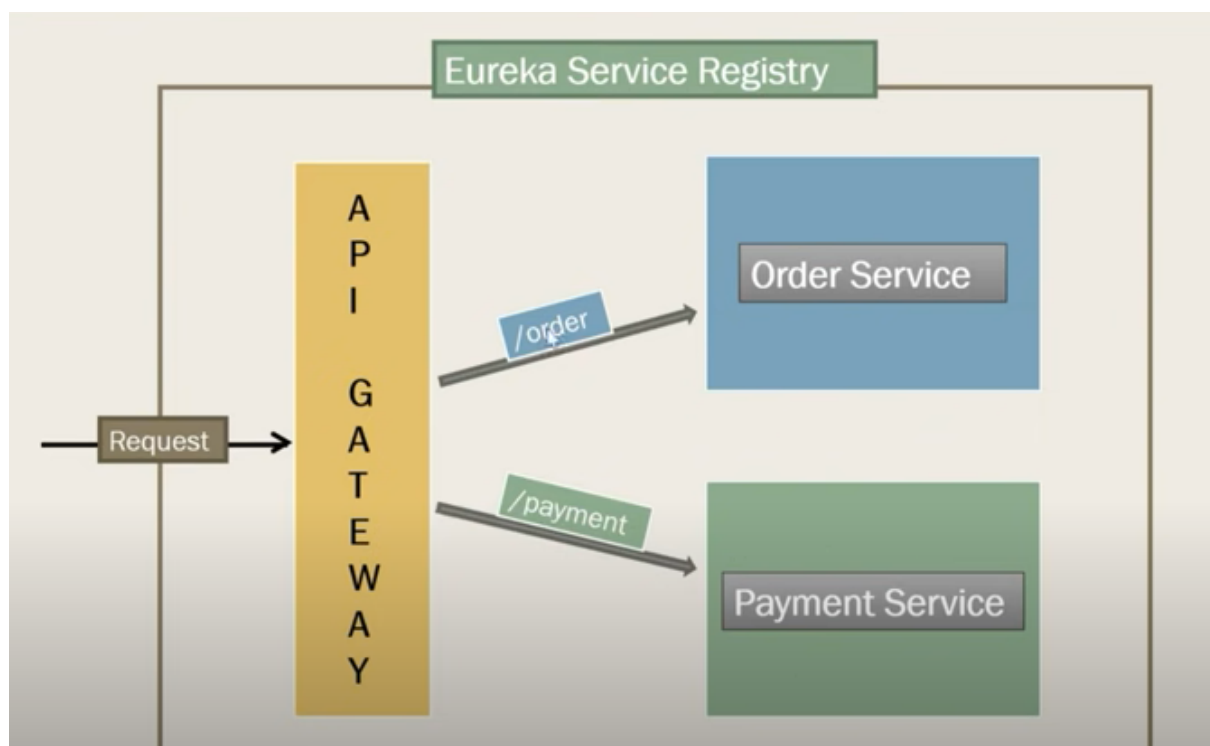


Microservice Using Spring Boot & Spring Cloud 2H:

What should we do? :

Agenda of this tutorial :

1. Create 2 microservice from scratch.
2. Register microservice in Eureka Service Discovery.
3. integrate Spring Cloud Gateway for routing user requests.
4. Integrate Hystrix & Hystrix Dashboard to identify failure for downstream services.
5. Spring cloud config server using Git to Centralize configuration across applications.
6. ELK Stack to centralize logging across all microservices.
7. Zipkin & Sleuth to centralize tracing in microservice architecture.



1 : Create 2 microservice from scratch :

Creating Order Service :

The screenshot shows the H2 console interface. The left sidebar displays the database structure: jdbc:h2:mem:testdb, INFORMATION_SCHEMA, Users, and H2 2.2.224 (2023-09-17). The main area shows the SQL statement: `SELECT * FROM ORDER_TB`. Below the statement, the results are displayed as a table with columns ID, PRICE, QTY, and NAME. The first row contains the values 103, 8000.0, 1, and Mobile. The execution time is 8 ms.

ID	PRICE	QTY	NAME
103	8000.0	1	Mobile

Creating Payment Service :

The screenshot shows the H2 console interface. The left sidebar displays the database structure: jdbc:h2:mem:testdb, INFORMATION_SCHEMA, Sequences, Users, and H2 2.2.224 (2023-09-17). The main area shows the SQL statement: `SELECT * FROM PAYMENT_TB`. Below the statement, the results are displayed as a table with columns PAYMENT_ID, PAYMENT_STATUS, and TRANSACTION_ID. The first row contains the values 1, Success, and 558ca821-9624-4de3-84b7-70ef7de6b2cd. The execution time is 21 ms.

