

# **BANCO DE DADOS**

## **MongoDB – Parte 2**

```
_id: ObjectId("573a1392f29313caabcda636")
plot: "In the 1920s, the Provence is a magnet for immigrants seeking work in ..."
✓ genres: Array
  0: "Drama"
  runtime: 81
✓ cast: Array
  0: "Charles Blavette"
  1: "Celia Montalvén"
  2: "èdouard Delmont"
  3: "Max Dalban"
poster: "https://m.media-amazon.com/images/M/MV5BMjYwZDlhMjMtMWYyZC00NzY5LTNmMj..."
title: "Toni"
fullplot: "In the 1920s, the Provence is a magnet for immigrants seeking work in ..."
✓ languages: Array
  0: "French"
  1: "Italian"
  2: "Spanish"
  released: 1936-11-04T00:00:00.000+00:00
> directors: Array
✓ writers: Array
  0: "Jean Renoir (screenplay)"
  1: "Jacques Levert (story)"
> awards: Object
  lastupdated: "2015-08-20 01:13:03.983000000"
  year: 1935
> imdb: Object
> countries: Array
  type: "movie"
> tomatoes: Object
```

# SELECIONAR DADOS

- O processo de pesquisa no MongoDB é bem diferente dos bancos relacionais tradicionais.
- Na plataforma Cloud (Atlas), podemos usar a ferramenta “Filter” na opção “Find” para os filtros.

# sample\_mflix.movies

COLLECTION SIZE: 35.88MB    TOTAL DOCUMENTS: 23530    INDEXES TOTAL SIZE: 13.4MB

Find

Indexes

Schema Anti-Patterns 0

Aggregation

Search Indexes ●

INSERT DOCUMENT

**FILTER** {runtime:33}

Find

Reset

QUERY RESULTS 1-2 OF 2

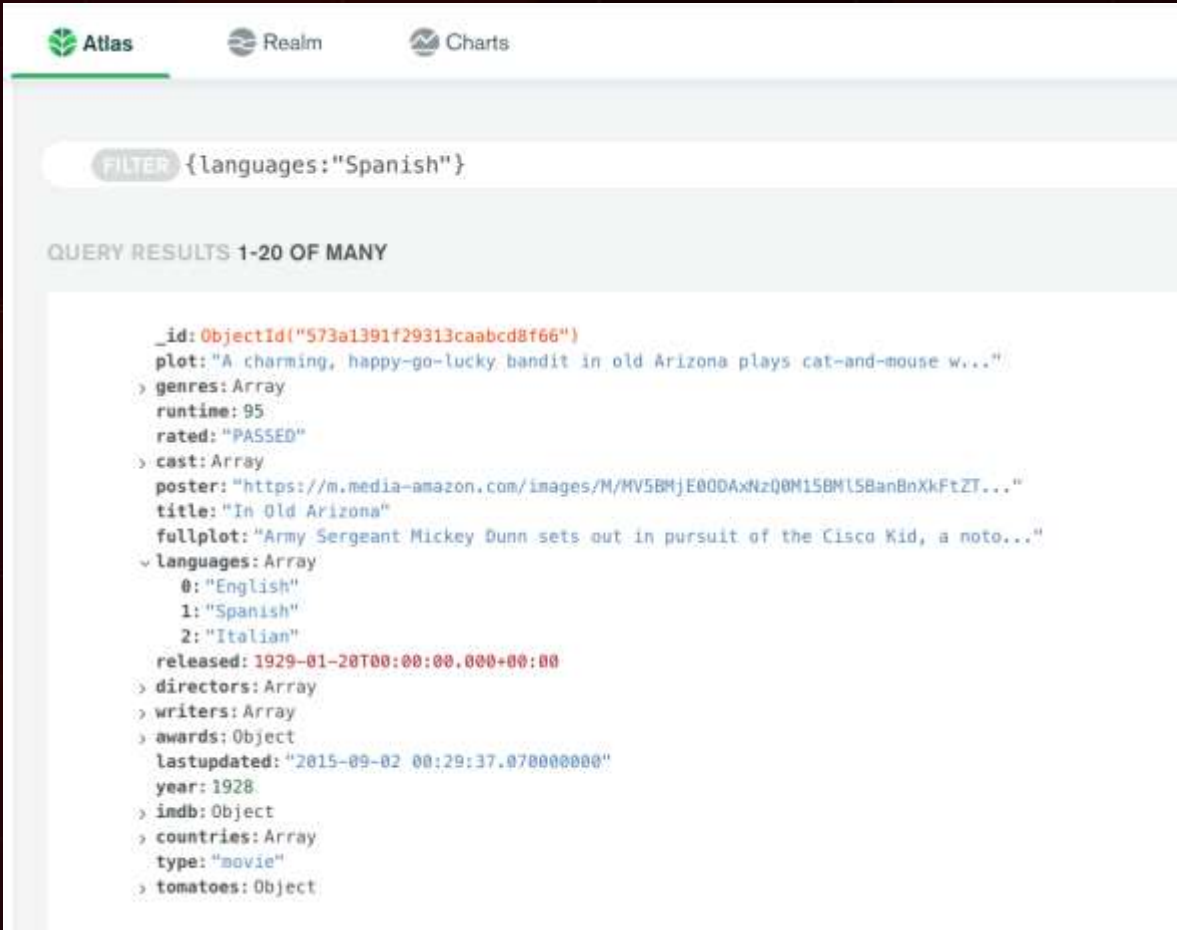


```
_id: ObjectId("573a13a1f29313caabd067b4")
plot: "At a New Millennium Eve party Blackadder and Baldrick test their new t..."
> genres: Array
  runtime: 33
> cast: Array
  poster: "https://m.media-amazon.com/images/M/MV5BMTY1MTc4Njc4Nl5BMl5BanBnXkFtZT..."
  title: "Blackadder Back & Forth"
  lastupdated: "2015-09-16 00:23:03.747000000"
> languages: Array
  released: 1999-12-06T00:00:00.000+00:00
> directors: Array
> writers: Array
> awards: Object
  year: 1999
> imdb: Object
> countries: Array
  type: "movie"
> tomatoes: Object
```



