



UNITAU
Universidade de Taubaté





MongoDB Find



Exercícios



1) Relacionar o título e ISBN dos livros de 2010

```
> db.livro.find( {ano: "2010"},  
  {_id:0,titulo:1,isbn:1} )  
{ "titulo" : "Programming With QT", "isbn" :  
  "9781449390938" }  
{ "titulo" : "Programacao em Linguagem C",  
  "isbn" : "8573939494" }
```


Exercícios



2) Relacionar o nome e nota da disciplina de matematica

```
> db.notas.find( {disciplina: "matematica"},  
  {_id:0,nome:1,nota:1})  
{ "nome" : "Andre", "nota" : 87 }  
{ "nome" : "Ana", "nota" : 78 }  
{ "nome" : "Joao", "nota" : 98 }
```

Exercícios



3) Encontrar os empregados que nao se reportam ao Eliot

```
> db.employees.find( {reportsTo: {$ne:  
"Eliot"}} } )  
{ "_id" : 1, "name" : "Eliot" }  
{ "_id" : 4, "name" : "Asya", "reportsTo" :  
"Ron" }  
{ "_id" : 5, "name" : "Dan", "reportsTo" :  
"Andrew" }
```

Exercícios



4) Relacionar o assunto e autor dos artigos com 90 visualizações ou mais

```
> db.articles.find( {views: {$gte: 90}},  
  {_id:0,subject:1,author:1} )  
{ "subject" : "Baking a cake", "author" :  
  "Jeff" }  
{ "subject" : "Café Con Leche", "author" :  
  "Jeff" }  
{ "subject" : "baking", "author" : "Ann" }
```


Exercícios



5) Relacionar o id, assunto e autor dos artigos do Ben, Jeff e Betty

```
> db.articles.find( {author: {$in: ["Ben",  
"Jeff", "Betty"]}}, {subject:1, author:1} )  
{ "_id" : 2, "subject" : "Coffee Shopping",  
"author" : "Ben" }  
{ "_id" : 3, "subject" : "Baking a cake",  
"author" : "Jeff" }  
{ "_id" : 5, "subject" : "Café Con Leche",  
"author" : "Jeff" }  
{ "_id" : 6, "subject" : "Blueberry pie",  
"author" : "Betty" }  
{ "_id" : 7, "subject" : "coffee and cream",  
"author" : "Ben" }
```

Exercícios



6) Encontrar as notas entre 70 e 90

```
> db.notas.find( {nota: {$gte: 70, $lte: 90}}
)
{ "_id" : 5, "nome" : "Andre", "disciplina" :
"matematica", "nota" : 87 }
{ "_id" : 6, "nome" : "Andre", "disciplina" :
"portugues", "nota" : 89 }
{ "_id" : 8, "nome" : "Ana", "disciplina" :
"matematica", "nota" : 78 }
{ "_id" : 9, "nome" : "Ana", "disciplina" :
"portugues", "nota" : 90 }
{ "_id" : 3, "nome" : "Joao", "disciplina" :
"portugues", "nota" : 77 }
{ "_id" : 10, "nome" : "Paula",
"disciplina" : "portugues", "nota" : 85 }
```


Exercícios



7) Encontrar as notas de matematica entre 70 e 90

```
> db.notas.find( {$and:  
  [{disciplina:"matematica"},{nota:{$gte:70}},  
  {nota:{$lte:90}}] } )  
{ "_id" : 5, "nome" : "Andre", "disciplina" :  
  "matematica", "nota" : 87 }  
{ "_id" : 8, "nome" : "Ana", "disciplina" :  
  "matematica", "nota" : 78 }
```

Exercícios



8) Encontrar as notas de matematica abaixo de 80 e acima de 90

```
> db.notas.find( {$and:  
  [{disciplina:"matematica"},{$or:[{nota:  
    {$lt:80}}, {nota:{$gt:90}}]}} ] } )  
{ "_id" : 8, "nome" : "Ana", "disciplina" :  
  "matematica", "nota" : 78 }  
{ "_id" : 2, "nome" : "Joao", "disciplina" :  
  "matematica", "nota" : 98 }
```

Exercícios



9) Encontrar os aeroportos que nao sejam JFK, ORD ou LHR

```
> db.airports.find( {$nor: [{airport: "JFK"},  
{airport: "ORD"}, {airport:"LHR"}]} )  
{ "_id" : 3, "airport" : "PWM", "connects" :  
[ "BOS", "LHR" ] }  
{ "_id" : 1, "airport" : "BOS", "connects" :  
[ "JFK", "PWM" ] }
```


Exercícios



10) Encontrar os empregados que nao se reportam ao Eliot, mas que se reportam a alguem

```
> db.employees.find( {reportsTo: {$ne:  
"Eliot", $exists: 1}} )  
{ "_id" : 4, "name" : "Asya", "reportsTo" :  
"Ron" }  
{ "_id" : 5, "name" : "Dan", "reportsTo" :  
"Andrew" }
```