DataStax

Developers

App Development with Apache Cassandra™

Bootcamp Week 3:
Full Stack Back End App Development with Apache Cassandra

Cedrick Lunven | Aleksandr Volochnev | Jack Fryer | Kirsten Hunter | Stefano Lottini | David Gilardi | Ryan Welford | Rags Srinivas | Sonia Siganporia

R S

DataStax Developers Crew

Senior Developer Advocate

- Training
- Teaching
- Support
- Public Speaking
- Reference Applications
- Cassandra Certified
- *Cough (Oracle 8 certified)

- Happy developer for 25 years
- Coded in a ton, but Java/js mainstays

- Advanced User

david-gilardi

SonicDMG

SonicDMG

David Jones-Gilardi

Director of Developer Relations

clunven

clunven

clun

Astra SDK

cassandra

- Trainer
- Public Speaker
- Developers Support
- Developer Applications
- Developer Tooling

Java

FF4J

- Creator of ff4j (ff4j.org)
- Maintainer for 8 years+

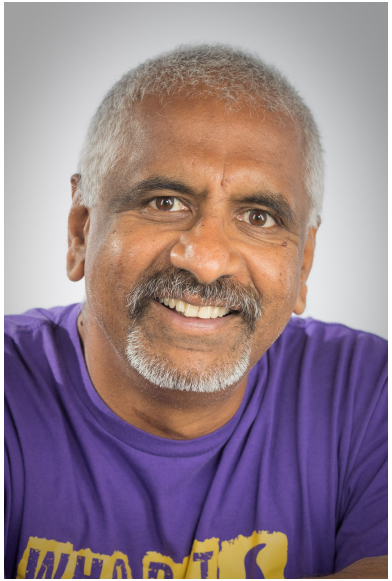
{ REST }

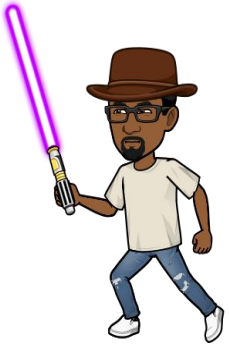
GRPC

- Happy developer for 14 years
- Spring Petclinic Reactive & Starters
- Implementing APIs for 8 years

Cédrick Lunven

Developer Advocate





cassandra

@rags

@ragsns

@ragss

- Developer/Architect
- Mechanical Engineer (so many moons ago)
- Distributed systems
- Love to teach and communicate
- Inner loop == developer productivity

Raghavan "Rags" Srinivas

Developer Advocate

@stefano-lottini

@hemidactylus

@rsprrs

- Developer/Architect
- Apache Cassandra™ certified

- Background in computational physics
- Distributed systems
- Love to teach and communicate

dtsx.io/stefano

Stefano Lottini

01

Weeks 1 &2 Reminders
HouseKeeping

02

The Todo Application
Context and Demo

03

Connectivity to Cassandra
Introduction to Drivers

04

Execute Statements
CRUD

05

What's next?
Quiz, Homework, Next week

Agenda

WEEK 1

January 5th - January 11th

WEEK 3

January 19th-January 25th

WEEK 2

Ianuary 12th - January 18th

WEEK 4

January 26th-February 1st

#1 Introduction to NoSQL Databases and Apache Cassandra™

- ~2H – Live lessons (same but different timezones)
  - January 5th 6 PM CET (= 9am PST = 11 am CST = 12 pm EST)
  - January 6th 9 AM CET (= 12am PST = 1am CST = 3am EST)
- ~3H – Homeworks
  - Hands-on with DBaas Cassandra service AstraDB
  - Learning materials: Cassandra Fundamentals exercices 1 to 8
  - Fill the week 1 test

#2 Design Cassandra Data Models for Full-Stack Applications

- ~2H – Live lessons (twice the same for different timezones)
  - January 12th 6 PM CET (= 9am PST = 11 am CST = 12 pm EST)
  - January 13th 9 AM CET (= 12am PST = 1am CST = 3am EST)
- ~3H – Homeworks
  - Hands-on with DBaas Cassandra service AstraDB
  - Learning materials: Cassandra Fundamentals exercices 9 to 14
  - Learning materials: DS220 Modules 1 to 20 (Table Features)
  - Fill the week 2 test

WEEK 3

January 19th-January 25th

Build a C* Backend WORKSHOP DataStax Developers

WEEK 4

January 26th-February 1st

Build APIs + Microservices WORKSHOP DataStax Developers

#1 Introduction to NoSQL Databases and Apache Cassandra™

- ~2H – Live lessons (same but different timezones)
  - January 5th 6 PM CET (= 9am PST = 11 am CST = 12 pm EST)
  - January 6th 9 AM CET (= 12am PST = 1am CST = 3am EST)
- ~3H – Homeworks
  - Hands-on with DBaas Cassandra service AstraDB
  - Learning materials: Cassandra Fundamentals exercices 1 to 8
  - Fill the week 1 test

#2 Design Cassandra Data Models for Full-Stack Applications

- ~2H – Live lessons (twice the same for different timezones)
  - January 12th 6 PM CET (= 9am PST = 11 am CST = 12 pm EST)
  - January 13th 9 AM CET (= 12am PST = 1am CST = 3am EST)
- ~3H – Homeworks
  - Hands-on with DBaas Cassandra service AstraDB
  - Learning materials: Cassandra Fundamentals exercices 9 to 14
  - Learning materials: DS220 Modules 1 to 20 (Table Features)
  - Fill the week 2 test

#3 Build a Full-Stack Backend with a NoSQL database

- ~2H – Live lessons (same but different timezones)
  - January 19th 6 PM CET (= 9am PST = 11 am CST = 12 pm EST)
  - January 20th 9 AM CET (= 12am PST = 1am CST = 3am EST)
- ~3H – Homeworks
  - Coding with DBaas Cassandra service AstraDB
  - DS220 Modules 21 to 40 (Collections-> Physical Data Model)
  - Fill the week 3 test

WEEK 4

January 26th-February 1st

#1 Introduction to NoSQL Databases and Apache Cassandra™

- ~2H - Live lessons (same but different timezones)
  - January 5th 6 PM CET (= 9am PST = 11 am CST = 12 pm EST)
  - January 6th 9 AM CET (= 12am PST = 1am CST = 3am EST)
- ~3H - Homeworks
  - Hands-on with DBaas Cassandra service AstraDB
  - Learning materials: Cassandra Fundamentals exercices 1 to 8
  - Fill the week 1 test

#2 Design Cassandra Data Models for Full-Stack Applications

- ~2H - Live lessons (twice the same for different timezones)
  - January 12th 6 PM CET (= 9am PST = 11 am CST = 12 pm EST)
  - January 13th 9 AM CET (= 12am PST = 1am CST = 3am EST)
- ~3H - Homeworks
  - Hands-on with DBaas Cassandra service AstraDB
  - Learning materials: Cassandra Fundamentals exercices 9 to 14
  - Learning materials: DS220 Modules 1 to 20 (Table Features)
  - Fill the week 2 test

#3 Build a Full-Stack Backend with a NoSQL database

- ~2H - Live lessons (same but different timezones)
  - January 19th 6 PM CET (= 9am PST = 11 am CST = 12 pm EST)
  - January 20th 9 AM CET (= 12am PST = 1am CST = 3am EST)
- ~3H - Homeworks
  - Coding with DBaas Cassandra service AstraDB
  - DS220 Modules 21 to 40 (Collections-> Physical Data Model)
  - Fill the week 3 test

