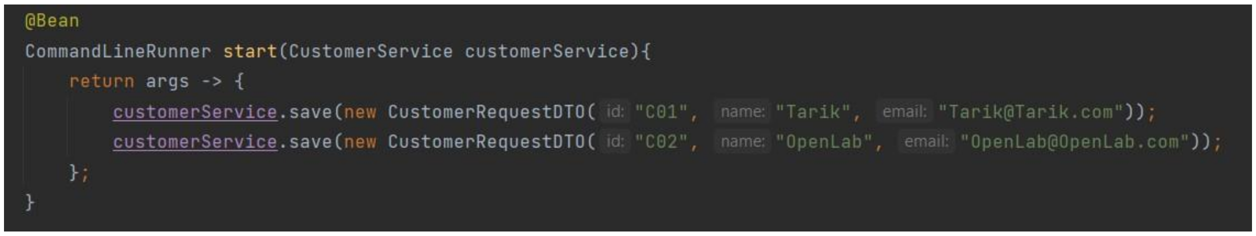
Tarik FERTAHI
II-BDCC

Année universitaire : 2022/2023

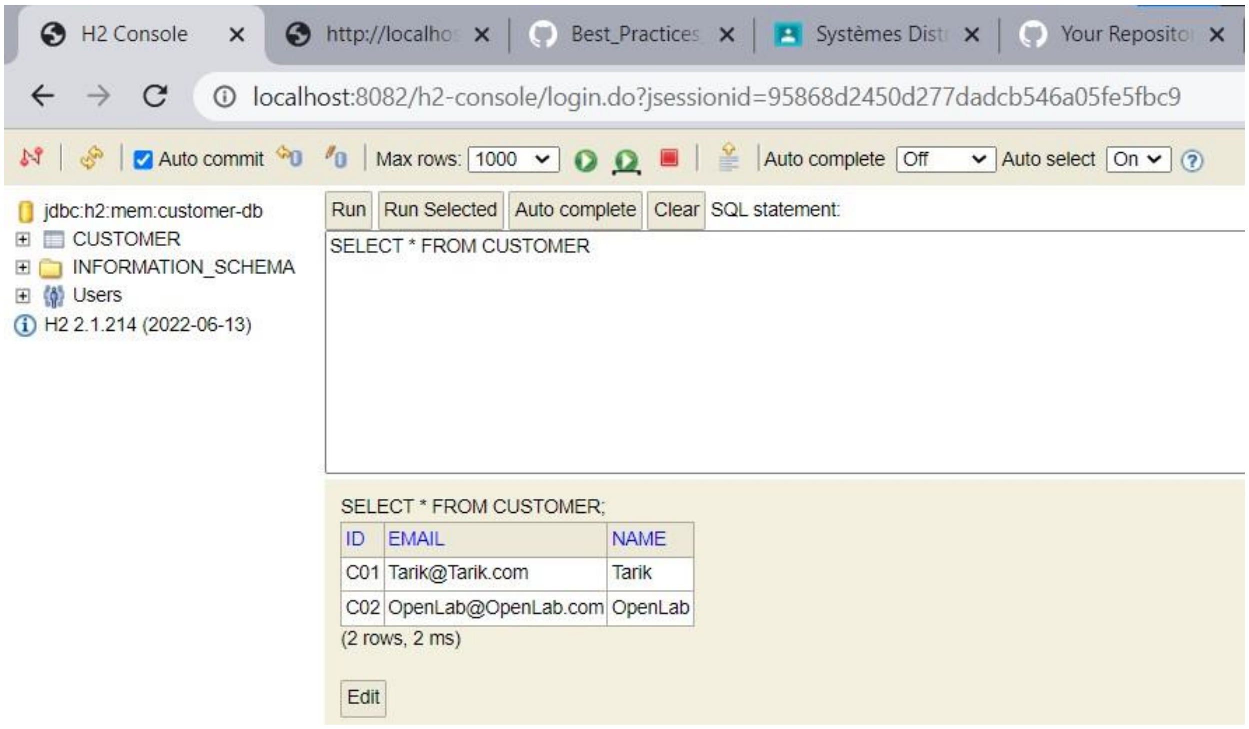
Pour le Micro-Service « **CUSTOMER-SERVICE** »

La création de deux Customers



```
@Bean
CommandLineRunner start(CustomerService customerService){
    return args -> {
        customerService.save(new CustomerRequestDTO( id: "C01",  name: "Tarik",  email: "Tarik@Tarik.com"));
        customerService.save(new CustomerRequestDTO( id: "C02",  name: "OpenLab",  email: "OpenLab@OpenLab.com"));
    };
}
```

Vérification des clients au niveau de la base de données



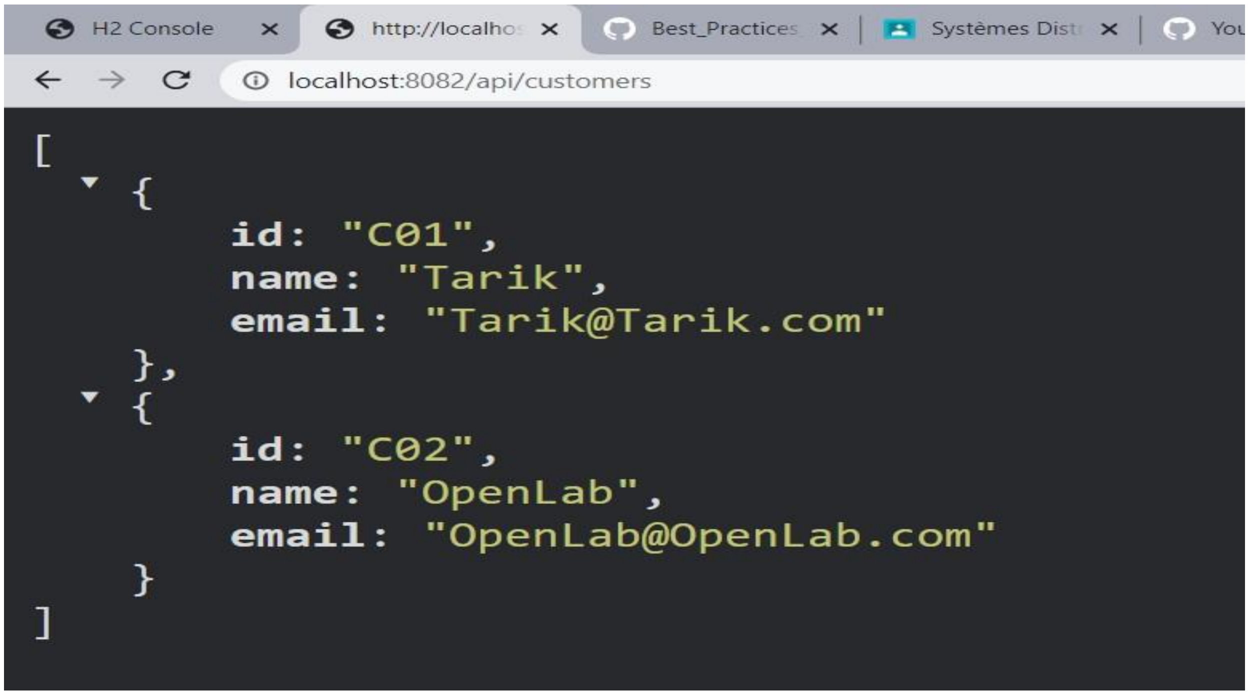
```
SELECT * FROM CUSTOMER
```

SELECT * FROM CUSTOMER;

ID	EMAIL	NAME
C01	Tarik@Tarik.com	Tarik
C02	OpenLab@OpenLab.com	OpenLab

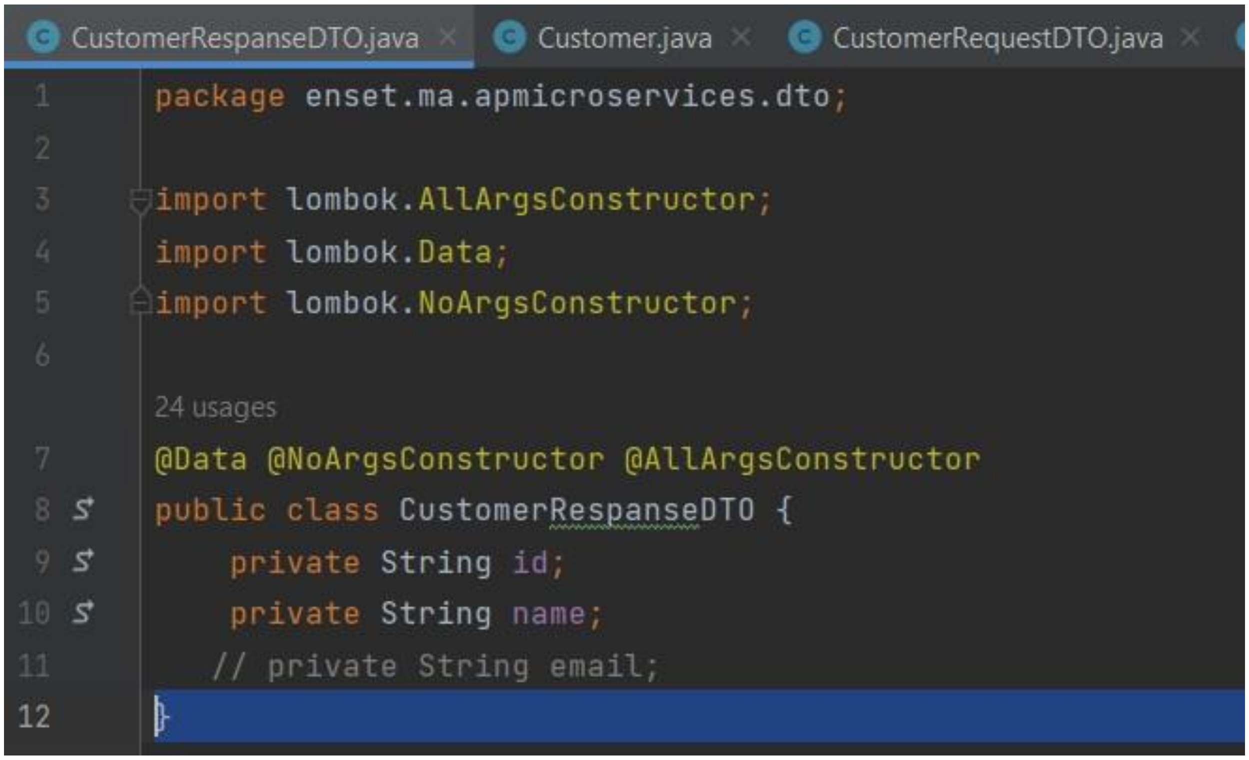
(2 rows, 2 ms)

