

**Lab. Basis Data**  
**Modul 6**

**Anggota Kelompok :**

Mahesa Alghifari (212310006)

Angga Parulian Harahap (212310043)

Muhammad Alfani (212310017)

Hera Dwi Pradita (212310054)

Frila Cahyani (212310014)

## Lab. Basis Data | Modul 6

1. Buatlah dan jalankan query untuk membuat setiap jenis tipe index pada database rentalfilm yang berbeda dengan contoh pada tahapan kerja.

### Unique

```
38  unique
39  db.customer.createIndex({"age":1}, {unique:true})
40
```

1 age\_1

### Sparse

```
JS use('film'); d:\Semester 6\Lab. Pemrograman Web Lanjut\Tugas\Pertemuan 5\todolist_212310006\playground-2.m ... Playground Result X
1 use('film');
2
3 db.address.createIndex({"CityID" : 1}, {sparse : true})
```

1 CityID\_1

### expireAfterSeconds

```
JS use('film'); d:\Semester 6\Lab. Pemrograman Web Lanjut\Tugas\Pertemuan 5\todolist_212310006\playground-2.mongodb.js ... Playground Result X
1 use('film');
2
3 db.address.createIndex({"last_update" : 1}, {expireAfterSeconds : 30})
```

1 last\_update\_1

### V

```
JS use('film'); d:\Semester 6\Lab. Pemrograman Web Lanjut\Tugas\Pertemuan 5\todolist_2 Playground Result X
1 use('film');
2
3 db.address.createIndex({ Postal_Code: 1 }, { v: 2 });
4
```

1 Postal\_Code\_1

### Weights

```
Currently connected to localhost:27017. Click here to change connection.
use('film');

db.address.createIndex(
  { Address: "text", District: "text" },
  { weights: { Address: 2, District: 1 } }
);
```

1 Address\_text\_District\_text

## Lab. Basis Data | Modul 6

### default\_language

```
Currently connected to localhost:27017. Click here to change connection.
use('film');

db.address.createIndex(
  { Name: "text" },
  { default_language: "english", weights: { Name: 1 },
    language_override: "language" }
);
```

1	Name_text
---	-----------

### Language\_override

```
JS use('film'); d:\Semester 6\Lab. Pemrograman Web Lanjut\Tugas\Pertemuan 5\todolist_2
1 use('film');
2
3 db.address.createIndex({ Name: "text" }, {
4   language_override: "language" });
```

1	Name_text
---	-----------

## Lab. Basis Data | Modul 6

2. Buatlah dan jalankan setiap query aggregate pada database rentalfilm yang berbeda dengan contoh yang telah diberikan.

### \$Sum

```
58
59 // AGGREGATION
60 // $sum
61 db.customer.aggregate([ {$group: { _id: null, age: { $sum: '$age' } } } ] )
62
63 // $avg
64 // db.customer.aggregate([ {$group: { _id: null, age: { $avg: '$age' } } } ] )
```

```
1  [
2    {
3      "_id": null,
4      "age": 64
5    }
6  ]
```

### \$Avg

```
61 // db.customer.aggregate([ {$group: { _id: null, age: { $sum: '$age' } } } ] )
62
63 // $avg
64 db.customer.aggregate([ {$group: { _id: null, age: { $avg: '$age' } } } ] );
65
66 // Min
67 // db.customer.aggregate([ {$group: { _id: null, Total_Replacement_Cost: { $min: 'Total_Replacement_Cost' } } } ] );
```

```
1  [
2    {
3      "_id": null,
4      "age": 21.333333333333332
5    }
6  ]
```

### \$Min

```
65 // db.customer.aggregate([ {$group: { _id: null, age: { $sum: '$age' } } } ] );
66 // Min
67 db.customer.aggregate([ {$group: { _id: null, age: { $min: '$age' } } } ] );
68
69 // Max
```

```
1  [
2    {
3      "_id": null,
4      "age": 14
5    }
6  ]
```

### \$Max

```
68
69 // Max
70 db.customer.aggregate([ {$group: { _id: null, age: { $max: '$age' } } } ] );
71
72 // Push
```

```
1  [
2    {
3      "_id": null,
4      "age": 30
5    }
6  ]
```

### \$Push

```
70 // db.customer.aggregate([ {$group: { _id: null, age: { $max: '$age' } } } ] );
71
72 // Push
73 db.actor.updateOne({ID : 1}, {$push : {Usia: 100}})
74
75 // AddToSet
76 // db.actor.updateOne({ID : 1}, {$addToSet : {Usia: 99}})
77
```

```
1  Click here to ask Blackbox to help you
2  {
3    "acknowledged": true,
4    "insertedId": null,
5    "matchedCount": 1,
6    "modifiedCount": 1,
7    "upsertedCount": 0
8  }
```

