We are working with a MongoDB collection called publishers where each document has an array field called books that contains embedded documents with data about the author and the year the book was published. The documents have the following structure:

```
{
  title: "The Lord Of The Rings: The Fellowship of The Ring"
  author: "J.R.R. Tolkien"
  year: 1954
  popular: true
}
```

Fill in the MongoDB command below so that it correctly queries the collection for all documents where the books array contains at least one document where the author is "Stephen King" and the published year is 2000 or greater.

We have a MongoDB database that stores data about outdoor gear that our e-commerce store sells. Inside the database, we have a collection named customers where each document has a shopping_cart field that stores product names that customers put into their cart. Here is an example document:

```
{
  customer_id: 3242,
  first_name: "Leonardo",
  shopping_cart: ["tent", "backpack", "kettle"]
}
```

Which of the following documents would NOT be matched by this query on our customers collection?

```
db.customers.find({ shopping_cart: { $all: [ "tent", "backpack", "water bottle" ] } });
```

```
{
    _id: ObjectId(...),
    customer_id: 2614,
    cart: ["tent", "water bottle", "cooler", "rain jacket"]
}
```



That's right! This document wouldn't be matched because it doesn't contain all the elements mentioned in our query.

We are working with a MongoDB database that has a collection named trucks. Each truck document has an array field named cargo with an embedded document with information about the items it is transporing. Each document follows a similar structure:

```
{
  truck_id: 342349,
  driver: "Abdel Jen",
  cargo: [{
    item_name: "Milk",
    perishable: true
  },
  {
    item_name: "Coffee Beans",
    perishable: false
  }]
}
```

Which of the following MongoDB queries would return all trucks that are carrying a perishable (where perishable is true) item?

```
db.trucks.find({ "cargo.perishable": true })
```



Correct! This would find all the truck documents that have an embedded document inside of the cargo field that is perishable.

Consider a MongoDB collection called movies where each document has an array field called cast. Here is an example document:

```
{
  name: "Star Wars"
  director: "George Lucas"
  cast: [ "Mark Hamill", "Harrison Ford", "Carrie Fisher"]
}
```

What command should we run if we want to search the collection for a movie featuring "Lupita Nyong'o" as a cast member?

```
db.movies.find( cast, "Lupita Nyong'o" )

db.movies.find({ }, { cast: "Lupita Nyong'o" })

db.movies.find({ cast: ["Lupita Nyong'o"] })

db.movies.find({ cast: "Lupita Nyong'o" })
```



Well done! This query would search the movies collection for the field cast with the value of "Lupita Nyong'o".

We are working with a MongoDB database that has a collection named students. Each student document has an array field named classes with an embedded document with information about the current classes the student is in. Each document follows a similar structure:

```
{
    student_id: 342349,
    name: "Mia Saf",
    classes: {
        name: "Chemistry 101",
        section: "4A"
    },
    {
        name: "Biology 101",
        section: "6C"
    }]
}
```

Which of the following MongoDB queries would return all students who are in a class named "Existential Philosophy" in section "28".

```
db.students( { classes: { name: "Existential Philosophy", section: "28"} } )

db.find( { classes: { name: "Existential Philosophy", section: "28" } } )

db.students.find( { classes: { section: "28", name: "Existential Philosophy"} } )

db.students.find( { classes: { name: "Existential Philosophy", section: "28" } } )

Excellent, that is correct! This query would find all the embedded documents in the classes field of the
```

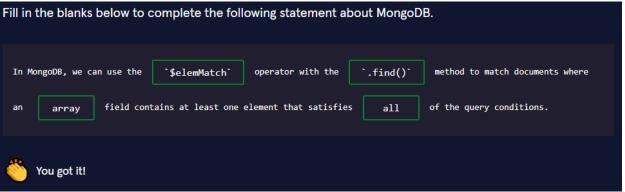
We are working with a MongoDB database and have a collection called trains. Which of the following commands would we run if we wanted to query this collection on an array field called transfer_stations with these exact values in this specific order: "Union Square", "Chambers Street", and "Fulton Street"?

students collection with a field name and section that has the exact values and in the exact ... Show more

```
db.trains.find({ transfer_stations: ["Union Square", "Chambers Street", "Fulton Street"] })
```



Excellent, that is correct! This query would find all the records in the trains collection with a field transfer_stations that has the exact values in the exact order we specified.



```
We are working with a MongoDB collection called nba_teams. The documents inside the collection look like
this:
     _id: ObjectId(...),
team: "Brooklyn Nets",
championship_wins: [1974, 1976]
Which of the following documents would be matched by this query?
  db.nba_teams.find({ championship_wins: { $gt: 1990, $1t: 2000 } })
        _id: ObjectId(...),
team: "New York Knicks",
championship_wins: [1970, 1973]
        _id: ObjectId(...),
team: "Miami Heat",
championship_wins: [2006, 2012, 2013]
        _id: ObjectId(...),
team: "Philadelphia 76ers",
championship_wins: [1955, 1967, 1983]
        _id: ObjectId(...),
team: "Milwaukee Bucks",
championship_wins: [1971, 2021]
```