

Paralelizando execução de consulta - Prática

Considere a seguinte consulta LINQ:

```
var queryCodigos =  
listaFaixas  
.Select(f => new  
{  
    Arquivo = string.Format("{0}\\{1}.jpg", Imagens, f.FaixaId),  
    Imagem = barcodWriter.Write(string.Format("aluratunes.com/faixa/{0}", f.FaixaId))  
});
```

Supondo que o método `barcodWriter.Write` exija um grande esforço computacional, como você modificaria o código acima para utilizar LINQ paralelo?

Selecione uma alternativa

A

```
var queryCodigos =  
listaFaixas  
.Paralelize()  
.Select(f => new  
{  
    Arquivo = string.Format("{0}\\{1}.jpg", Imagens, f.FaixaId),  
    Imagem = barcodWriter.Write(string.Format("aluratunes.com/faixa/{0}", f.FaixaId))  
});
```

B

```
var queryCodigos =  
listaFaixas  
.Select(f => new  
{  
    Arquivo = string.Format("{0}\\{1}.jpg", Imagens, f.FaixaId),  
    Imagem = barcodWriter.Write(string.Format("aluratunes.com/faixa/{0}", f.FaixaId))  
})  
.AsParallel();
```

C

```
var queryCodigos =  
listaFaixas  
.Select(f => new  
{  
    Arquivo = string.Format("{0}\\{1}.jpg", Imagens, f.FaixaId),  
    Imagem = barcodWriter.Write(string.Format("aluratunes.com/faixa/{0}", f.FaixaId))  
});  
  
queryCodigos = queryCodigos.AsParallel();
```

D

```
var queryCodigos =  
new Parallel(listaFaixas  
.Select(f => new  
{  
    Arquivo = string.Format("{0}\\{1}.jpg", Imagens, f.FaixaId),  
    Imagem = barcodWriter.Write(string.Format("aluratunes.com/faixa/{0}", f.FaixaId))  
}));
```

E

```
var queryCodigos =  
listaFaixas  
.AsParallel()  
.Select(f => new  
{  
    Arquivo = string.Format("{0}\\{1}.jpg", Imagens, f.FaixaId),  
    Imagem = barcodWriter.Write(string.Format("aluratunes.com/faixa/{0}", f.FaixaId))  
}));
```