## Lab. Basis Data | Modul 6

### \$addToSet

72 // Push	
73 // db.actor.updateOne({ID : 1}, {\$push : {Usia: 100}})	
74	
75  AddToSet	
76 db.actor.updateOne({ID : 1}, {\$addToSet : {Usia: 99}})	1 {
77	2
78 // First	3 "acknowledged": true,
79 // db.customer.aggregate({\$group: {_id: null, first: {\$first: "\$\$ROOT"}}})	4 "insertedId": null,
	5 "matchedCount": 1,
	6 "modifiedCount": 0,
	7 "upsertedCount": 0
	}

### \$first

Click to add a breakpoint	
73 // db.actor.updateOne({ID : 1}, {\$push : {Usia: 100}})	1  Click here to ask Blackbox to help you code faster
74	2 {
75 // AddToSet	3
76 // db.actor.updateOne({ID : 1}, {\$addToSet : {Usia: 99}})	4 " _id": null,
77	5 "first": {
78  First	6 " _id": {
79 db.customer.aggregate({\$group: {_id: null, first: {\$first: "\$\$ROOT"}}})	7 "\$oid": "660b8d38cd1d045c518bf202"
80	8 },
81 // Last	9 "name": "John",
82 // db.customer.aggregate({\$group: {_id: null, last: {\$last: "\$\$ROOT"}}})	10 "age": 30,
83	11 "status": "active",
84 // \$Unwind	12 "joinedDate": {
85 // db.customer.aggregate([	13 "date": "2024-04-02T04:44:40.262Z"
86 // {	14 }
87 // \$unwind: {	15 }
88 // path: '\$age'	16 }

### \$last

76 // db.actor.updateOne({ID : 1}, {\$addToSet : {Usia: 99}})	1  Click here to ask Blackbox to help you code faster
77	2 {
78 // First	3
79 // db.customer.aggregate({\$group: {_id: null, first: {\$first: "\$\$ROOT"}}})	4 " _id": null,
80	5 "last": {
81  Last	6 " _id": {
82 db.customer.aggregate({\$group: {_id: null, last: {\$last: "\$\$ROOT"}}})	7 "\$oid": "66273b71c51e6fcd834c1253"
83	8 },
84 // \$Unwind	9 "id": "1",
85 // db.customer.aggregate([	10 "name": "John Doe",
86 // {	11 "age": 14,
87 // \$unwind: {	12 "status": "active"
88 // path: '\$age'	13 }
89 // }	14 }

## Lab. Basis Data | Modul 6

3. Buatlah dan jalankan setiap query pipeline pada database rentalfilm yang berbeda dengan contoh yang telah diberikan.

### \$Project

```
74 // AddToSet
75 // db.actor.updateOne({ID : 1}, {$addToSet : {Usia: 99}})
76 // First
77 // db.customer.aggregate({$group: {_id: null, first: {$first: "$$ROOT"}}})
78 // Last
79 // db.customer.aggregate({$group: {_id: null, last: {$last: "$$ROOT"}}})
80 // PIPELINE
81 // $Project
82 db.customer.aggregate([
83   {
84     $project: {
85       "joinedDate": 0
86     }
87   }
88 ])
89 // $Unwind
90 // db.customer.aggregate([
91 //   {
92 //     $unwind: {
93 //       path: '$age'
94 //     }
95 //   }
96 // ]);
97
98
99
100
101
102
103
104
105
```

```
1 Click here to ask blackbox to help you code faster
2 [
3   {
4     "_id": {
5       "$oid": "660b8d38cd1d045c518bf202"
6     },
7     "name": "John",
8     "age": 30,
9     "status": "active"
10  },
11  {
12    "_id": {
13      "$oid": "660b8d38cd1d045c518bf204"
14    },
15    "name": "Bob",
16    "age": 20,
17    "status": "active"
18  },
19  {
20    "_id": {
21      "$oid": "660b8d38cd1d045c518bf205"
22    },
23    "name": "Eve",
24    "age": null,
25    "status": "inactive"
26  },
27  {
28    "_id": {
29      "$oid": "66273b71c51e6fcd834c1253"
30    },
31    "id": "1",
32    "name": "John Doe",
33    "age": 14,
34    "status": "active"
35  }
36 ]
```

### \$Match

```
90 // "joinedDate": 0
91 // }
92 // }
93 // ])
94 // $Match
95 db.customer.aggregate([
96   {
97     $match: {
98       name: "John"
99     }
100   }
101 ])
102
103
104
105
```

```
1 Click here to ask Blackbox to help you code faster
2 [
3   {
4     "_id": {
5       "$oid": "660b8d38cd1d045c518bf202"
6     },
7     "name": "John",
8     "age": 30,
9     "status": "active",
10    "joinedDate": {
11      "$date": "2024-04-02T04:44:40.262Z"
12    }
13  }
14 ]
```