PAYMENT_ID	PAYMENT_STATUS	TRANSACTION_ID
1	Success	558ca821-9624-4de3-84b7-70ef7de6b2cd

Let's check whether our microservices have connected successfully :

POST http://localhost:9192/order/bookOrder

POST http://localhost:9192/order/bookOrder

Params Authorization Headers (8) **Body** Pre-request Script Tests Settings

none form-data x-www-form-urlencoded raw binary **JSON**

```
2  .... "order" : {
3  .... "id" : 103,
4  .... "name" : "Mobile",
5  .... "qty" : 1,
6  .... "price" : 8000
```

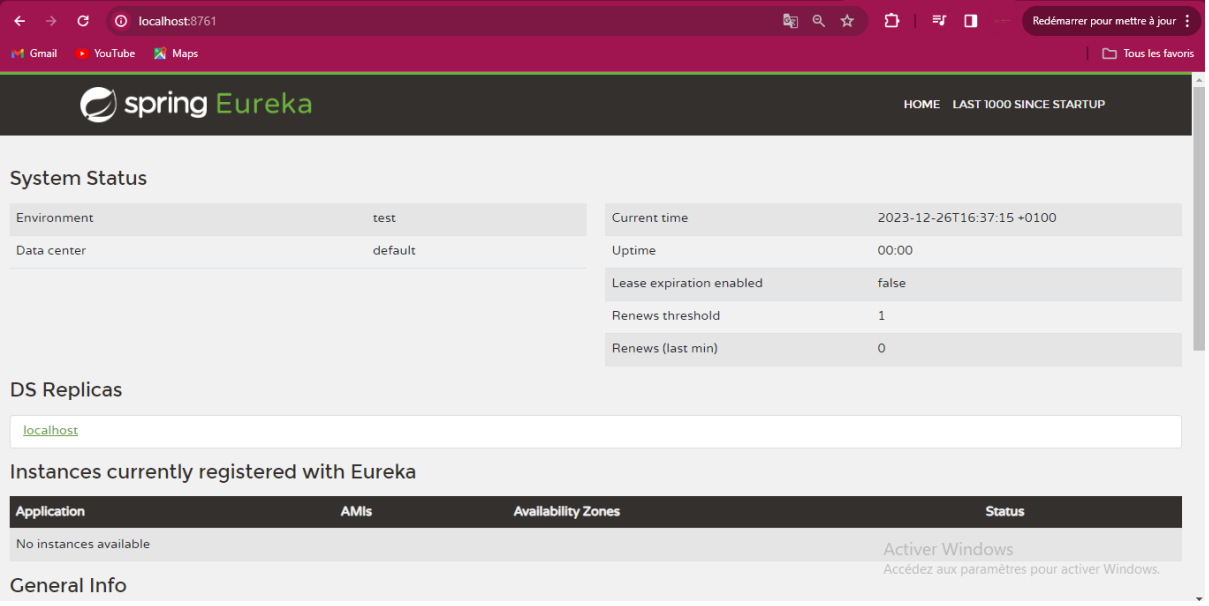
Body Cookies Headers (5) Test Results 200 OK 786 ms 352 B

Pretty Raw Preview Visualize **JSON**

```
1  {
2  "order": {
3    "id": 103,
4    "name": "Mobile",
5    "qty": 1,
6    "price": 8000.0
7  },
8  "amount": 8000.0,
9  "transactionId": "b715fd37-cc51-4b3c-ba8f-331fd9e591e3",
10 "message": "payment processing successful and order place"
```

Activer Windows
Accédez aux paramètres

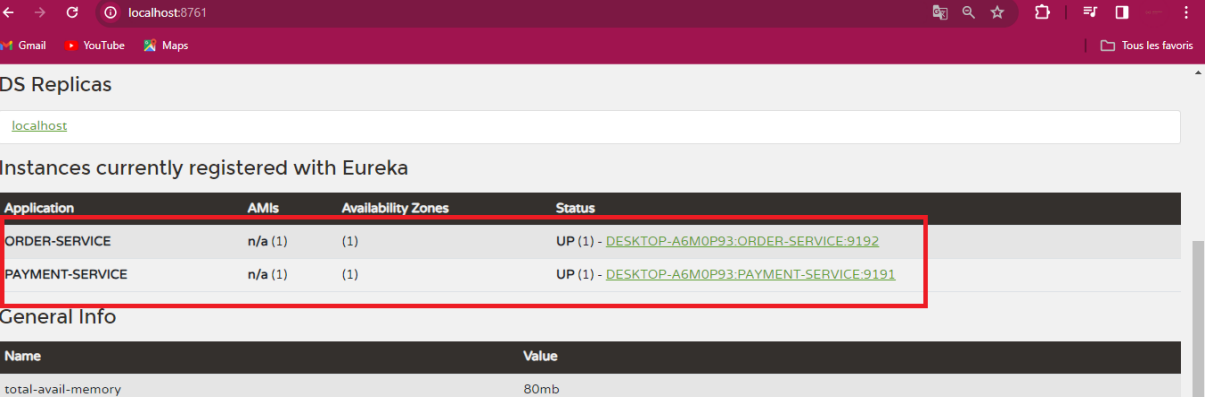
2 : Register microservice in Eureka Service Discovery :



The screenshot shows the Spring Eureka web interface at localhost:8761. The page has a dark header with the 'spring Eureka' logo and navigation links for 'HOME' and 'LAST 1000 SINCE STARTUP'. Below the header, the 'System Status' section displays two columns of information: Environment (test), Data center (default), Current time (2023-12-26T16:37:15 +0100), Uptime (00:00), Lease expiration enabled (false), Renew threshold (1), and Renew (last min) (0). The 'DS Replicas' section shows a search bar with 'localhost' entered. The 'Instances currently registered with Eureka' section contains a table with columns 'Application', 'AMIs', 'Availability Zones', and 'Status'. A message 'No instances available' is displayed below the table. A 'General Info' section is partially visible at the bottom. A Windows watermark 'Activer Windows' is present on the right side of the page.