Affichage des clients

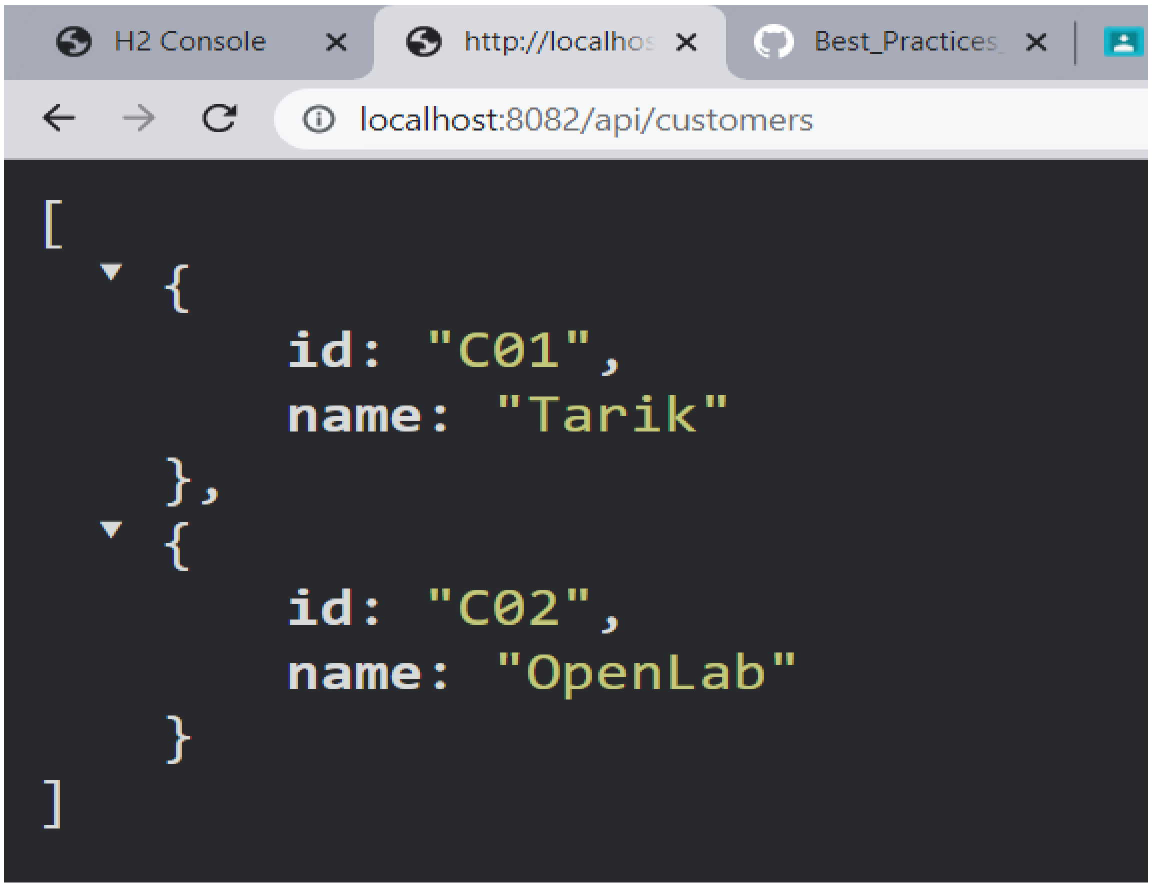


```
[
  {
      id: "C01",
      name: "Tarik",
      email: "Tarik@Tarik.com"
  },
  {
      id: "C02",
      name: "OpenLab",
      email: "OpenLab@OpenLab.com"
  }
]
```

Grace au DTOs on peut ignorer l’attribut email dans la partie UI

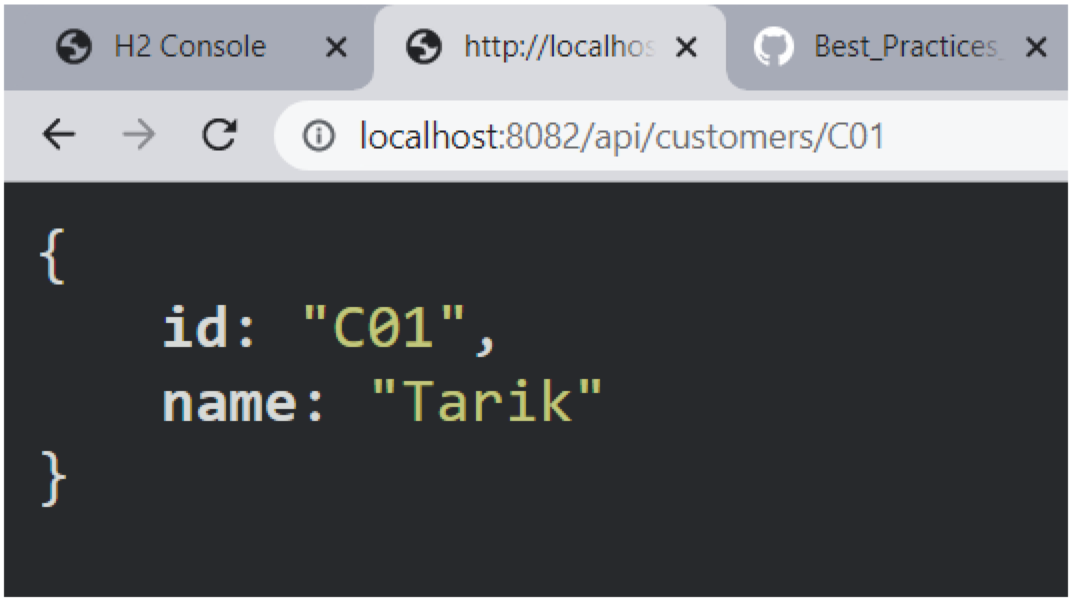


```
package enset.ma.apmicroservices.dto;

import lombok.AllArgsConstructor;
import lombok.Data;
import lombok.NoArgsConstructor;

@Data @NoArgsConstructor @AllArgsConstructor
public class CustomerRespanseDTO {
    private String id;
    private String name;
   // private String email;
}
```



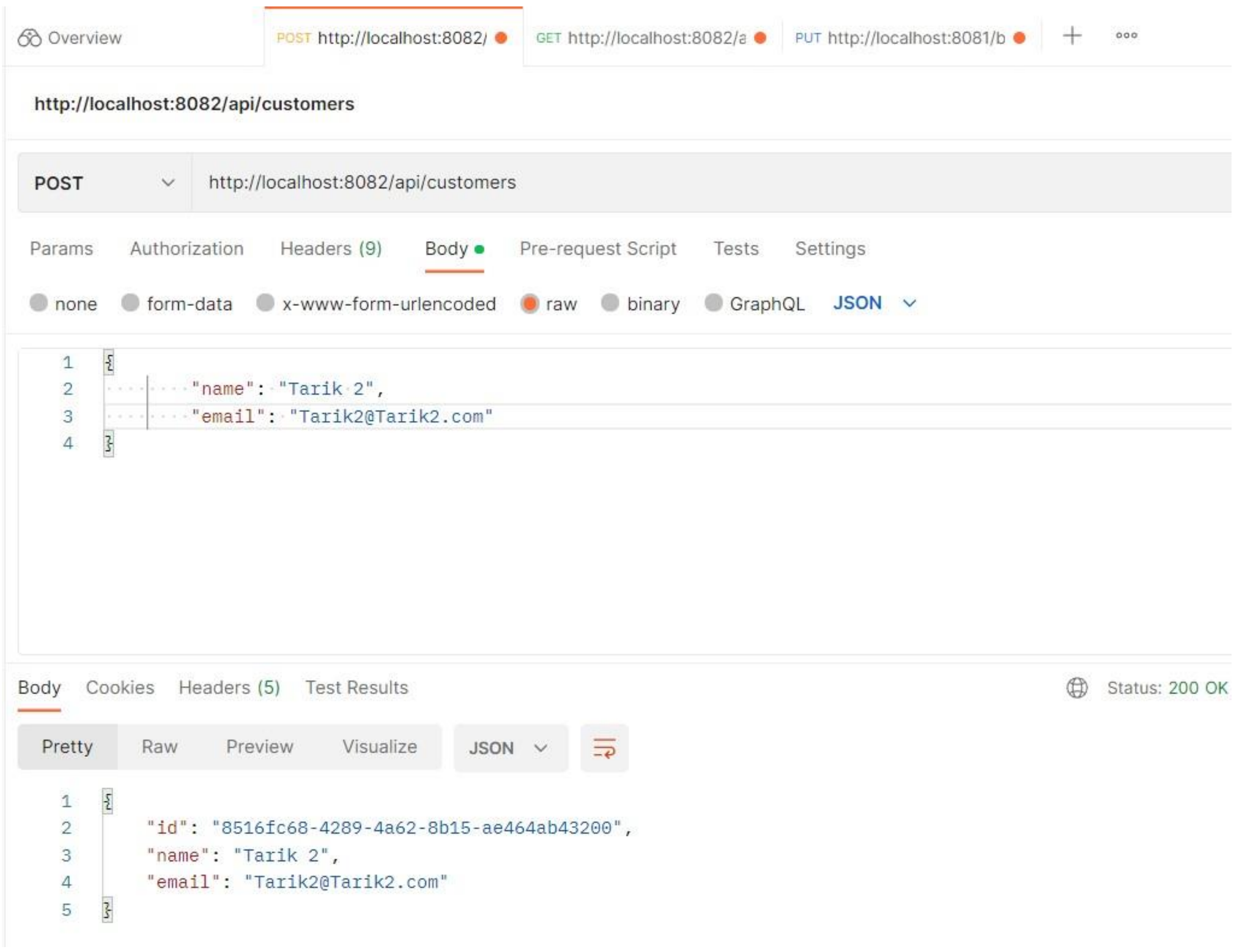
```
[
  {
    id: "C01",
    name: "Tarik"
  },
  {
    id: "C02",
    name: "OpenLab"
  }
]
```

Consulter un client par ID



```
{
  id: "C01",
  name: "Tarik"
}
```

Test de la méthode Save



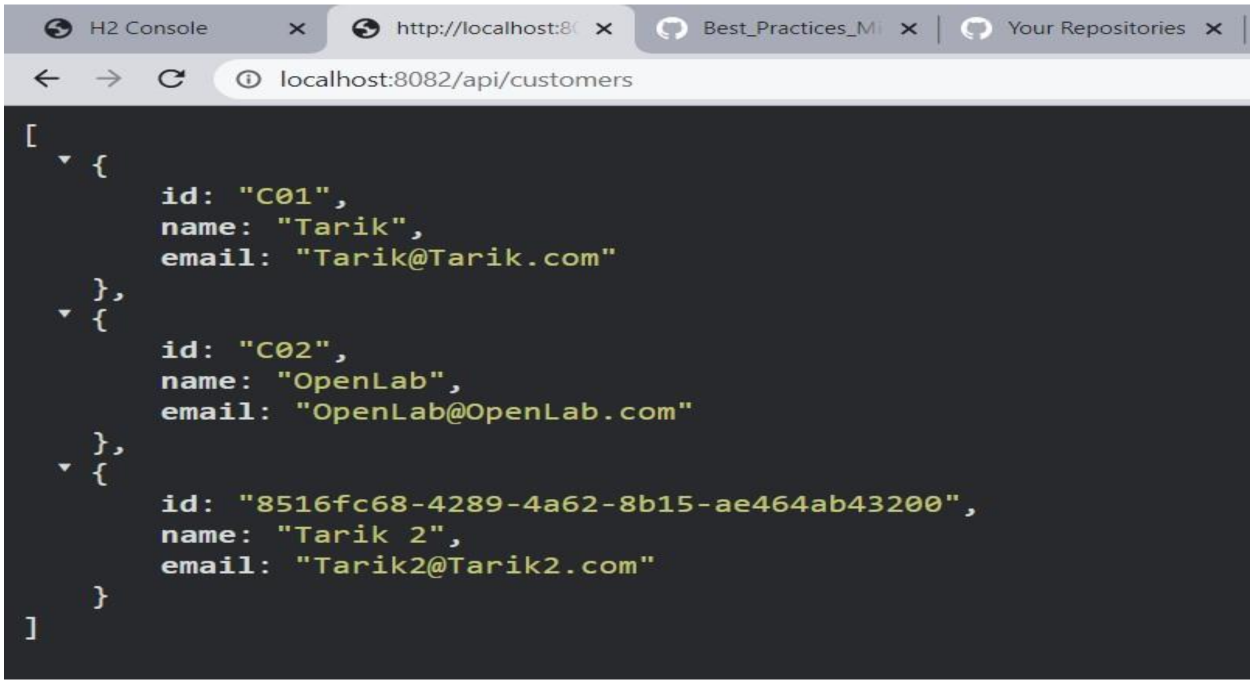
http://localhost:8082/api/customers

POST http://localhost:8082/api/customers

```
{
    "name": "Tarik 2",
    "email": "Tarik2@Tarik2.com"
}
```

