**LINQ and Lambda Expressions in C#**

**LINQ (Language Integrated Query)**

* A feature in C# for querying collections (e.g., arrays, lists).
* Allows querying data in a readable, declarative way.
* **Two styles:**

**Query Syntax** (SQL-like):

csharp

var result = from n in numbers

where n > 5

select n;

**Method Syntax** (uses lambda expressions):

csharp

var result = numbers.Where(n => n > 5);

**Lambda Expressions**

* Anonymous functions that define logic in a short, inline way.
* Often used as parameters in LINQ methods.
* **Syntax**: (parameters) => expression
* **Example**:

csharp

CopyEdit

n => n > 5

**Relationship:**

* **LINQ** is a feature for querying data.
* **Lambda expressions** are used inside LINQ methods (e.g., Where, Select).
* **LINQ method syntax** heavily depends on **lambda expressions**.

**✅ Type Safety in C#**

Before executing the code we know we assigned wrong data.(red error line throws)

* **Type safety** ensures that a variable is only assigned values that match its declared data type.
* It helps **prevent runtime errors** by catching **type mismatches at compile time**.
* C# does not allow operations between incompatible types without **explicit casting**.
* It **protects memory** by disallowing unsafe operations like treating one type as another incorrectly.

**Pass by value and Pass by refernce**