Application	AMIs	Availability Zones	Status
No instances available			

Let's register our microservice instance with Eureka :

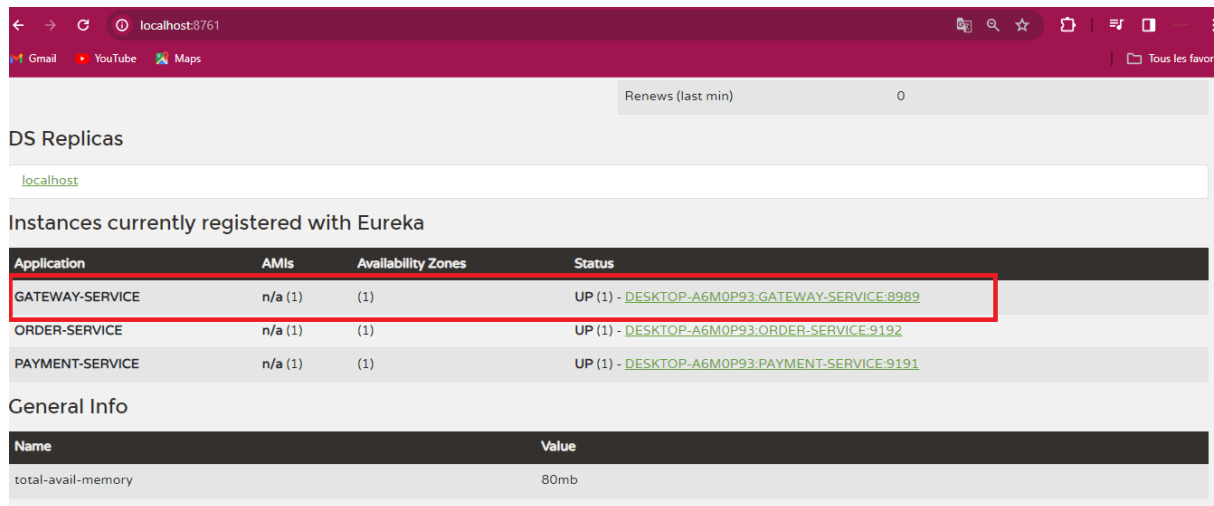


The screenshot shows the Spring Eureka web interface at localhost:8761, displaying two registered instances. The 'DS Replicas' section shows a search bar with 'localhost' entered. The 'Instances currently registered with Eureka' section contains a table with columns 'Application', 'AMIs', 'Availability Zones', and 'Status'. Two instances are listed: 'ORDER-SERVICE' and 'PAYMENT-SERVICE'. The 'ORDER-SERVICE' instance has status 'UP (1) - DESKTOP-A6MOP93:ORDER-SERVICE:9192' and the 'PAYMENT-SERVICE' instance has status 'UP (1) - DESKTOP-A6MOP93:PAYMENT-SERVICE:9191'. A red rectangle highlights these two rows. The 'General Info' section is partially visible at the bottom, showing 'total-avail-memory' with a value of '80mb'.

Application	AMIs	Availability Zones	Status
ORDER-SERVICE	n/a (1)	(1)	UP (1) - DESKTOP-A6MOP93:ORDER-SERVICE:9192
PAYMENT-SERVICE	n/a (1)	(1)	UP (1) - DESKTOP-A6MOP93:PAYMENT-SERVICE:9191