#4 Build Api and Microservices for Apache Cassandra™

- ~2H - Live lessons (same but different timezones)
  - January 26th 6 PM CET (= 9am PST = 11 am CST = 12 pm EST)
  - January 27th 9 AM CET (= 12am PST = 1am CST = 3am EST)
- ~3H - Homeworks
  - Coding with DBaas Cassandra service AstraDB
  - Learning materials: DS220 Modules 42 to 56
  - Fill the week 4 test

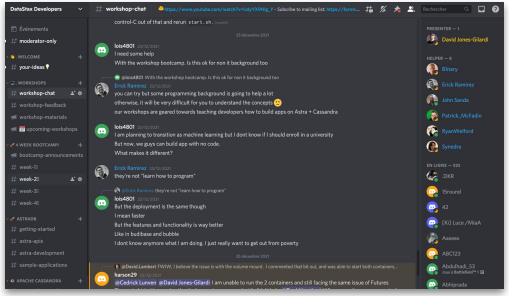
1 Attend one of the 2 LIVE STREAMED workshop (Wednesday or Thursday)

Livestream: youtube.com/DataStaxDevs

YouTube

Twitch

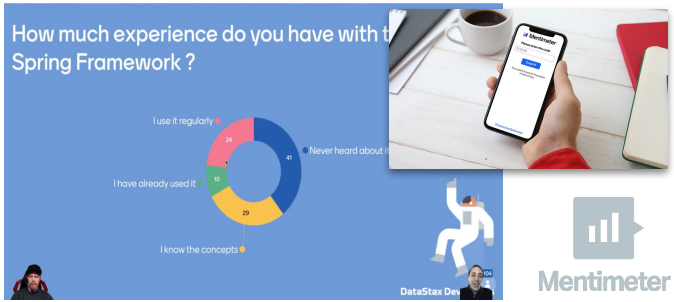
Questions: https://dtsx.io/discord



YouTube

Discord

Games and quizzes: menti.com



Mentimeter

1 Attend the live sessions

1. Attend one of the 2 LIVE STREAMED workshop (Wednesday or Thursday)
2. Complete the workshop labs

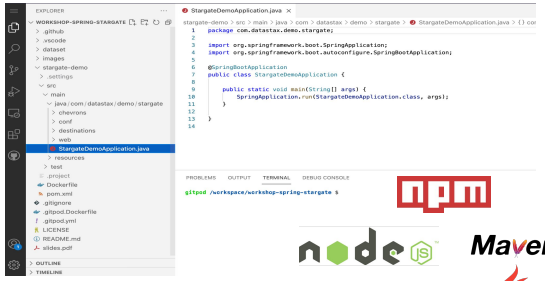
4-week bootcamp Housekeeping

AstraDB astra.datastax.com

Database + GraphQL + PlayGround

DataStax Astra DB

Gitpod IDE: https://gitpod.io



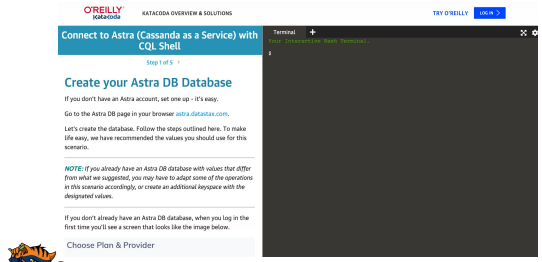
npm

node js

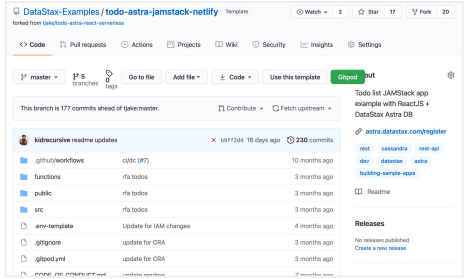
Maven

Gitpod

Katacoda katacoda.com/datastax



Katacoda



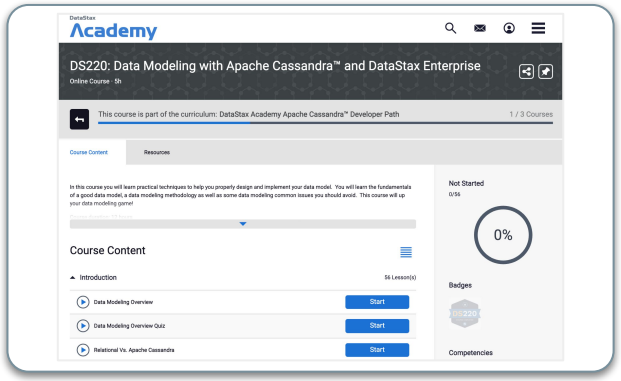
GitHub

2 Complete Workshops Labs

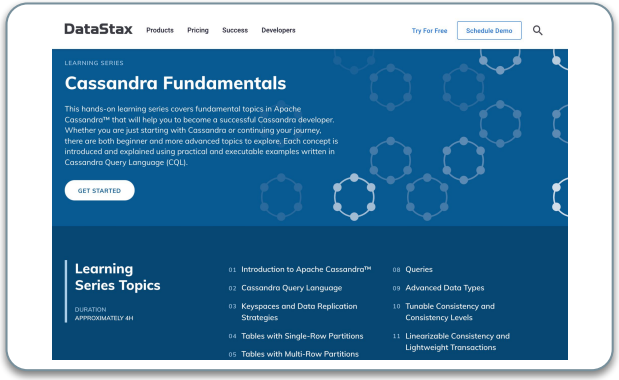
1. Attend one of the 2 LIVE STREAMED workshop (Wednesday or Thursday)
2. Complete the workshop labs
3. Complete the Learning materials

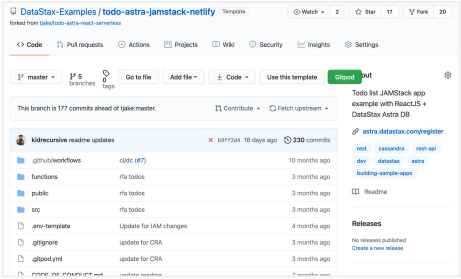
4-week bootcamp Housekeeping

Academy academy.datastax.com



Datastax /dev datastax.com/dev



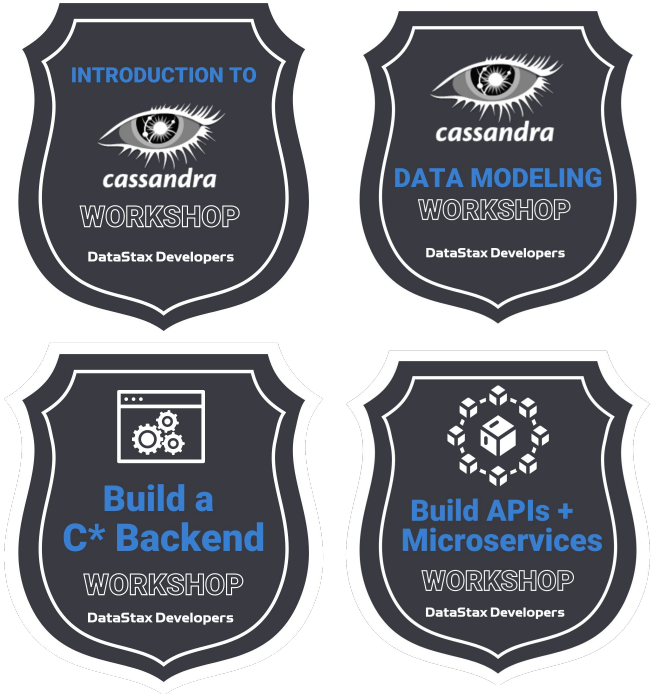


GitHub

3 **Do your homeworks**

1. Attend one of the 2 LIVE STREAMED workshop (Wednesday or Thursday)
2. Complete the workshop labs
3. Complete the Learning materials
4. Pass the weekly test (google form)