Status: 200 OK

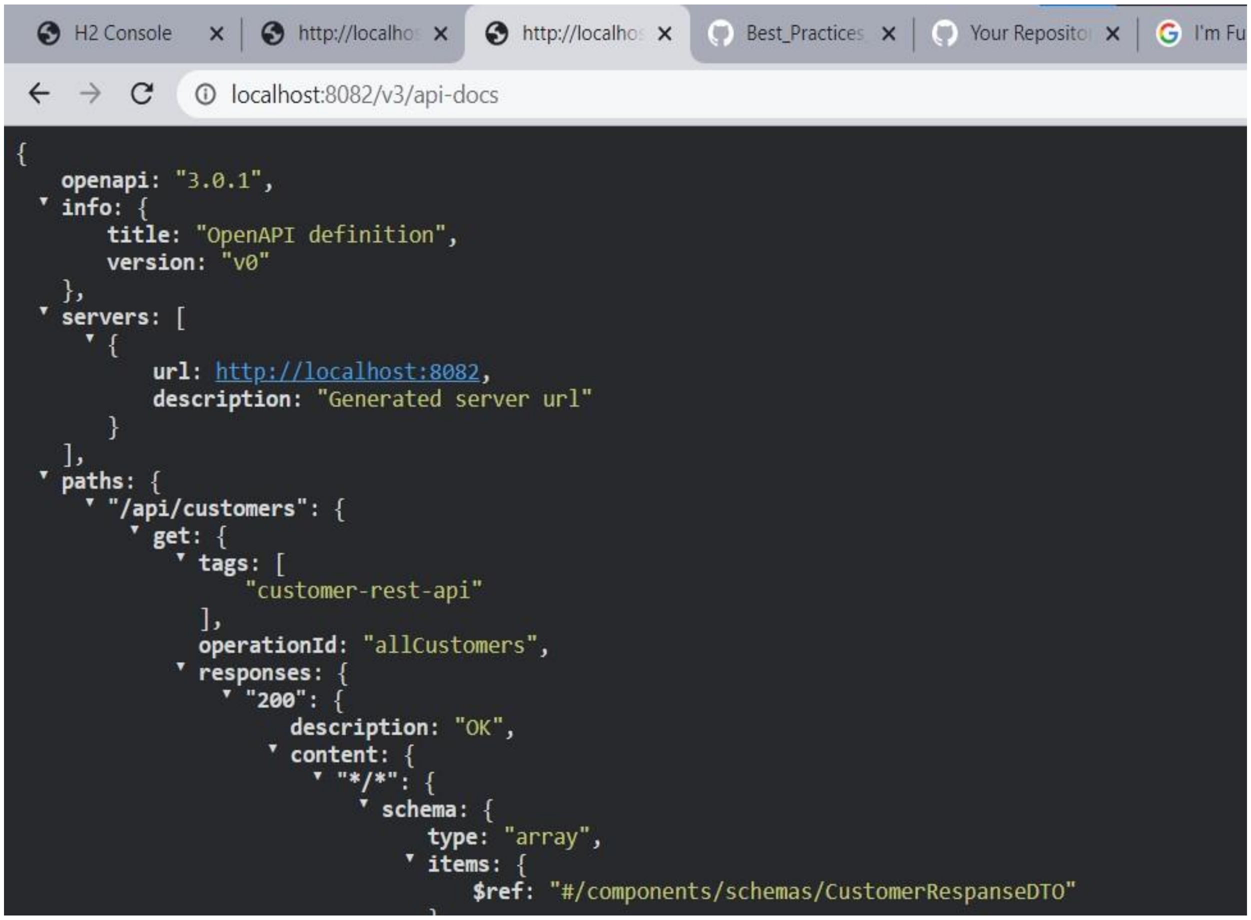
```
{
    "id": "8516fc68-4289-4a62-8b15-ae464ab43200",
    "name": "Tarik 2",
    "email": "Tarik2@Tarik2.com"
}
```



localhost:8082/api/customers

```
[
  {
      id: "C01",
      name: "Tarik",
      email: "Tarik@Tarik.com"
  },
  {
      id: "C02",
      name: "OpenLab",
      email: "OpenLab@OpenLab.com"
  },
  {
      id: "8516fc68-4289-4a62-8b15-ae464ab43200",
      name: "Tarik 2",
      email: "Tarik2@Tarik2.com"
  }
]
```

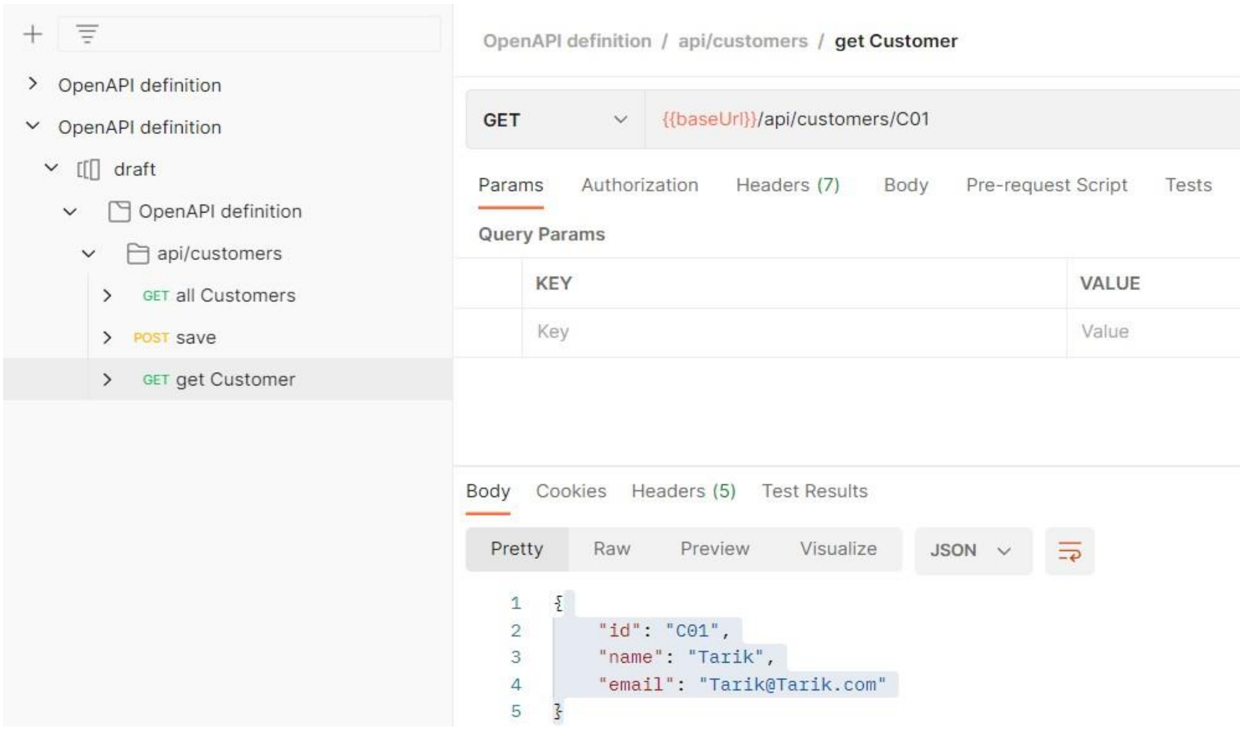
Documentation Api



H2 Console | http://localho | http://localho | Best_Practices | Your Reposito | I'm Fu

localhost:8082/v3/api-docs

```
{
    openapi: "3.0.1",
  ▾ info: {
        title: "OpenAPI definition",
        version: "v0"
    },
  ▾ servers: [
      ▾ {
            url: http://localhost:8082,
            description: "Generated server url"
        }
    ],
  ▾ paths: {
      ▾ "/api/customers": {
          ▾ get: {
              ▾ tags: [
                    "customer-rest-api"
                ],
                operationId: "allCustomers",
              ▾ responses: {
                  ▾ "200": {
                        description: "OK",
                      ▾ content: {
                          ▾ "*/*": {
                              ▾ schema: {
                                    type: "array",
                                  ▾ items: {
                                        $ref: "#/components/schemas/CustomerRespanseDTO"
```



OpenAPI definition
OpenAPI definition
draft
OpenAPI definition
api/customers
GET all Customers
POST save
GET get Customer

OpenAPI definition / api/customers / get Customer

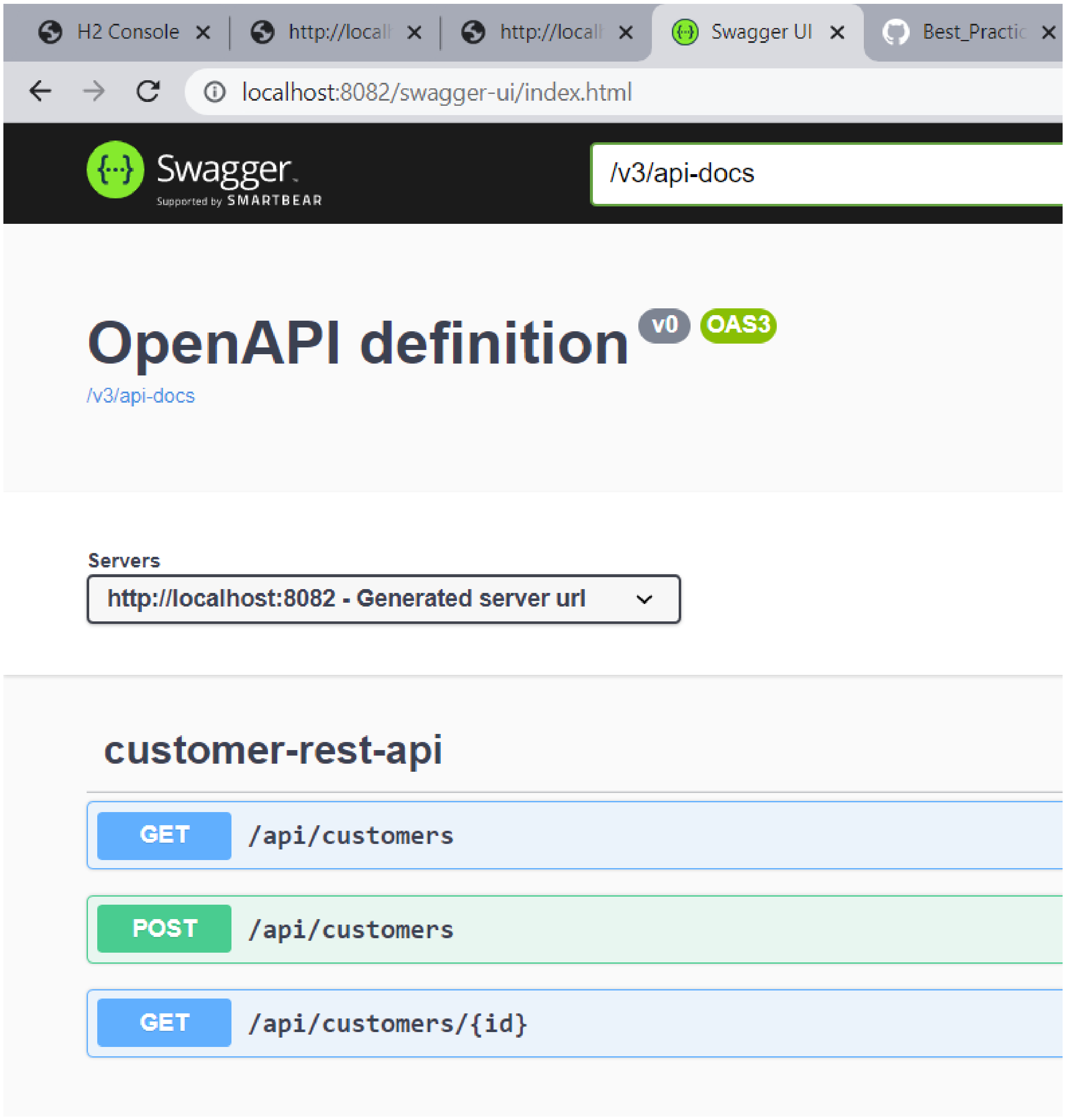
GET {{baseUrl}}/api/customers/C01

Params Authorization Headers (7) Body Pre-request Script Tests
Query Params

KEY	VALUE
Key	Value

Body Cookies Headers (5) Test Results
Pretty Raw Preview Visualize JSON

```
{
    "id": "C01",
    "name": "Tarik",
    "email": "Tarik@Tarik.com"
}
```