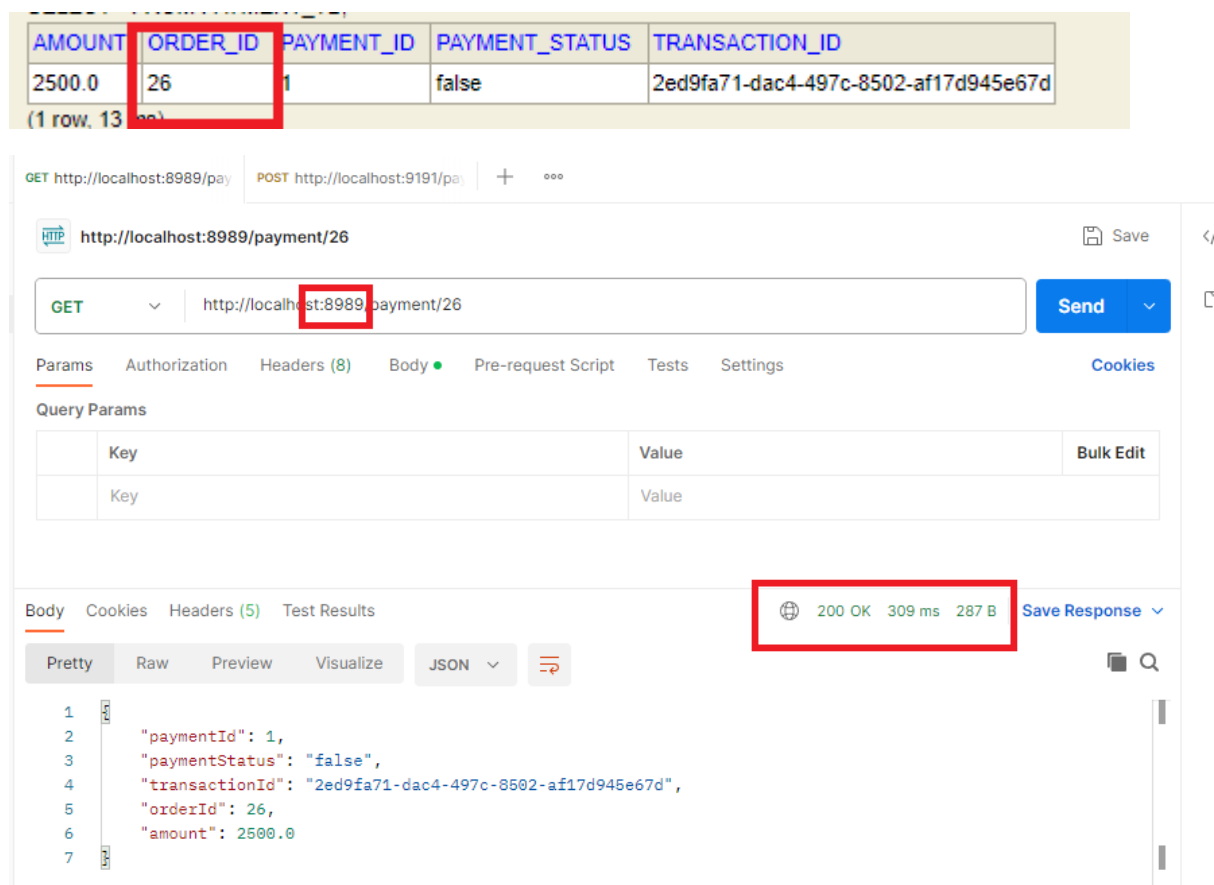
Name	Value
total-avail-memory	80mb

3 : integrate Spring Cloud Gateway for routing user requests :



Application	AMIs	Availability Zones	Status
GATEWAY-SERVICE	n/a (1)	(1)	UP (1) - DESKTOP-A6M0P93:GATEWAY-SERVICE-8989
ORDER-SERVICE	n/a (1)	(1)	UP (1) - DESKTOP-A6M0P93:ORDER-SERVICE-9192
PAYMENT-SERVICE	n/a (1)	(1)	UP (1) - DESKTOP-A6M0P93:PAYMENT-SERVICE-9191

Let's verify if the user request from the microservices successfully passes through the API gateway :



AMOUNT	ORDER_ID	PAYMENT_ID	PAYMENT_STATUS	TRANSACTION_ID
2500.0	26	1	false	2ed9fa71-dac4-497c-8502-af17d945e67d

GET http://localhost:8989/payment/26

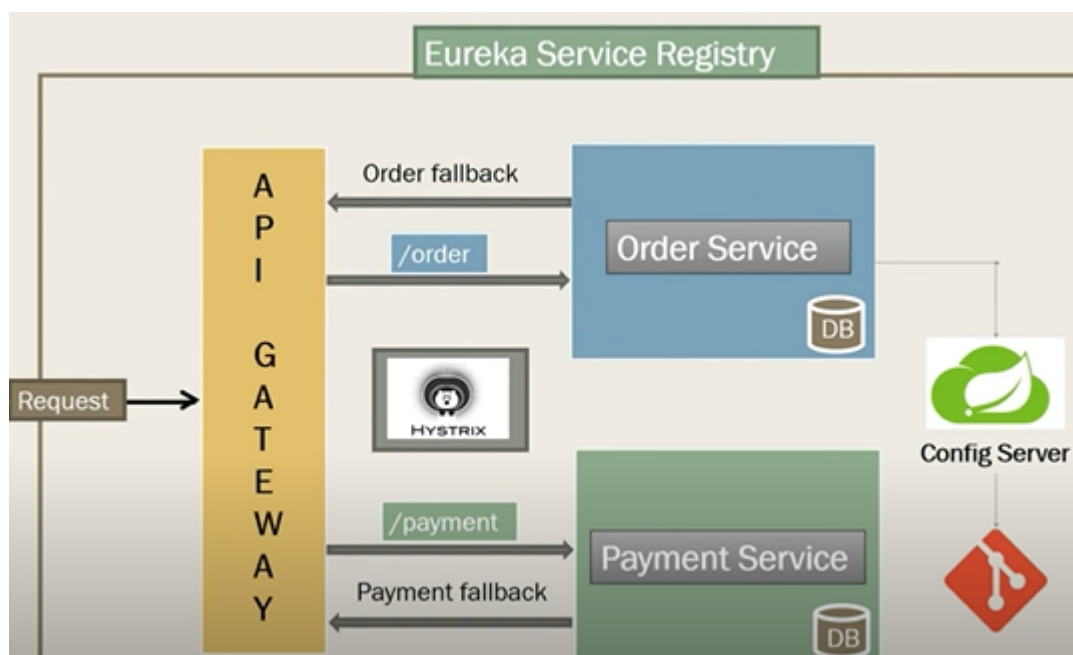
200 OK 309 ms 287 B

```
1 {
2   "paymentId": 1,
3   "paymentStatus": "false",
4   "transactionId": "2ed9fa71-dac4-497c-8502-af17d945e67d",
5   "orderId": 26,
6   "amount": 2500.0
7 }
```

