast to the specified type, and you want an exception if this is not the case.

We need to use the Cast Operator

When we want to cast all the elements in the collection

When we know for sure that the collection contains only elements of the specified type

LINQ OfType Method

* Purpose: The OfType method filters elements in the collection that are of or can be cast to a specified type.
* Behavior: It includes only those elements in the collection that can be cast to the specified type and silently ignores elements that cannot be cast.
* Error Handling: No exceptions are thrown if elements cannot be cast to the specified type; those elements are simply excluded from the result.
* Use Case: Use OfType when you want to safely extract elements of a specific type from a collection, ignoring elements of other types.

We need to use the OfType operator.

When we want to filter the elements and return only the specified type of elements.

Choose Cast and OfType based on whether the collection is expected to have a uniform type or if it contains a mix of types and you only need elements of a specific type.