4-week bootcamp Housekeeping



Intro to NoSQL Homework

Welcome and thank you! Here you can submit your homework for the datastax developers "Intro to NoSQL" workshop. In case of any questions please contact organisers at https://dtsx.io/aleks or just send an email to aleksandr.volochnev@datastax.com

- Workshop materials: https://github.com/datastaxdevs/workshop-introduction-to-nosql
- Discord chat: https://dtsx.io/discord

cedrick.lunven@datastax.com Switch account

The name and photo associated with your Google account will be recorded when you upload files and submit this form. Only the email you enter is part of your response.

* Required

Email *

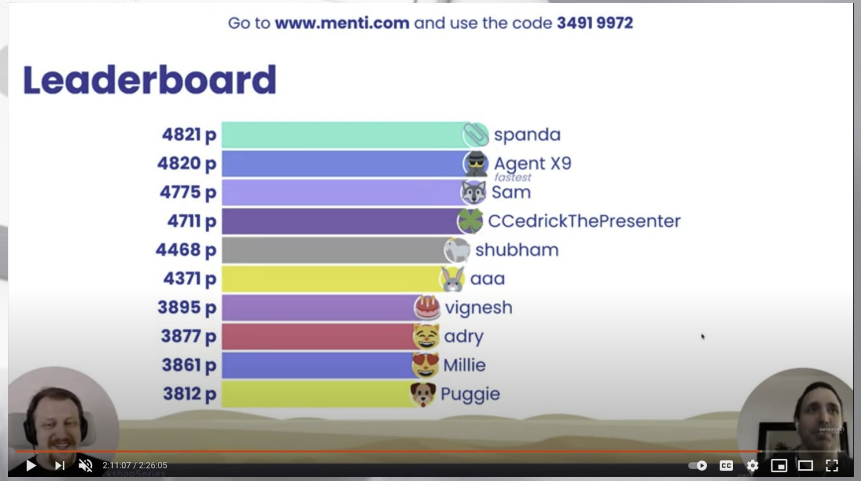
Your email

4 Pass the Weekly Test

menti.com

Apache
CASSANDRA
25
DC COMICS
DATASTAX

Go to www.menti.com and use the code 3491 9972
Inequality predicates are allowed on ...
4
3
5
1
All table columns
Partition key columns
clustering key columns
No inequality predicates are allowed
2:10:19 / 2:26:05
Big paycheck


Go to www.menti.com and use the code 3491 9972
Leaderboard
4821 p spanda
4820 p Agent X9
fastest
4775 p Sam
4711 p CCedrickThePresenter
4468 p shubham
4371 p aaa
3895 p vignesh
3877 p adry
3861 p Millie
3812 p Puggie
2:11:07 / 2:26:05