OpenAPI definition v0 OAS3

/v3/api-docs

Servers

http://localhost:8082 - Generated server url

customer-rest-api

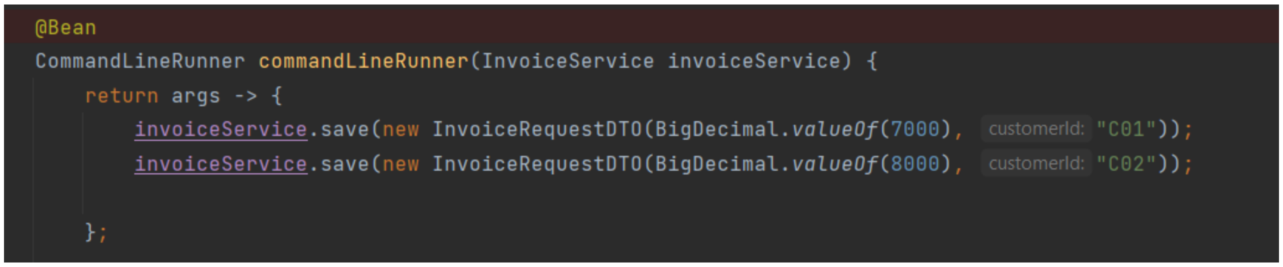
GET /api/customers

POST /api/customers

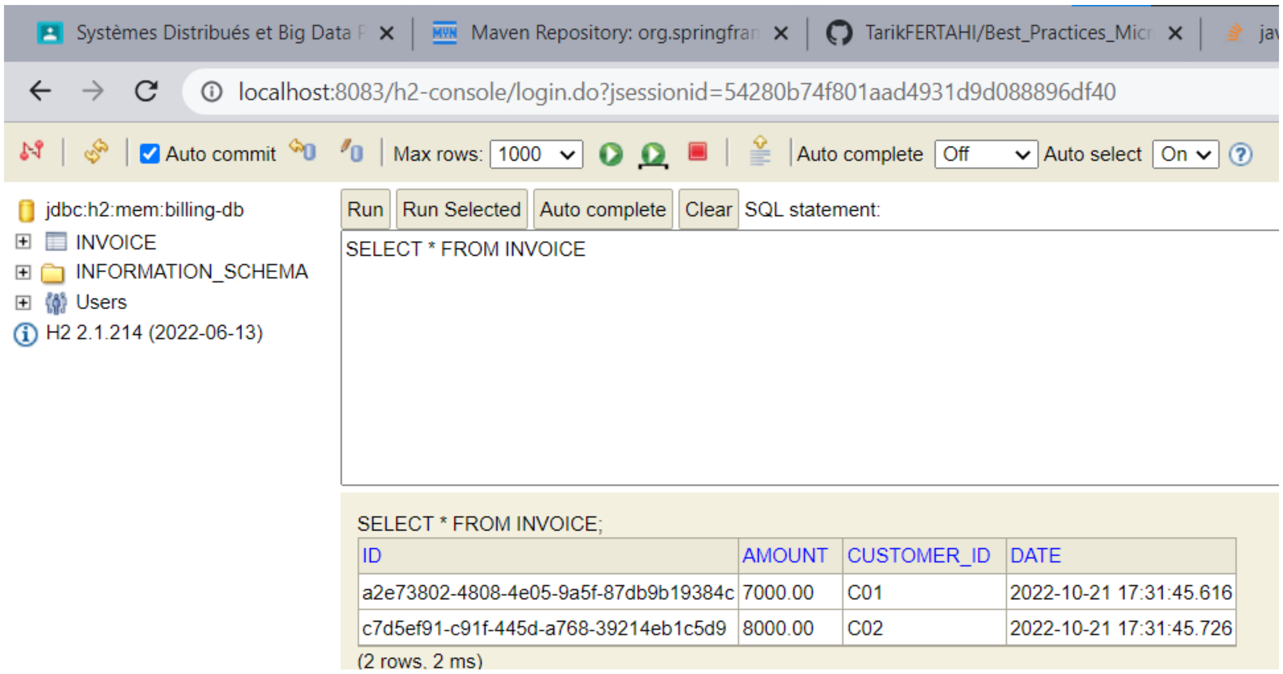
GET /api/customers/{id}

Pour le Micro-Service « **BILLING-SERVICE** »

La création de deux Invoices



```
@Bean
CommandLineRunner commandLineRunner(InvoiceService invoiceService) {
    return args -> {
        invoiceService.save(new InvoiceRequestDTO(BigDecimal.valueOf(7000), customerId: "C01"));
        invoiceService.save(new InvoiceRequestDTO(BigDecimal.valueOf(8000), customerId: "C02"));

    };
```

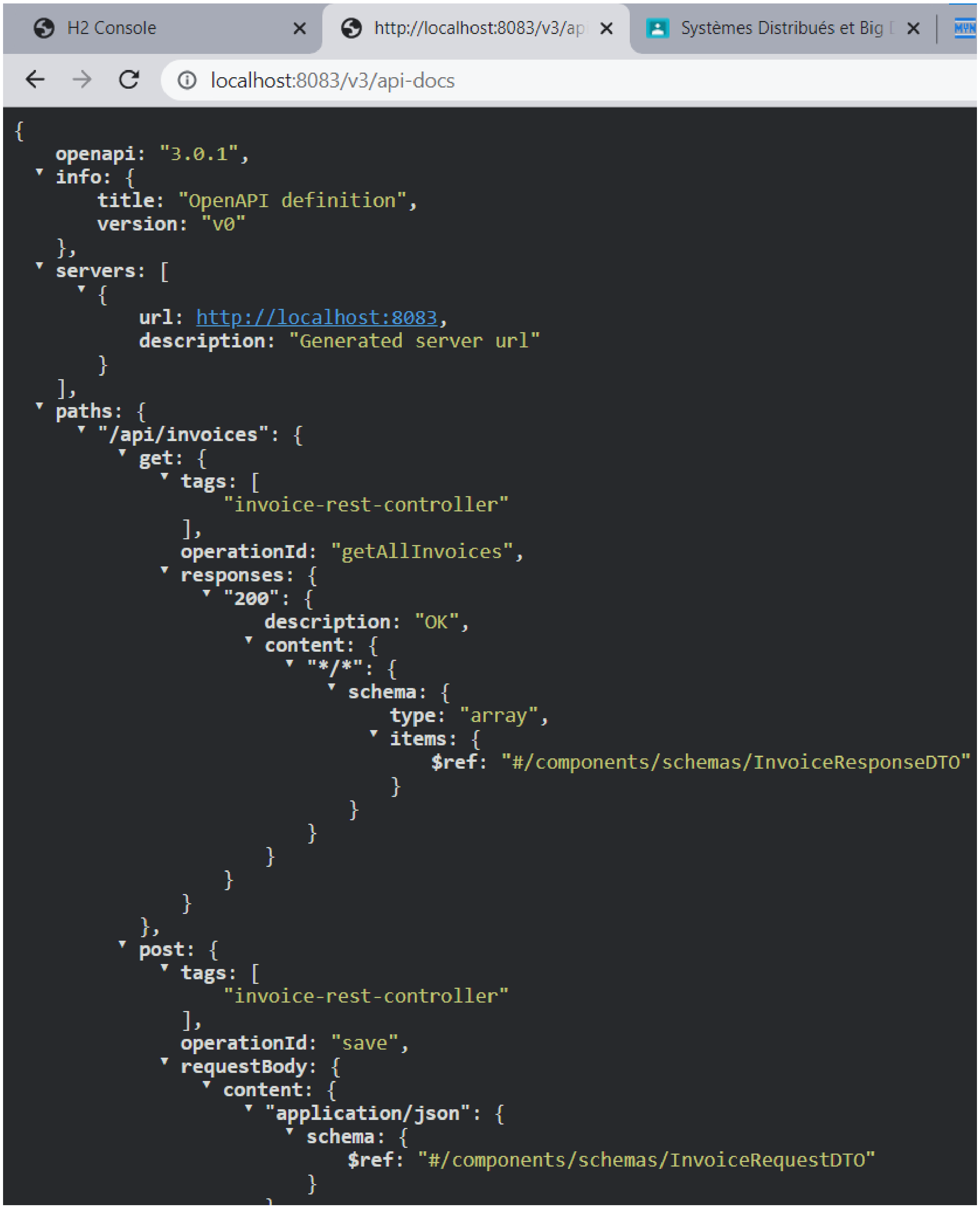


```
SELECT * FROM INVOICE
```

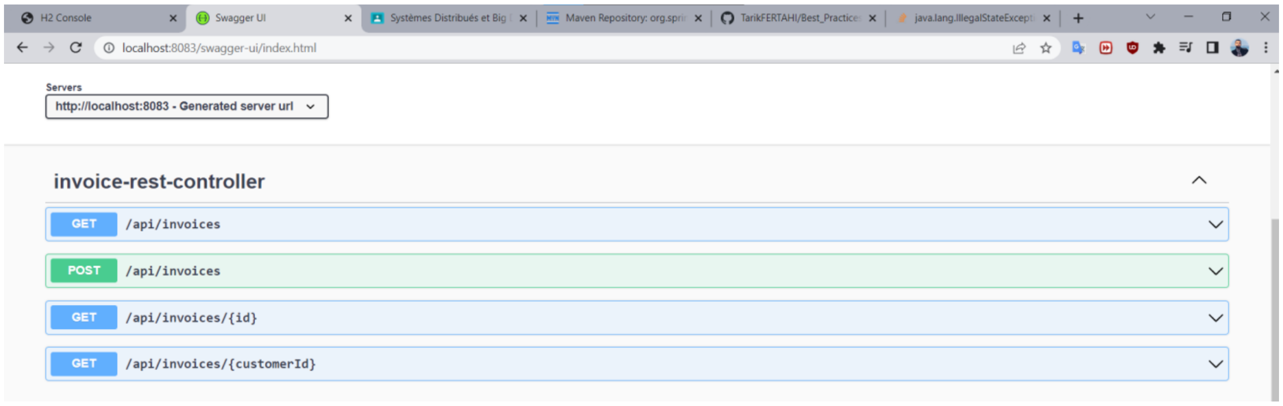
SELECT * FROM INVOICE;

ID	AMOUNT	CUSTOMER_ID	DATE
a2e73802-4808-4e05-9a5f-87db9b19384c	7000.00	C01	2022-10-21 17:31:45.616
c7d5ef91-c91f-445d-a768-39214eb1c5d9	8000.00	C02	2022-10-21 17:31:45.726

(2 rows, 2 ms)


