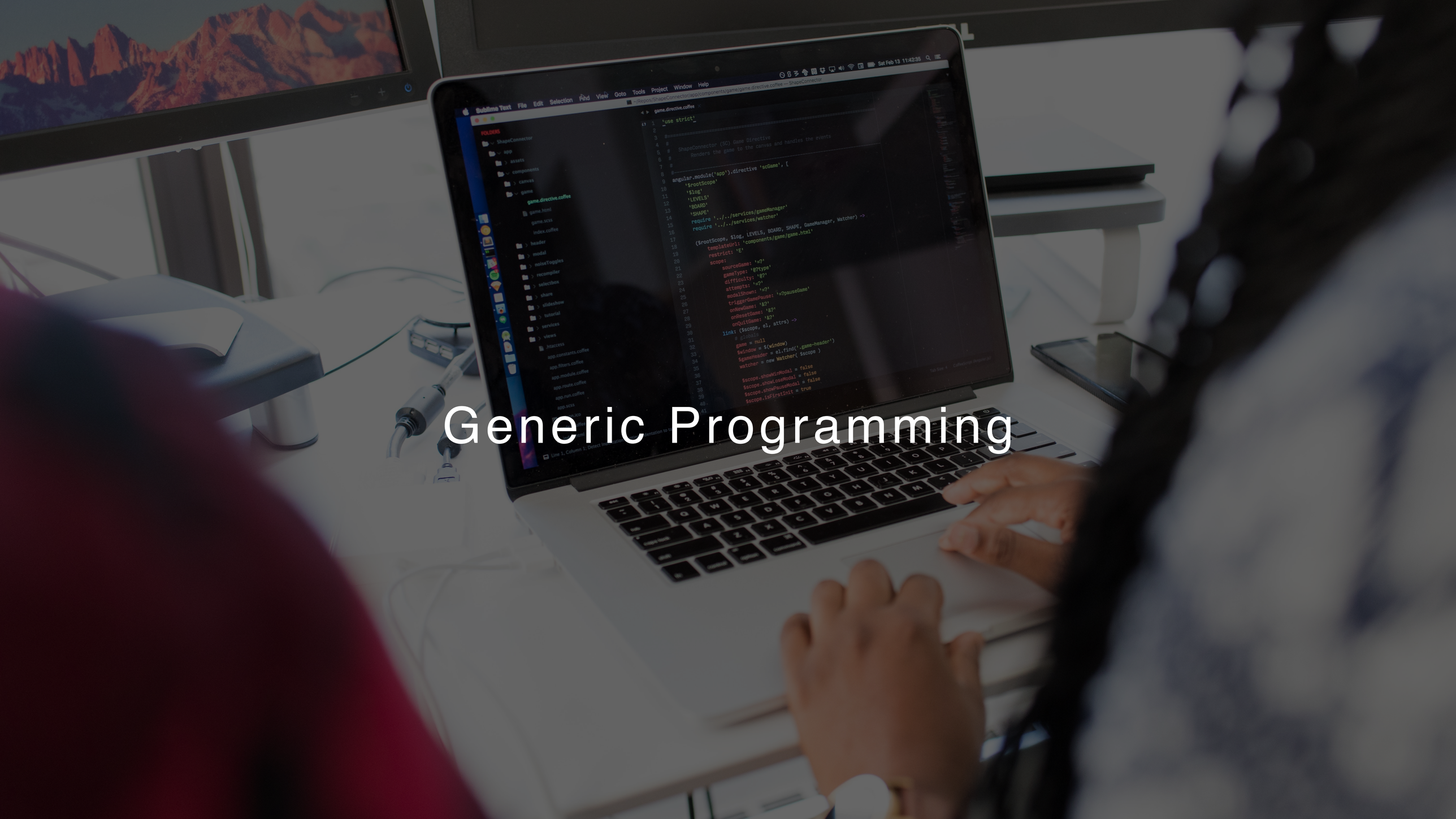
Advanced C#

LUKÁŠ VAVREK

Outline

Generic Programming

LINQ


Generic Programming

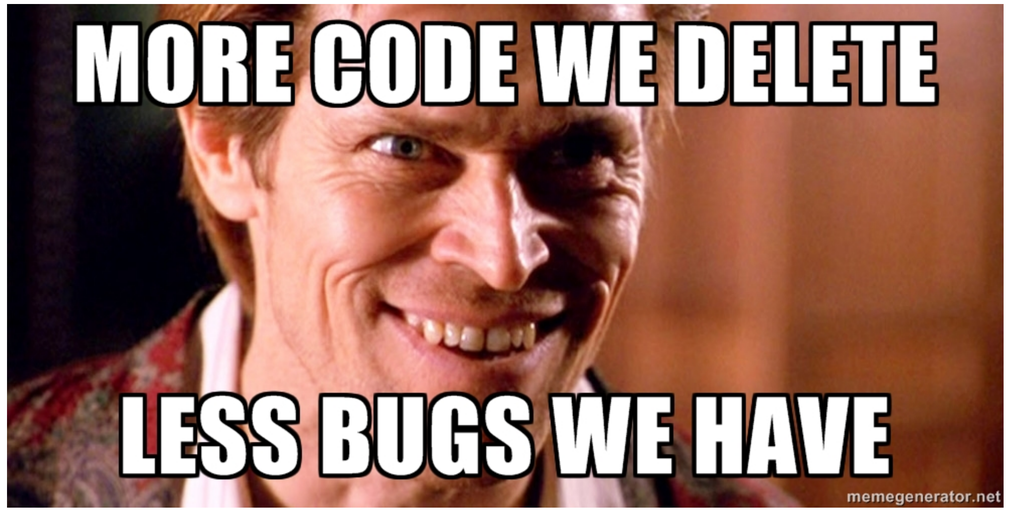
Awesome storage class

```
class IntStorage
{
    private int[] _data;

    public void add(int n) { ... }
    public int get(int index) { ... }
}
```

We should support strings

```
class StringStorage
{
    private string[] _data;

    public void add(string n) { ... }
    public string get(int index) { ... }
}
```

What is the problem?



- Repetitive code
- Shared bugs vs Specific bugs
- Classes have nothing in common

Generic code

```
class Storage<T>
{
    private T[] _data;

    public void add(T n) { ... }
    public T get(int index) { ... }
}
```

