

Mão na massa: Alterando nossas classes

Vamos começar a construção do nosso **Blog**.

- 1) Baixe o arquivo do projeto [aqui \(https://caelum-online-public.s3.amazonaws.com/712-dotnet-mongodb-parte2/01/projetoBlog01-inicial-para-importar.zip\)](https://caelum-online-public.s3.amazonaws.com/712-dotnet-mongodb-parte2/01/projetoBlog01-inicial-para-importar.zip), e descompacte-o no diretório que você está utilizando para os projetos.
- 2) Abra o arquivo **projetoBlog** no Visual Studio. Você aproveitará também os exemplos que fez no primeiro curso!
- 3) Procure por comentários com "XXX", nesses pontos específicos do programa você incluirá a interação Blog - MongoDB
- 4) Adicione a **MongoDriver**, clique em **Ferramentas**, escolha **Gerenciador de Pacotes NuGet** e então **Gerenciar Pacotes do NuGet para a Solução....** Procure por **Mongo** e instale o **MongoDB.Driver**.
- 5) Em **Models**, comece a trabalhar na classe **AcessoMongoDB**.
- 6) Antes de tudo, adicione a linha referente ao **MongoDB.Driver** e a **MongoDB.Bson**;

```
using MongoDB.Driver;  
using MongoDB.Bson;
```

Copie o exemplo do último curso e cole na classe para que façamos as mudanças.

- 7) Primeiro, modifique o nome da coleção e adicione uma linha parecida logo abaixo dela:

```
public const string NOME_DA_COLECAO_PUBLICACAO = "Publicacoes";  
public const string NOME_DA_COLECAO_USUARIO = "Usuarios";
```

- 8) Altere o nome da função **conectandoMongoDB** para o mesmo nome da classe (**AcessoMongoDB**)
- 9) Duplicue a função **IMongoCollection** e altere, ficando da seguinte forma:

```
public IMongoCollection<Usuario> Usuarios  
{  
    get { return _BaseDeDados.GetCollection<Usuario>(NOME_DA_COLECAO_USUARIO); }  
}  
  
public IMongoCollection<Publicacao> Publicacoes  
{  
    get { return _BaseDeDados.GetCollection<Usuario>(NOME_DA_COLECAO_PUBLICACAO); }  
}
```

- 10) Prosseguindo, crie as classes de **Comentario**, **Publicacao** e **Usuario**. Lembre-se que você pode utilizar o site [json2csharp \(http://json2csharp.com/\)](http://json2csharp.com/) para facilitar o trabalho.

11) Na classe **Usuario**, adicione 2 linhas logo abaixo do último `using`

```
using MongoDB.Bson;  
using MongoDB.Bson.Serialization.Attributes;
```

12) Dentro da classe **Usuario**, adicione o seguinte trecho de código:

```
[BsonRepresentation(BsonType.ObjectId)]  
public string Id { get; set; }
```

Abaixo disso cole o que foi gerado no site [json2csharp \(http://json2csharp.com/\)](http://json2csharp.com/)! Repita o processo para a classe **Publicacao**.

13) Na classe **Comentario**, não há a necessidade de adicionar as linhas `using`, bastando adicionar apenas o que foi gerado no [json2csharp \(http://json2csharp.com/\)](http://json2csharp.com/). Também não é necessário criar um ID.