# SELECIONAR DADOS

- Realizando uma busca em um campo do tipo array.



The screenshot shows the Atlas web interface. At the top, there are tabs for 'Atlas', 'Realm', and 'Charts'. Below the tabs, there is a 'FILTER' button and a text input containing the JSON query: `{languages:"Spanish"}`. Below the filter, it says 'QUERY RESULTS 1-20 OF MANY'. The results are displayed in a collapsible tree view. The first result is expanded, showing the following fields: `_id` (ObjectId), `plot` (A charming, happy-go-lucky bandit in old Arizona plays cat-and-mouse w...), `genres` (Array), `runtime` (95), `rated` (PASSED), `cast` (Array), `poster` (https://m.media-amazon.com/images/M/MV5BMjE0ODAxNzQ0M15BM15BanBnXkFtZT...), `title` (In Old Arizona), `fullplot` (Army Sergeant Mickey Dunn sets out in pursuit of the Cisco Kid, a noto...), `languages` (Array), `released` (1929-01-20T00:00:00.000+00:00), `directors` (Array), `writers` (Array), `awards` (Object), `lastupdated` (2015-09-02 00:29:37.070000000), `year` (1928), `indb` (Object), `countries` (Array), `type` (movie), and `tomatoes` (Object).

Árabe ou Russo

`FILTER {languages:{$in:["Arabic","Russia"]}}`

# AGGREGATION FRAMEWORK

- Base para realizar consultas.
- Conjunto de ferramentas de análise que permite realizar vários tipos de consultas em documentos em uma ou mais coleções.
- Baseada no conceito de Pipeline.



# AGGREGATION FRAMEWORK - PIPELINE

- **Pipeline** (imagine uma esteira industrial).
- A entrada é uma **coleção MongoDB**, que vai sendo submetida a etapas sequenciais, cada uma com sua operação específica.
- **Cada etapa** recebe, como entrada, o resultado da saída da etapa anterior, independente do resultado que ela tiver gerado.
- E as entradas e saídas para todas as etapas são documentos, ou seja, um fluxo de documentos.

# AGGREGATION FRAMEWORK - PIPELINE

- O resultado final do pipeline é justamente o conjunto de documentos que se espera.
- É possível que as etapas se repitam no pipeline, dependendo do resultado que se espera.
- Há diversas possibilidades de operações que podem ser realizadas no pipeline MongoDB.
- A ferramenta Atlas possibilita construir pipelines de agregação.