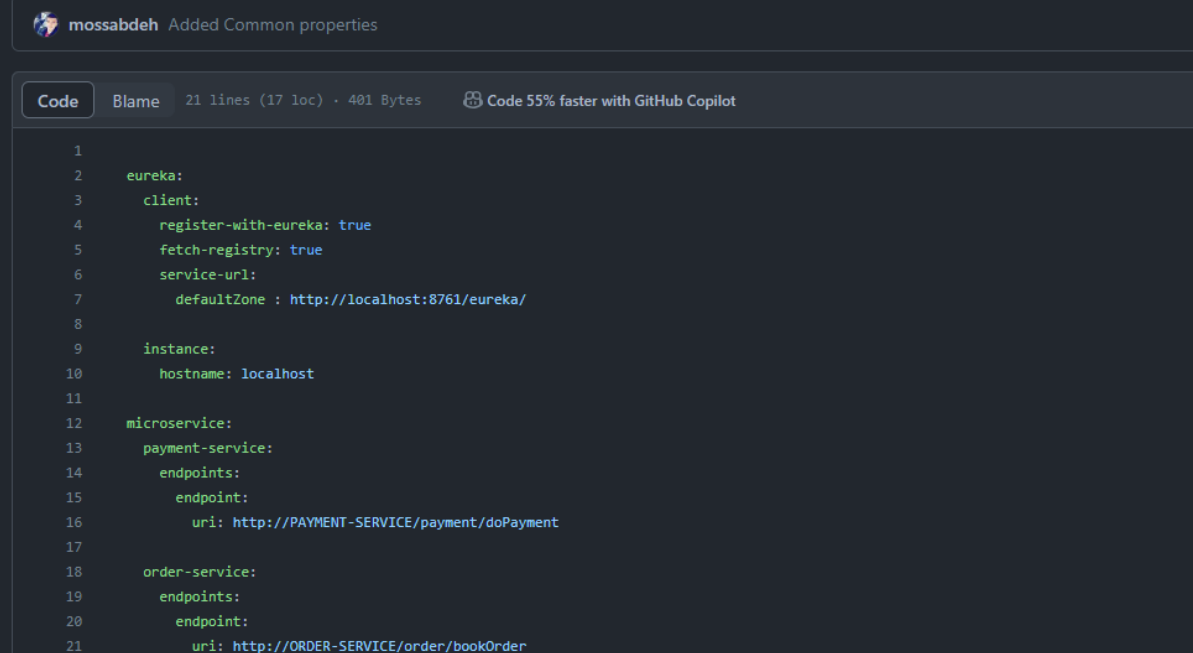
4 : Integrate Hystrix & Hystrix Dashboard to identify failure for downstream services :

Spring Cloud Hystrix project is deprecated. So new applications should not use this project. Resilience4j is a new option for Spring developers to implement the circuit breaker pattern. Resilience4j comes with other features like Rate Limiter, Retry and Bulkhead along with Circuit Breaker pattern. 10 déc. 2019

5 : Spring cloud config server using Git to Centralize configuration across applications :



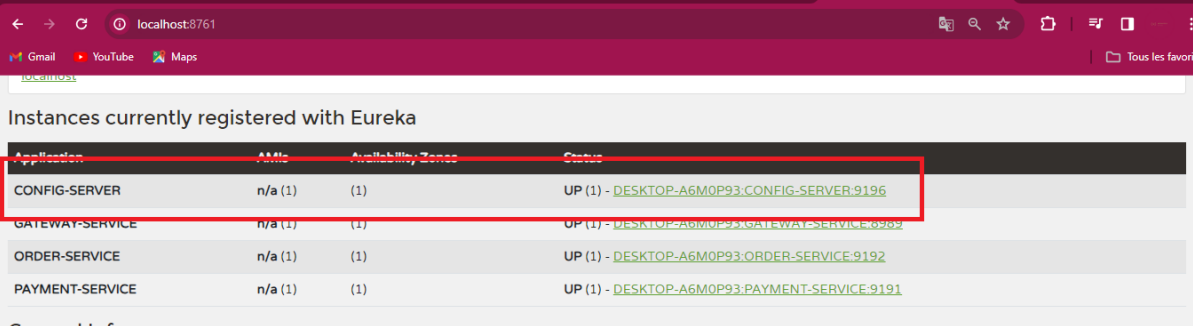
Let's establish a Git repository to centralize configuration using Spring Cloud Config :