## Lab. Basis Data | Modul 6

### \$Group

```
103
104 // $Group
105 db.customer.aggregate([
106   {
107     $group: {
108       age: 20
109     }
110   }
111 ]);
112 [ {age: 20}, {age:14}]
113
```

```
1 Click here to ask Blackbox
2 [
3   {
4     "age": 20
5   },
6   {
7     "age": 14
8   }
9 ]
```

### \$Sort

```
111 // ];
112 // [ {age: 20}, {age:14}]
113
114 // $Sort
115 db.customers.aggregate([
116   {
117     $sort: {
118       fullname: 1
119     }
120   }
121 ]);
122
```

```
1 Click here to ask Blackbox to help you code faster
2 [
3   {
4     "_id": {
5       "$oid": "660b82451723992d618bf202"
6     },
7     "id": "Astrid",
8     "fullname": "Astrid Tiar ",
9     "email": "astrid.tiar@ibik.ac.id"
10  }
11 ]
```

### \$Limit

```
113
114 // $Sort
115 // db.customers.aggregate([
116 //   {
117 //     $sort: {
118 //       fullname: 1
119 //     }
120 //   }
121 // ]);
122
123 // $Limit
124 db.customer.aggregate([
125   {
126     $limit: 2
127   }
128 ]);
129
130 // $Unwind
131 db.customer.aggregate([
132   {
133     $unwind: {
134       path: '$age'
135     }
136   }
137 ]);
```

```
1 Click here to ask Blackbox to help you code faster
2 [
3   {
4     "_id": {
5       "$oid": "660b8d38cd1d045c518bf202"
6     },
7     "name": "John",
8     "age": 30,
9     "status": "active",
10    "joinedDate": {
11      "$date": "2024-04-02T04:44:40.262Z"
12    }
13  },
14  {
15    "_id": {
16      "$oid": "660b8d38cd1d045c518bf204"
17    },
18    "name": "Bob",
19    "age": 20,
20    "status": "active",
21    "joinedDate": "2022-01-01"
22  }
23 ]
```

## Lab. Basis Data | Modul 6

4. Buatlah dan jalankan minimal 3 buah gabungan query pipeline pada database rentalfilm.

Mengambil semua aktor yang memiliki nama belakang Hanks :

```
JS use('film'); d:\Semester 6\Lab. Pemrograman Web Lanjut\Tugas\Pertemuan 6\... Playground Result X
1 Currently connected to localhost:27017. Click here to change connector
2 use('film');
3 db.actor.aggregate([
4   { $match: { Last_Name: "Hanks" } }
5 ])
6
1 [
2   {
3     "_id": {
4       "$oid": "6627c5b3546d59e513b48122"
5     },
6     "ID": 1,
7     "First_Name": "Tom",
8     "Last_Name": "Hanks",
9     "Last_Update": {
10      "$date": "2024-04-23T14:29:07.523Z"
11    }
12  },
13  {
14    "_id": {
15      "$oid": "6627d07e64d62b953ce7311d"
16    },
17    "ID": 2,
18    "First_Name": "Parulian",
19    "Last_Name": "Hanks",
20    "Last_Update": {
21      "$date": "2024-04-23T15:15:10.461Z"
22    }
23  }
24 ]
```