# AGGREGATION FRAMEWORK - PIPELINE

## sample\_mflix.movies

COLLECTION SIZE: 35.88MB   TOTAL DOCUMENTS: 23530   INDEXES TOTAL SIZE: 13.4MB

Find   Indexes   Schema Anti-Patterns 0   **Aggregation**   Search Indexes •

COLLATION

23530 Documents in the Collection

Select an operator to construct expressions used in the aggregation pipeline stages. [Learn more](#)

Preview of Documents in the Collection

```
cast: Array
  title: "Blacksmith Scene"
  fullplot: "A stationary camera looks at a large an
    with a blacksmith behind it..."
directors: Array
  rated: "UNRATED"
awards: Object
runtime: 1
num_mflix_comments: 1
```

```
runtime: 11
fullplot: "Among the earliest existing films in Am
  cinema - notable as the ..."
languages: Array
directors: Array
  _id: ObjectId("573a1390f29313caabcd42e8")
plot: "A group of bandits stage a brazen train hol
  up, only to find a determ..."
cast: Array
```

Select...

☒

+


1

ADD STAGE

A sample of the aggregated results from this stage will be shown below

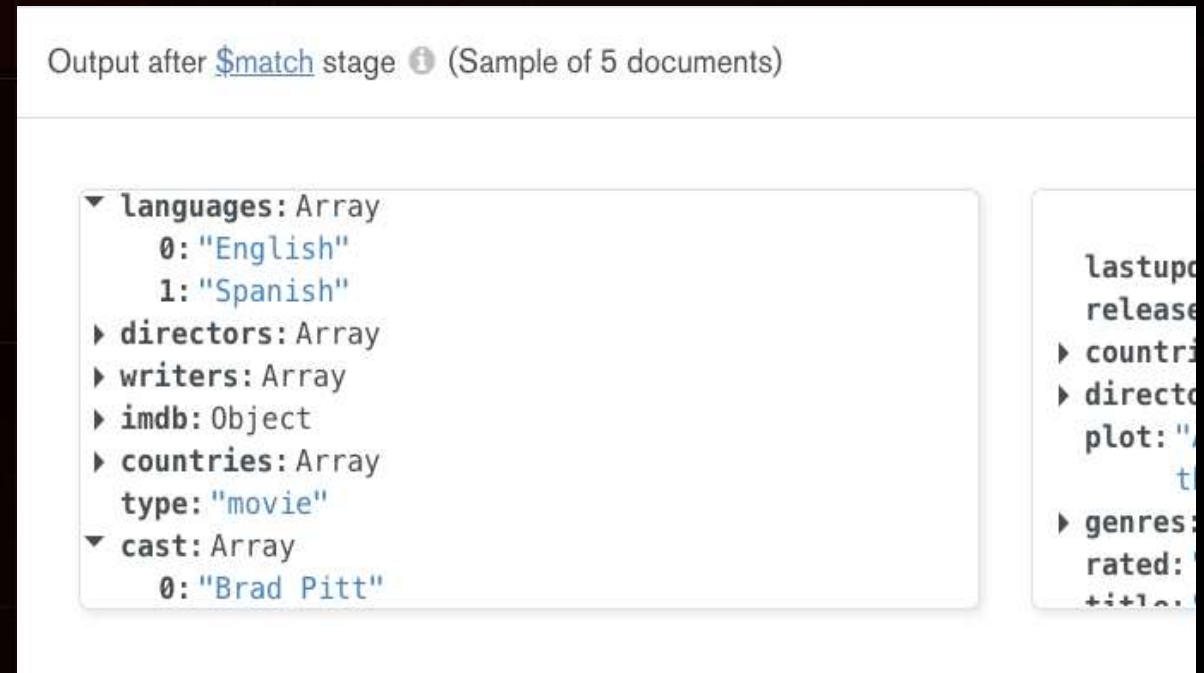
# ESTÁGIO/ETAPA - \$match

- Uma das operações mais simples que existem no MongoDB, ela aplica um filtro ao conjunto de documentos.
- Exemplo: filtro em dois campos (cast e languages)