The screenshot shows a code editor with a dark theme. At the top, a notification bar says "mossabdeh Added Common properties". Below it, the editor displays a configuration file with the following content:

```
1
2   eureka:
3     client:
4       register-with-eureka: true
5       fetch-registry: true
6       service-url:
7         defaultZone : http://localhost:8761/eureka/
8
9     instance:
10      hostname: localhost
11
12  microservice:
13    payment-service:
14      endpoints:
15        endpoint:
16          uri: http://PAYMENT-SERVICE/payment/doPayment
17
18    order-service:
19      endpoints:
20        endpoint:
21          uri: http://ORDER-SERVICE/order/bookOrder
```

Let's register our spring cloud config instance with Eureka :



The screenshot shows a web browser window with the URL "localhost:8761". The page title is "EUREKA". The main content area is titled "Instances currently registered with Eureka". Below this title is a table with the following data:

Application	Instances	Availability Zones	Status
CONFIG-SERVER	n/a (1)	(1)	UP (1) - DESKTOP-A6M0P93-CONFIG-SERVER-9196
GATEWAY-SERVICE	n/a (1)	(1)	UP (1) - DESKTOP-A6M0P93-GATEWAY-SERVICE-8988
ORDER-SERVICE	n/a (1)	(1)	UP (1) - DESKTOP-A6M0P93-ORDER-SERVICE-9192
PAYMENT-SERVICE	n/a (1)	(1)	UP (1) - DESKTOP-A6M0P93-PAYMENT-SERVICE-9191

Below the table, there is a section titled "General Info".

Let's check whether our microservices have connected successfully using spring cloud config :

HTTP <http://localhost:8989/order/bookOrder> Save

POST <http://localhost:8989/order/bookOrder> Send

Params Authorization Headers (8) **Body** Pre-request Script Tests Settings Cookies Beautify

none form-data x-www-form-urlencoded raw binary JSON

```
1 {
2   "order": {
3     "id": 26,
4     "name": "EarPhones",
5     "qty": 25,
6     "price": 2500
7   },
8   "payment": {}
9 }
```

Body Cookies Headers (5) Test Results

Status: 200 OK Time: 1530 ms Size: 355 B Save Response

Pretty Raw Preview Visualize JSON

```
1 {
2   "order": {
3     "id": 26,
4     "name": "EarPhones",
5     "qty": 25,
6     "price": 2500.0
7   },
8   "amount": 2500.0,
9 }
```

Activer Windows
Accédez aux paramètres pour activer Windows.

HTTP <http://localhost:8989/payment/26> Save

GET <http://localhost:8989/payment/26> Send

Params Authorization Headers (8) **Body** Pre-request Script Tests Settings Cookies

Query Params

Key	Value	Bulk Edit
Key	Value	

Body Cookies Headers (5) Test Results

Status: 200 OK Time: 295 ms Size: 289 B Save Response

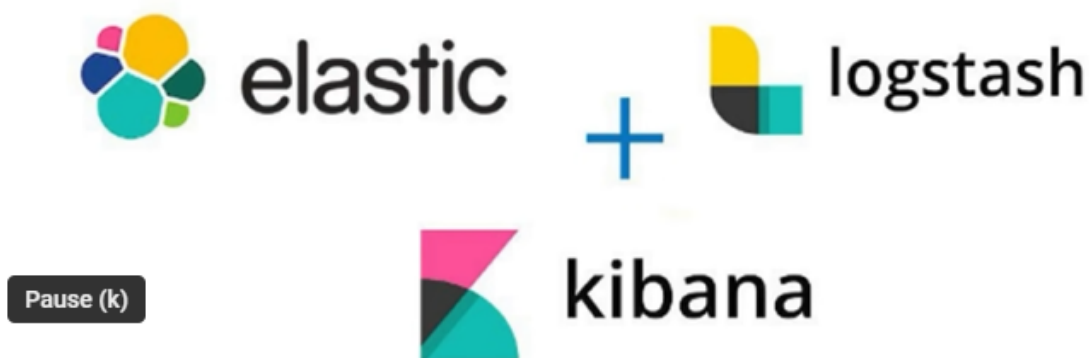
Pretty Raw Preview Visualize JSON

```
1 {
2   "paymentId": 1,
3   "paymentStatus": "success",
4   "transactionId": "4af450a5-2b57-408b-a5a1-65b92cf714c2",
5   "orderId": 26,
6   "amount": 2500.0
7 }
```

Activer Windows
Accédez aux paramètres pour activer Windows.

6 : *ELK Stack to centralize logging across all microservices. :*

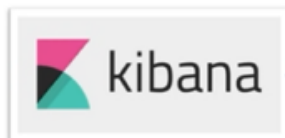
What is ELK ?