H2 Console
http://localhost:8083/v3/ap
Systèmes Distribués et Big
localhost:8083/v3/api-docs

```
{
    openapi: "3.0.1",
  ▾ info: {
        title: "OpenAPI definition",
        version: "v0"
    },
  ▾ servers: [
      ▾ {
            url: http://localhost:8083,
            description: "Generated server url"
        }
    ],
  ▾ paths: {
      ▾ "/api/invoices": {
          ▾ get: {
              ▾ tags: [
                    "invoice-rest-controller"
                ],
                operationId: "getAllInvoices",
              ▾ responses: {
                  ▾ "200": {
                        description: "OK",
                      ▾ content: {
                          ▾ "*/*": {
                              ▾ schema: {
                                    type: "array",
                                  ▾ items: {
                                        $ref: "#/components/schemas/InvoiceResponseDTO"
                                    }
                                }
                            }
                        }
                    }
                }
            },
          ▾ post: {
              ▾ tags: [
                    "invoice-rest-controller"
                ],
                operationId: "save",
              ▾ requestBody: {
                  ▾ content: {
                      ▾ "application/json": {
                          ▾ schema: {
                                $ref: "#/components/schemas/InvoiceRequestDTO"
                            }
```



localhost:8083/swagger-ui/index.html

Servers

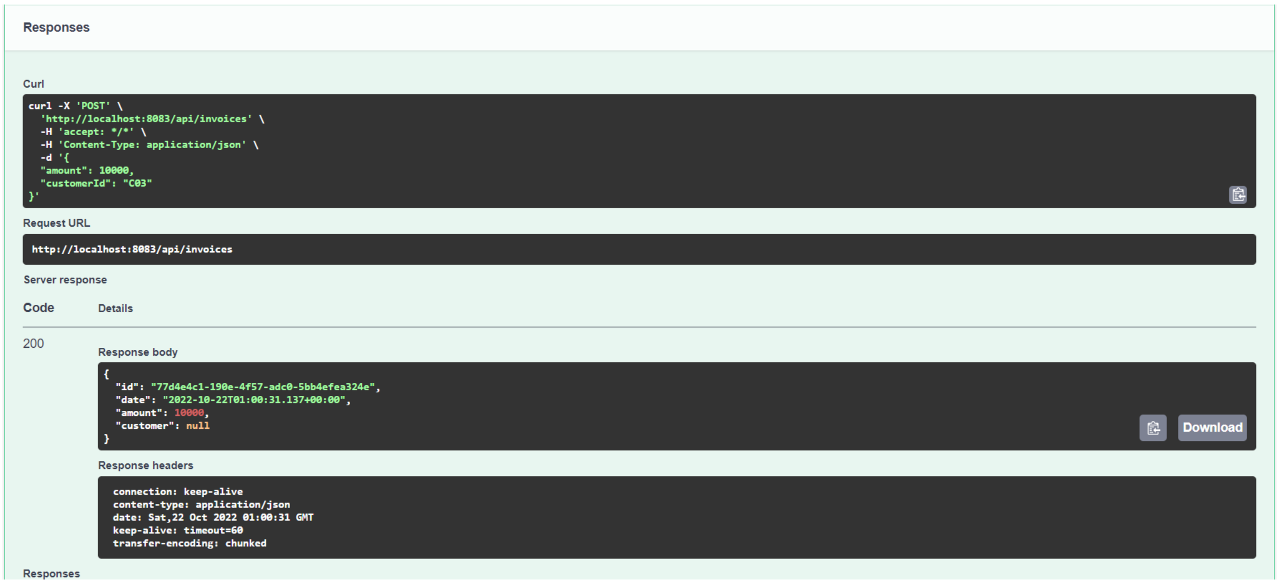
http://localhost:8083 - Generated server url

invoice-rest-controller

```
GET   /api/invoices
POST  /api/invoices
GET   /api/invoices/{id}
GET   /api/invoices/{customerId}
```

Méthode Get

Méthode Save



Responses

Curl

```
curl -X 'POST' \
  'http://localhost:8083/api/invoices' \
  -H 'accept: */*' \
  -H 'Content-Type: application/json' \
  -d '{
  "amount": 10000,
  "customerId": "C03"
}'
```

Request URL

```
http://localhost:8083/api/invoices
```

Server response

Code Details

200

Response body

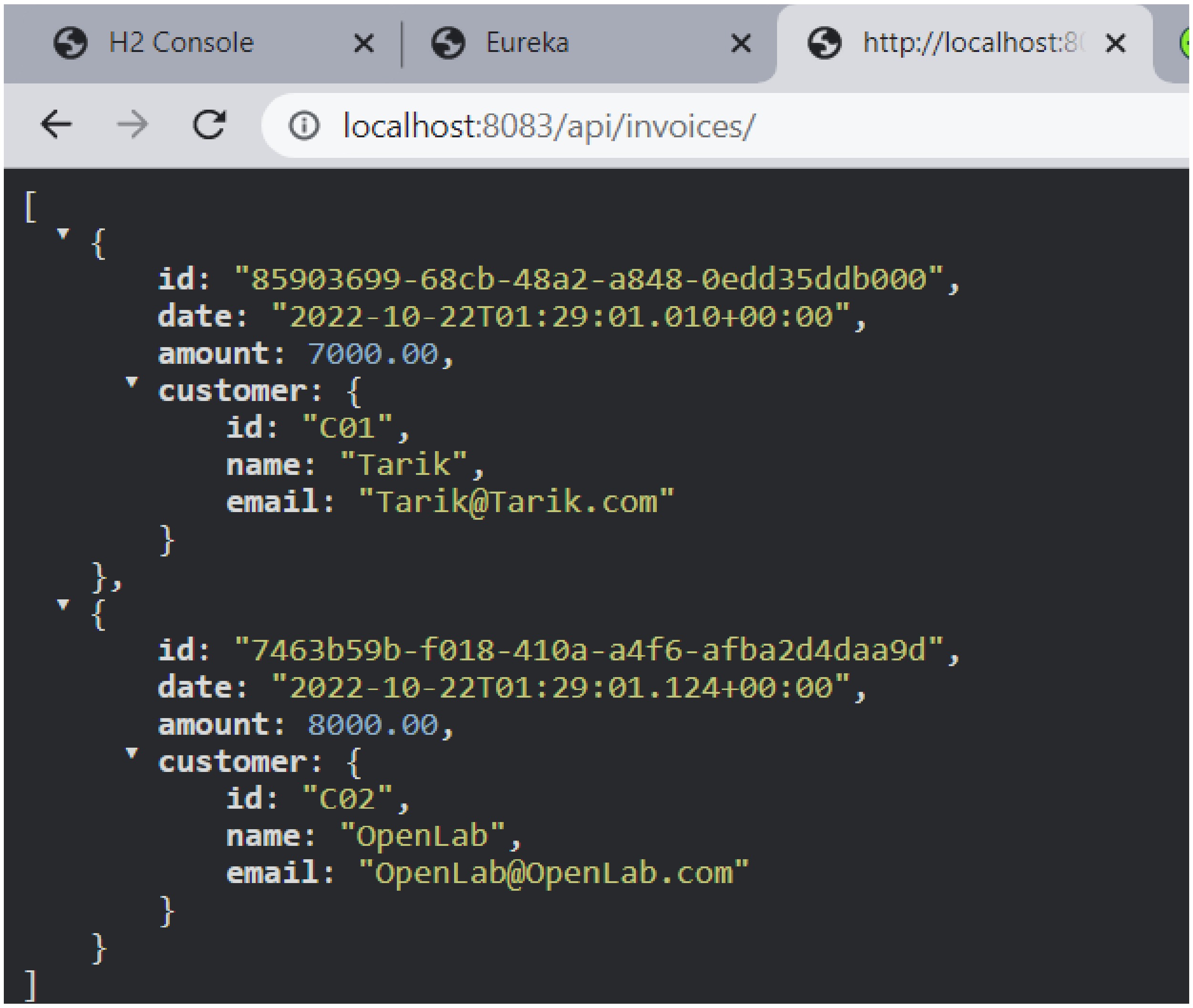
```
{
  "id": "77d4e4c1-190e-4f57-adc0-5bb4efea324e",
  "date": "2022-10-22T01:00:31.137+00:00",
  "amount": 10000,
  "customer": null
}
```

Response headers

```
 connection: keep-alive
 content-type: application/json
 date: Sat,22 Oct 2022 01:00:31 GMT
 keep-alive: timeout=60
 transfer-encoding: chunked
```

Responses

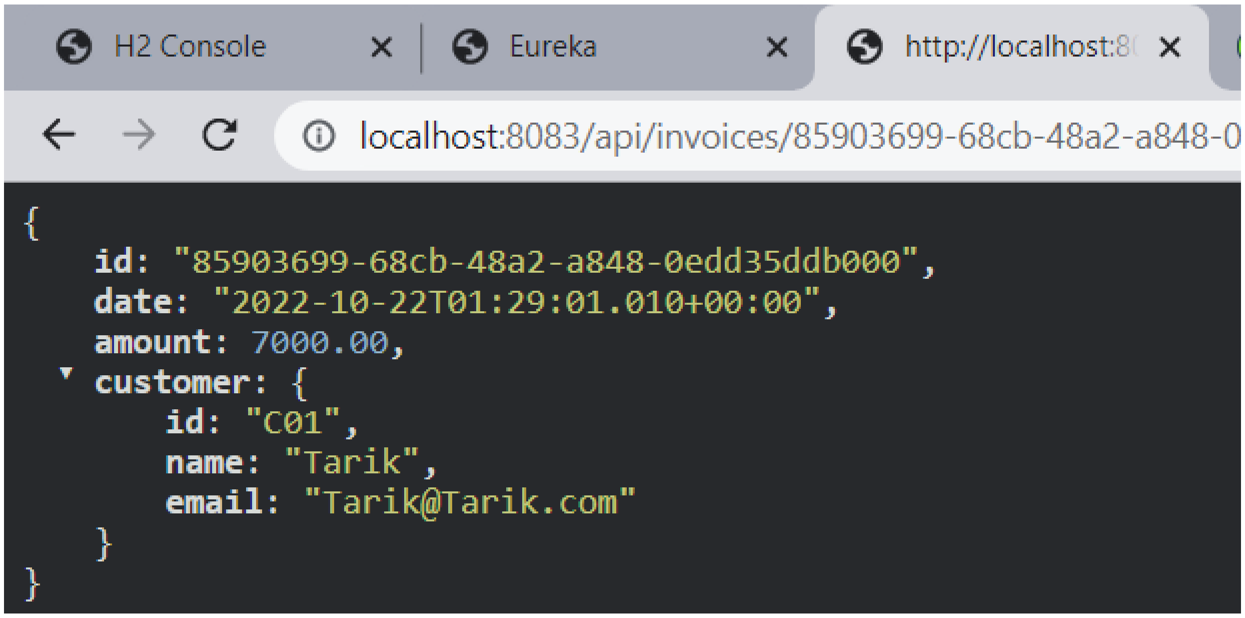
Récupérer toutes les factures



localhost:8083/api/invoices/

```
[
  {
      id: "85903699-68cb-48a2-a848-0edd35ddb000",
      date: "2022-10-22T01:29:01.010+00:00",
      amount: 7000.00,
      customer: {
          id: "C01",
          name: "Tarik",
          email: "Tarik@Tarik.com"
      }
  },
  {
      id: "7463b59b-f018-410a-a4f6-afba2d4daa9d",
      date: "2022-10-22T01:29:01.124+00:00",
      amount: 8000.00,
      customer: {
          id: "C02",
          name: "OpenLab",
          email: "OpenLab@OpenLab.com"
      }
  }
]
```