The image shows the MongoDB Compass interface. At the top, there is a dropdown menu with '\$match' selected and a green toggle switch. Below this, a code editor displays a query in MQL format:

```
1 /**  
2  * query: The query in MQL.  
3  */  
4 {  
5   cast: "Brad Pitt",  
6   languages: "Spanish"  
7 }  
8
```



The image shows the output of the \$match stage in MongoDB Compass. The title bar reads 'Output after \$match stage (Sample of 5 documents)'. The output is displayed in a tree view format, showing the following structure:

```
▼ languages: Array  
  0: "English"  
  1: "Spanish"  
  ▶ directors: Array  
  ▶ writers: Array  
  ▶ imdb: Object  
  ▶ countries: Array  
  type: "movie"  
▼ cast: Array  
  0: "Brad Pitt"
```

On the right side, there is a partial view of another document structure, showing fields like 'lastupdate', 'release', 'country', 'director', 'plot', 'genres', and 'rated'.

# ESTÁGIO/ETAPA - \$group

- Uma etapa de um pipeline possibilita uma saída que pode ser bem diferente do documento original.
- É muito comum que, ao passar por uma etapa, o conjunto de documentos passe a ter uma nova construção estrutural (esquema) diferente do seu documento original.
- A etapa **\$group** pode aplicar vários operadores de grupo (parecido com SQL):
  - \$sum, \$avg, \$max, \$min, \$accumulator, entre outros

# ESTÁGIO/ETAPA - \$group

- Exemplo: agrupamento por idiomas (languages).
- No agrupamento é realizada a agregação de valor por minutos e também são quantificados os grupos.



```
1 /**
2  * _id: The id of the group.
3  * fieldN: The first field name.
4  */
5 {
6   _id: '$languages',
7   minutos: {$sum: '$runtime'},
8   qtde: {$sum: 1}
9 }
```

Output after `$group` stage ⓘ (Sample of 4 documents)

```
qtde: 1
_id: Array
  0: "English"
  1: "French"
  2: "Spanish"
minutos: 122
```

```
_id: Array
  0: "English"
  1: "Spanish"
minutos: 243
qtde: 2
```

```
_id: Array
  0: "E
  1: "S
  2: "H
  3: "A
minutos
qtde: 1
```

23530 Documents in the Collection

**C**

### Preview of Documents in the Collection

Select an operator to construct expressions used in the aggregation pipeline stages. [Learn more](#)

```
runtime: 1
num_mflix_comments: 1
lastupdated: "2015-08-26 00:03:50.133000000"
type: "movie"
plot: "Three men hammer on an anvil and pass a
      bottle of beer around."
countries: Array
_id: ObjectId("573a1390f29313caabcd4135")
genres: Array
released: 1893-05-09T00:00:00.000+00:00
```

```
runtime: 11
fullplot: "Among the earliest existing films in Am...
cinema - notable as the ..."
languages: Array
directors: Array
_id: ObjectId("573a1390f29313caabcd42e8")
plot: "A group of bandits stage a brazen train hol...
up, only to find a determ..."
cast: Array
```

```

▶ awards: Object
  type: "movie"
  poster: "https://m
    amazon.com/
▶ directors: Array
▶ writers: Array
  num_mflix_comments
  title: "The Land Be
  released: 1912-10-

```

|||  \$match  




Output after **Smatch** stage ⑤ (Sample of 5 documents)

```
1 //**
2  * query: The query in MQ...
3  */
4 {
5     cast:"Brad Pitt",
6     languages:"Spanish"
7 }
8 }
```

```
_id: ObjectId("573a13a2f29313caabd0c19d")
released: 2001-03-02T00:00:00.000+00:00
▶ awards: Object
year: 2001
plot: "A man tries to transport an ancient gun
      called The Mexican, believed t..."
▶ genres: Array
runtime: 123
metacritic: 43
```

```
nn and Jane Smith are a normal married
ple, living a normal life ..."
ect

S
y
t
s://m.media-
n.com/images/M/MV5BMTUxMzcwNzQzOF5BMl5BanBnXkFtZT...
```

```
("573a13b3f29313caabd
y strikes a married c
Moroccan desert, t...
"
ments:1
t
interlocking stories
gle gun all converge
```

