

PROJECTION OPERATORS :

Projection operators in MongoDB are used to specify or restrict the fields that should be included in the documents returned by a query. By default, MongoDB returns all fields in the matching documents, but projection operators allow you to control which fields are included or excluded.

- Create new collection called candidates.
- Upload the dataset [link](#)

```
_id: ObjectId('665752830959f4120ac93d06')
name : "Emily Jones"
age : 21
▶ courses : Array (3)
gpa : 3.6
home_city : "Houston"
blood_group : "AB-"
is_hotel_resident : false
```

Basic Projection

When you perform a query in MongoDB, you can include a projection to specify the fields you want to include in the result set. The projection is specified in the second parameter of the find method.

*Include fields:specify the fields to exclude with a value 0.

```
db.collection.find({}, { name: 1, age:1})
```

*Exclude fields: specify the fields to exclude with a value of 0.

```
db.collection.find({}, {password:0})
```

PROJECTION OPERATION:

\$slice

\$elemMatch

\$filter.

Example 1: Retrieve Name, Age, and GPA

JavaScript

```
db.candidates.find({}, { name: 1, age: 1, gpa: 1 });
```

Example 2: Find Candidates Enrolled in "Computer Science" with Specific Projection

JavaScript

```
db.candidates.find({ courses: { $elemMatch: { $eq: "Computer Science" }  
  { name: 1, "courses.$": 1 } }); // Include only matched course
```

Here, we use operations like \$elemMatch.

OUTPUT:

```
test> db.candidate.find({courses:{$elemMatch:{$eq:"Computer Science"}}},{name:1,"courses.$":1});
[
  {
    _id: ObjectId('6682d28ec38b2a00d2064ae3'),
    name: 'Bob Johnson',
    courses: [ 'Computer Science' ]
  },
  {
    _id: ObjectId('6682d28ec38b2a00d2064ae8'),
    name: 'Gabriel Miller',
    courses: [ 'Computer Science' ]
  },
  {
    _id: ObjectId('6682d28ec38b2a00d2064aec'),
    name: 'Kevin Lewis',
    courses: [ 'Computer Science' ]
  }
]
```

\$SelemMatch:

```
test> db.players.insertOne( {
...   name: "player1",
...   games: [ { game: "abc", score: 8 }, { game: "xyz", score: 5 } ],
...   joined: new Date("2020-01-01"),
...   lastLogin: new Date("2020-05-01")
... } )
{
  acknowledged: true,
  insertedId: ObjectId('6682dbaeec4204f7fecdcdf6')
}
```

```
test> db.players.find({}, {games:{$elemMatch:{score:{$gt:5}}},joined:1,lastLogin:1})
;
[
  {
    _id: ObjectId('6682dbaeec4204f7fecdcdf6'),
    joined: ISODate('2020-01-01T00:00:00.000Z'),
    games: [ { game: 'abc', score: 8 } ]
  }
]
```

