The background of the slide shows a laptop screen displaying a dark-themed dashboard. The dashboard features a line graph at the top showing a steady upward trend with some fluctuations, and a pie chart below it divided into two main segments. The overall aesthetic is professional and modern.

Accelerated Spring Boot 2 & MongoDB

#01 Introduction

Complete Spring Boot 2 + MongoDB Course

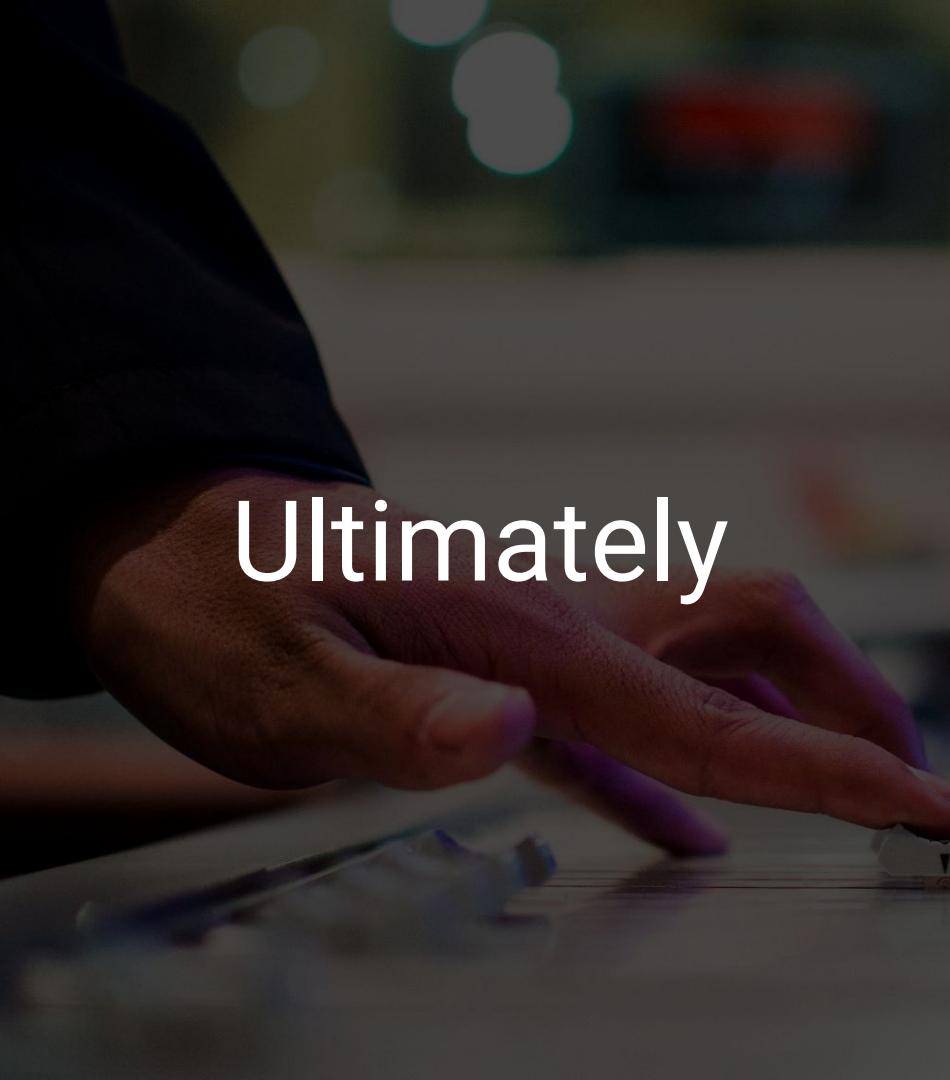
Understand the Spring Data MongoDB Annotations

Implement MongoDB CRUD functionality using Java and Spring Data

Create efficient MongoDB queries using the built in MongoTemplate, custom Mongo repositories or Query DSL

Learn how to implement incremental data migrations in Java using Mongobee

Create MongoDB integration tests in Spring Boot applications

A close-up photograph of a person's hands typing on a white keyboard. The background is dark and out of focus, showing some blurred lights.

Ultimately

Efficiently use MongoDB in
your Java projects right
away

Step by step coding guide



A dark, semi-transparent background image showing several people sitting on a set of wide wooden steps. Some are looking down at their phones, while others are talking. The scene is set outdoors in what appears to be a public space.

You have a 30 day money
back guarantee

Requirements

Basic Java Skills

Some familiarity with
Spring Boot

Desire to write some code
:)



Accelerated Spring Boot 2 & MongoDB

#02 About Me

Dan Geabunea

“Romanian Coder”



Senior Software Developer

In this field for 9 years, working with .NET, Java in cool projects (aviation, banking)

MongoDB Enthusiast

Been using MongoDB since 2012 with most of the big technology stacks

Blogger & Teacher

I just ❤️ teaching. I use my blog, YouTube channel and Twitter for sharing my dev experience

Dan Geabunea

"Romanian Coder"



I'm also social



@romaniancoder



www.youtube.com/romaniancoder



www.romaniancoder.com



Accelerated Spring Boot 2 & MongoDB

#03 Why Learn MongoDB?

Trusted by many



“

As part of our infrastructure redesign, we needed to ensure that new app development was never waiting on the back-end.”

— Luke Kolin, The Weather Channel

Tweet



“

The flexibility of MongoDB has put a lot of power in the hands of our developers.”

— Brian Massey, Under Armour



Tweet



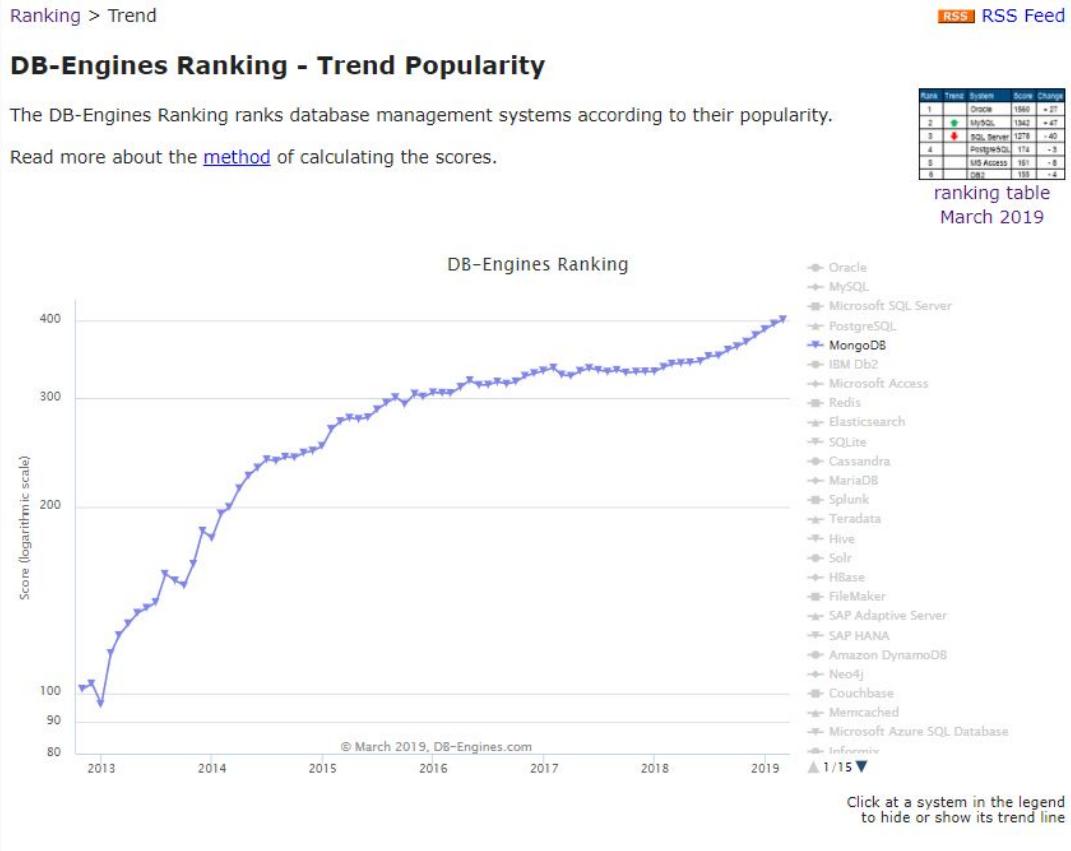
Enterprise Ready

Enterprise Support

ACID Transactions
(multi document)

Deal with massive
amounts of data

Very popular and still rising



Very popular and still rising

DB-Engines Ranking

The DB-Engines Ranking ranks database management systems according to their popularity. The ranking is updated monthly.

Read more about the [method](#) of calculating the scores.

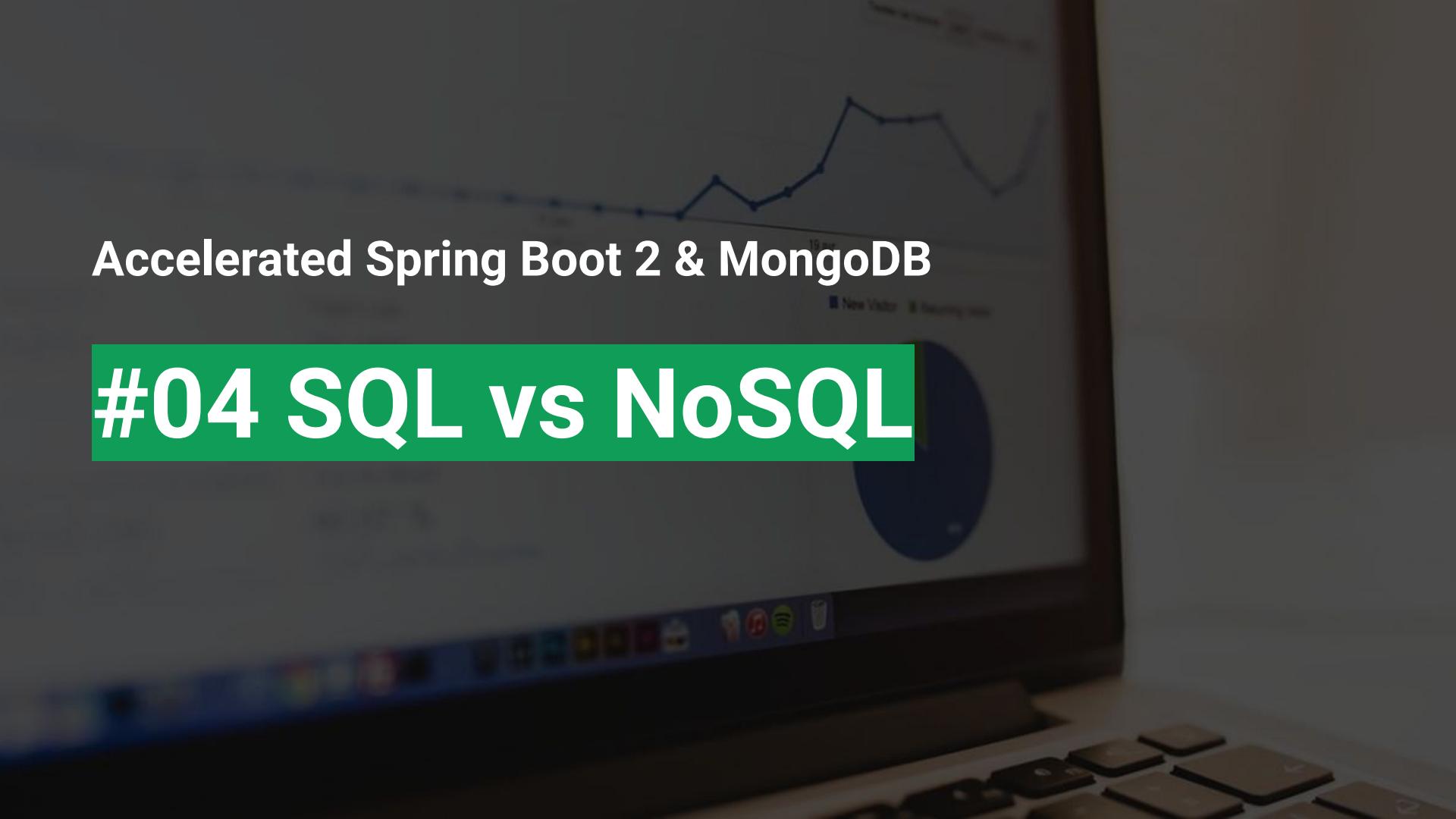


345 systems in ranking, March 2019

Rank				Database Model	Score		
	Mar 2019	Feb 2019	Mar 2018		Mar 2019	Feb 2019	Mar 2018
1.	1.	1.	1.	Oracle	Relational, Multi-model	1279.14	+15.12 -10.47
2.	2.	2.	2.	MySQL	Relational, Multi-model	1198.25	+30.96 -30.62
3.	3.	3.	3.	Microsoft SQL Server	Relational, Multi-model	1047.85	+7.79 -56.94
4.	4.	4.	4.	PostgreSQL	Relational, Multi-model	469.81	-3.75 +70.46
5.	5.	5.	5.	MongoDB	Document	401.34	+6.24 +60.82
6.	6.	6.	6.	IBM Db2	Relational, Multi-model	177.20	-2.23 -9.47
7.	↑ 9.	7.	7.	Microsoft Access	Relational	146.20	+2.18 +14.26
8.	↓ 7.	8.	8.	Redis	Key-value, Multi-model	146.12	-3.32 +14.90
9.	↓ 8.	9.	9.	Elasticsearch	Search engine, Multi-model	142.79	-2.46 +14.25
10.	10.	↑ 11.	11.	SQLite	Relational	124.87	-1.29 +10.06
11.	11.	↓ 10.	Cassandra	Wide column	122.80	-0.58 -0.69	
12.	12.	↑ 15.	MariaDB	Relational, Multi-model	84.31	+0.89 +21.21	
13.	13.	13.	Splunk	Search engine	83.10	+0.29 +17.44	
14.	14.	↓ 12.	Teradata	Relational	75.22	-0.75 +2.76	
15.	15.	↑ 18.	Hive	Relational	73.00	+0.71 +16.00	
16.	16.	↓ 14.	Solr	Search engine	60.01	-0.95 -4.80	
17.	17.	17.	HBase	Wide column	58.80	-1.48 -2.14	
18.	18.	↑ 19.	FileMaker	Relational	58.13	+0.34 +3.00	
19.	↑ 20.	↓ 16.	SAP Adaptive Server	Relational	56.03	+0.29 -6.58	

The background of the slide features a dark, atmospheric city skyline at night. The Empire State Building stands prominently in the center, its Art Deco spire reaching towards a sky filled with scattered clouds. Other skyscrapers are visible, their windows glowing with various shades of light. The overall mood is mysterious and professional.