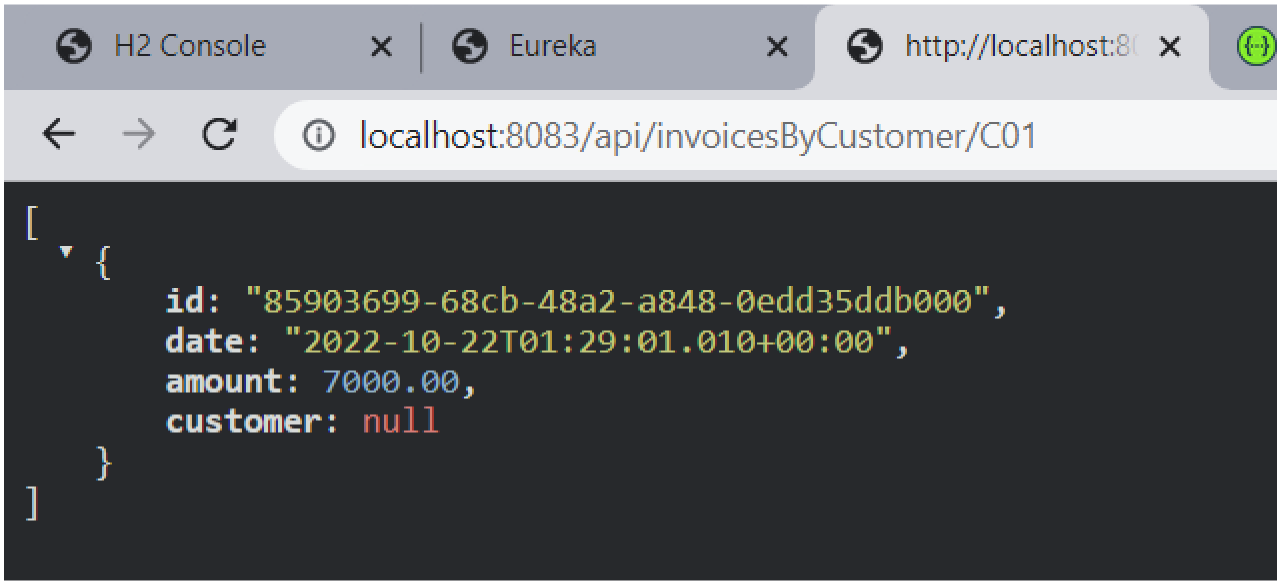
Récupérer une facture par Id



localhost:8083/api/invoices/85903699-68cb-48a2-a848-0

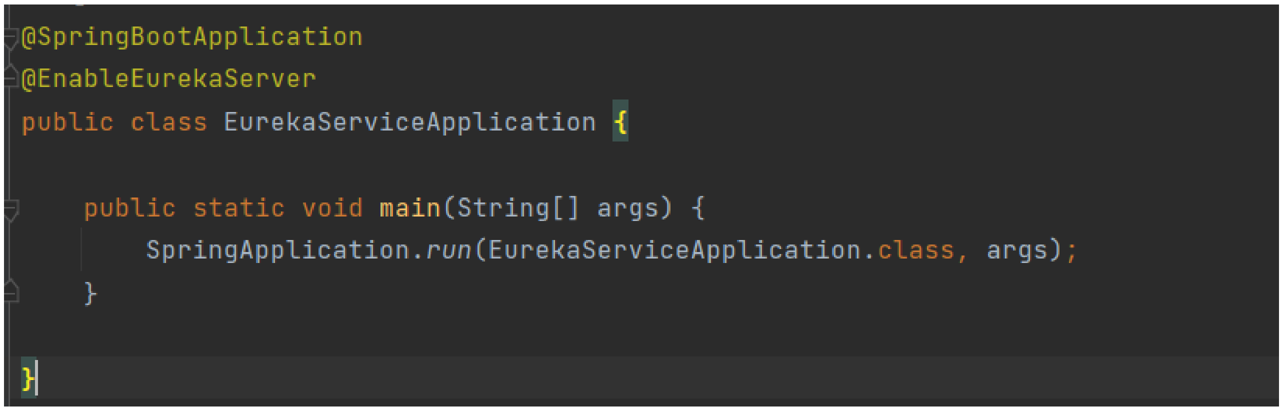
```
{
    id: "85903699-68cb-48a2-a848-0edd35ddb000",
    date: "2022-10-22T01:29:01.010+00:00",
    amount: 7000.00,
    customer: {
        id: "C01",
        name: "Tarik",
        email: "Tarik@Tarik.com"
    }
}
```

Récupérer toutes les factures associe a un client

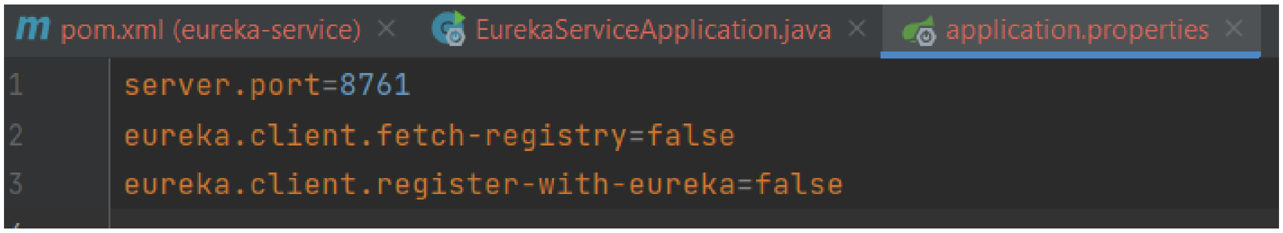


```
[
  ▾ {
      id: "85903699-68cb-48a2-a848-0edd35ddb000",
      date: "2022-10-22T01:29:01.010+00:00",
      amount: 7000.00,
      customer: null
    }
]
```

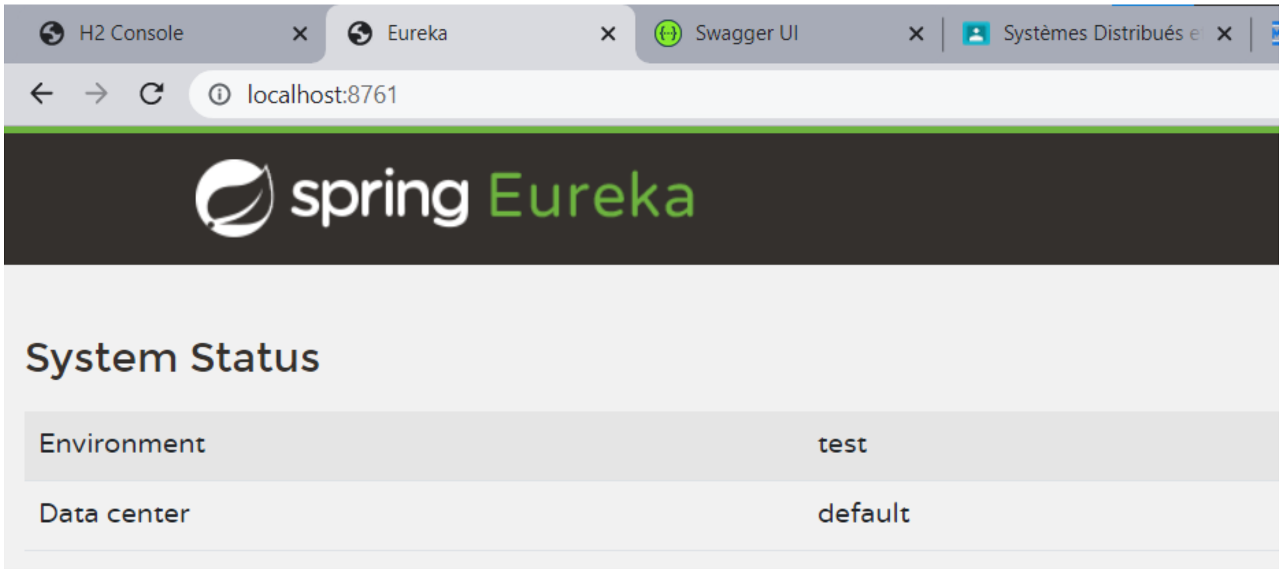
Pour le Micro-Service « **EUREKA-SERVICE** »



```java
@SpringBootApplication
@EnableEurekaServer
public class EurekaServiceApplication {

    public static void main(String[] args) {
        SpringApplication.run(EurekaServiceApplication.class, args);
    }

}
```

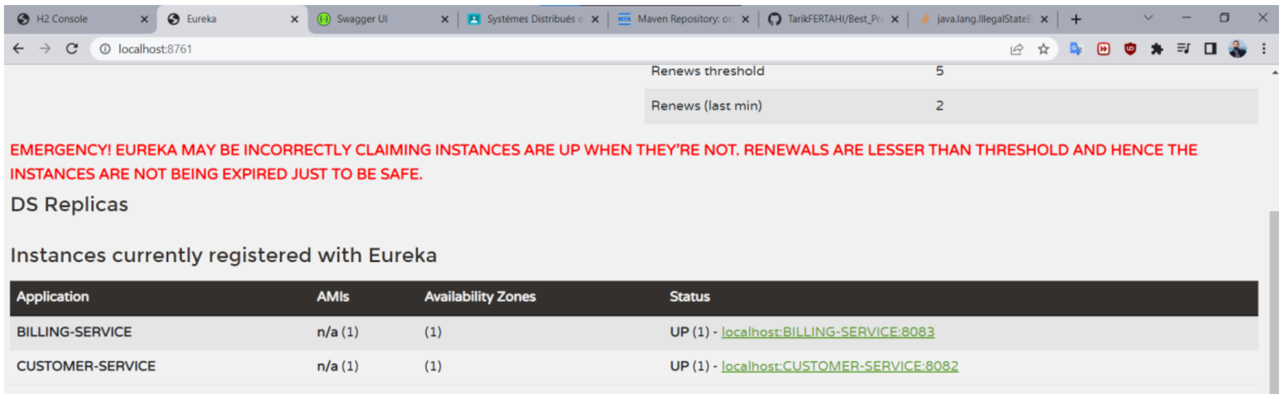


```properties
server.port=8761
eureka.client.fetch-registry=false
eureka.client.register-with-eureka=false
```



System Status

Environment	test
Data center	default



Renews threshold	5
Renews (last min)	2

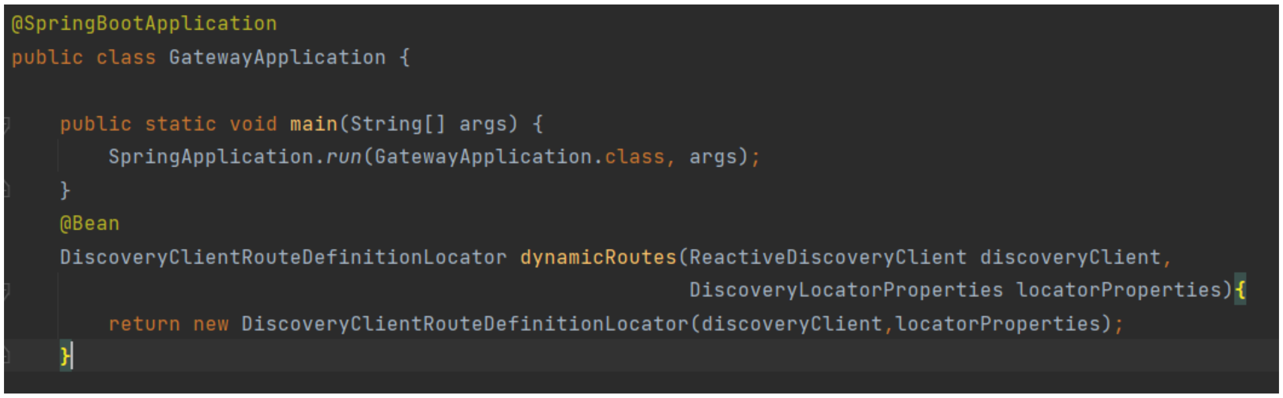
EMERGENCY! EUREKA MAY BE INCORRECTLY CLAIMING INSTANCES ARE UP WHEN THEY'RE NOT. RENEWALS ARE LESSER THAN THRESHOLD AND HENCE THE INSTANCES ARE NOT BEING EXPIRED JUST TO BE SAFE.