Elasticsearch is a NoSQL database that is based on the Lucene search engine which will helps us to store inputs/logs



Logstash is a log pipeline tool that accepts inputs/logs from various sources, & exports the data to various targets



Kibana is a visualization UI layer , which will helps developer to monitor application logs



Let's start with Elasticsearch :
turn on server :

```

Invite de commandes - elasticsearch
https://www.elastic.co/guide/en/elasticsearch/reference/8.11/important-settings.html#initial_master_nodes
[2024-01-02T13:41:28,693][INFO ][o.e.c.s.MasterService] [DESKTOP-AGM0P93] elected-as-master ([1] nodes joined in term 4)[_FINISH_ELECTION_, {DESKTOP-AGM0P93}{NchruR
p4Qg5dm57vpBX1dA}{qqYIKeBrSEiRFubm71_TTg}{DESKTOP-AGM0P93}{127.0.0.1}{127.0.0.1:9300}{cdfhilmrstw}{8.11.3}{7000099-8500003} completing election], term: 4, version: 55,
delta: master node changed {previous [], current [{DESKTOP-AGM0P93}{NchruR4Qg5dm57vpBX1dA}{qqYIKeBrSEiRFubm71_TTg}{DESKTOP-AGM0P93}{127.0.0.1}{127.0.0.1:9300}{cdfhilmr
stw}{8.11.3}{7000099-8500003}}]
[2024-01-02T13:41:28,986][INFO ][o.e.c.s.ClusterApplierService] [DESKTOP-AGM0P93] master node changed {previous [], current [{DESKTOP-AGM0P93}{NchruR4Qg5dm57vpBX1dA}{q
qYIKeBrSEiRFubm71_TTg}{DESKTOP-AGM0P93}{127.0.0.1}{127.0.0.1:9300}{cdfhilmrstw}{8.11.3}{7000099-8500003}}], term: 4, version: 55, reason: Publication{term=4, version=55
}
[2024-01-02T13:41:29,079][INFO ][o.e.c.f.AbstractFileWatchingService] [DESKTOP-AGM0P93] starting file watcher ...
[2024-01-02T13:41:29,227][INFO ][o.e.c.c.NodeJoinExecutor] [DESKTOP-AGM0P93] node-join: [{DESKTOP-AGM0P93}{NchruR4Qg5dm57vpBX1dA}{qqYIKeBrSEiRFubm71_TTg}{DESKTOP-AGM0
P93}{127.0.0.1}{127.0.0.1:9300}{cdfhilmrstw}{8.11.3}{7000099-8500003}}] with reason [completing election]
[2024-01-02T13:41:29,237][INFO ][o.e.h.AbstractHttpServerTransport] [DESKTOP-AGM0P93] publish_address {172.28.32.1:9200}, bound_addresses {[::]:9200}
[2024-01-02T13:41:29,255][INFO ][o.e.n.Node] [DESKTOP-AGM0P93] started {DESKTOP-AGM0P93}{NchruR4Qg5dm57vpBX1dA}{qqYIKeBrSEiRFubm71_TTg}{DESKTOP-AGM0P93}
{127.0.0.1}{127.0.0.1:9300}{cdfhilmrstw}{8.11.3}{7000099-8500003}{ml.allocated_processors_double=4.0, ml.max_jvm_size=4223664128, ml.config_version=11.0.0, xpack.instal
led=true, transform.config_version=10.0.0, ml.machine_memory=8444350464, ml.allocated_processors=4}
[2024-01-02T13:41:29,294][INFO ][o.e.c.f.AbstractFileWatchingService] [DESKTOP-AGM0P93] file settings service up and running [tid=69]
[2024-01-02T13:41:30,076][INFO ][o.e.l.ClusterStateLicenseService] [DESKTOP-AGM0P93] license [a9a9458c-feed-4ca8-a616-d42d0ce4eb0c] mode [basic] - valid
[2024-01-02T13:41:30,127][INFO ][o.e.x.s.a.Realms] [DESKTOP-AGM0P93] license mode is [basic], currently licensed security realms are [reserved/reserved,file/de
fault,file,native/default,native]
[2024-01-02T13:41:30,139][INFO ][o.e.g.GatewayService] [DESKTOP-AGM0P93] recovered [1] indices into cluster_state
[2024-01-02T13:41:30,793][INFO ][o.e.i.m.MapperService] [DESKTOP-AGM0P93] [.security-7] reloading search analyzers
[2024-01-02T13:41:31,110][INFO ][o.e.h.n.s.HealthNodeTaskExecutor] [DESKTOP-AGM0P93] Node [{DESKTOP-AGM0P93}{NchruR4Qg5dm57vpBX1dA}] is selected as the current health
node.
[2024-01-02T13:41:31,124][INFO ][o.e.n.r.a.AllocationService] [DESKTOP-AGM0P93] current.health="GREEN" message="Cluster health status changed from [RED] to [GREEN] (rea
son: [shards started [[.security-7][0]]], previous.health="RED" reason="shards started [[.security-7][0]]"
[2024-01-02T13:45:18,796][INFO ][o.e.x.s.a.f.FileUserPasswdStore] [DESKTOP-AGM0P93] users file [C:\Tools\elasticsearch-8.11.3-windows-x86_64\elasticsearch-8.11.3\config
\users] changed. updating users...
  