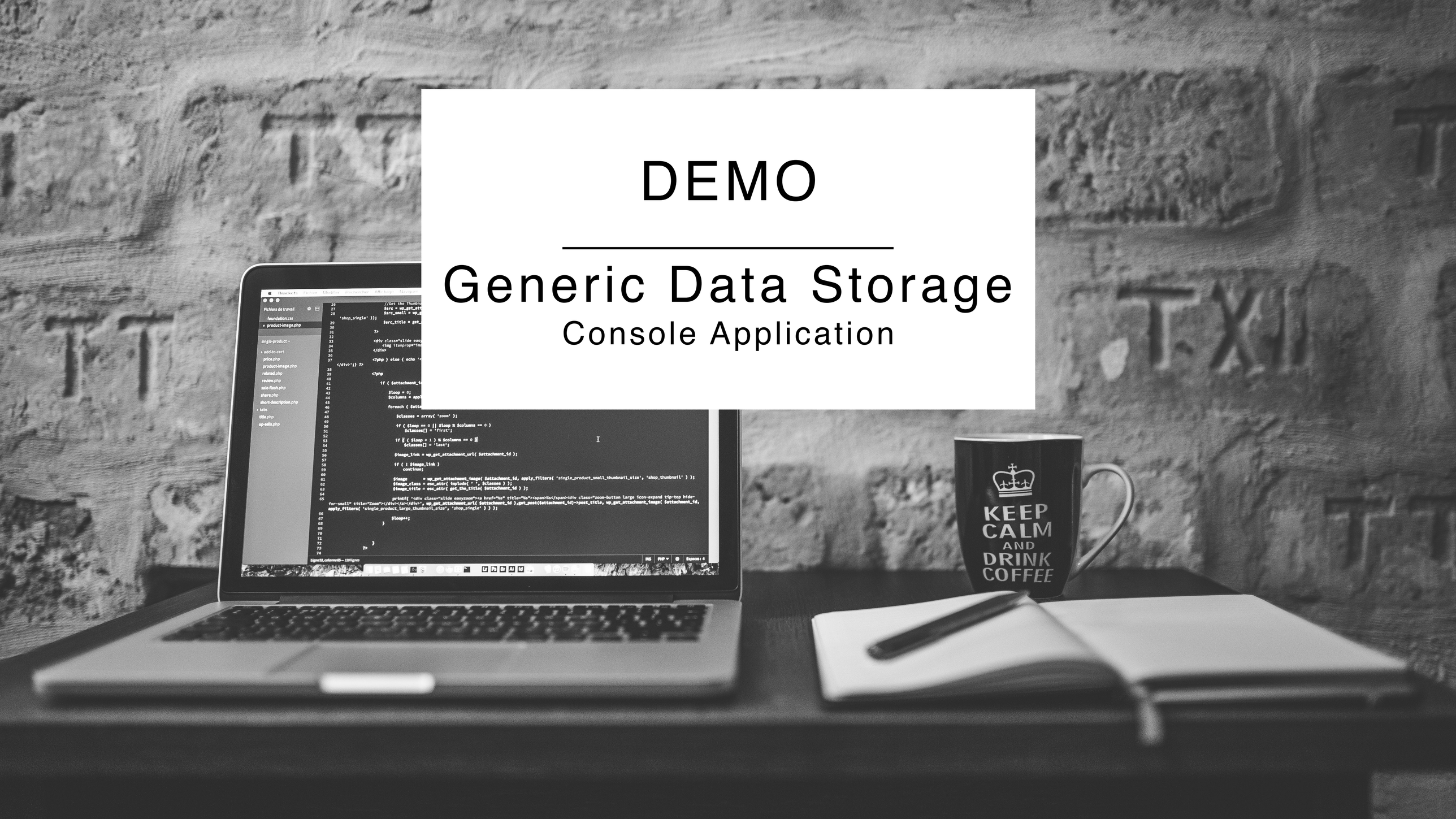
Instantiation

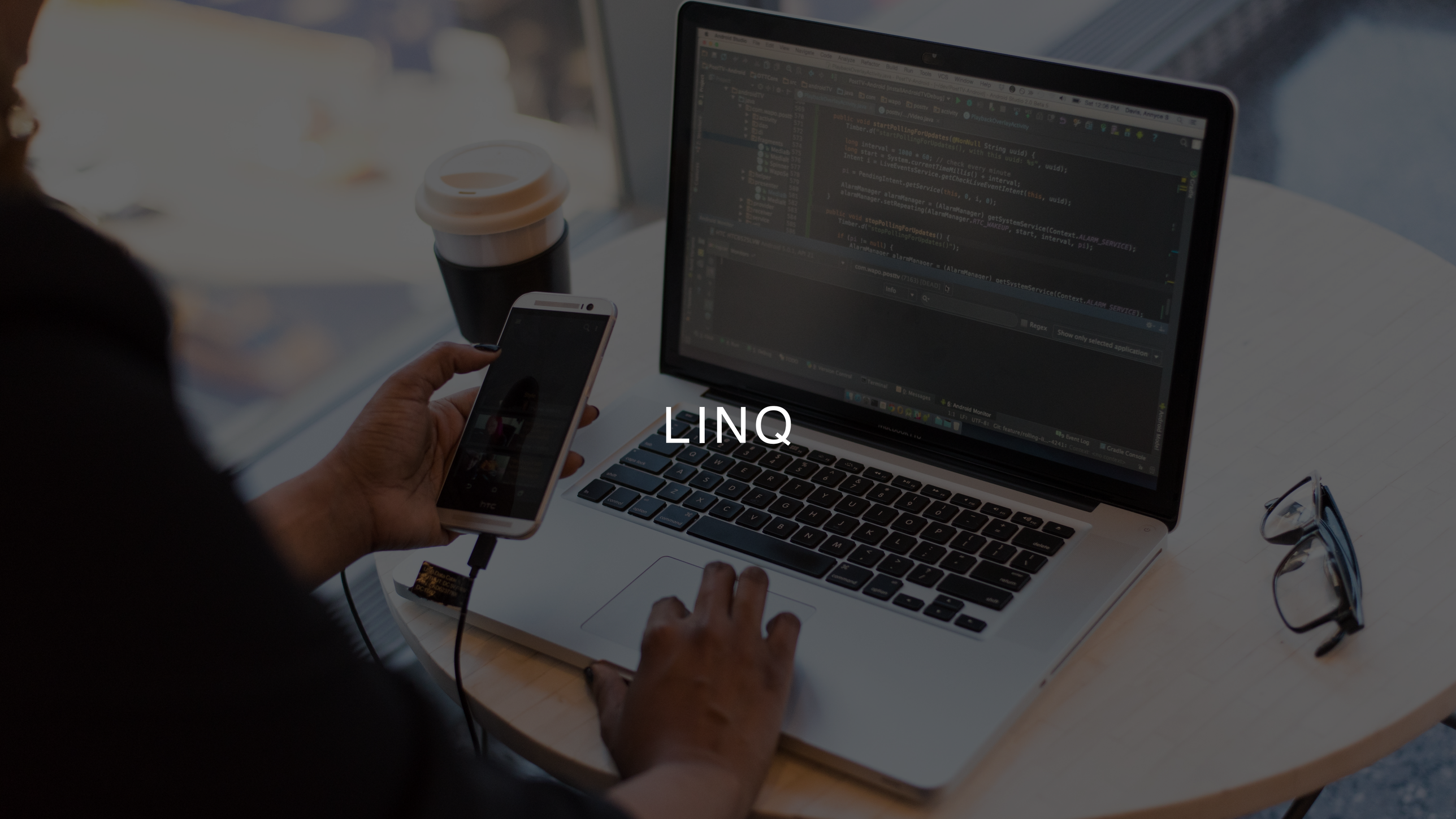
```
var intStorage = new Storage<int>()
var stringStorage = new Storage<string>()
```

DEMO

Generic Data Storage

Console Application




LINQ

SQL

SQL (Structured Query Language) is a domain-specific language used in programming and designed for managing data held in a relational database management systems (RDBMS), or for stream processing in a relational data stream management system (RDSMS).

SQL

```
SELECT p.Name AS ProductName,
NonDiscountSales = (OrderQty * UnitPrice),
Discounts = ((OrderQty * UnitPrice) *
UnitPriceDiscount)
FROM Production.Product AS p
INNER JOIN Sales.SalesOrderDetail AS sod
ON p.ProductID = sod.ProductID
ORDER BY ProductName DESC;
```