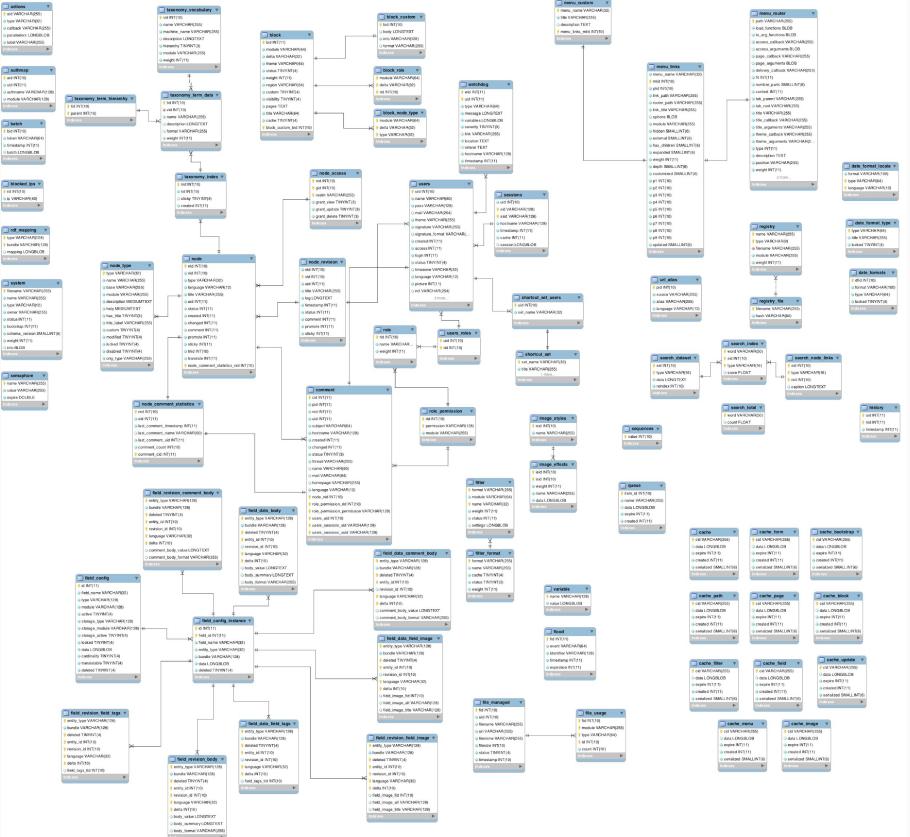
MongoDB is a must have in
your Java coding arsenal

A dark laptop screen is visible in the background, showing a line graph with a blue line and some text at the bottom.

Accelerated Spring Boot 2 & MongoDB

#04 SQL vs NoSQL

The “Relational Problem”



On a 1 to 10 scale, how easy would it be to dramatically change this model?

Not easy :(



Changes to your business
domain will involve a data
model redesign

NoSQL (MongoDB) vs SQL

✓ Easier to evolve the data model, which makes maintaining and evolving applications easier

✓ Scale rapidly

✓ Can handle huge amounts of data with lower costs

✗ Multi document transactions are limited

✗ Not ideal for advanced searching, reporting, business intelligence

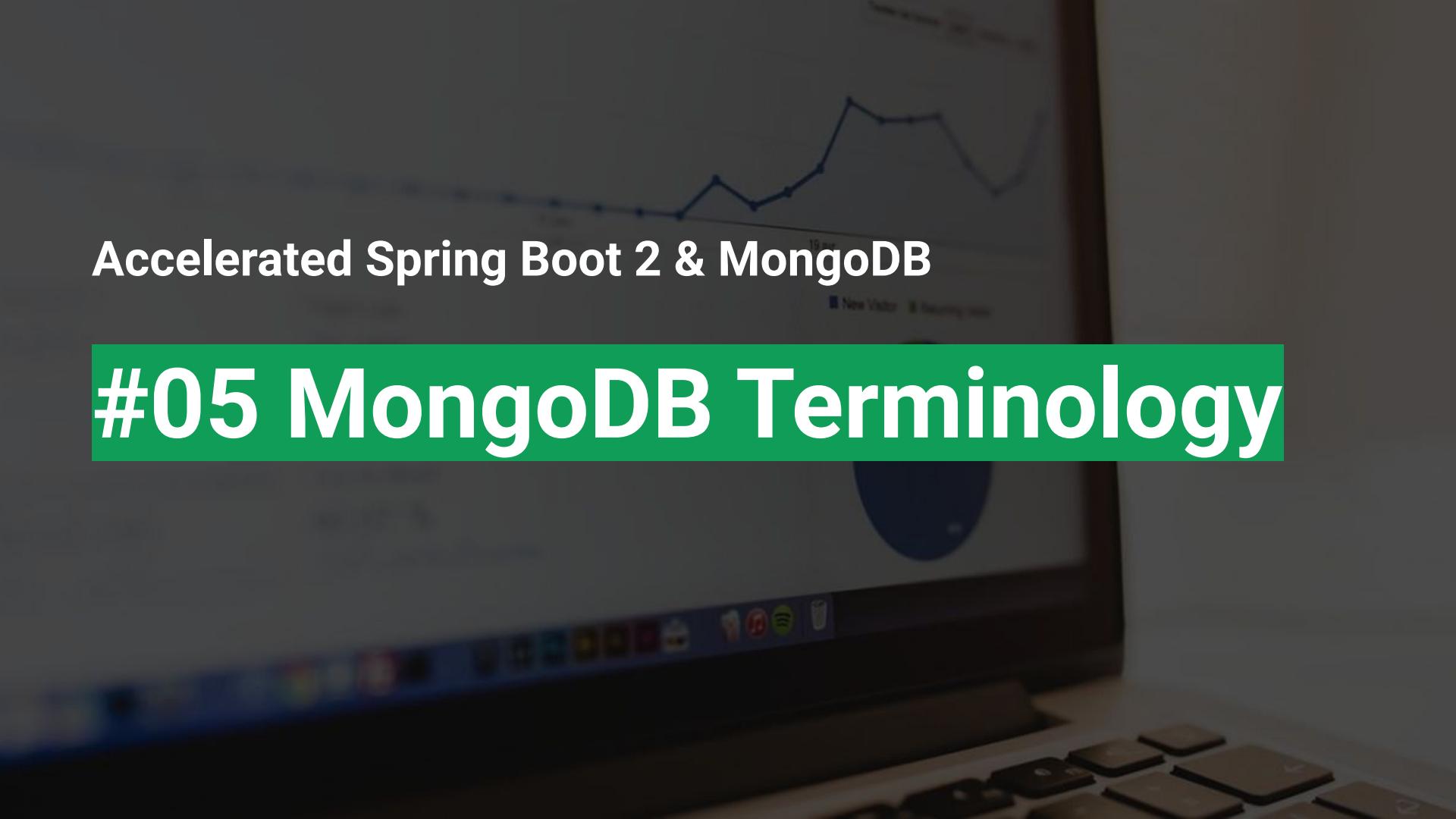
✓ Works very well when the data model is stable

✓ Full transaction support

✓ Extremely good for data mining, business intelligence, etc

✗ Difficult to evolve when model gets complicated

✗ More difficult to scale

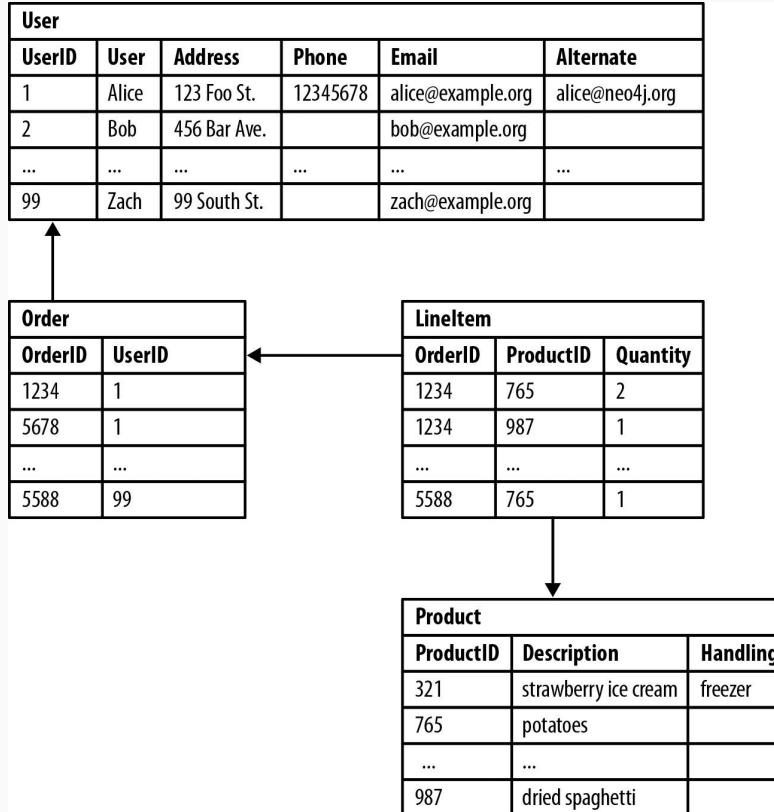
A dark laptop screen is visible in the background, showing a line graph with a blue line and some text at the bottom.

Accelerated Spring Boot 2 & MongoDB

#05 MongoDB Terminology

Relational Data Structure

Relations



<http://www.odbms.org/2015/08/graph-databases-for-beginners-why-data-relationships-matter/>

A dark, slightly grainy photograph showing several people sitting on a set of wide wooden steps. They are seen from behind, looking out over a body of water. The scene is dimly lit, suggesting it might be early morning or late afternoon.

In NoSQL we need to think
differently in terms of data

Object Graphs / Object Trees

Person

- **Id**
- Name
- Age

Address

- Str
- City
- Zip

Accounts

- Nb.
- Bank

Accounts

- Nb.
- Bank

Accounts

- Nb.
- Bank

Documents

Fields

Primary types

Sub-documents

Arrays

No relationships

NoSQL (MongoDB) vs SQL

Database

Table

Rows

Columns

Indexes

Relationships

Database

Collection

Document

Fields

Indexes

Sub-documents (references)

A close-up photograph of a person's hands writing on a piece of paper with a pen. The hands are positioned over a white sheet of paper, with one hand holding a pen and the other hand resting nearby. The background is blurred, showing some greenery and possibly a building.