01

Weeks 1 &2 Reminders
HouseKeeping

02

The Todo Application
Context and Demo

03

Connectivity to Cassandra
Introduction to Drivers

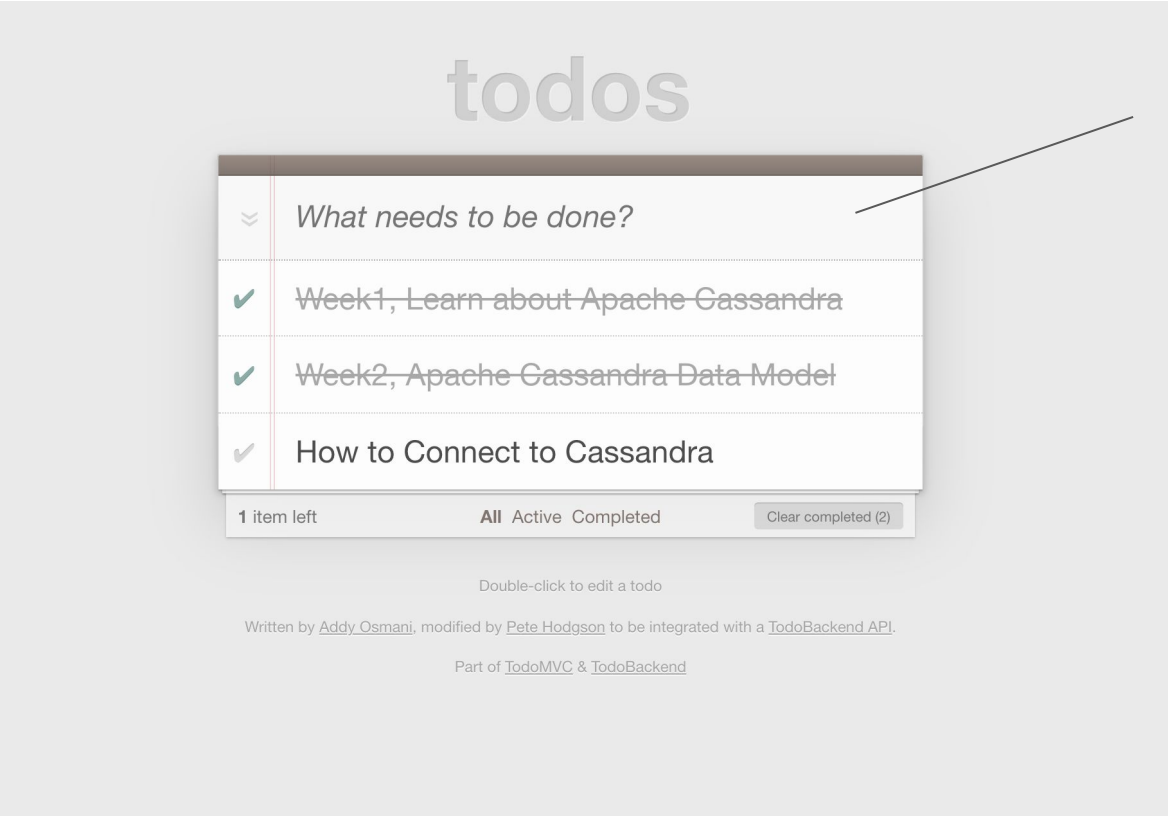
04

Execute Statements
CRUD

05

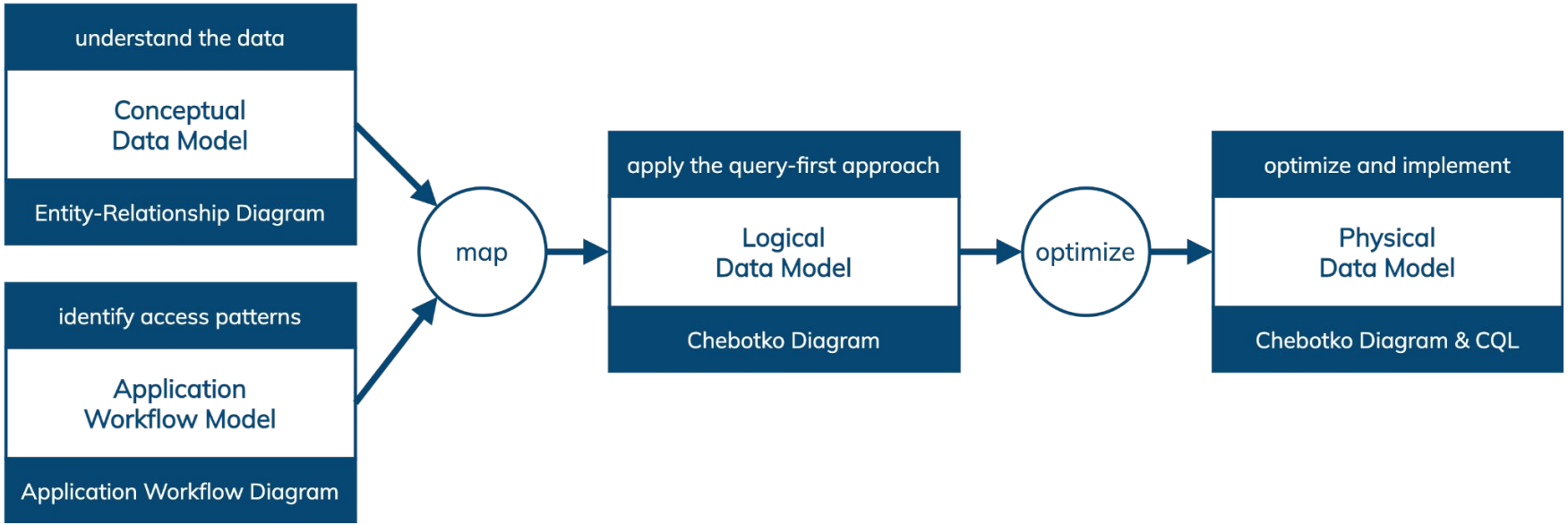
What's next?
Quiz, Homework, Next week

Agenda



- user id
- item id
- title
- completed
- order

Demo TodoApplication



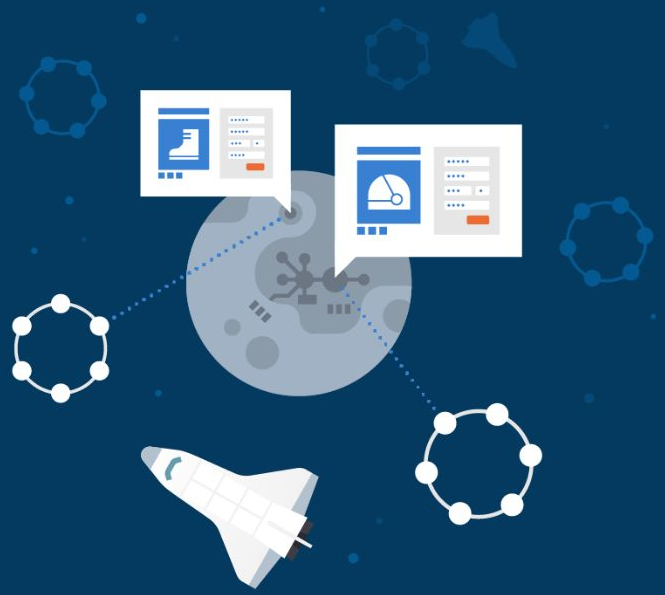
Data Modelling Methodology

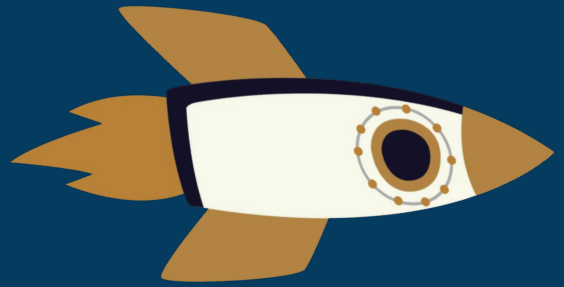
todoitems		
user_id	**TEXT**	**K**
item_id	TIMEUUID	C↑
completed	BOOLEAN	
title	TEXT	
offset	INT	

```
CREATE TABLE todos.todoitems (
    user_id     text,
    item_id     timeuuid,
    completed   boolean,
    title       text,
    offset      int,
    PRIMARY KEY ((user_id),item_id)
);
```

```
token@cqlsh:native_java> select * from todoitems where user_id='john';

 user_id | item_id                              | completed | offset | title
---------+--------------------------------------+-----------+--------+---------------------
    john | 11111111-5cff-11ec-be16-1fedb0dfd057 |      True |   null |        Walk the dog
    john | 22222222-5cff-11ec-be16-1fedb0dfd057 |     False |   null | Have lunch tomorrow
    john | 708520d0-76af-11ec-9bee-6f21c69b6a43 |     False |      0 |        Sample task1

(3 rows)
token@cqlsh:native_java>
```

TodoApplication DataModel





Hands-on

Setup your Database

#1 Connect to your Astra instance (Create/UnHibernate your Astra instance as needed)