☰ ▾ \$group ▾ ☒




Output after **\$group** stage (Sample of 4 documents)

```
1 ▾ /**
2   * _id: The id of the group.
3   * fieldN: The first field name.
4   */
5 ▾ {
6   _id: '$languages',
7   minutos: {$sum: '$runtime'},
8   qtde: {$sum: 1}
9 }
```

```
qtde: 1
_id: Array
  0: "English"
  1: "French"
  2: "Spanish"
minutos: 122
```

```

    _id: Array
      0: "English"
      1: "Spanish"
    minutos: 243
    qtde: 2
  
```

```

    _id: Array
      0: "English"
      1: "Spanish"
      2: "Hebrew"
      3: "Arabic"
    minutos: 116
    qtde: 1
  
```

ADD STAGE

|||  \$match  ☒



Output after [\\$match](#) stage ⓘ (Sample of 20 documents)

```
1 ▾ /**
2   * query: The query in MQL.
3   */
4 ▾ {
5
6   languages: "Russian"
7
8 }
```

```
▸ writers: Array
  lastupdated: "2015-08-18 00:22:19.750000000"
  _id: ObjectId("573a1391f29313caabcd7e30")
  rated: "UNRATED"
▸ cast: Array
  poster: "https://m.media-
    amazon.com/images/M/MV5BMTEyMTQzMjQ0MTJeQT.
▸ languages: Array
  _id: ObjectId("573a1391f29313caabcd7e30")
```

```
▸ countries: Array
  type: "movie"
  plot: "In 1918 a simple Mongolian herdsman escapes
    to the hills after brawlin..."
▸ cast: Array
  fullplot: "In 1918 a simple Mongolian herdsman esc
    to the hills after brawlin..."
▸ languages: Array
  _id: ObjectId("573a1391f29313caabcd7e30")
```

```
▸ languages: Array
  lastupdated: "2015-07-20 00:26:02.200000000"
▸ awards: Object
▸ countries: Array
  _id: ObjectId("573a1391f29313caabcd91d7")
▸ genres: Array
  rated: "PASSED"
▸ cast: Array
  _id: ObjectId("573a1391f29313caabcd91d7")
```

|||  \$group  ☒



Output after [\\$group](#) stage ⓘ (Sample of 20 documents)

```
1 ▾ /**
2   * _id: The id of the group.
3   * fieldN: The first field name.
4   */
5 ▾ {
6   _id: '$year',
7   media_minutos: {$avg: '$runtime'},
8   qtde_filmes: {$sum: 1}
9 }
```

```
qtde_filmes: 6
_id: 1960
media_minutos: 90.66666666666667
```

```
_id: 1979
media_minutos: 135.2
qtde_filmes: 10
```

```
_id: 1930
media_minutos: 87
qtde_filmes: 1
```

|||  \$sort  ☒



Output after [\\$sort](#) stage ⓘ (Sample of 20 documents)

```
1 ▾ /**
2   * Provide any number of field/order pairs.
3   */
4 ▾ {
5   _id: 1
6 }
```

```
_id: 1925
media_minutos: 66
qtde_filmes: 1
```