Annotations

@Document

@Field

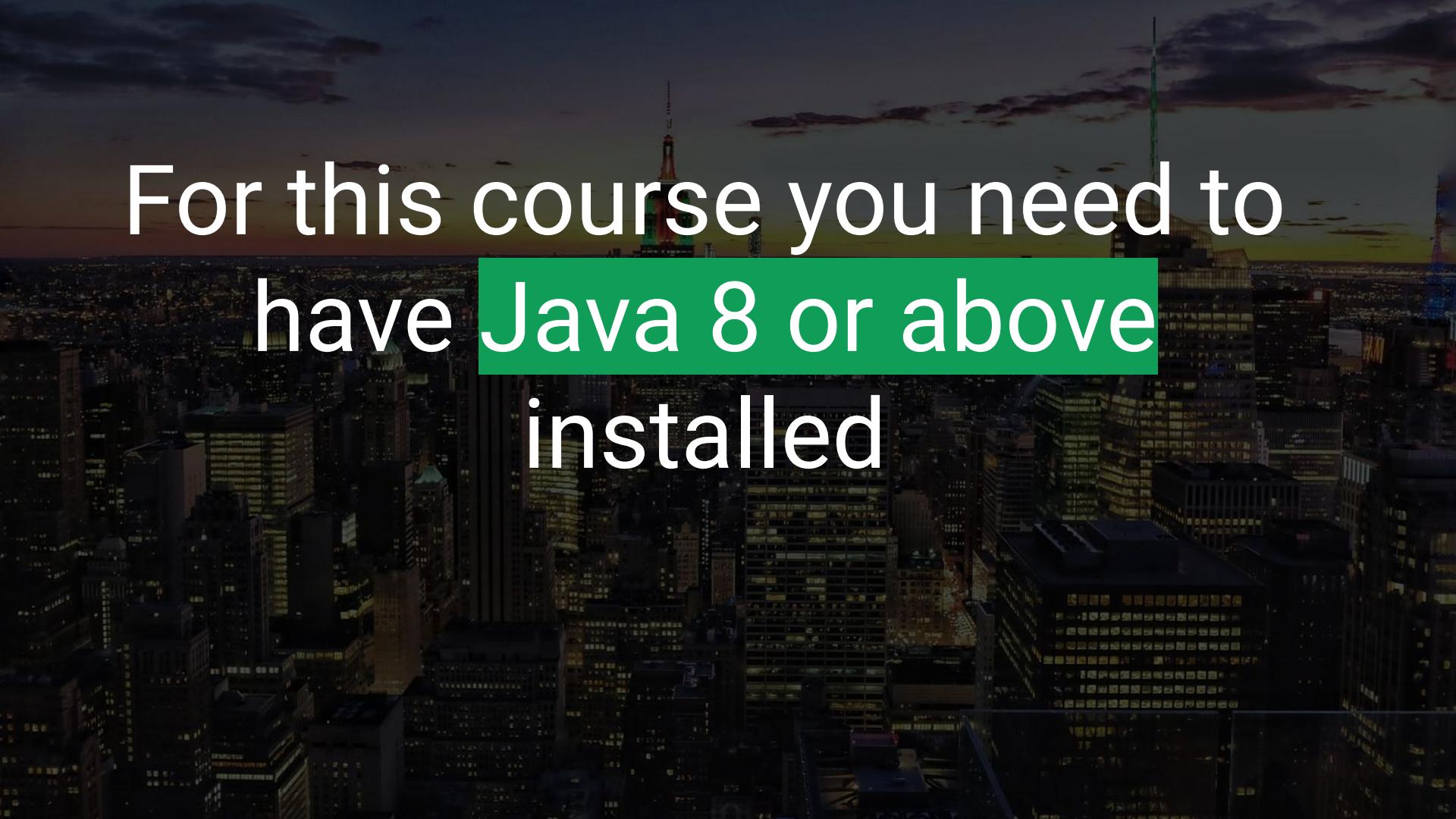
@Indexed

...

A dark laptop screen is visible in the background, showing a line graph with a blue line and some text at the bottom.

Accelerated Spring Boot 2 & MongoDB

#06 Java Requirements



For this course you need to
have Java 8 or above
installed

A dark laptop screen is visible in the background, showing a line graph with a blue line and some text at the bottom.

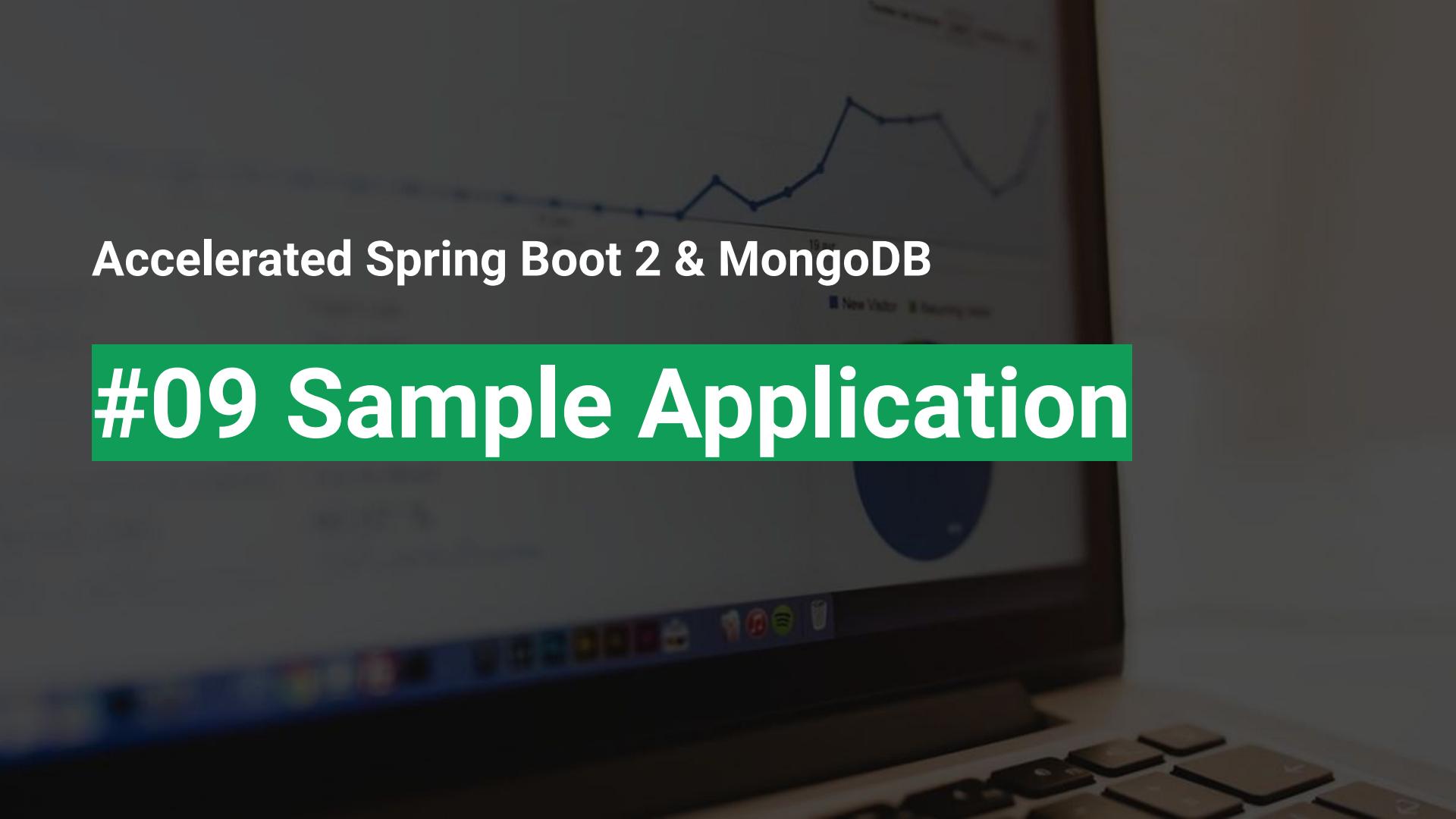
Accelerated Spring Boot 2 & MongoDB

#07 Install MongoDB Server

A dark laptop screen is visible in the background, showing a line graph with a blue line and some text at the bottom.

Accelerated Spring Boot 2 & MongoDB

#08 Install Robo3T Client

A dark laptop screen is visible in the background, showing a line graph with a blue line and some text at the bottom.

Accelerated Spring Boot 2 & MongoDB

#09 Sample Application

Online Lego Store Api :)



App Requirements

Support basic CRUD operations

Filtering (simple and advanced)

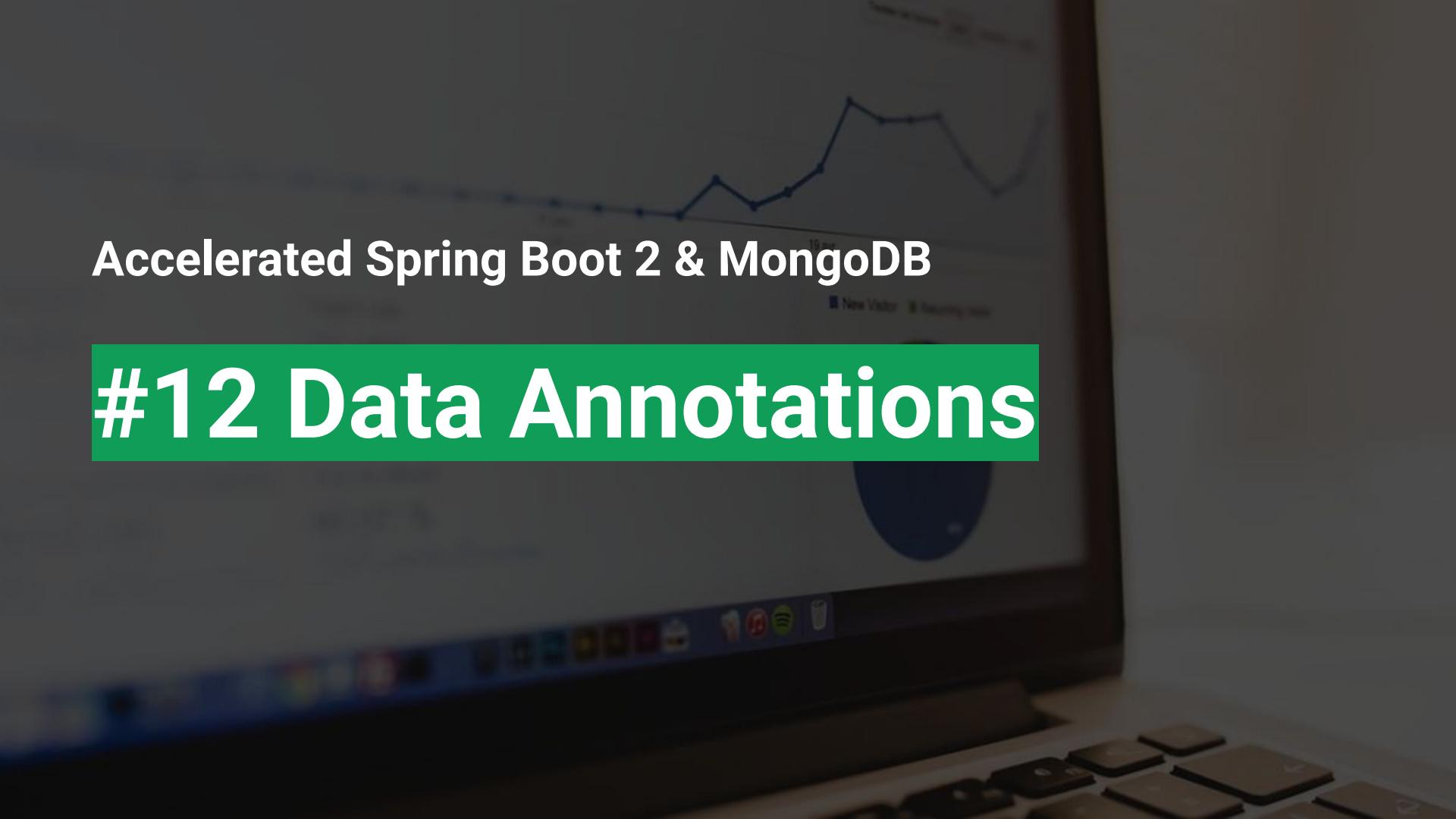
Data Aggregation for Reports

Data Migrations

Integration Testing

Accelerated Spring Boot 2 & MongoDB

#10 Create a SpringBoot & MongoDB Project

A dark, slightly blurred background image of a laptop screen. The screen displays a line graph with a blue line showing fluctuations over time, and a green bar chart below it. The overall theme is data analysis or software development.

Accelerated Spring Boot 2 & MongoDB

#12 Data Annotations

Annotations

Help us to map our classes to our MongoDB documents.