01

Weeks 1 &2 Reminders
HouseKeeping

02

The Todo Application
Context and Demo

03

Connectivity to Cassandra
Introduction to Drivers

04

Execute Statements
CRUD

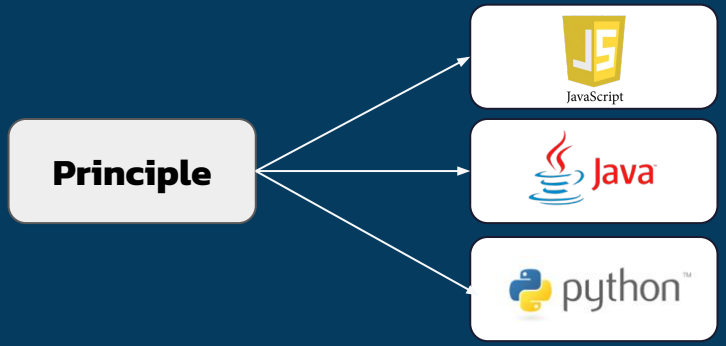
05

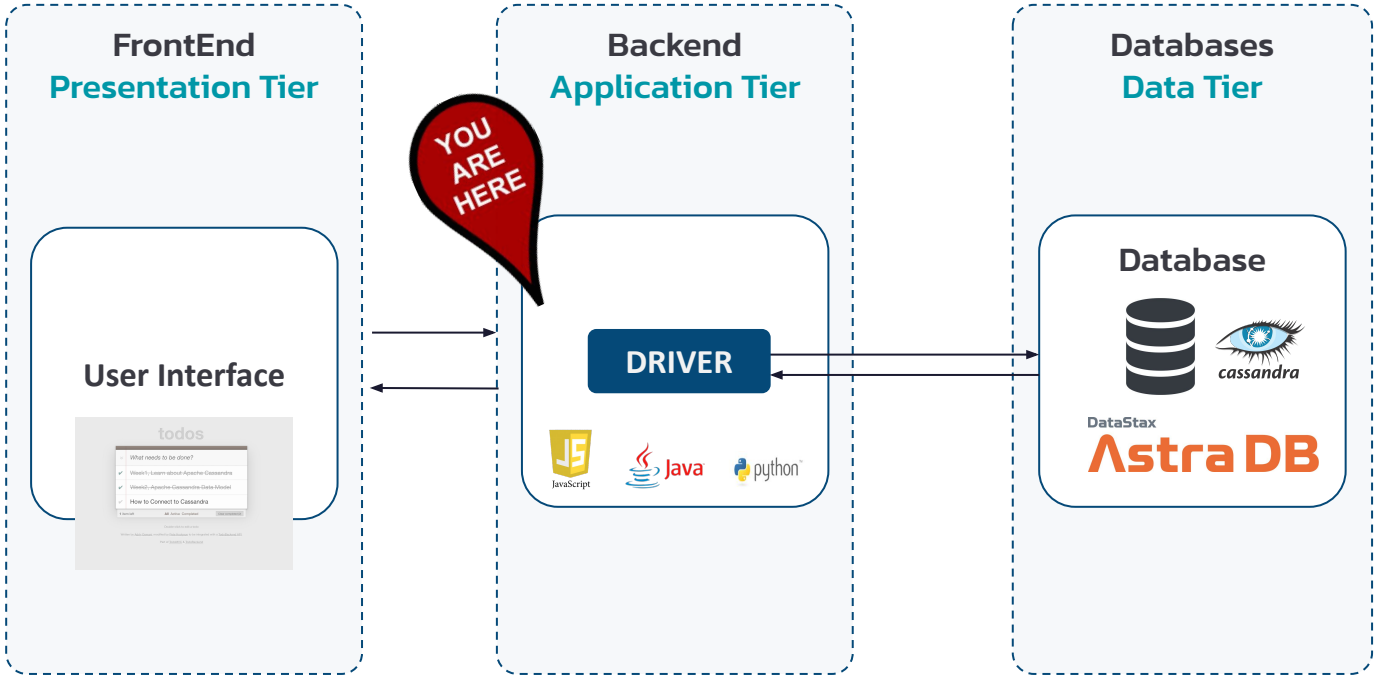
What's next?
Quiz, Homework, Next week

Agenda

Having a scalable database is great

Learn how to use it and implement your own applications with 3 languages it is even better.





Application Architecture

Connectivity

- ★ Token & Datacenter Aware
- ★ Load Balancing Policies
- ★ Retry Policies
- ★ Reconnection Policies
- ★ Connection Pooling
- ★ Health Checks
- ★ Authentication | Authorization
- ★ SSL

Query

- ★ CQL Support
- ★ Schema Management
- ★ Sync/Async/Reactive API
- ★ Query Builder
- ★ Compression
- ★ Paging

Parsing Results

- ★ Lazy Load
- ★ Object Mapper
- ★ Spring Support
- ★ Paging

Drivers

Maven

```
<dependency>
    <groupId>com.datastax.oss</groupId>
    <artifactId>java-driver-core</artifactId>
    <version>4.13.1</version>
</dependency>
```

Java

4.6.3 npm

```
npm install cassandra-driver
```

```
{
  "dependencies": {
      "cassandra-driver": "^4.6.3"
  }
}
```

JavaScript

3.24.0 python Package Index

```
pip install cassandra-driver
```

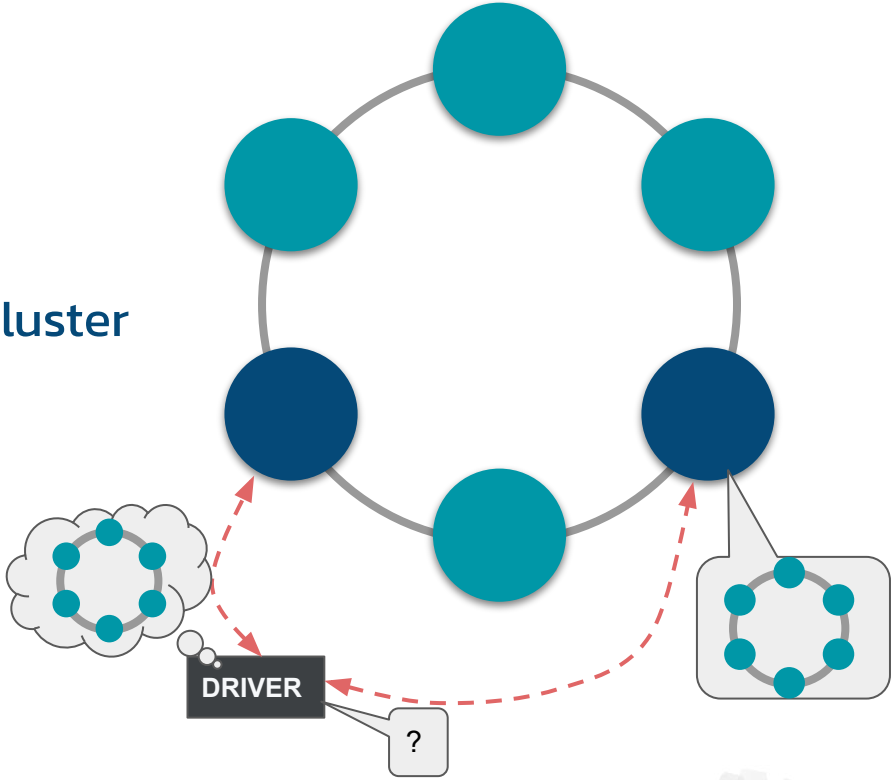
python

nuget v3.15.0

```
Install-Package CassandraCSharpDriver -Version 3.15.0
```

Installing the drivers

- Only one necessary
- Unless that node is down
- Better ~3 nodes for resilience
- From there, drivers discover whole cluster



Apache Cassandra™ Contact Point

```java
CqlSession cqlSession = CqlSession.builder()
 .addContactPoint(new InetSocketAddress("127.0.0.1", 9042))
 .withKeyspace("killrvideo")
 .withLocalDatacenter("dc1")
 .withAuthCredentials("U","P")
 .build();
```

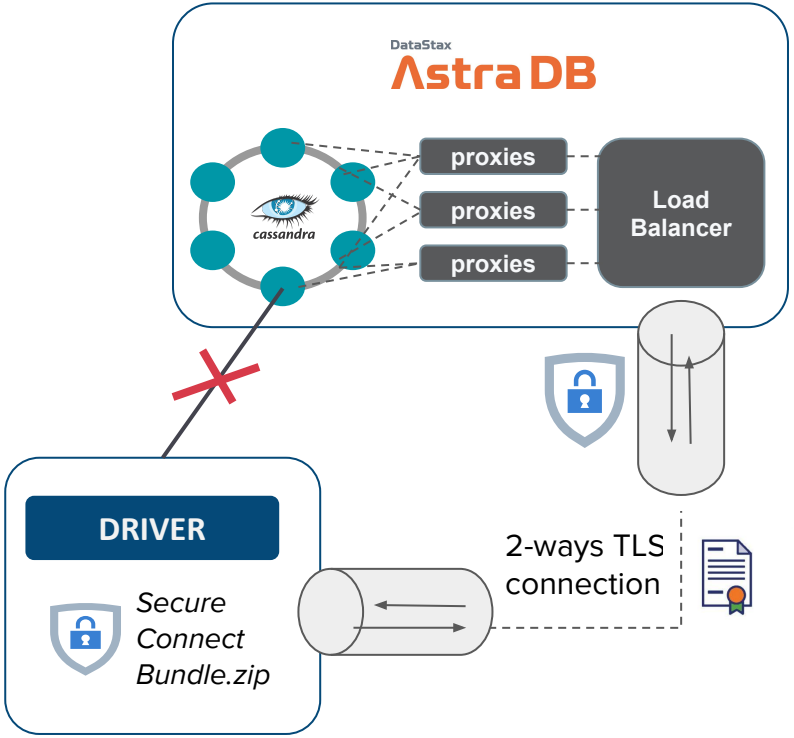
```node
const client = new cassandra.Client({
  contactPoints: ['127.0.0.1'],
  localDataCenter: 'dc1',
  keyspace: 'killrvideo',
  credentials: { username: 'U', password: 'P' }
});
```

```python
uth_provider = PlainTextAuthProvider(
   username='U', password='P')
cluster = Cluster(['127.0.0.1'],
  auth_provider=auth_provider, protocol_version=2)
session = cluster.connect('todos')
```

```dotnet
Cluster cluster = Cluster.Builder()
.AddContactPoint("127.0.0.1")
.WithCredentials("U", "P")
.Build();
session = cluster.Connect("todos");
```