DS Replicas

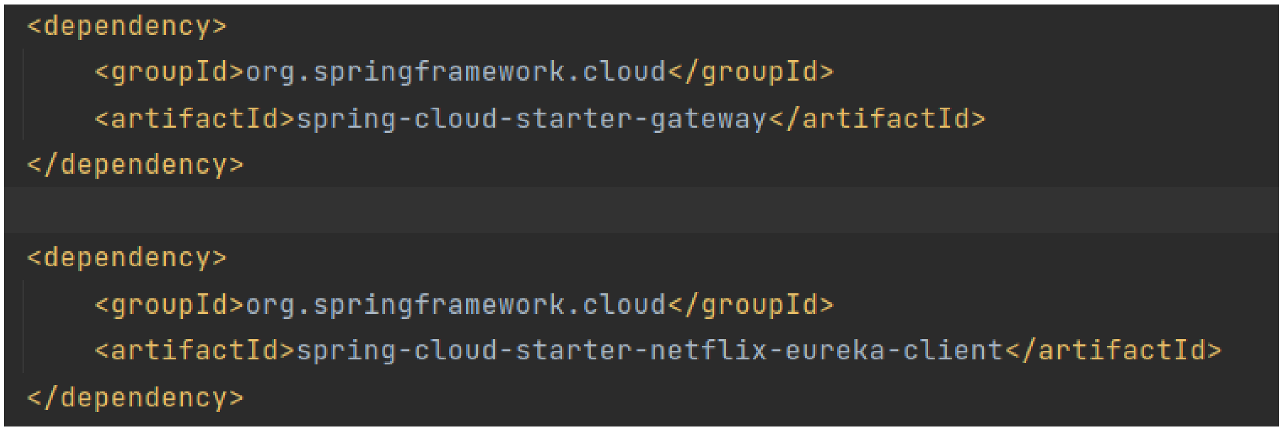
Instances currently registered with Eureka

Application	AMIs	Availability Zones	Status
BILLING-SERVICE	n/a (1)	(1)	UP (1) - localhost:BILLING-SERVICE:8083
CUSTOMER-SERVICE	n/a (1)	(1)	UP (1) - localhost:CUSTOMER-SERVICE:8082

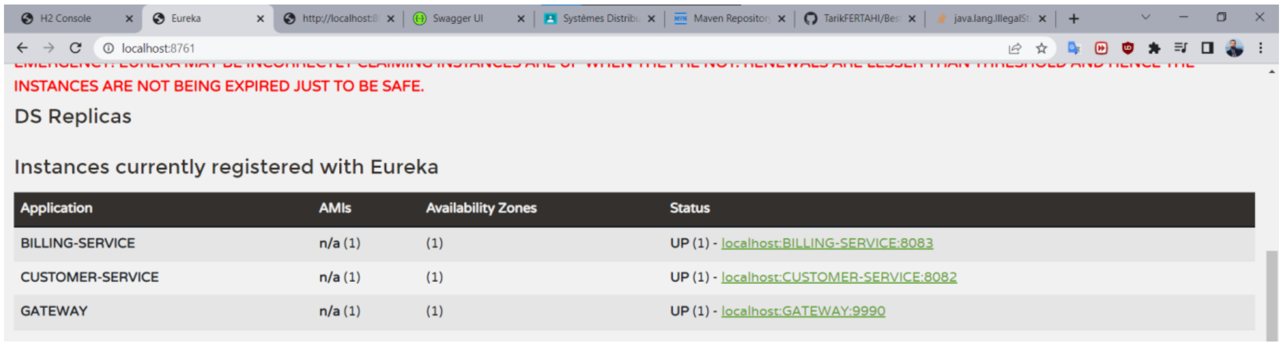
Pour le Micro-Service « **GATEWAY** »



```java
@SpringBootApplication
public class GatewayApplication {

    public static void main(String[] args) {
        SpringApplication.run(GatewayApplication.class, args);
    }
    @Bean
    DiscoveryClientRouteDefinitionLocator dynamicRoutes(ReactiveDiscoveryClient discoveryClient,
                                                        DiscoveryLocatorProperties locatorProperties){
        return new DiscoveryClientRouteDefinitionLocator(discoveryClient,locatorProperties);
    }
```



```xml
<dependency>
    <groupId>org.springframework.cloud</groupId>
    <artifactId>spring-cloud-starter-gateway</artifactId>
</dependency>

<dependency>
    <groupId>org.springframework.cloud</groupId>
    <artifactId>spring-cloud-starter-netflix-eureka-client</artifactId>
</dependency>
```



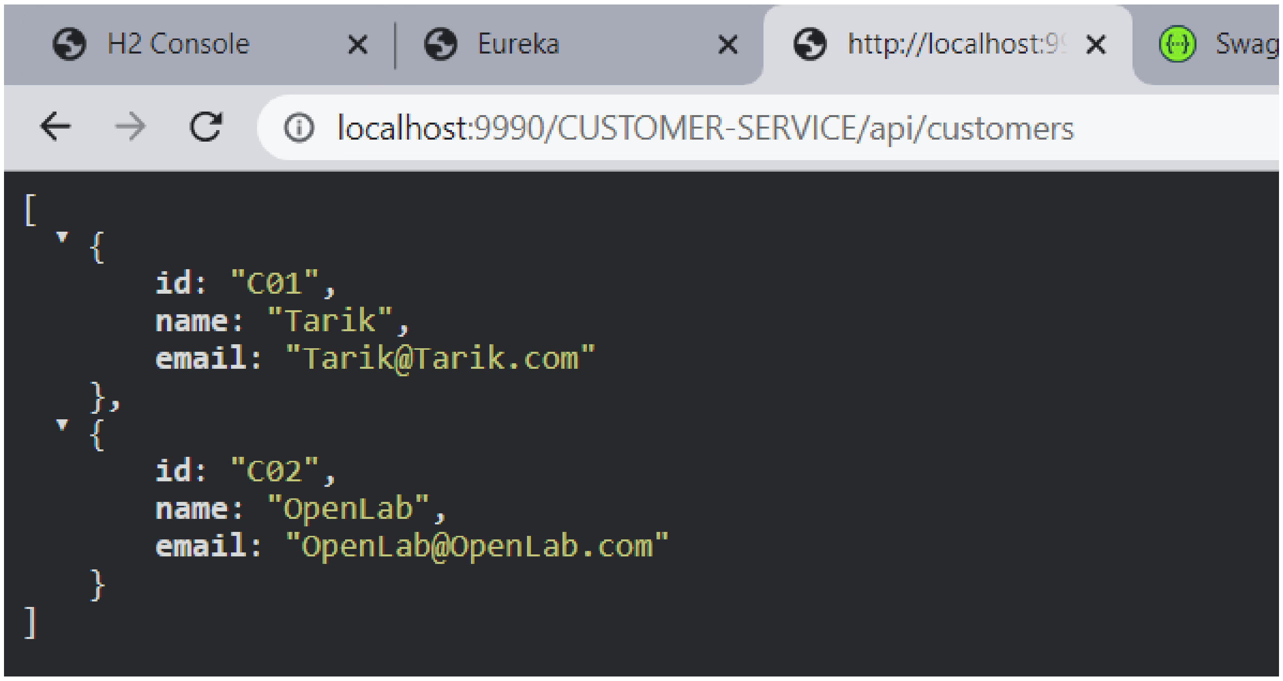
EMERGENCY! EUREKA MAY BE INCORRECTLY CLAIMING INSTANCES ARE UP WHEN THEY'RE NOT. RENEWALS ARE LESSER THAN THRESHOLD AND HENCE THE
INSTANCES ARE NOT BEING EXPIRED JUST TO BE SAFE.

DS Replicas

Instances currently registered with Eureka

Application	AMIs	Availability Zones	Status
BILLING-SERVICE	n/a (1)	(1)	UP (1) - localhost:BILLING-SERVICE:8083
CUSTOMER-SERVICE	n/a (1)	(1)	UP (1) - localhost:CUSTOMER-SERVICE:8082
GATEWAY	n/a (1)	(1)	UP (1) - localhost:GATEWAY:9990

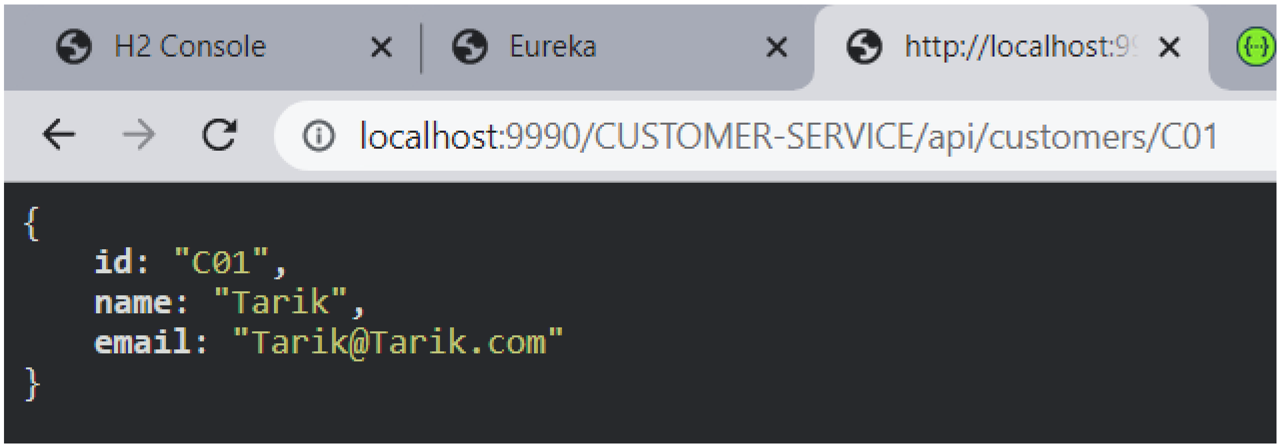
Récupérer la liste des clients en passant par **Gateway**



localhost:9990/CUSTOMER-SERVICE/api/customers

```
[
  ▼ {
        id: "C01",
        name: "Tarik",
        email: "Tarik@Tarik.com"
    },
  ▼ {
        id: "C02",
        name: "OpenLab",
        email: "OpenLab@OpenLab.com"
    }
]
```

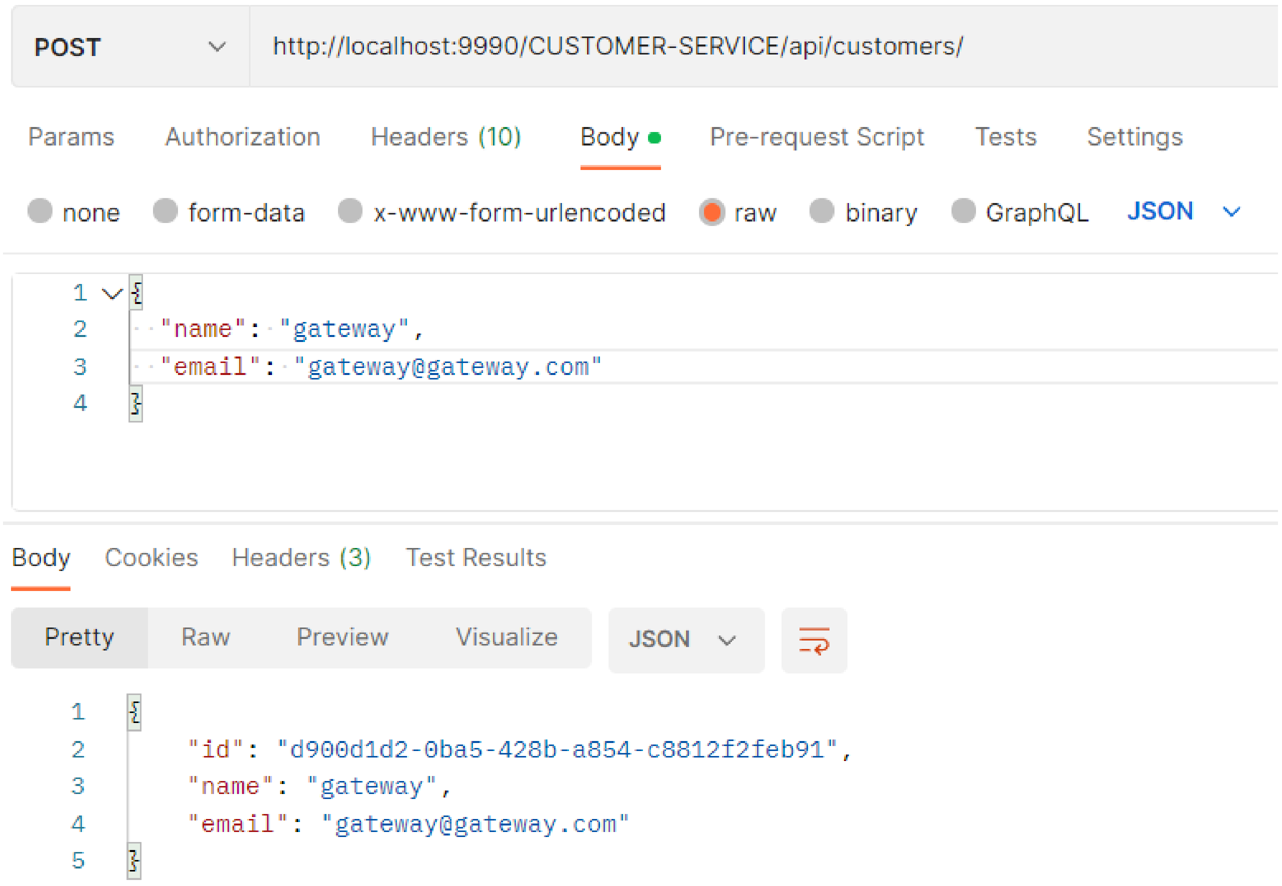
Récupérer un client par id en passant par **Gateway**