Person.java

Name

Age

Address

Street

Zip

Email

Mongo Representation

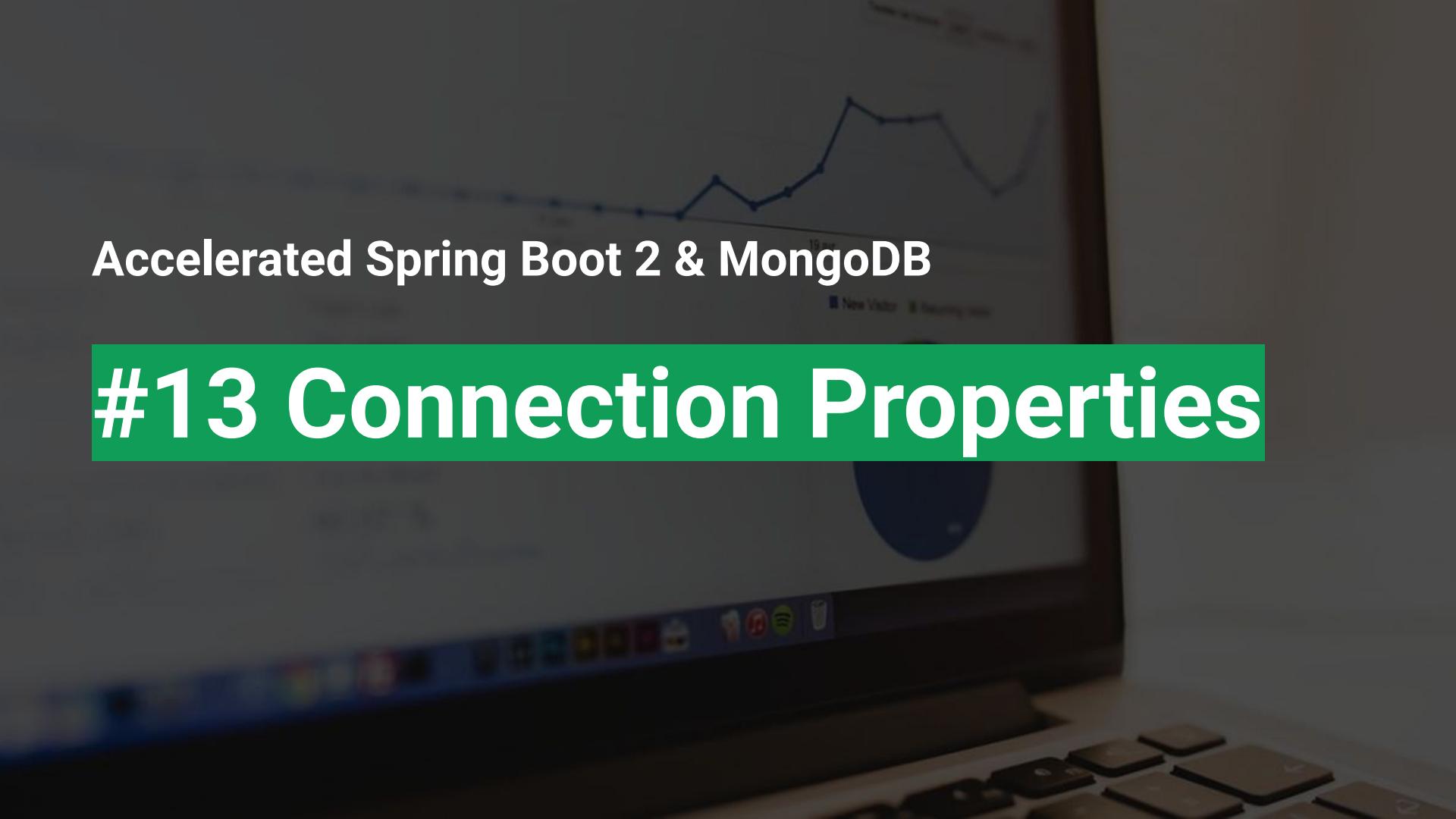
```
{  
  "_id",  
  "name",  
  "mail",  
  "addr": {  
    "str"  
  }  
}
```

Annotation Examples

@Document
@Field

@Id

@Indexed
@TextIndexed
@Transient
@PersistenceCons
tructor

A dark laptop screen is visible in the background, showing a line graph with a blue line and some text at the bottom.

Accelerated Spring Boot 2 & MongoDB

#13 Connection Properties



Accelerated Spring Boot 2 & MongoDB

#14 Create DB/Collections



Accelerated Spring Boot 2 & MongoDB

#15 Insert



Accelerated Spring Boot 2 & MongoDB

#16 Get All



Accelerated Spring Boot 2 & MongoDB

#17 Update

Insert vs Save

INSERT

- If the object has no Id (Id is null), a new Id will get generated
- If the object has an Id value:
 - If Id value not already present in the collection, will use it to insert
 - **Else, MongoDB will throw a Duplicate Key Exception**

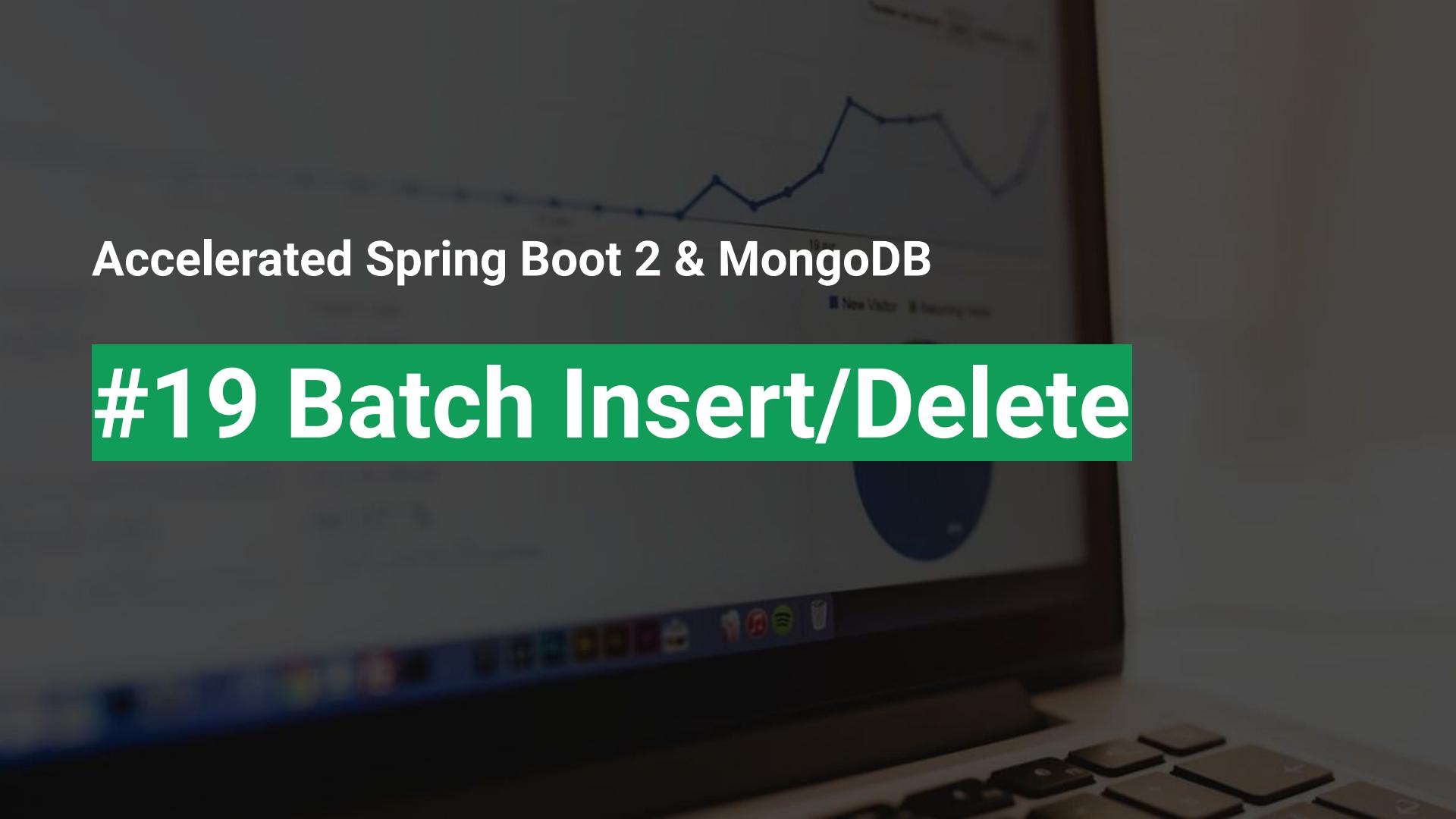
SAVE

- If the object has no Id (Id is null), a new Id will get generated
- If the object has Id value:
 - If Id value not already present, will use it
 - **If Id value present, will update the document**



Accelerated Spring Boot 2 & MongoDB

#18 Delete

A dark laptop screen is visible in the background, showing a line graph with a blue line and some text at the bottom.

Accelerated Spring Boot 2 & MongoDB

#19 Batch Insert/Delete

Pre-populates a database

Application Starts

Delete All
Documents (DB)

Insert Documents
in Collections (DB)

A dark, slightly blurred background image of a laptop screen. The screen displays a line graph with a blue line showing an upward trend over time, and a green bar chart below it. The overall theme is data analysis or software development.

Accelerated Spring Boot 2 & MongoDB

#20 Mongo Repository

MongoTemplate vs MongoRepository

✓ More flexible and powerful (more complex queries, aggregations)

✗ Low level; You need to know how Mongo queries work

✓ Easier to use because they are a higher abstraction (90% cases)

✓ Friendly syntax for filtering data

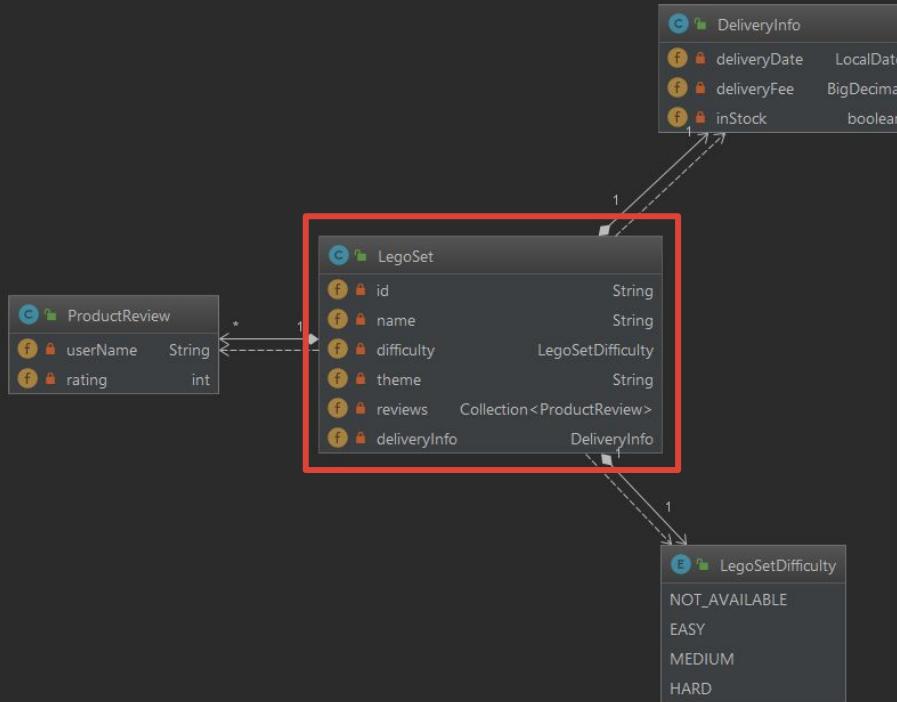
✓ Built on top of JPA, consistent syntax, same methods as repositories over SQL

✗ Do not work on all use cases; Sometimes you need to fall back to the MongoTemplate

Method Name Composition

Auto generates queries
based on method names

Our Object Graph





Accelerated Spring Boot 2 & MongoDB

#21 Find By Id

Accelerated Spring Boot 2 & MongoDB

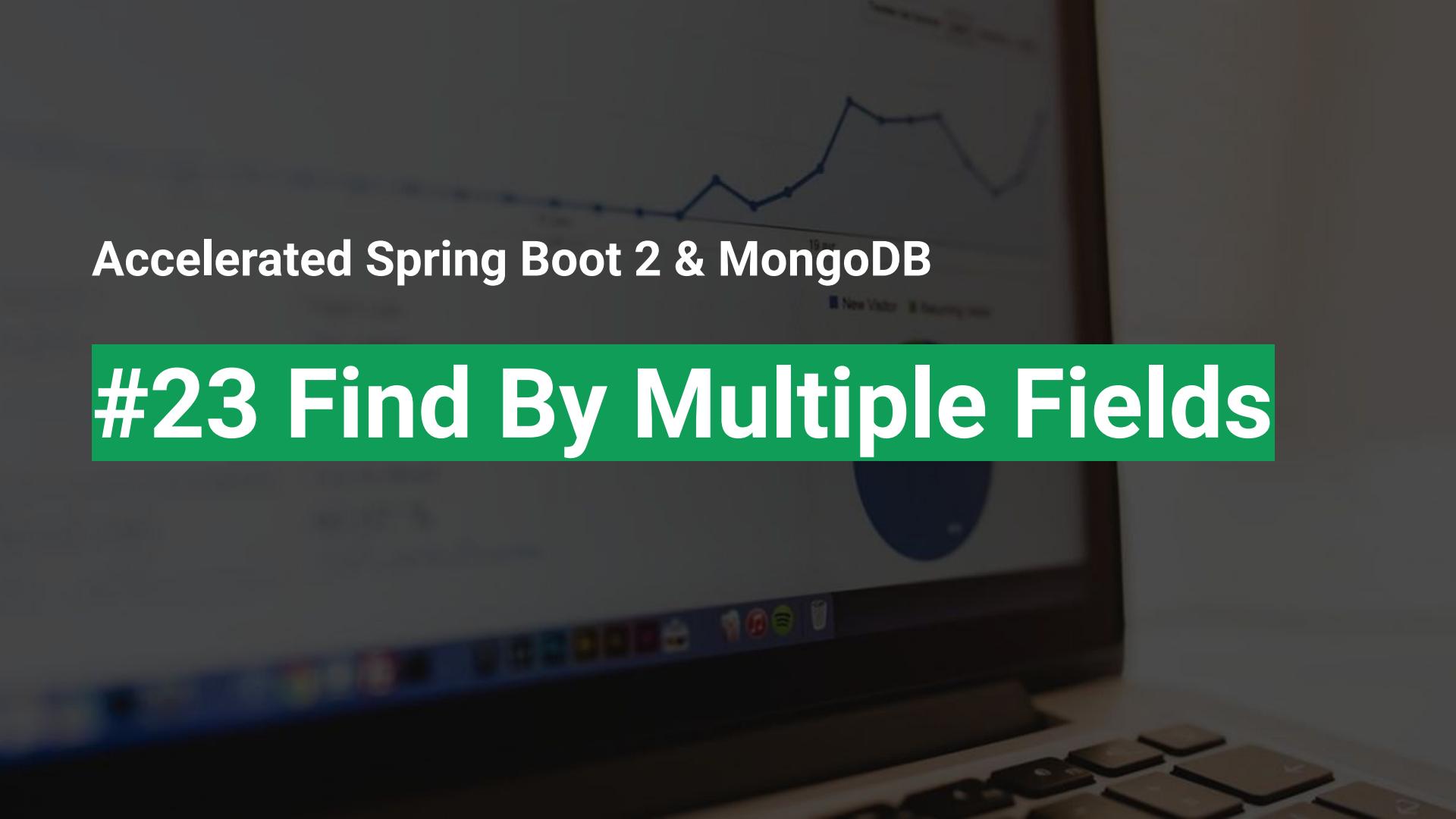
#22 Find By Field

Method Name Composition

Spring Data can provide proxy implementation of queries based on method names:

- **findByLastname**
- **findByAgeLessThan**
- **findByActiveTrue**

<https://docs.spring.io/spring-data/jpa/docs/current/reference/html/>

A dark laptop screen is visible in the background, showing a line graph with a blue line and some text at the bottom.

Accelerated Spring Boot 2 & MongoDB

#23 Find By Multiple Fields

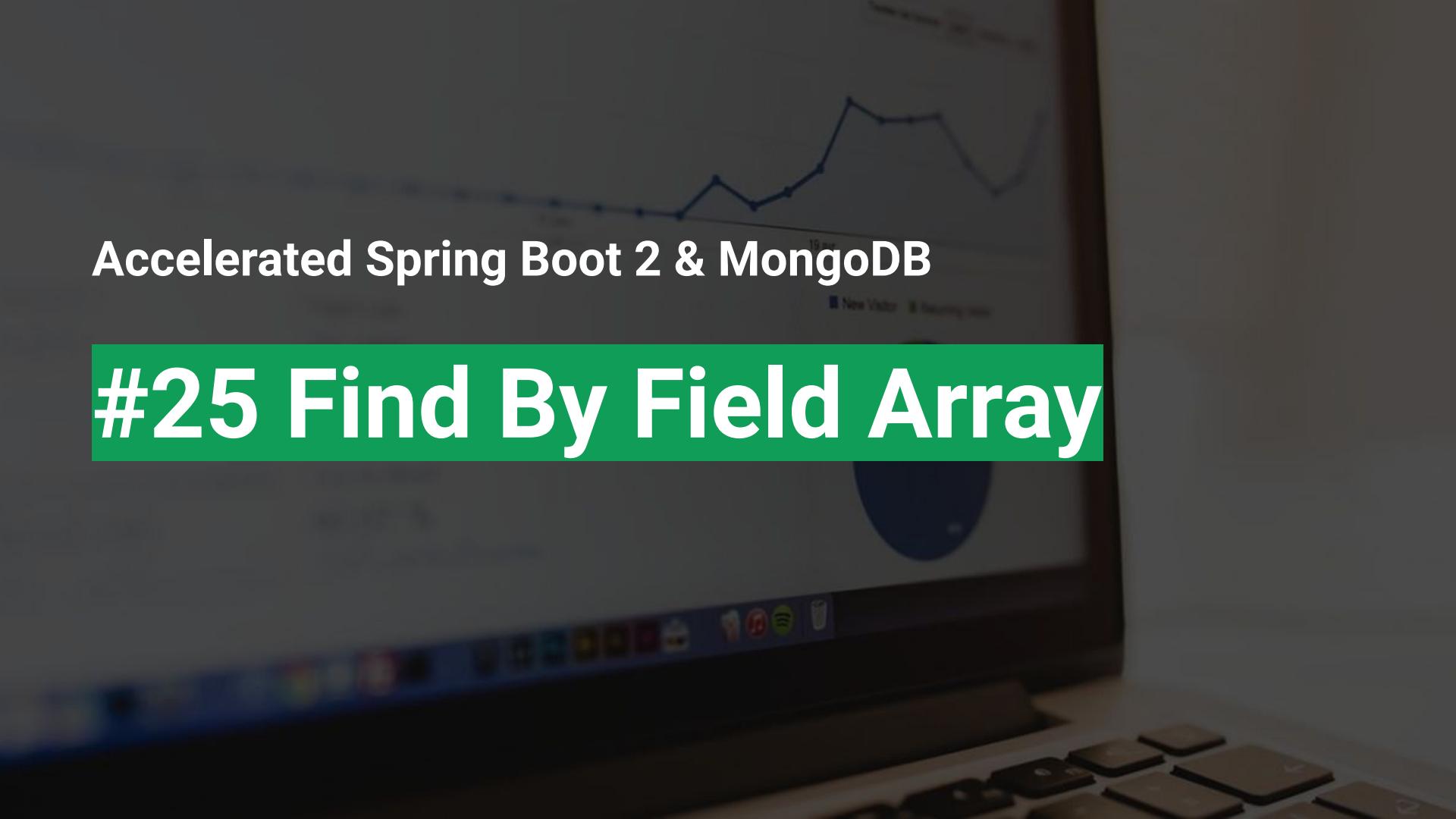
Accelerated Spring Boot 2 & MongoDB

#24 Find By Field In Sub-document

@Query

You can build custom Mongo Queries and execute them when a Repository method gets called

```
@Query("{'_id' : '123'})
```

A dark, semi-transparent background image of a laptop screen. The screen displays a line graph with a blue line showing fluctuations over time, and a green bar chart below it. The overall theme is data analysis or web development.

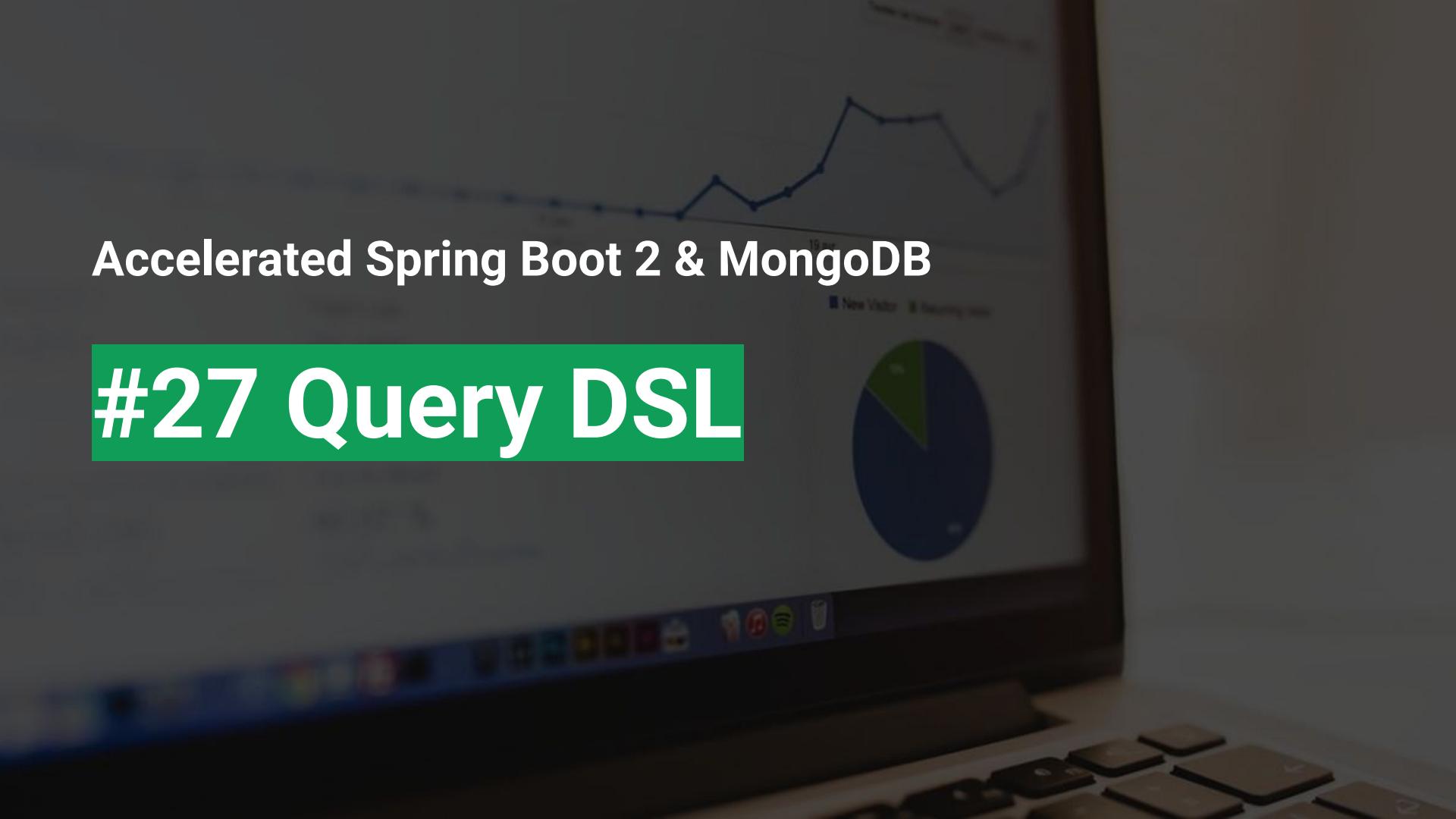
Accelerated Spring Boot 2 & MongoDB

#25 Find By Field Array



Accelerated Spring Boot 2 & MongoDB

#26 Sorting

A laptop screen is visible in the background, displaying a dark-themed dashboard. The dashboard features a line chart with a blue line showing a general upward trend over time, and a pie chart divided into two main segments. The laptop is resting on a desk with a keyboard partially visible.

Accelerated Spring Boot 2 & MongoDB

#27 Query DSL

@Query

You can build custom Mongo Queries and execute them when a Repository method gets called

```
@Query("{'_id' : '123'}")
```

```
@Query("{'delivery.inStock' : {$eq : true}}")
```

QueryDSL allows us to write
queries that are type safe. No
more magic strings.

Process

Scans all classes annotated with @Document, and their children

Generates “query types” during the process goal of Maven build. For example, for a LegoSet class => QLegoSet class

We can then use query types in our code to build complex queries in a type safe manner

Accelerated Spring Boot 2 & MongoDB

#28 Query DSL Maven Dependencies



Dependencies

To start using QueryDSL
for Mongo we need to
bring in:

- Library Dependencies
- Plugin

Accelerated Spring Boot 2 & MongoDB

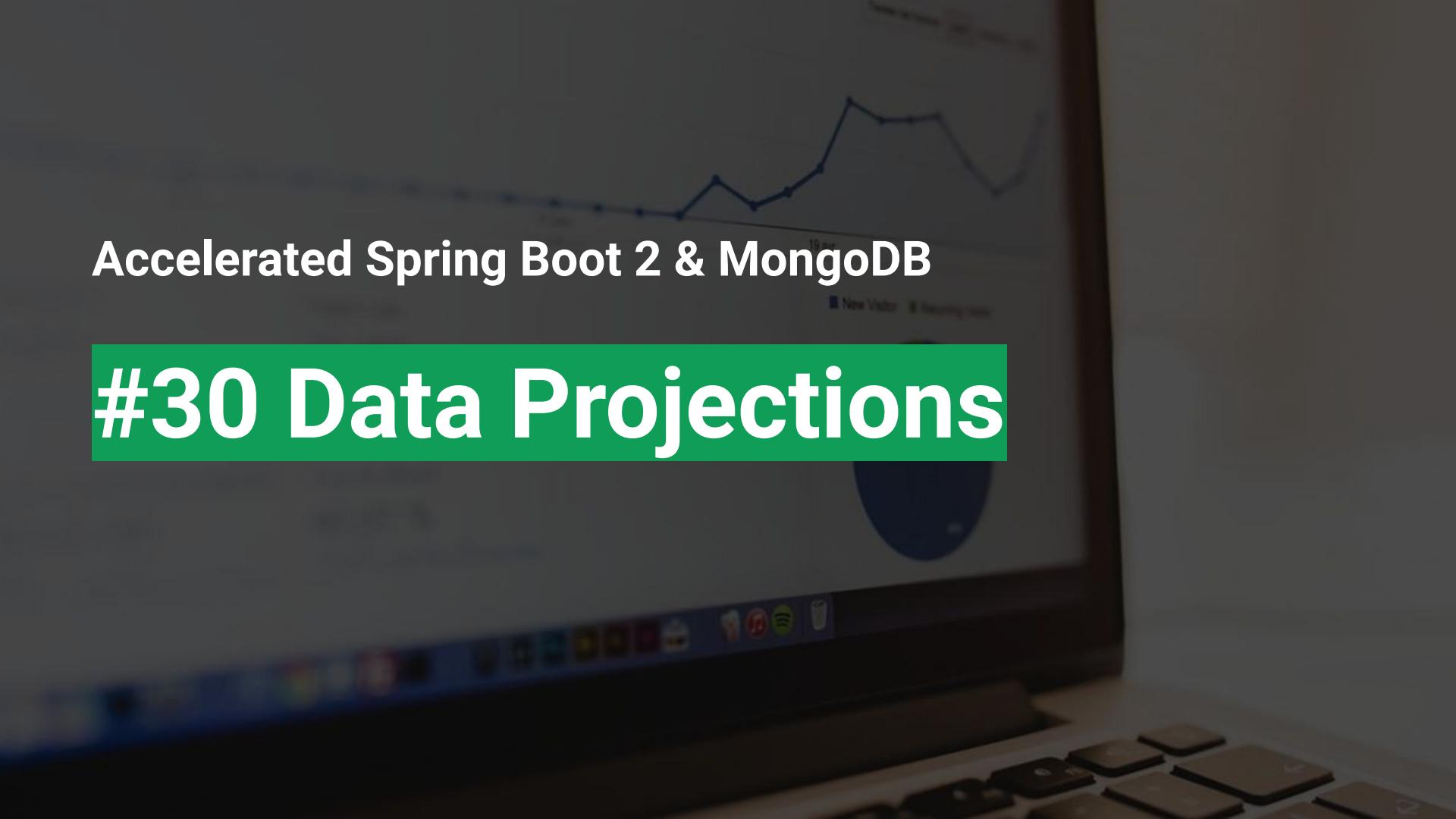
#29 Filter Data Using QueryDSL

A close-up photograph of a person's hands working on a LEGO set. The hands are positioned over a light-colored surface, likely a table or desk, and are carefully placing small blue LEGO pieces. In the background, more blurred LEGO elements are visible.

Best Buys

Query all lego sets that:

- Are in stock
- Have delivery fee less than 50
- Have at least one review with a 10 rating

A dark laptop screen is visible in the background, showing a line graph with a blue line and some text at the bottom.

Accelerated Spring Boot 2 & MongoDB

#30 Data Projections



Reports

Our manager wants a report on all the lego sets and their **average rating** so he can optimize sales.

Data Transformations

```
{  
  "_id": "123",  
  "name": "Millenium Falcon",  
  "reviews": [  
    {"rating": 8},  
    {"rating": 10}  
  ]  
}  
  
{  
  "_id": "123",  
  "productName": "Millenium Falcon",  
  "avgReview": 9  
}
```



Accelerated Spring Boot 2 & MongoDB

#31 Average Rating Projection In Mongo Query Language

Mongo Projections

```
db.legosets.aggregate([  
    {$project : {  
        legoSetName: "$name", ←  
        avgRating : {$avg : "$reviews.rating"} ←  
    }  
})
```

Accelerated Spring Boot 2 & MongoDB

#32 Implement Average Rating Report

Implementation

Average Rating Report Feature

Create the aggregation/projection in Mongo Query Language

Define a class model for the projection (AvgRatingModel)

Build projection using MongoTemplate class

Execute the projection and convert to AvgRatingModel



Implement Average Rating Report Feature

Create the
aggregation/projection in
Mongo Query Language

Define a model for
projection
(AvgRatingModel)

Build projection using
MongoTemplate class

A dark, slightly blurred background image of a laptop screen. The screen displays a line graph with a blue line showing an upward trend over time, and a green bar chart below it. The overall theme is data analysis or web development.

Accelerated Spring Boot 2 & MongoDB

#33 Full Text Search

A close-up photograph of a person's hands resting on a laptop keyboard. The hands are positioned as if the person is about to type or has just finished. The background is blurred, showing some colorful shapes.

@TextIndexed

You can add it on root level properties

You can add it on sub-documents and array elements

Accelerated Spring Boot 2 & MongoDB

#34 Changing The Data Structure



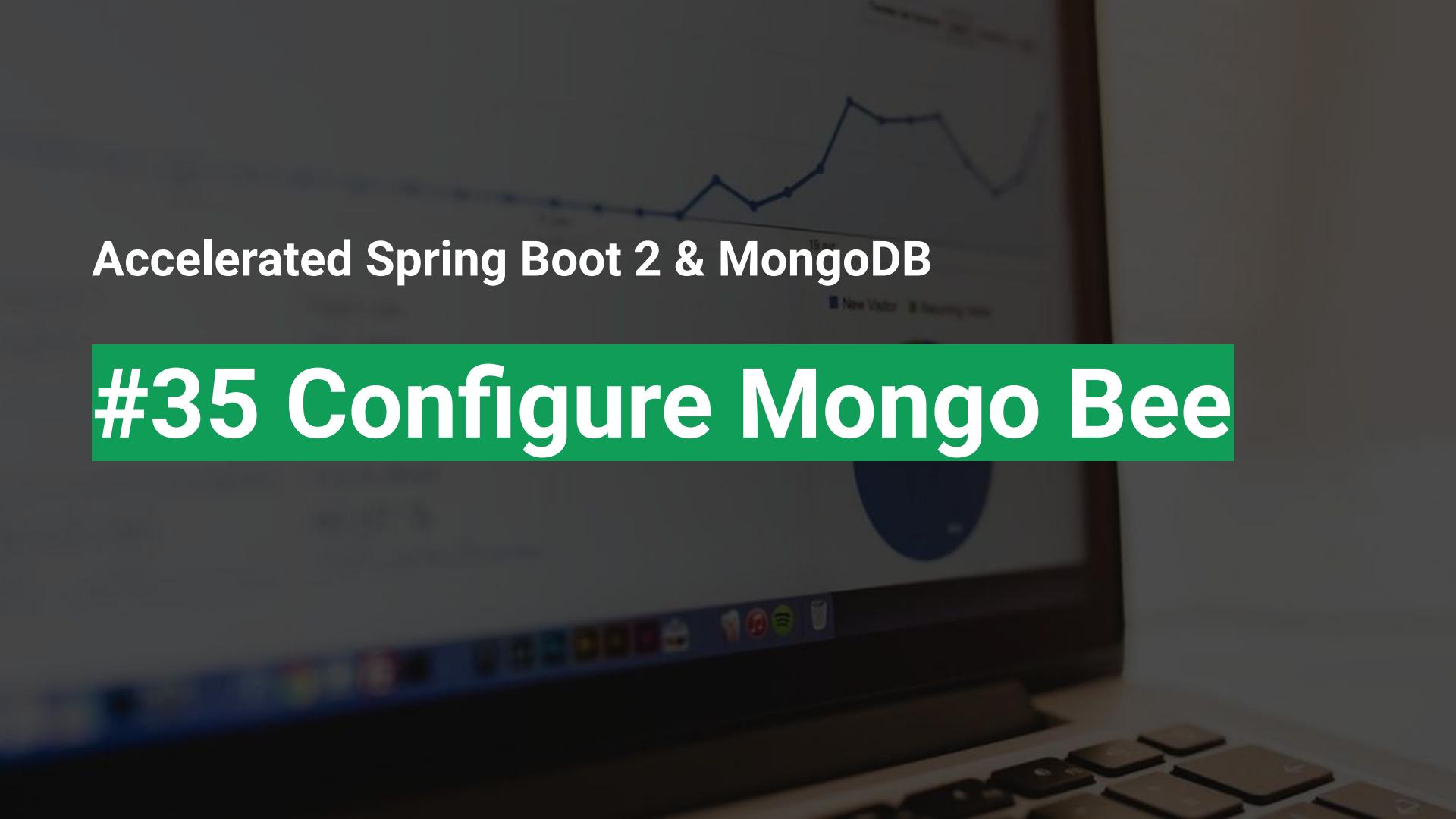
Data Structure

Add new collections

Add new fields

Remove fields

Change name of existing
fields



Accelerated Spring Boot 2 & MongoDB

#35 Configure Mongo Bee

Accelerated Spring Boot 2 & MongoDB

#36 Create Migrations Using Mongo Bee

Accelerated Spring Boot 2 & MongoDB

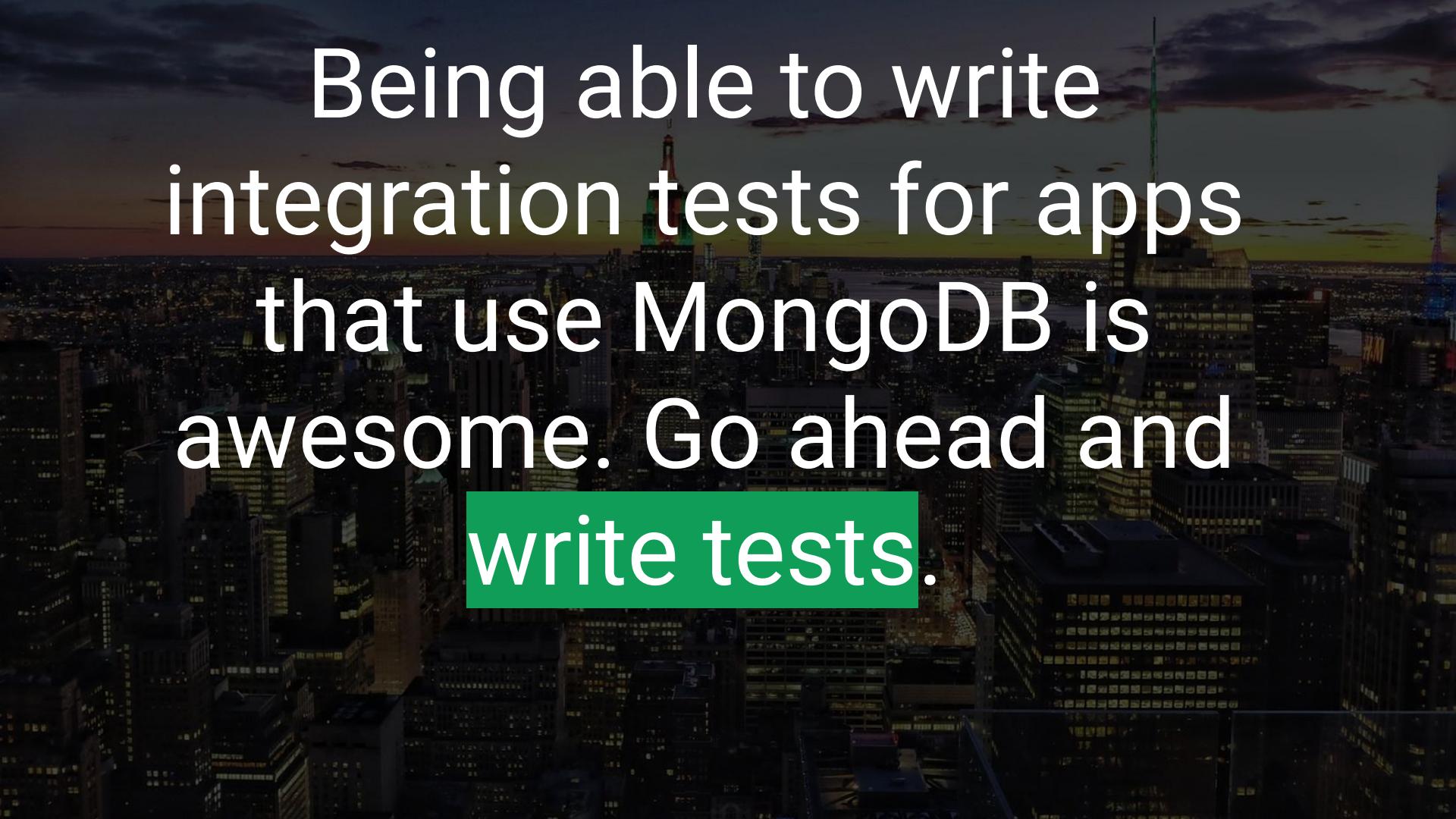
#37 Integration Testing Overview

How?

How can you run integration tests? Against which server/database?

There is full Spring Support for writing integration tests

We can use an embedded (in memory) MongoDB Server for testing

The background of the slide features a dark, atmospheric city skyline at either dusk or night. The city lights are visible through the windows of numerous skyscrapers, creating a dense grid of glowing points against a dark sky. In the center, the Eiffel Tower is partially visible, its silhouette standing out. The overall mood is mysterious and modern.