Mengambil lima film dengan rating tertinggi:

```
JS use('film'); d:\Semester 6\Lab. Pemrograman Web Lanjut\Tugas\Pertemuan 6\... Playground Result X
1 Currently connected to localhost:27017. Click here to change connector
2 use('film');
3 db.film.aggregate([
4   { $sort: { Rating: -1 } },
5   { $limit: 5 }
6 ])
7
1 [
2   {
3     "_id": {
4       "$oid": "6627d07e64d62b953ce73120"
5     },
6     "ID": 1,
7     "Title": "Forrest Gump",
8     "Release_Year": 1994,
9     "Rating": 5,
10    "Last_Update": {
11      "$date": "2024-04-23T15:15:10.509Z"
12    }
13  },
14  {
15    "_id": {
16      "$oid": "6627c0e3294e52508f8e730e"
17    },
18    "Language_Id": 1,
19    "Title": "Gundala",
20    "Description": "Film tentang pahlawan super Indonesia",
21    "Release_Year": 2019,
22    "Rental_Duration": 7,
23    "Rental_Rate": {
24      "$numberDecimal": "10.99"
25    },
26    "Length": 120,
27    "Replacement_Cost": {
28      "$numberDecimal": "20.99"
29    },
30    "Rating": 5,
31    "Last_Update": {
32      "$date": "2024-04-23T14:08:35.157Z"
33    },
34    "Special_Features": "Behind the scenes footage",
35    "Fulltext": "Gundala Film Indonesia"
36  },
37  {
38    "_id": {
39      "$oid": "6627c0e3294e52508f8e730e"
40    },
41    "Language_Id": 1,
42    "Title": "Gundala",
43    "Description": "Film tentang pahlawan super Indonesia",
44    "Release_Year": 2019,
45    "Rental_Duration": 7,
46    "Rental_Rate": {
47      "$numberDecimal": "10.99"
48    },
49    "Length": 120,
50    "Replacement_Cost": {
51      "$numberDecimal": "20.99"
52    },
53    "Rating": 5,
54    "Last_Update": {
55      "$date": "2024-04-23T14:08:35.157Z"
56    },
57    "Special_Features": "Behind the scenes footage",
58    "Fulltext": "Gundala Film Indonesia"
59  },
60  {
61    "_id": {
62      "$oid": "6627c0e3294e52508f8e730e"
63    },
64    "Language_Id": 1,
65    "Title": "Gundala",
66    "Description": "Film tentang pahlawan super Indonesia",
67    "Release_Year": 2019,
68    "Rental_Duration": 7,
69    "Rental_Rate": {
70      "$numberDecimal": "10.99"
71    },
72    "Length": 120,
73    "Replacement_Cost": {
74      "$numberDecimal": "20.99"
75    },
76    "Rating": 5,
77    "Last_Update": {
78      "$date": "2024-04-23T14:08:35.157Z"
79    },
80    "Special_Features": "Behind the scenes footage",
81    "Fulltext": "Gundala Film Indonesia"
82  },
83  {
84    "_id": {
85      "$oid": "6627c0e3294e52508f8e730e"
86    },
87    "Language_Id": 1,
88    "Title": "Gundala",
89    "Description": "Film tentang pahlawan super Indonesia",
90    "Release_Year": 2019,
91    "Rental_Duration": 7,
92    "Rental_Rate": {
93      "$numberDecimal": "10.99"
94    },
95    "Length": 120,
96    "Replacement_Cost": {
97      "$numberDecimal": "20.99"
98    },
99    "Rating": 5,
100   "Last_Update": {
101     "$date": "2024-04-23T14:08:35.157Z"
102   },
103   "Special_Features": "Behind the scenes footage",
104   "Fulltext": "Gundala Film Indonesia"
105 }
106 ]
```



## Lab. Basis Data | Modul 6

Menghitung jumlah film yang dirilis pada setiap tahun:

```
use('film');
db.film.aggregate([
  { $group: { _id: "$Release_Year", count: { $sum: 1 } } }
])
```

```
{
  "_id": 2010,
  "count": 1
},
{
  "_id": 2017,
  "count": 1
},
{
  "_id": null,
  "count": 3
},
{
  "_id": 2008,
  "count": 1
},
{
  "_id": 1999,
  "count": 1
},
{
  "_id": 2002,
  "count": 1
},
{
  "_id": 1994,
  "count": 1
},
{
  "_id": 2019,
  "count": 1
},
{
  "_id": 2011,
  "count": 1
}
```

Menggabungkan data aktor dan film yang mereka bintangi:

```
use('film');
db.aktor.aggregate([
  { $lookup:
    {
      from: "actor_film",
      localField: "ID",
      foreignField: "Actor_ID",
      as: "films"
    }
  })
])
```

```
{
  "_id": {
    "oid": "6627c1c33988b3403c04e8f2"
  },
  "type": "header",
  "version": "5.2.1",
  "comment": "Export to JSON plugin for PHPMyAdmin",
  "films": []
},
{
  "_id": {
    "oid": "6627c1c33988b3403c04e8f3"
  },
  "type": "database",
  "name": "film",
  "films": []
},
{
  "_id": {
    "oid": "6627c5b3546d59e513b48122"
  },
  "ID": 1,
  "First_Name": "Tom",
  "Last_Name": "Hanks",
  "last_update": {
    "$date": "2024-04-23T14:29:07.523Z"
  },
  "films": [
    {
      "_id": {
        "oid": "6627d07e64d62b953ce73123"
      },
      "Actor_ID": 1,
      "film_ID": 1
    },
    {
      "_id": {
```

Menghitung rata-rata panjang film:

```
use('film');
db.film.aggregate([
  { $group: { _id: null, averageLength: { $avg: "$Length" } } }
])
```

```
{
  "_id": null,
  "averageLength": 118
}
```