Create Session/Client with Contact Points

- Secured connection over HTTP required.
- A *secureConnectBundle* ZIP is expected
- Your username is clientId
- Your Password is clientSecret
- No Single Point of failure (SPOF)



Contact Points with ASTRA

```java
CqlSession cqlSession = CqlSession.builder()
 .withCloudSecureConnectBundle(Paths.get("secure.zip"))
 .withAuthCredentials("U","P")
 .withKeyspace("todos")
 .build();
```

```node
const client = new cassandra.Client({
  cloud: { secureConnectBundle: 'secure.zip' },
  credentials: { username: 'u', password: 'p' }
});
```

```python
auth_provider = PlainTextAuthProvider(
   username='U', password='P')
cluster = Cluster(
   Cloud ={ Secure_connect_bundle: 'secure.zip'},
   auth_provider=auth_provider, protocol_version=2)
session= cluster.connect('todos')
```

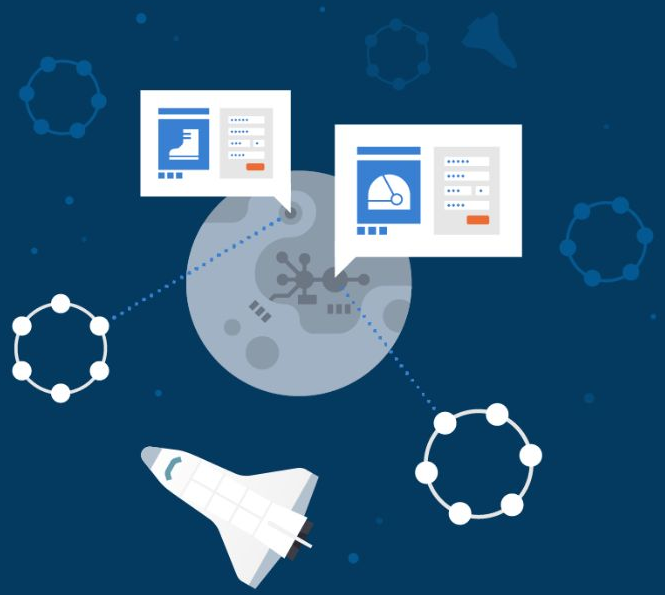
```dotnet
var cluster = Cluster.Builder()
 .WithCloudSecureConnectionBundle("secure.zip")
 .WithCredentials("u", "p")
 .Build();
var session = cluster.Connect("todos");
```

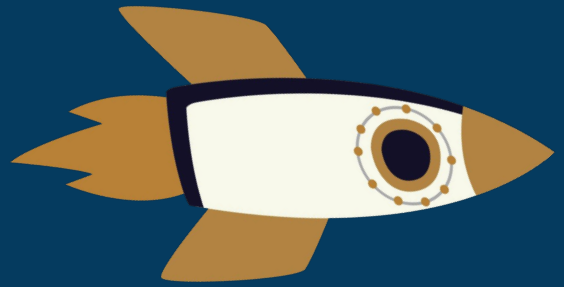
Create Session/Client with ASTRA

- Stateful object handling communications with each node
- Should be unique in the Application (*Singleton*)
- Should be closed at application shutdown (*shutdown hook*) in order to free opened TCP sockets (*stateful*)

```
Java:       cqlSession.close();
Python:     session.shutdown();
Node:       client.shutdown();
CSharp:     IDisposable
```

IMPORTANT about Client/Session





Hands-on

Setup your Application

#2 Create your Token
#3 Launch Gitpod
#4 Know your Gitpod
#5 Setup your Application
#6 Connectivity

01

Weeks 1 &2 Reminders
HouseKeeping

02

The Todo Application
Context and Demo

03

Connectivity to Cassandra
Introduction to Drivers

04

Execute Statements
CRUD

05

What's next?
Quiz, Homework, Next week

Agenda

First job of CqlSession is to execute queries using the execute method.

```
cqlSession.execute("SELECT * FROM todos.todoitems");
```

Statement

```java
Statement statement = ...

// (1) Explicit SimpleStatement Definition
SimpleStatement.newInstance("select * from t1 where c1 = 5");

// (2) Externalize Parameters (no name)
SimpleStatement.builder("select * from t1 where c1 = ?")
               .addPositionalValue(5);

// (3) Externalize Parameters (name)
SimpleStatement.builder("select * from t1 where c1 = :myVal")
               .addNamedValue("myVal”, 5);

cqlSession.execute(statement);
```

Parameters in Statements

```java
Statement statement = SimpleStatement
  .builder("select * from t1 where c1 = ?")
  .addPositionalValue(5)
session.execute(stmt);
```

```node
client.execute('select * from t1 where c1 = ?', [5]);
```

```python
session.execute("select * from t1 where c1 = %s", 5);
```

```dotnet
var statement = new SimpleStatement("select * from t1
  where c1 = ?"", 5);
session.Execute(statement);
```

Execute Statements

```java
PreparedStatement prepared = session.prepare(
    "select * from t1 where c1 = ?");
BoundStatement bound = prepared.bind("5");
session.execute(bound);
```

```node
const query = 'select * from t1 where c1 =?;';
await client.execute(query, [5], { prepare: true })
```

```python
prepared = session.prepare(
    "select * from t1 where c1=?")
result = session.execute(prepared, [5])
```

```dotnet
var ps = session.Prepare(
      "select * from t1 where c1 = ?");
var statement = ps.Bind("5");
session.Execute(statement);
```

Prepared Statements

- ResultSet is the object returned for executing query. It contains ROWS (data) and EXECUTION INFO.
- ResultSet is iterable and as such you can navigate from row to row.
- Results are always paged for you (avoiding memory and response time issues)

```
ResultSet rs = cqlSession.execute(myStatement);

ExecutionInfo info = rs.getExecutionInfo();
int executionTime = info.getQueryTrace().getDurationMicros();

Iterator<Row> iterRow = rs.iterator();
int itemsFirstCall = rs.getAvailableWithoutFetching();
```

ResultSet and Rows

```java
ResultSet resultSet = session.execute("select * from
  t1 where c1 = 5");

Row row = resultSet.one();
```

```node
const result = await client.execute("select * from
  t1 where c1 = 5");

const row = result.first();
```

```python
#ResultSet

result = session.execute("select * from t1 where c1 = 5")

#Row

result.one()
```

```dotnet
var resultSet = session.Execute("select * from t1 where
  c1 = 5");

var row = resultSet.First();
```