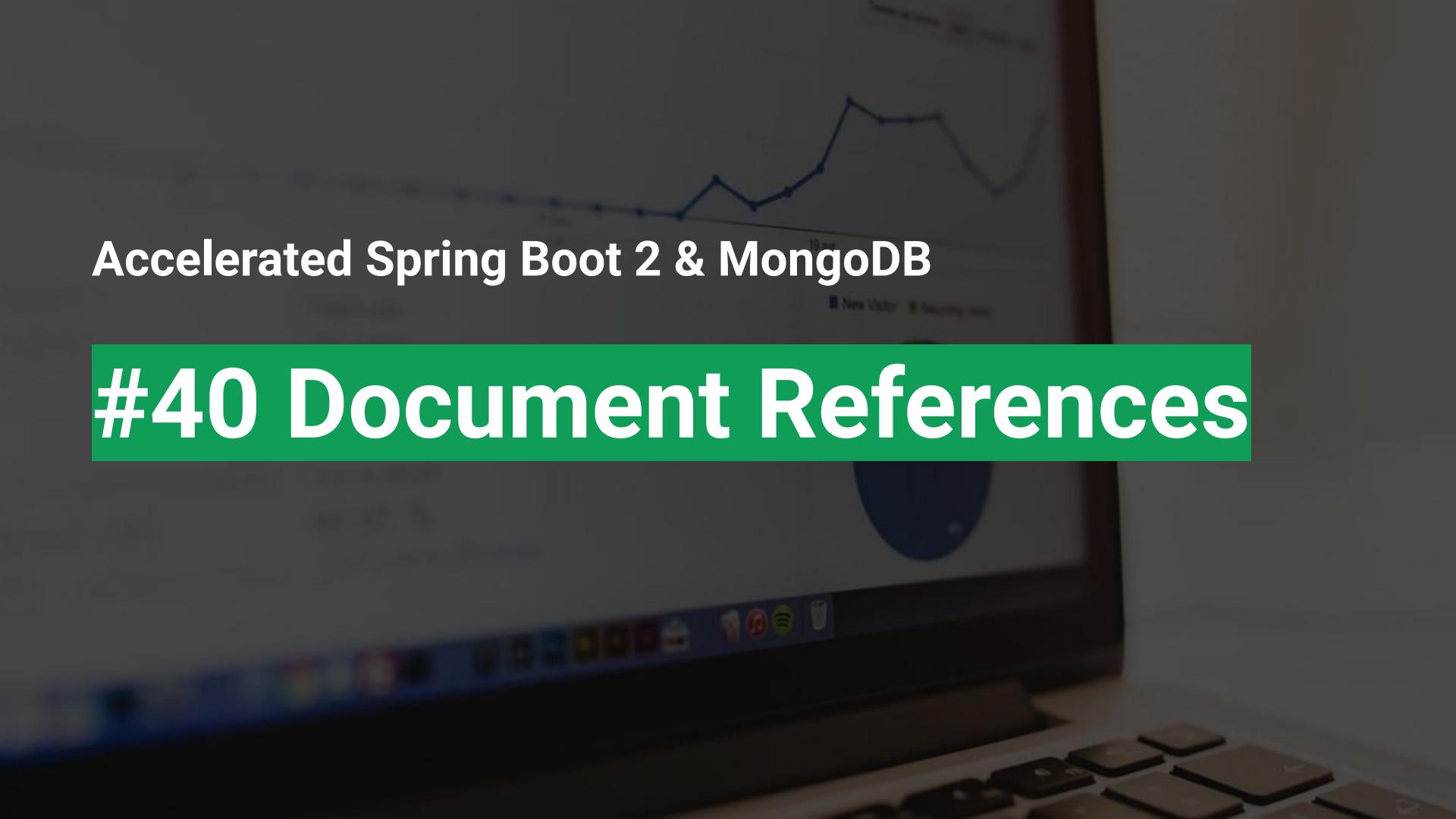
Being able to write
integration tests for apps
that use MongoDB is
awesome. Go ahead and
write tests.

Accelerated Spring Boot 2 & MongoDB

#38 Add In Memory Mongo Database

Accelerated Spring Boot 2 & MongoDB

#39 Writing Integration Tests

A dark laptop screen is visible in the background, showing a line graph with a blue line and some text at the bottom.

Accelerated Spring Boot 2 & MongoDB

#40 Document References

Embedded Documents

LegoSets Collection

```
{  
  "_id" : "123",  
  "name" : "Millenium Falcon",  
  "paymentOption" : {  
    "type" : "CreditCard",  
    "fee" : 0  
  }  
}
```

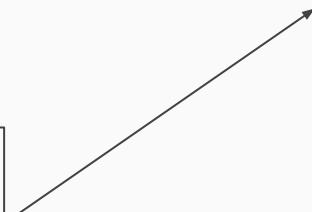
Referenced Documents

LegoSets Collection

```
{  
  "_id" : "123",  
  
  "name" : "Millenium Falcon",  
  
  "paymentOption" : {  
    "$ref" : "PaymentOptions",  
    "$id" : "567"  
  }  
}
```

PaymentOptions Collection

```
{  
  "_id" : "567",  
  
  "type" : "CreditCard",  
  "fee" : 0  
}
```



Good To Know

```
{  
  "_id": "123",  
  "name": "Millenium Falcon",  
  "paymentOption": {  
    "$ref": "PaymentOptions",  
    "$id": "567"  
  }  
}
```

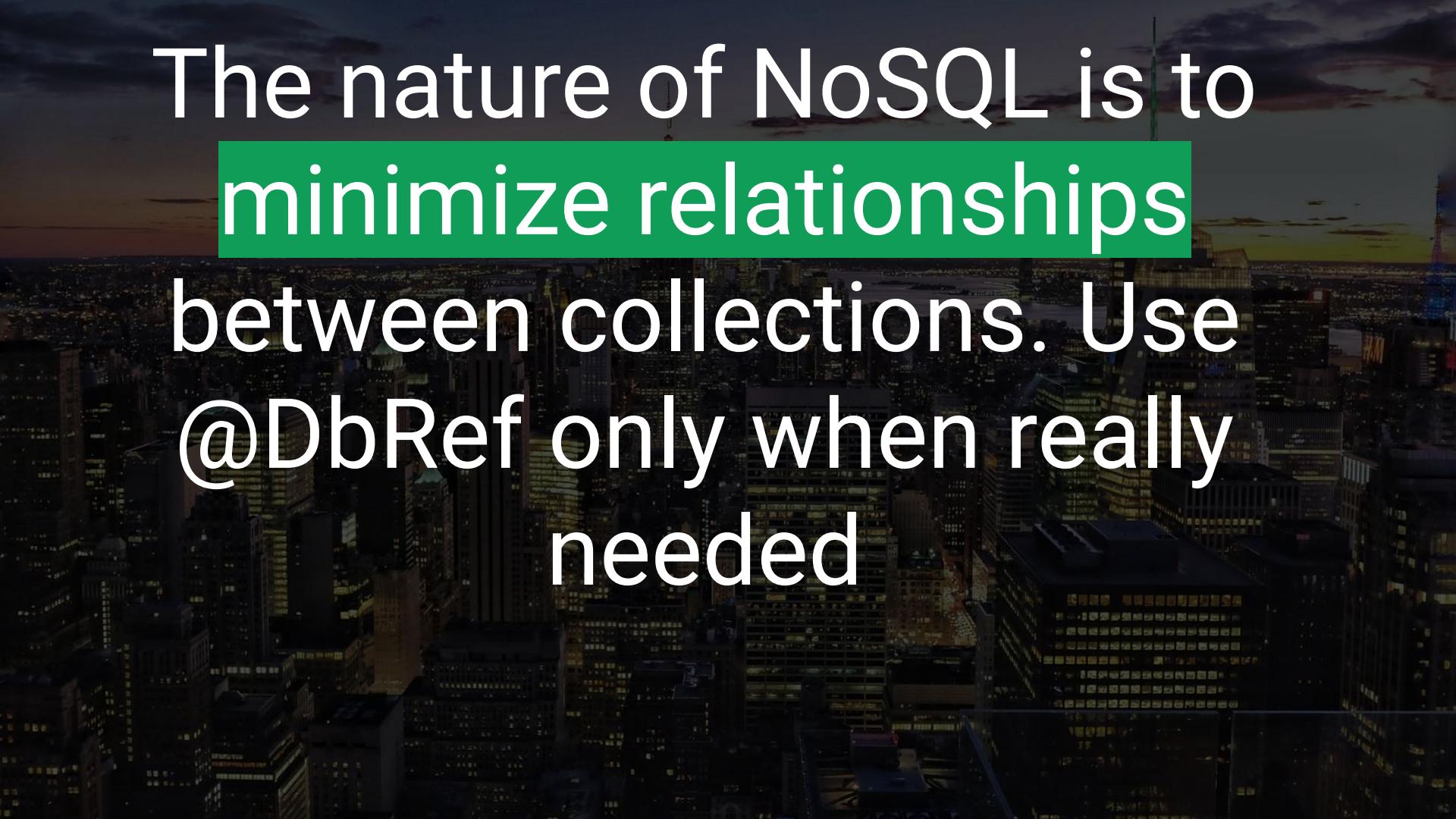
Eager loading/serialization in Spring

You can not traverse the object tree to query based on payment options

`{"paymentOptions.fee" : 0}`

No cascading operations by default on parent-child relationships. You need to manually manage them.

But you can implement cascading operations via Mongo event listeners like `onBeforeConvert`



The nature of NoSQL is to
minimize relationships
between collections. Use
@DbRef only when really
needed

Accelerated Spring Boot 2 & MongoDB

#41 Create Document References Using @DbRef

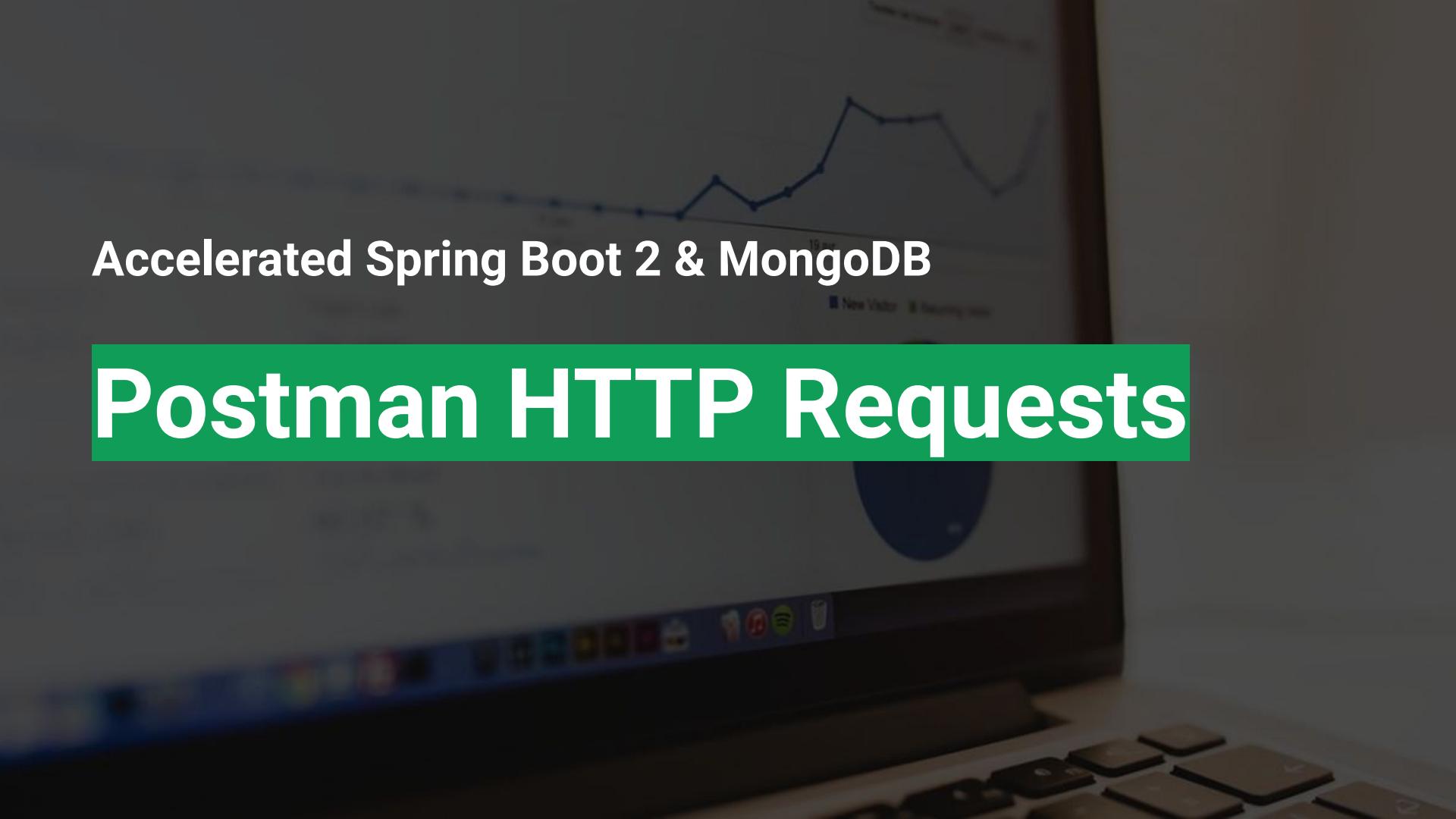
Accelerated Spring Boot 2 & MongoDB

#42 Find By Document Reference Id

A laptop screen is visible in the background, showing a dark-themed dashboard. The dashboard features a line graph with several data points, a pie chart divided into two main segments (dark blue and light green), and some small icons or text at the bottom. The overall aesthetic is professional and modern.

