

Tarefa 02

ALUNO : URIAS ABREU

LINGUAGEM C#

LANGUAGE-INTEGRATED QUERY (LINQ)

O que é LINQ?

- ▶ O LINQ (consulta integrada à linguagem) é o nome de um conjunto de tecnologias com base na integração de recursos de consulta diretamente na linguagem C#.

Porque usá-lo ?

Você precisará aprender uma linguagem de consulta diferente para cada tipo de fonte de dados: Bancos de dados SQL, documentos XML, vários serviços Web e assim por diante.

Você escreve consultas em coleções fortemente tipadas de objetos usando palavras-chave da linguagem e operadores familiares. A família de tecnologias LINQ fornece uma experiência de consulta consistente para objetos (LINQ to Objects), bancos de dados relacionais (LINQ to SQL) e XML (LINQ to XML).

Porque usá-lo ?

Para um desenvolvedor que escreve consultas, a parte mais visível "integrada à linguagem" do LINQ é a expressão de consulta.

As expressões de consulta são uma *sintaxe declarativa de consulta*.

Usando a sintaxe de consulta, você pode executar operações de filtragem, ordenação e agrupamento em fontes de dados com o mínimo de código.

Você pode usar os mesmos padrões de expressão de consulta básica para consultar e transformar dados em bancos de dados SQL, conjuntos de dados do ADO.NET, documentos XML e fluxos e coleções .NET.

LINGUAGEM JAVA

API STREAM

VISÃO GERAL

- ▶ Uma das principais características da Stream API é que ela concentra em disponibilizar um trio de operações conhecido como:
- ▶ Filter
- ▶ Map
- ▶ Reduce.
- ▶ Essas operações servem para filtrar dados em uma coleção e retorná-los de acordo com a necessidade da aplicação.

100

[illegible]

EXEMPLO EM JAVASCRIPT

```
var cores = ['vermelho', 'verde', 'azul'];  
cores.forEach(cor => console.log(cor));  
// vermelho  
// verde  
// azul
```



QUAL A DIFERENÇA?

PODER DA ABSTRAÇÃO

LINQ (LANGUAGE INTEGRATED QUERY) É UMA SINTAXE DE CONSULTA UNIFORME EM C # E VB.NET PARA RECUPERAR DADOS DE DIFERENTES FONTES E FORMATOS. ELE É INTEGRADO EM C # OU VB, ELIMINANDO ASSIM A INCOMPATIBILIDADE ENTRE LINGUAGENS DE PROGRAMAÇÃO E BANCOS DE DADOS, ALÉM DE FORNECER UMA ÚNICA INTERFACE DE CONSULTA PARA DIFERENTES TIPOS DE FONTES DE DADOS.

Standard Query Operators (Operadores de consulta padrão)

- ▶ Where
- ▶ OrderBy
- ▶ ThenBy
- ▶ GroupBy, ToLookup
- ▶ Join
- ▶ GroupJoin
- ▶ FirstOrDefault
- ▶ Select
- ▶ All, Any
- ▶ Contains
- ▶ Aggregate
- ▶ Average
- ▶ Count

SELECT

[illegible]

GroupJoin

Example: GroupJoin in Method syntax C#

```
IList<Student> studentList = new List<Student>() {  
    new Student() { StudentID = 1, StudentName = "John", StandardID =1 },  
    new Student() { StudentID = 2, StudentName = "Moin", StandardID =1 },  
    new Student() { StudentID = 3, StudentName = "Bill", StandardID =2 },  
    new Student() { StudentID = 4, StudentName = "Ram", StandardID =2 },  
    new Student() { StudentID = 5, StudentName = "Ron" }  
};  
  
IList<Standard> standardList = new List<Standard>() {  
    new Standard(){ StandardID = 1, StandardName="Standard 1"},  
    new Standard(){ StandardID = 2, StandardName="Standard 2"},  
    new Standard(){ StandardID = 3, StandardName="Standard 3"}  
};  
  
var groupJoin = standardList.GroupJoin(studentList, //inner sequence  
    std => std.StandardID, //outerKeySelector  
    s => s.StandardID, //innerKeySelector  
    (std, studentsGroup) => new // resultSelector  
    {  
        Students = studentsGroup,  
        StandarFullIdName = std.StandardName  
    });
```



FIM