LINQ

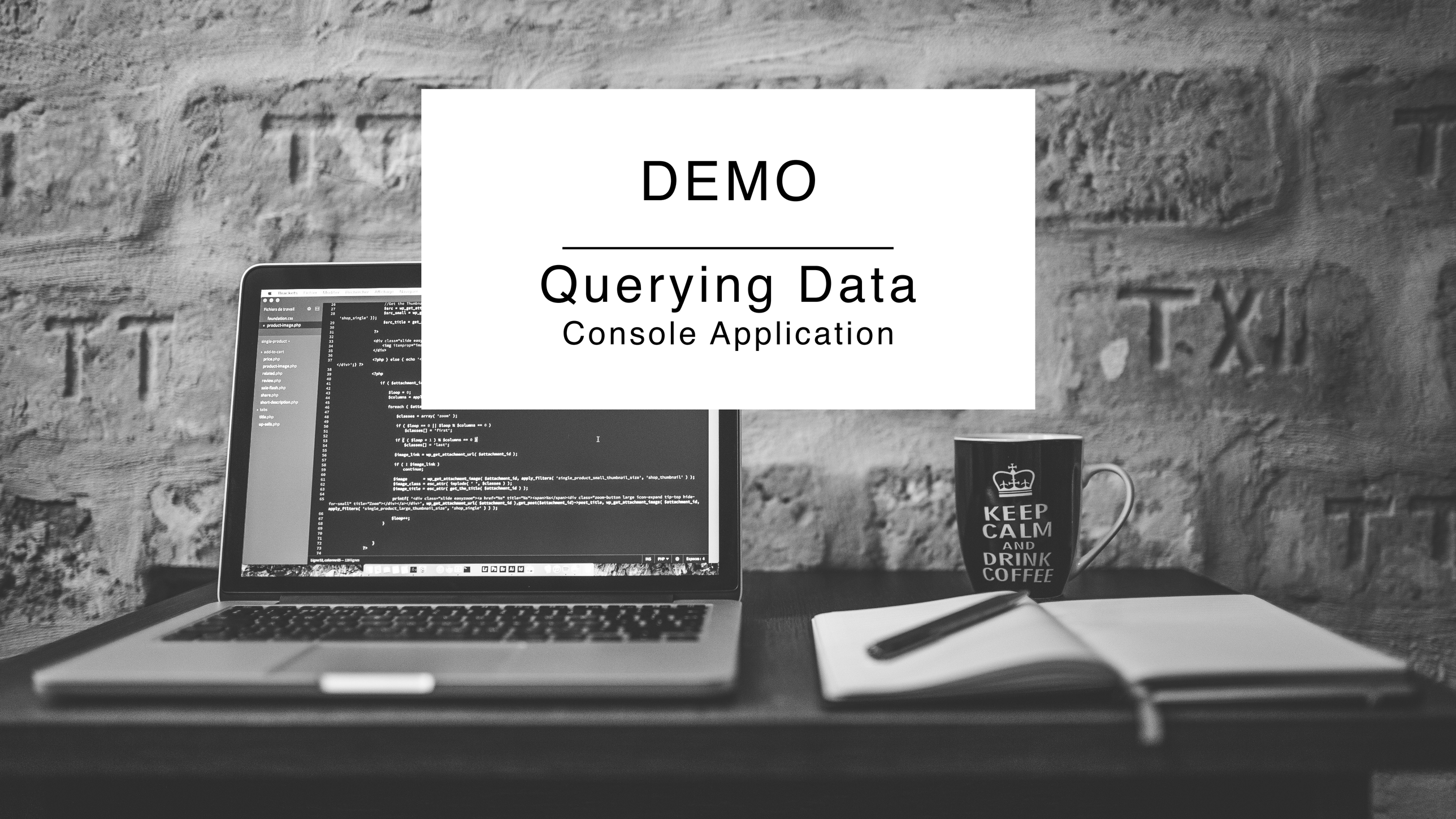
LINQ (Language Integrated Query) is a Microsoft .NET Framework component that adds native data querying capabilities to .NET languages. LINQ extends the language by the addition of query expressions, which are akin to SQL statements and can be used to conveniently extract and process data from arrays, enumerable classes, XML documents, relational databases, and third-party data sources.

LINQ

```
var query_join4 =
    from a in svcContext.AccountSet
    join c in svcContext.ContactSet
      on a.PrimaryContactId.Id equals c.ContactId
    join l in svcContext.LeadSet
      on a.OriginatingLeadId.Id equals l.LeadId
    select new
    {
        contact_name = c.FullName,
        account_name = a.Name,
        lead_name = l.FullName
    };
```

LINQ

```
var studentNames = studentList.Where(s => s.Age > 18)
                              .Select(s => s)
                              .Where(st => st.StandardID > 0)
                              .Select(s => s.StudentName);
```



DEMO

Querying Data

Console Application


A 3
B 3
A 4
B 4
Q & A