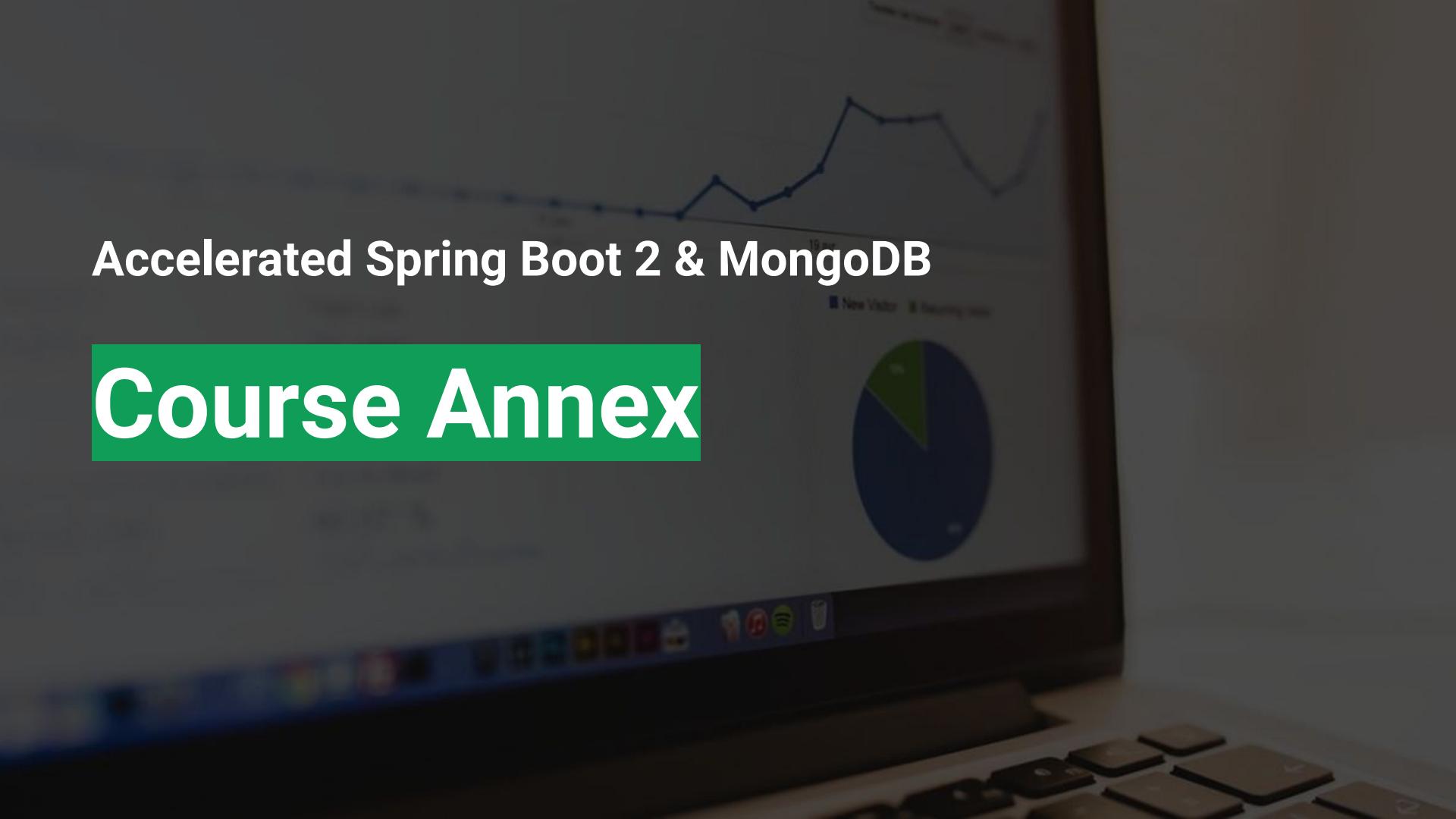
Accelerated Spring Boot 2 & MongoDB

Source Code



Accelerated Spring Boot 2 & MongoDB

Postman HTTP Requests

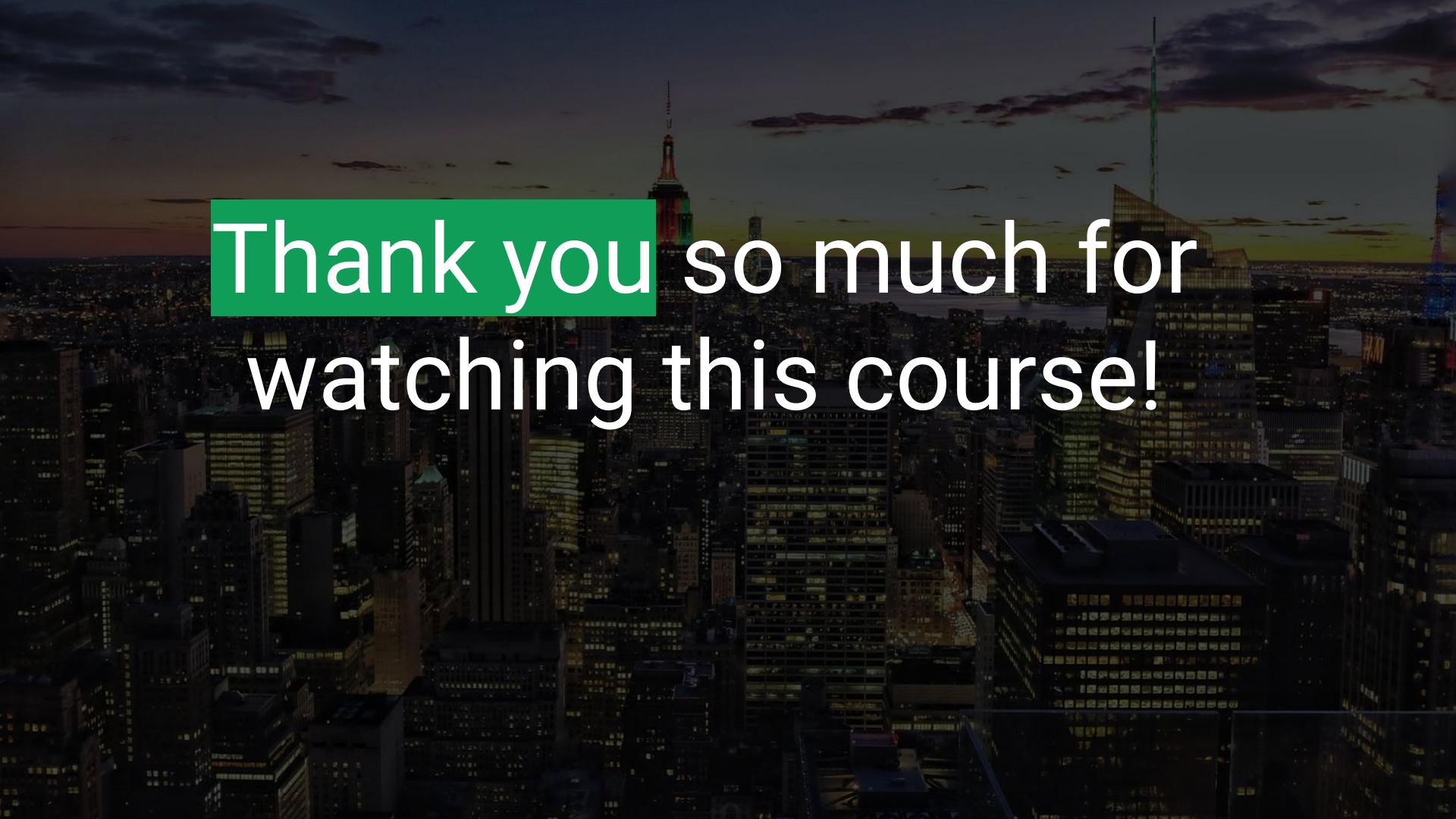
A laptop screen is visible in the background, showing a dark-themed dashboard. The dashboard features a line graph with several data points, a pie chart divided into two main segments (dark blue and light green), and some small icons or text at the bottom. The overall theme is professional and technical.

Accelerated Spring Boot 2 & MongoDB

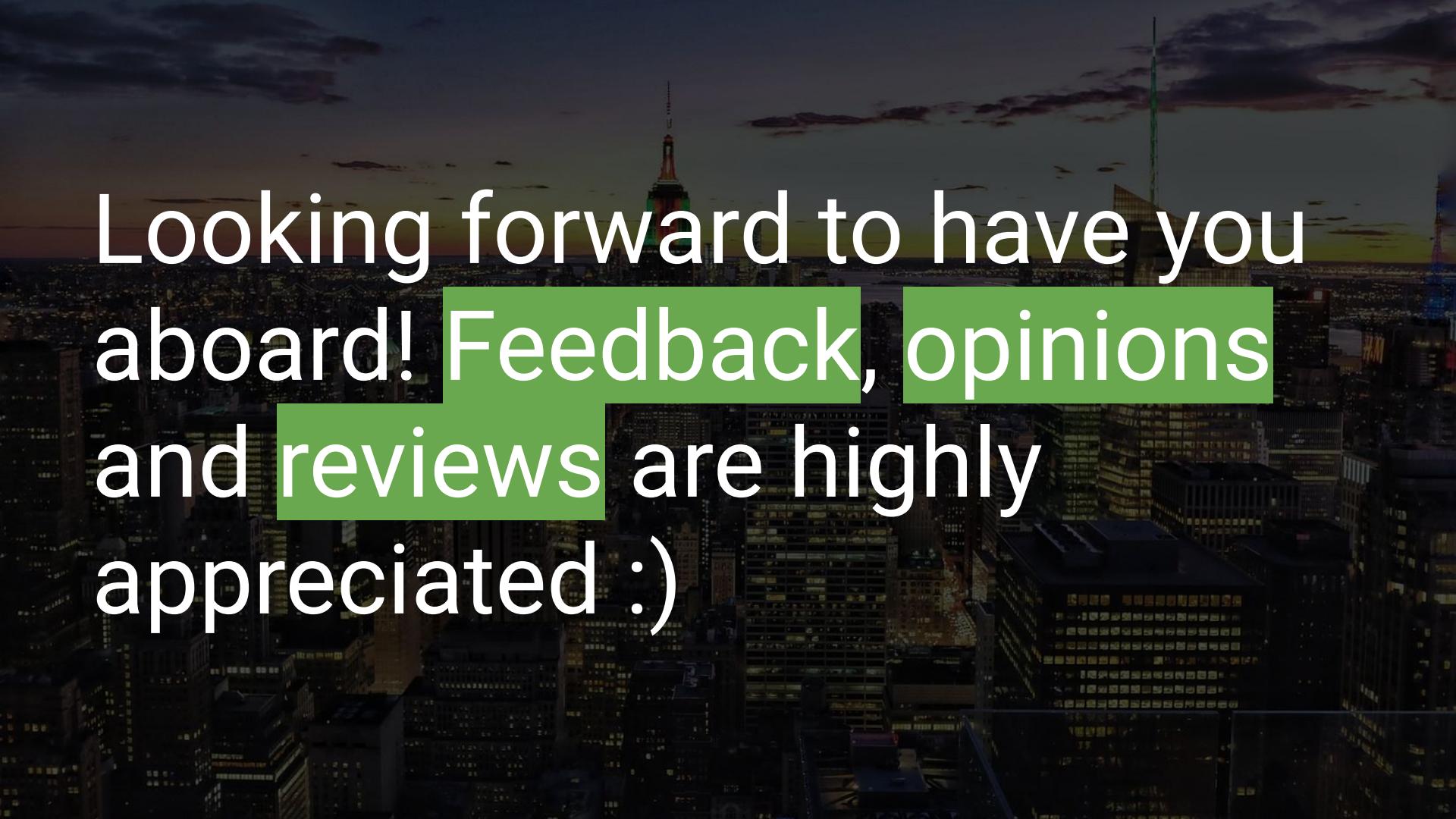
Course Annex

Accelerated Spring Boot 2 & MongoDB

Final Thoughts

A dark, atmospheric photograph of a city skyline at dusk or night. The sky is filled with dramatic, colorful clouds in shades of orange, yellow, and purple. In the foreground, numerous skyscrapers are silhouetted against the light, their windows reflecting some light. Two prominent towers stand out: the Empire State Building on the left, its Art Deco spire reaching towards the top, and One World Trade Center on the right, its tall, slender spire reaching upwards. The overall mood is moody and celebratory.

Thank you so much for
watching this course!



Looking forward to have you
aboard! Feedback, opinions
and reviews are highly
appreciated :)