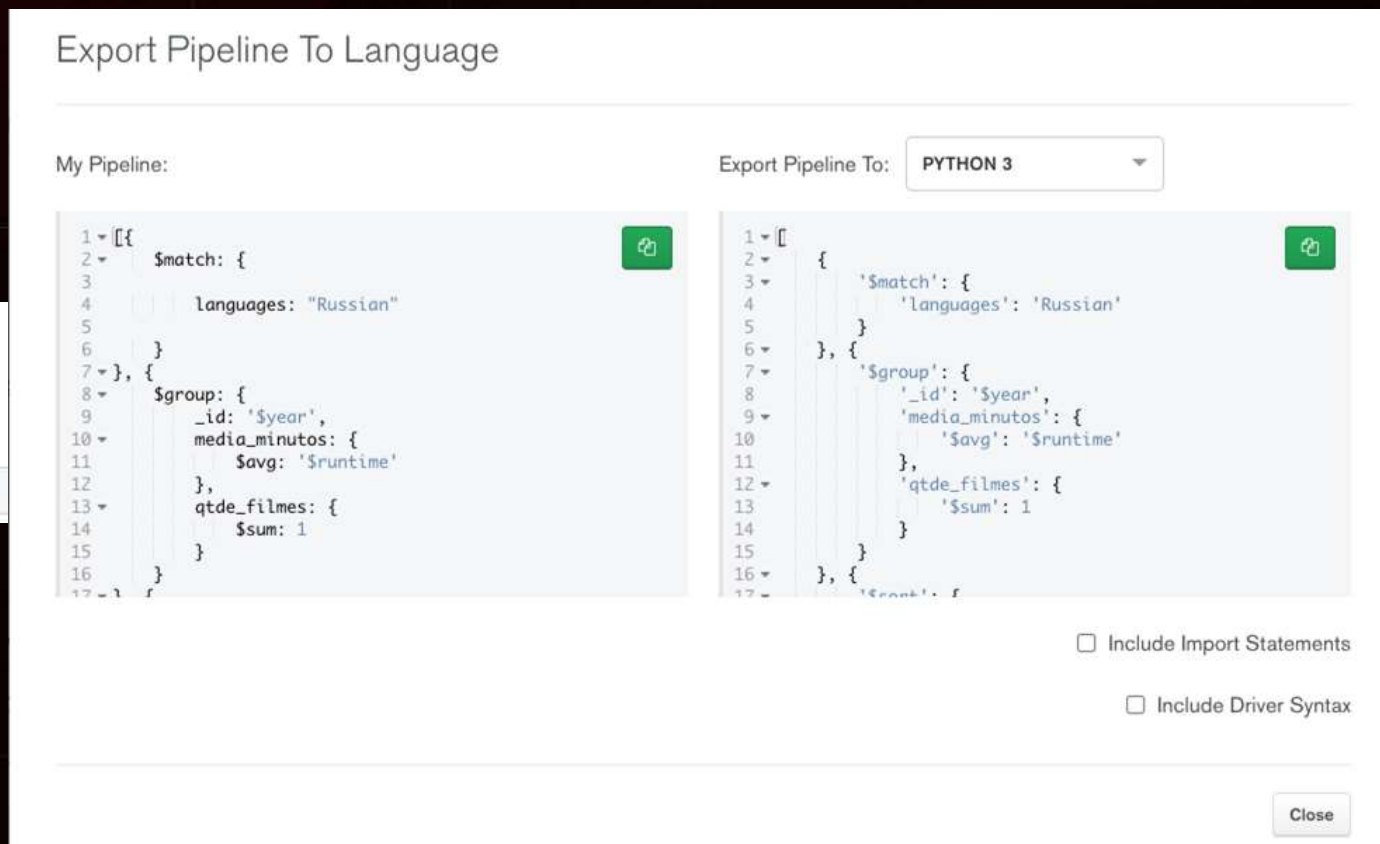
```
_id: 1928
media_minutos: 74
qtde_filmes: 1
```

```
_id: 1930
media_minutos: 87
qtde_filmes: 1
```



# ATLAS – EXPORTAÇÃO PIPELINE

- O Atlas permite que a estrutura do Pipeline seja exportada como código do próprio mongo ou já adaptado para algumas linguagens.