ResultSet

```java
Row row = resultSet.one();
Boolean isUserNameNull = row.isNull("userName");
var value = row.getString("userName");
```

```node
row = result.first();
//false when null
Boolean(row.userName);
value = row.userName.toString();
```

```python
row = resultSet.one();
#false if None
bool(row.userName)
value = str(row.userName)
```

```dotnet
var row = resultSet.First();
Boolean isUserNameNull = row.IsNull("userName");
var value = row.GetValue<string>("userName");
```

Parsing Rows

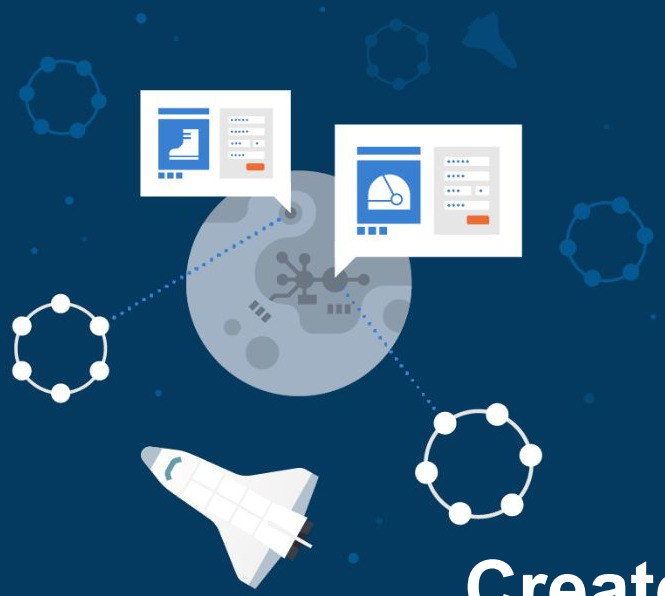
```java
@Dao
public interface SpacecraftJourneyDao {
   PagingIterable<SpacecraftJourneyCatalog> findAll();
}
 @Mapper
 public interface SpacecraftMapper {
 @DaoFactory SpacecraftJDao dao(@DaoKeyspace CqlIdentifier keyspace);
}
```

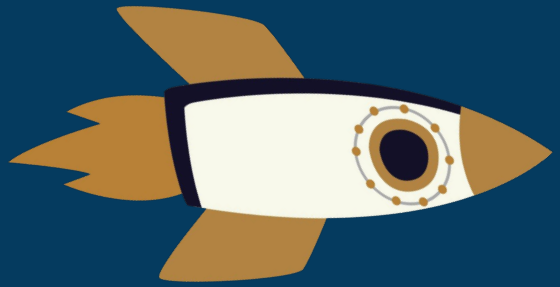
```node
const mapper = new cassandra.mapping.Mapper(client, {
models: {
 'Journey': { tables: [‘spacecraft_journey_catalog’],
    keyspace: 'killrvideo', columns: { 'journey_id': 'journeyId'},
    mappings: new UnderscoreCqlToCamelCaseMappings()
} } });
const journeyMapper = mapper.forModel('Journey');
journeyMapper.insert({ videoId });
```

```python
from cassandra.cqlengine import columns
from cassandra.cqlengine.models import Model
from cassandra.cqlengine.management import sync_table
class Journey(Model):
    journey_id = columns.UUID(primary_key=True)
    spacecraft_name = columns.Text()
sync_table(Journey)
j = Journey.create(journey_id=...)
```

```dotnet
MappingConfiguration.Global.Define(new Map<Journey>()
  .TableName("spacecraft_journey_catalog")
  .PartitionKey(k => j.journeyId, )
  .Column(u => u.journeyId, cm => cm.WithName("journey_id")
  .Column(u => u.spacecraftName, cm => cm.WithName("spacecraft_name")));

IEnumerable<Journey> journey = mapper.Fetch<Journey>("WHERE spacecraft_name
   = ? AND journey_id =?", a, b);
```

Object Mapping





Hands-on

Create, Read, Update, Delete (CRUD)

Exercises #6a to 6f

01

Weeks 1 &2 Reminders
HouseKeeping

02

The Todo Application
Context and Demo

03

Connectivity to Cassandra
Introduction to Drivers

04

Execute Statements
CRUD

05

What's next?
Quiz, Homework, Next week

Agenda

Homework

!homework

```
FXHtFByhHAjqNjdcLgZNYEll@cqlsh:todos> select * from todoitems;

 user_id | item_id                              | completed | title
---------+--------------------------------------+-----------+----------------------
    mary | 33333333-5cff-11ec-be16-1fedb0dfd057 |      True | Attend the workshop
    john | 11111111-5cff-11ec-be16-1fedb0dfd057 |      True |        Walk the dog
    john | 22222222-5cff-11ec-be16-1fedb0dfd057 |     False | Have lunch tomorrow

(3 rows)
FXHtFByhHAjqNjdcLgZNYEll@cqlsh:todos>
```

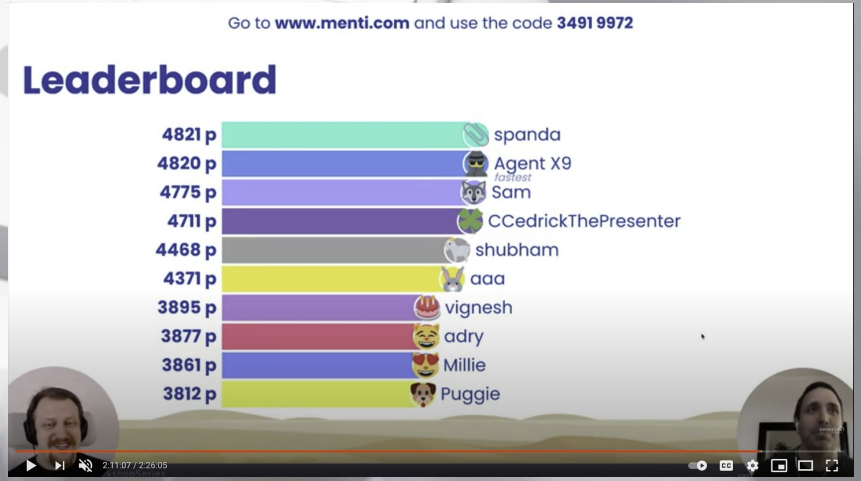
```
gitpod /workspace/bootcamp-fullstack-apps-with-cassandra/week3-app-development/python $ pyt
hon Ex6g_Transfer_Todos.py
========================================
Start exercise
mary
INSERT INTO todoitems (user_id, item_id, completed, title)
1ec-be16-1fedb0dfd057, false , 'Attend the workshop');
john
INSERT INTO todoitems (user_id, item_id, completed, title)
1ec-be16-1fedb0dfd057, false , 'Walk the dog');
john
INSERT INTO todoitems (user_id, item_id, completed, title)
1ec-be16-1fedb0dfd057, false , 'Have lunch tomorrow');
Success
========================================
gitpod /workspace/bootcamp-fullstack-apps-with-cassandra/w
```

```
FXHtFByhHAjqNjdcLgZNYEll@cqlsh:todos> select * from todoitems;

 user_id | item_id                              | completed | title
---------+--------------------------------------+-----------+----------------------
    rags | 11111111-5cff-11ec-be16-1fedb0dfd057 |     False |        Walk the dog
    rags | 22222222-5cff-11ec-be16-1fedb0dfd057 |     False | Have lunch tomorrow
    rags | 33333333-5cff-11ec-be16-1fedb0dfd057 |     False | Attend the workshop
    mary | 33333333-5cff-11ec-be16-1fedb0dfd057 |      True | Attend the workshop
    john | 11111111-5cff-11ec-be16-1fedb0dfd057 |      True |        Walk the dog
    john | 22222222-5cff-11ec-be16-1fedb0dfd057 |     False | Have lunch tomorrow

(6 rows)
FXHtFByhHAjqNjdcLgZNYEll@cqlsh:todos>
```

menti.com

Apache
CASSANDRA
25
DC COMICS
DATASTAX

Go to www.menti.com and use the code 3491 9972
Inequality predicates are allowed on ...
4
3
5
1
All table columns
Partition key columns
clustering key columns
No inequality predicates are allowed
2:10:19 / 2:26:05
Big paycheck


