

# Tutoring coordinator

## Objective

To gather the number of lessons each tutor made with every of their tutees in a given month and send invoices to tutees.

## Screencast

<https://drive.google.com/file/d/1vDWckcgL2zt2vY7ldep0vcpWn4VnLh8L/view?usp=sharing>

## Changes to make in the code

In the tutoring-database project in *CoordinatorApplication* class change attribute *yourEmail* from 28 line to your email address.

In the tutoring-invoices project in *GlobalProjectProperties* class change attributes in the following way:

- *filePath* - an absolute path to the folder *tutoring-coordinator/tutoring-invoices/tmp*
- *tuteesSampleEmail* - your personal email address

## Launching

We need 5 terminals for Zookeeper, Apache Kafka, and 3 Microservices

In folder *tutoring-coordinator/kafka/kafka\_2.13-2.8.0* run 2 terminals. In the first one type:

```
bin/zookeeper-server-start.sh config/zookeeper.properties
```

to launch Zookeeper. In the second type:

```
bin/kafka-server-start.sh config/server.properties
```

to launch Kafka. The beneath table shows us how to launch consecutive microservices. Warning: order matters!

Service name	Folder	Command
Reminder	tutoring-coordinator/tutoring-reminder	gradle bR
Database	tutoring-coordinator/tutoring-database	SERVER_PORT=8082 gradle bR
Invoices	tutoring-coordinator/tutoring-invoices	SERVER_PORT=8081 gradle bR

## Microservices

Reminder

- Gets triggered after run and every month after that
- Requests data from database

- Sends emails to tutors requesting them to update how many lessons they had with every student during last month

#### Database

- Calmly waits for Apache Kafka requests
- Answers the requests
- when information about all #lessons for a student are updated, triggers invoice sender

#### Invoices

- Gets GET requests about the number of lessons every tutor had with their students
- Updates data in database
- Gets informed when all #lessons for a student are updated
- Sends email with invoice to the student

### Architecture

Apache Camel Email -- Tutoring reminder -- Database -- Invoice Sender -- Apache Camel Email

### Program flow

1. [Reminder] Starts with trigger
2. [Reminder] Requests data about all tutorings from database
3. [Database] Answers request
4. [Reminder] Sends emails to tutors requesting them to update how many lessons they had with every student during last month. The email contains links where a person should put number of lessons with a corresponding student
5. [Manual] Tutor uses a link and states how many hours of lessons he made
6. [Invoices] Receives GET request and sends update to database
7. [Database] Updates information about #lessons made by Tutor for Tutee this month. If information about lessons for the tutee in this month becomes complete, sends request to Invoice
8. [Invoices] Generates invoices and sends emails to given students