# NOVA BASE POR PROJEÇÃO

## sample\_mflix.movies

COLLECTION SIZE: 35.88MB   TOTAL DOCUMENTS: 23530   INDEXES TOTAL SIZE: 13.4MB

Find   Indexes   Schema Anti-Patterns 0   **Aggregation**   Search Indexes ●

>

+

▼

COLLATION

🔗

>

📄

23530 Documents in the Collection

🔄

⋮

>

\$limit

▼

🟢

🗑️

+

⋮

>

\$project

▼

🟢

🗑️

+

⋮

>

\$out

▼

🟢

🗑️

+

ADD STAGE

Preview of Documents in the Collection

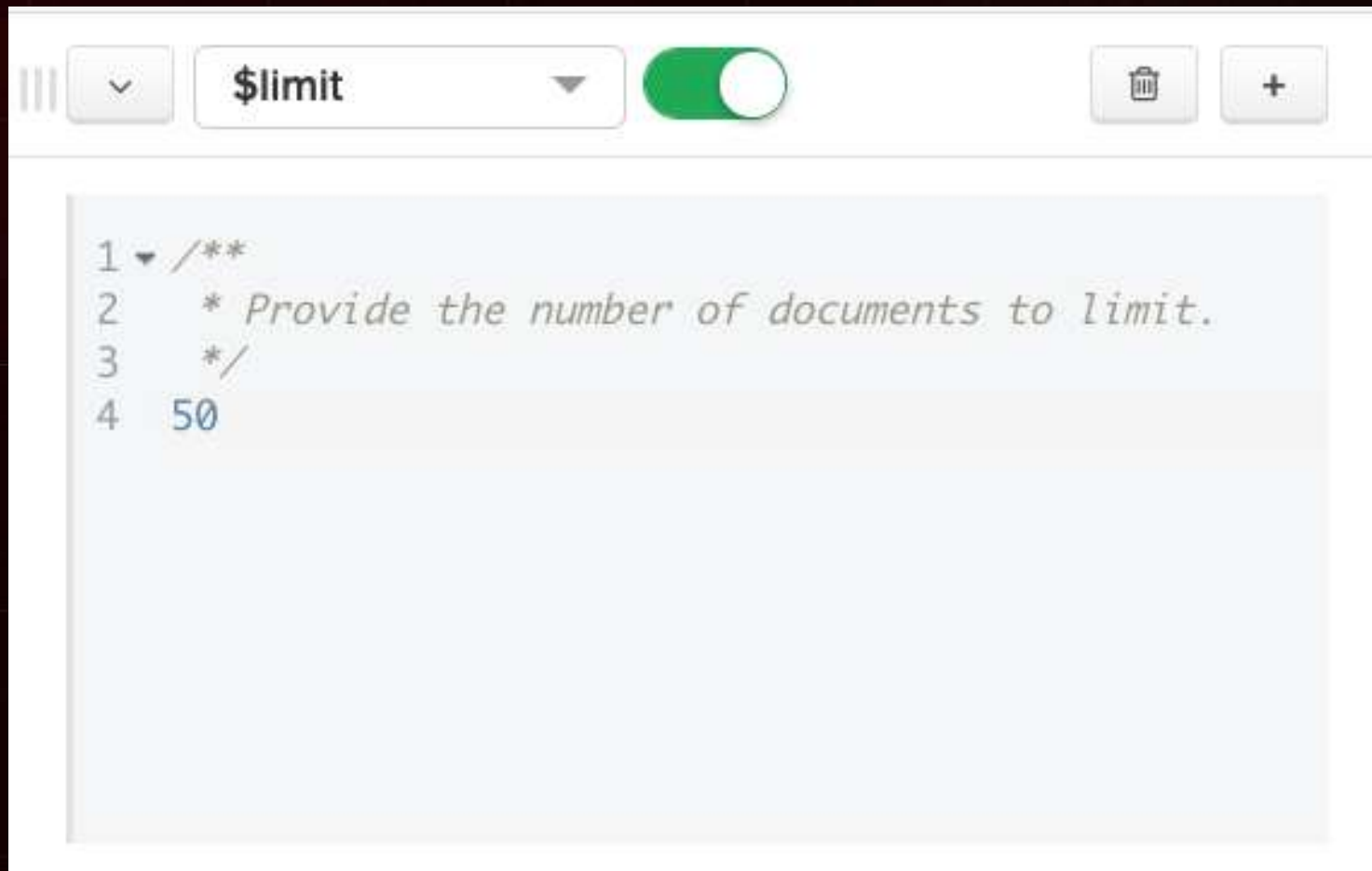
Output after [\\$limit](#) stage ⓘ (Sample of 20 documents)

Output after [\\$project](#) stage ⓘ (Sample of 20 documents)

Documents will be saved to the collection: "filmes"



# NOVA BASE POR PROJEÇÃO



# NOVA BASE POR PROJEÇÃO

☰

▼

\$project

▼

🗑️

+

1 ▾ {

2    titulo: "\$title",

3    title: {\$split:["\$title"," "]},

4    year: 1,

5    actors: "\$cast",

6    plot: 1,

7    fullPlot: '\$fullplot',

8    rated: '\$rating',

9    released: 1,

10    runtime: 1,

11    poster: 1,

12    metacritic: 1,

13    awards: 1,

14    type: 1,

15    lastUpdated: '\$lastupdated'

16 }

Output after \$project stage ⓘ (Sample of 20 documents)

\_id: ObjectId("573a1390f29313caabcd4135")

▶ awards: Object

year: 1893

titulo: "Blacksmith Scene"

▾ title: Array

0: "Blacksmith"

1: "Scene"

▶ actors: Array

plot: "Three men hammer on an anvil and pass a bottle of beer around."