Go to www.menti.com and use the code 3491 9972
Leaderboard
4821 p spanda
4820 p Agent X9
fastest
4775 p Sam
4711 p CCedrickThePresenter
4468 p shubham
4371 p aaa
3895 p vignesh
3877 p adry
3861 p Millie
3812 p Puggie
2:11:07 / 2:26:05

SWAG WINNERS 🎉

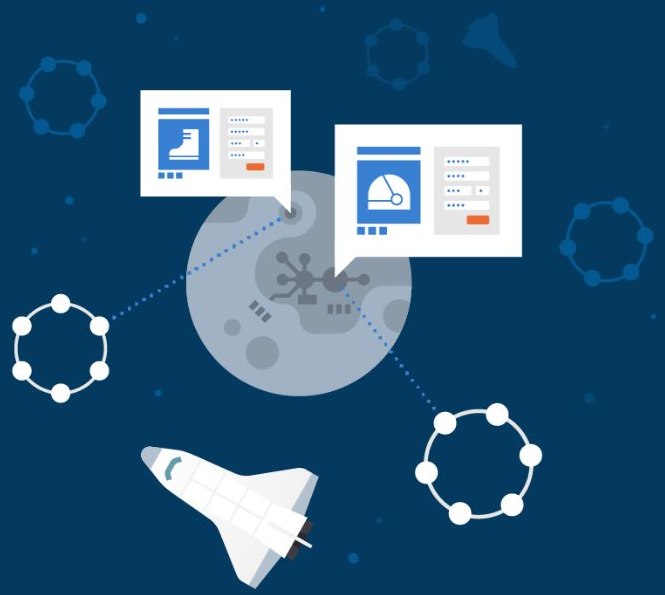
Congratulations to 1st, 2nd and 3rd place on the Menti quiz!

To claim your prize, please send an email to:

gary.harvey@datastax.com

**** Include a screenshot of your Menti screen**

Next Week



DataStax
Developers

Build Microservices on Apache Cassandra with APIs

26 January, 2022

#NoSQL, #Apache Cassandra, #APIs, #DataModeling #Certification

Java JS

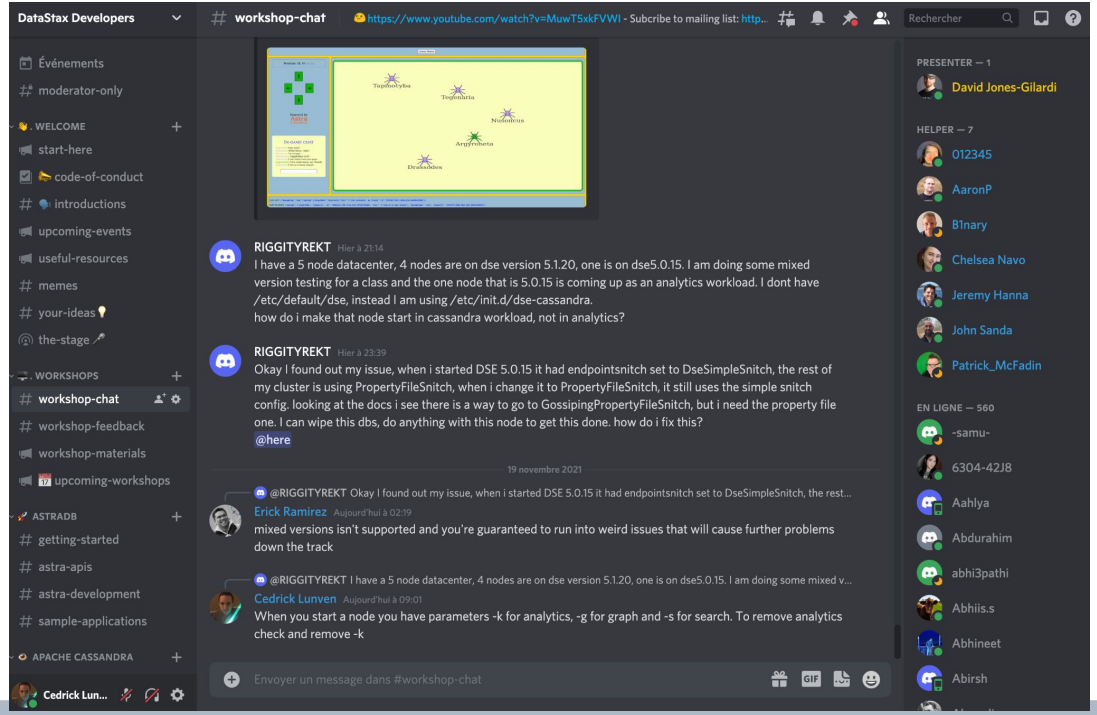
WED JAN 26 2022

Bootcamp: Build Microservices on Apache Cassandra™ with APIs

Register Now

Join our 18k Discord Community

DataStax Developers



!discord

dtsx.io/discord

Discord Community

Subscribe

to NoSQL

ds •
ed 1 week ago

Crash Course | Introduction to Cassandra for Developers

331 views • 1 week ago

Introduction to NoSQL Databases

3.5K views •
Streamed 1 week ago

Introduction to NoSQL Databases

10K views •
Streamed 1 week ago

Build your own NETFLIX clone!

4K views •
Streamed 2 weeks ago

Build your own NETFLIX clone!

7.4K views •
Streamed 2 weeks ago

Astra Streaming Demo

177 views • 2 weeks ago

Kubernetes Ingress Management with Traefik...

496 views •
Streamed 2 weeks ago

Build your own TikTok clone!

1.9K views •
Streamed 3 weeks ago

Build your own TikTok Clone!

4K views •
Streamed 3 weeks ago

How to use the Connect Driver in Astra DB

113 views • 4 weeks ago

How to use the CQL Console in Astra DB

39 views • 4 weeks ago

How to create an Authentication Token in...

37 views • 4 weeks ago

How to use the Data Loader in Astra DB

62 views • 4 weeks ago

Astra DB Sample App Gallery

36 views • 4 weeks ago

How to use Secure Connect in Astra DB

42 views • 4 weeks ago

Cassandra Day India: CL Room (Workshops)

2.4K views •
Streamed 4 weeks ago

Cassandra Da
Room (Talk

1.3K view
Stream

Subscribe

DataStax Developers

Thank You!