

Exemplos com ArrayLists e suas Propriedades e Métodos – Coleções: Estrutura da Linguagem – Parte 2

Veja neste artigo exemplos com ArrayLists, suas propriedades e seus métodos.

Olá pessoal, neste artigo veremos alguns exemplos com o **ArrayList**. Veremos também suas principais propriedades. Acompanhem:

Uma das classes do .NET disponíveis para gerenciar uma coleção é a classe **ArrayList**, que se encontra dentro do namespace **System.Collections**.

Inserção de um elemento no final da coleção – Usando o método **Add**, é possível adicionarmos um elemento ao final de um **ArrayList**, o qual é redimensionado automaticamente. Veja no exemplo abaixo como:

```
ArrayList listaEstados = new ArrayList();

listaEstados.Add("São Paulo");

listaEstados.Add("Rio de Janeiro");

listaEstados.Add("Minas Gerais");
```

Remoção de um elemento – Usando o método **Remove**, podemos remover um elemento de um **ArrayList**, a qual reordena seus elementos de maneira automática. Veja no exemplo:

```
ArrayList listaEstados = new ArrayList();

listaEstados.Add("São Paulo");

listaEstados.Add("Rio de Janeiro");

listaEstados.Add("Minas Gerais");

listaEstados.Remove("São Paulo");
```

Outro método usado para inserir elementos é o **Insert**. A diferença é que aqui podemos especificar onde será inserido o elemento, usando índice. Como nos outros métodos, o **ArrayList** será redimensionado. Veja o exemplo:

```
ArrayList listaEstados = new ArrayList();

listaEstados.Add("São Paulo");
```

```
listaEstados.Add("Rio de Janeiro");
```

```
listaEstados.Add("Minas Gerais");
```

```
listaEstados.Insert(3, "Pernambuco");
```

Principais Propriedades – Abaixo descrevo a principal propriedade e os métodos da classe **ArrayList**:

- **Propriedade**

- **Count** – Retorna o número de elementos.

- **Métodos**

- **Add** – Adiciona um item no final da coleção;
- **AddRange** – Adiciona um array ou uma coleção no final da coleção;
- **BinarySearch** – Realiza uma pesquisa binária em uma coleção que deve estar ordenada. A pesquisa binária é muito rápida porque testa blocos de elementos de uma só vez;
- **Contains** – Retorna o índice de um elemento se ele estiver na coleção;
- **IndexOf** – Igual ao Contains, retorna o índice de um determinado elemento;
- **Clone** – Duplica a ArrayList
- **LastIndexOf** – Retorna a última ocorrência de um item em um array;
- **Insert** – Insere um elemento na coleção, no local especificado pelo usuário;
- **Remove** – Remove um item da coleção.

Fonte de Consulta: Apostila de C#, da Impacta Tecnologia.

Assim finalizo o artigo. Na próxima e última parte veremos as classes **Stack**, **HashTable** e **Queue**, não percam!