

09

Faça o que eu fiz na aula

Vamos adicionar os comentários nos artigos. Em um arquivo novo com o nome de `modelando-comentarios.mongo`, vamos fazer um update:

```
db.artigos.update({}, {
  $unset: {
    comentarios: []
  }
}, { multi: true })
```

Para adicionarmos os dados, execute os comandos que estão dentro do arquivo `insercoes/inserir-comentarios.mongo`.

No arquivo `modelando-comentarios.mongo`, vamos fazer uma agregação para mover os comentários para uma nova coleção:

```
db.artigos.aggregate([
  {
    $unwind: "$comentarios"
  },
  {
    $project: {
      _id: 0,
      id_artigo: "$_id",
      nome: "$comentarios.nome",
      email: "$comentarios.email",
      texto: "$comentarios.texto",
      data: "$comentarios.data"
    }
  },
  {
    $out: "comentarios"
  }
])
```

Agora precisamos manter somente os cinco comentários mais recentes no documento do artigo, vamos ordenar os comentários pela data e se houver mais do que cinco comentários, os excedentes são removidos.

Vamos utilizar o seguinte comando na linha de comando do MongoDB:

```
db.artigos.find({
  comentarios: {
    $exists: true
  }
}).forEach(function(artigo) {

  artigo.comentarios = db.comentarios.find({
    id_artigo: artigo._id
  }, {
    _id: 0,
    nome: 1,
  })
})
```

```

    email: 1,
    texto: 1,
    data: 1
}).sort({ data: 1 }).limit(5).toArray();

db.artigos.save(artigo)

})

```

Depois disso, o processo de inserção ficou um pouco mais longo, envolve três comandos que devem ser executados na linha de comando do MongoDB. Primeiro abrimos espaço nos cinco comentários mais recentes e inserimos no começo do array:

```

db.artigos.update({
  slug: "visualizando-dados-de-frequencia"
}, {
  $pop: {
    comentarios: 1
  }
})

var documento = db.artigos.findOneAndUpdate({
  slug: "visualizando-dados-de-frequencia"
}, {
  $push: {
    comentarios: {
      $each: [
        {
          nome: "Giovanni",
          email: "giovanni@alura.com.br",
          texto: "Gostei do artigo!",
          data: ISODate('2019-09-24')
        }
      ],
      $position: 0
    }
  }
}, { returnNewDocument: true })

```

E para inserir o comentário na coleção de comentários:

```

db.comentarios.insertOne({
  id_artigo: documento._id,
  nome: "Giovanni",
  email: "giovanni@alura.com.br",
  texto: "Gostei do artigo!",
  data: ISODate('2019-09-24')
})

```

Com isso a modelagem do nosso banco de dados está completa! Vamos fazer agora uma inserção de um autor, artigo e comentário com essa nova estrutura, em um arquivo chamado inserindo.mongo, vamos digitar os seguintes comandos:

Para inserirmos um autor:

```
db.infoAutores.insertOne({
  nome: "Yan Orestes",
  email: "yan@caelum.com.br",
  senha: "yan1234",
  descricao: "Instrutor e desenvolvedor na Alura"
})
```

Para inserirmos um artigo deste autor, vamos precisar do documento do autor para preencher algumas informações:

```
var autorInfo = db.infoAutores.findOne({
  email: "yan@caelum.com.br"
})
```

Agora sim podemos inserir o artigo:

```
db.artigos.insertOne({
  autor: {
    nome: autorInfo.nome,
    descricao: autorInfo.descricao,
    id_info: autorInfo._id
  },
  slug: "dicas-politicas-de-senha",
  titulo: "Algumas dicas importantes sobre políticas de senha",
  conteudo: "O fato é que, por algum motivo, meus usuários estavam usando senhas fracas para suas",
  dataCriado: ISODate('2018-07-18'),
  categorias: [
    'Arquitetura',
    'Programação'
  ],
  comentarios: []
})
```

Para inserirmos um comentário neste artigo:

```
var documento = db.artigos.findOneAndUpdate({
  slug: "dicas-politicas-de-senha"
}, {
  $push: {
    comentarios: {
      $each: [
        {
          nome: "Giovanni",
          email: "giovanni@alura.com.br",
          texto: "Achei muito legal esse artigo!",
          data: ISODate('2019-09-24')
        }
      ],
      $position: 0
    }
  },
  { returnNewDocument: true })
})
```

E por fim, precisamos inserir o comentário na coleção de comentários:

```
db.comentarios.insertOne({  
    id_artigo: documento._id,  
    nome: "Giovanni",  
    email: "giovanni@alura.com.br",  
    texto: "Achei muito legal esse artigo!",  
    data: ISODate('2019-09-24')  
})
```