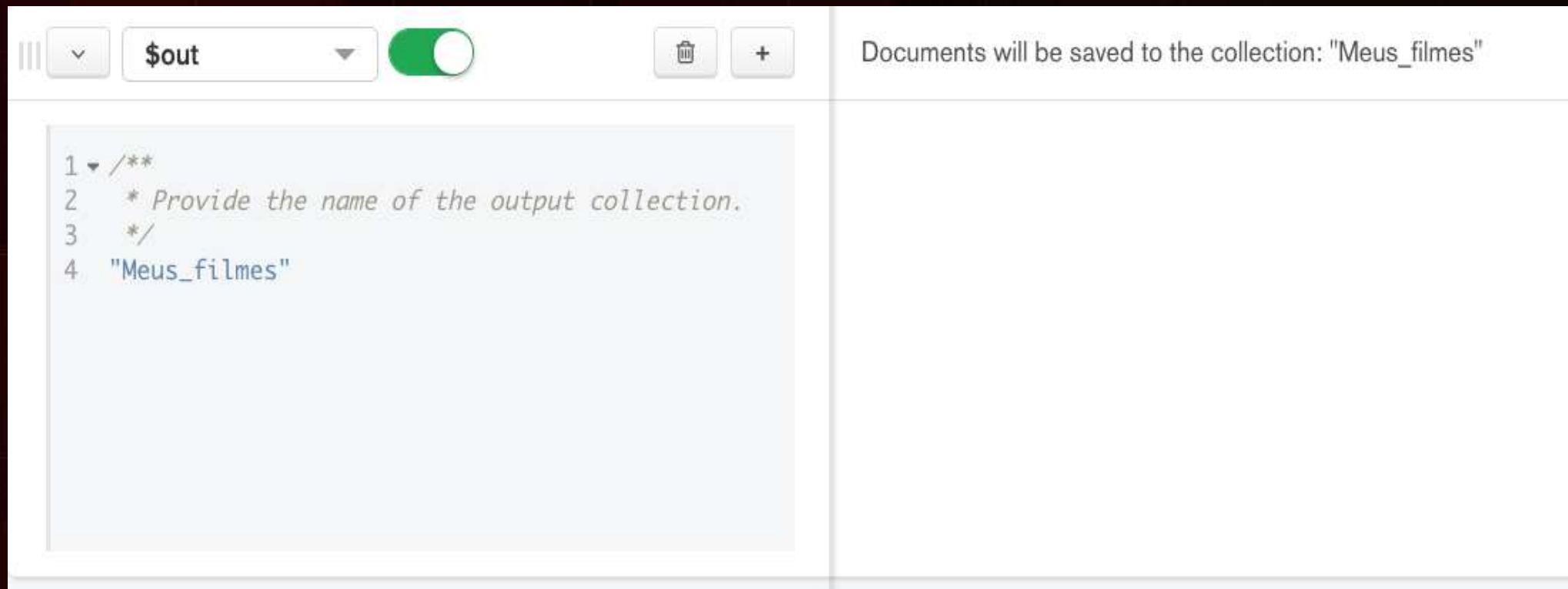
runtime: 1

released: 1893-05-09T00:00:00.000+00:00

type: "movie"

lastUpdated: "2015-08-26 00:03:50.133000000"

# NOVA BASE POR PROJEÇÃO



# **BANCOS NoSQL e MongoDB**

- **A ideia de apresentar o MongoDB e uma pequena parte de suas funcionalidades é demonstrar que os bancos NoSQL podem ter paradigmas completamente diferentes do que é atualmente o tradicional.**
- **Outros produtos terão características específicas e também demandarão muito estudo para compreensão de funcionamento.**
- **Deixem a mente aberta para aprender!**

# **BANCO DE DADOS**

## **MongoDB – Parte 2**