localhost:9990/CUSTOMER-SERVICE/api/customers/C01

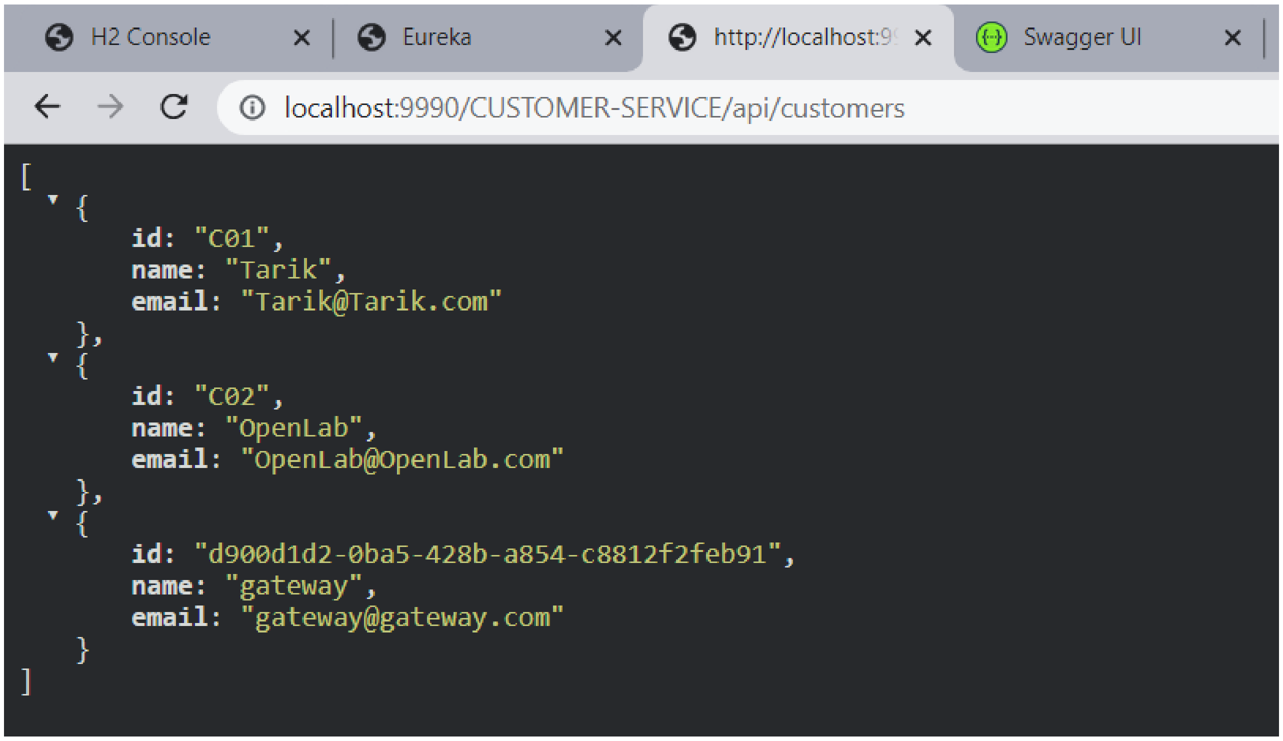
```
{
    id: "C01",
    name: "Tarik",
    email: "Tarik@Tarik.com"
}
```

Méthode Post


POST http://localhost:9990/CUSTOMER-SERVICE/api/customers/

```
{
  "name": "gateway",
  "email": "gateway@gateway.com"
}
```

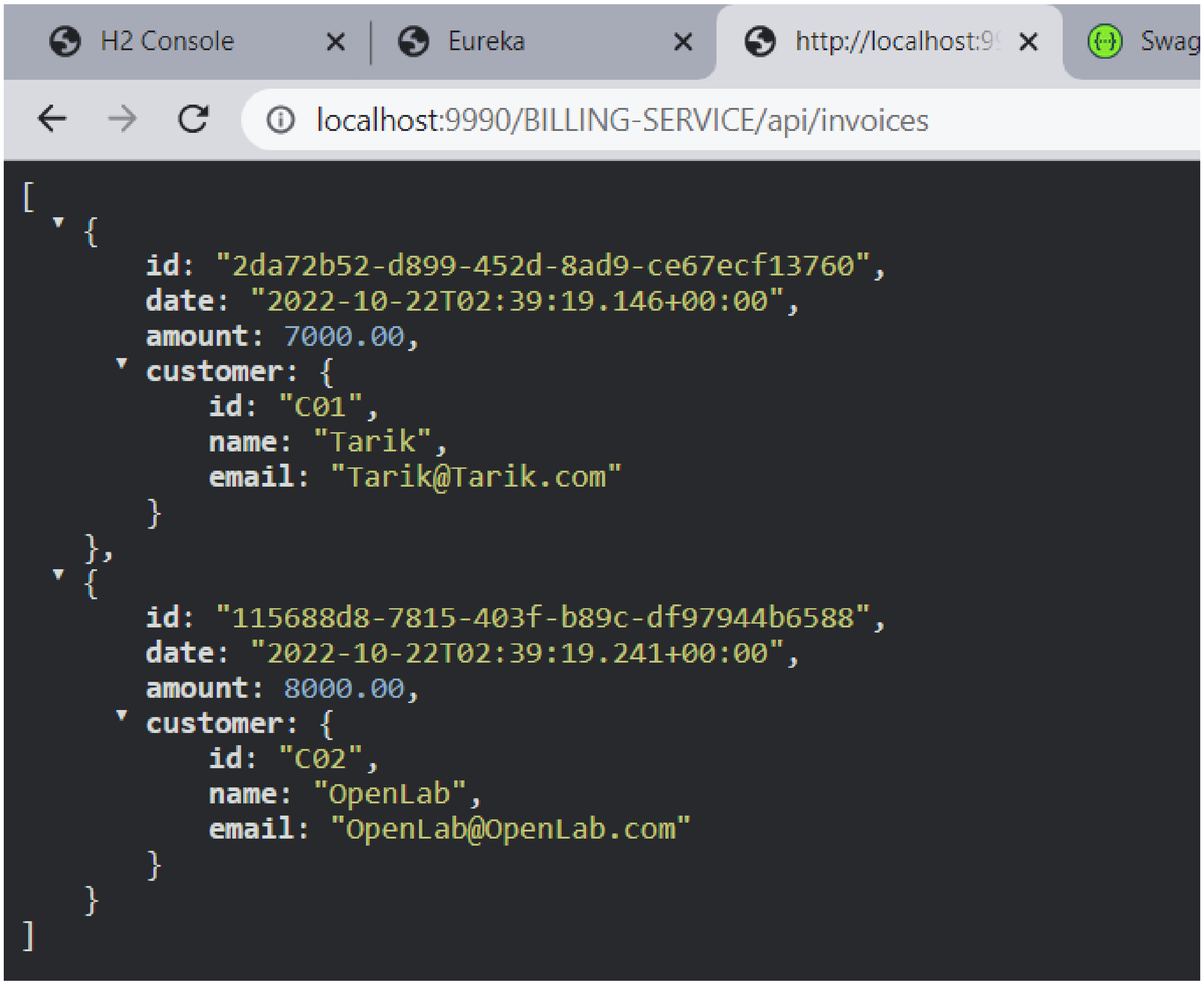
```
{
    "id": "d900d1d2-0ba5-428b-a854-c8812f2feb91",
    "name": "gateway",
    "email": "gateway@gateway.com"
}
```



localhost:9990/CUSTOMER-SERVICE/api/customers

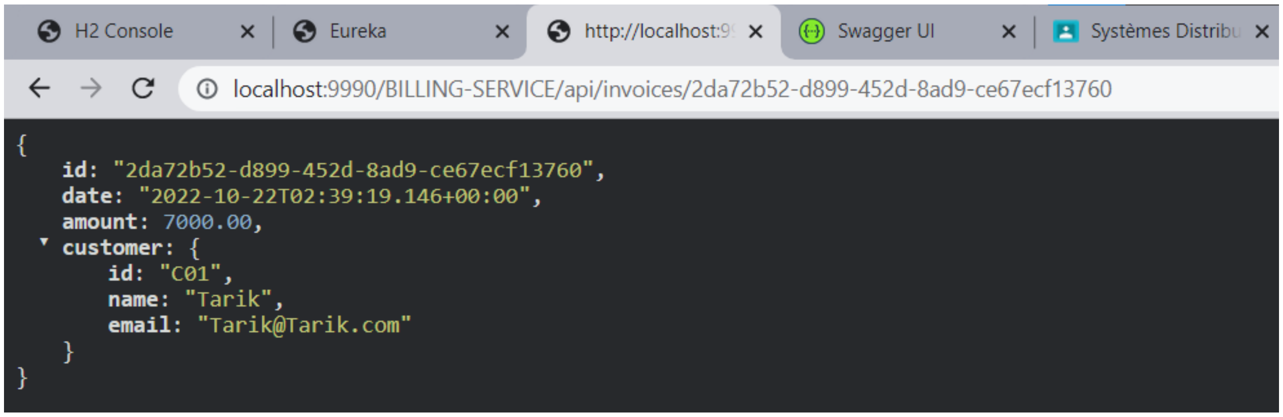
```
[
  ▾ {
        id: "C01",
        name: "Tarik",
        email: "Tarik@Tarik.com"
    },
  ▾ {
        id: "C02",
        name: "OpenLab",
        email: "OpenLab@OpenLab.com"
    },
  ▾ {
        id: "d900d1d2-0ba5-428b-a854-c8812f2feb91",
        name: "gateway",
        email: "gateway@gateway.com"
    }
]
```

Récupérer la liste des factures en passant par **Gateway**



```
[
  ▾ {
        id: "2da72b52-d899-452d-8ad9-ce67ecf13760",
        date: "2022-10-22T02:39:19.146+00:00",
        amount: 7000.00,
      ▾ customer: {
            id: "C01",
            name: "Tarik",
            email: "Tarik@Tarik.com"
        }
    },
  ▾ {
        id: "115688d8-7815-403f-b89c-df97944b6588",
        date: "2022-10-22T02:39:19.241+00:00",
        amount: 8000.00,
      ▾ customer: {
            id: "C02",
            name: "OpenLab",
            email: "OpenLab@OpenLab.com"
        }
    }
]
```

Récupérer une facture par id en passant par **Gateway**


H2 Console
Eureka
http://localhost:9
Swagger UI
Systèmes Distribu
localhost:9990/BILLING-SERVICE/api/invoices/2da72b52-d899-452d-8ad9-ce67ecf13760

```
{
    id: "2da72b52-d899-452d-8ad9-ce67ecf13760",
    date: "2022-10-22T02:39:19.146+00:00",
    amount: 7000.00,
  ▾ customer: {
        id: "C01",
        name: "Tarik",
        email: "Tarik@Tarik.com"
    }
}
```