```

works fine on windows (default port of elastic search is 9200)
and the **system username = elastic** :

```

localhost:9200
Gmail YouTube Maps
{
  "error": {
    "root_cause": [
      {
        "type": "security_exception",
        "reason": "action [cluster:monitor/main] is unauthorized for user [mossab] with effective roles [], this action is granted by the cluster privileges [monitor,manage,all]"
      }
    ],
    "type": "security_exception",
    "reason": "action [cluster:monitor/main] is unauthorized for user [mossab] with effective roles [], this action is granted by the cluster privileges [monitor,manage,all]"
  },
  "status": 403
}
  
```

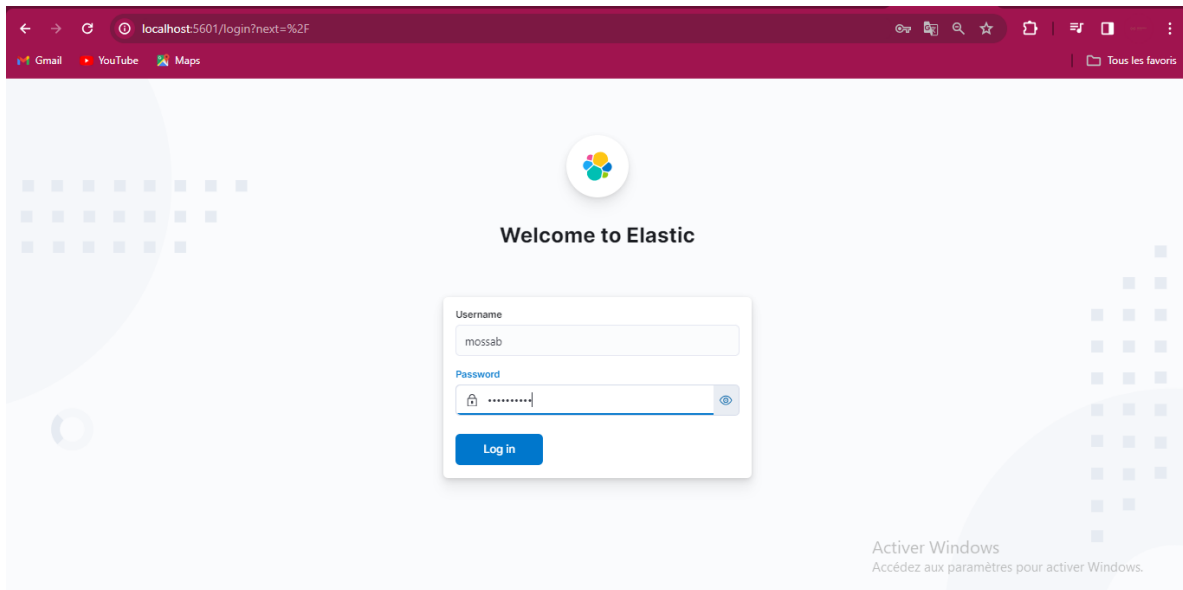
```
localhost:9200/_cat/indices
Raw Parsed
{
  "error": {
    "root_cause": [
      {
        "type": "security_exception",
        "reason": "action [indices:monitor/settings/get] is unauthorized for user [mossab] with effective roles [], this action is granted by the index privileges [monitor,view_index_metadata,manage,all]",
        "suppressed": [
          {
            "type": "security_exception",
            "reason": "action [cluster:monitor/state] is unauthorized for user [mossab] with effective roles [], this action is granted by the cluster privileges [read_ccr,transport_client,cross_cluster_replication,manage_ccr,monitor,manage,all]"
          },
          {
            "type": "security_exception",
            "reason": "action [indices:monitor/stats] is unauthorized for user [mossab] with effective roles [], this action is granted by the index privileges [monitor,cross_cluster_replication,manage,all]"
          }
        ]
      }
    ],
    "type": "security_exception",
    "reason": "action [indices:monitor/settings/get] is unauthorized for user [mossab] with effective roles [], this action is granted by the index privileges [monitor,view_index_metadata,manage,all]"
  },
  "suppressed": [
    {
      "type": "security_exception",
      "reason": "action [cluster:monitor/state] is unauthorized for user [mossab] with effective roles [], this action is granted by the cluster privileges [read_ccr,transport_client,cross_cluster_replication,manage_ccr,monitor,manage,all]"
    },
    {
      "type": "security_exception",
      "reason": "action [indices:monitor/stats] is unauthorized for user [mossab] with effective roles [], this action is granted by the index privileges [monitor,cross_cluster_replication,manage,all]"
    }
  ]
}
```

Activier Windows
Accédez aux paramètres pour activer Windows.

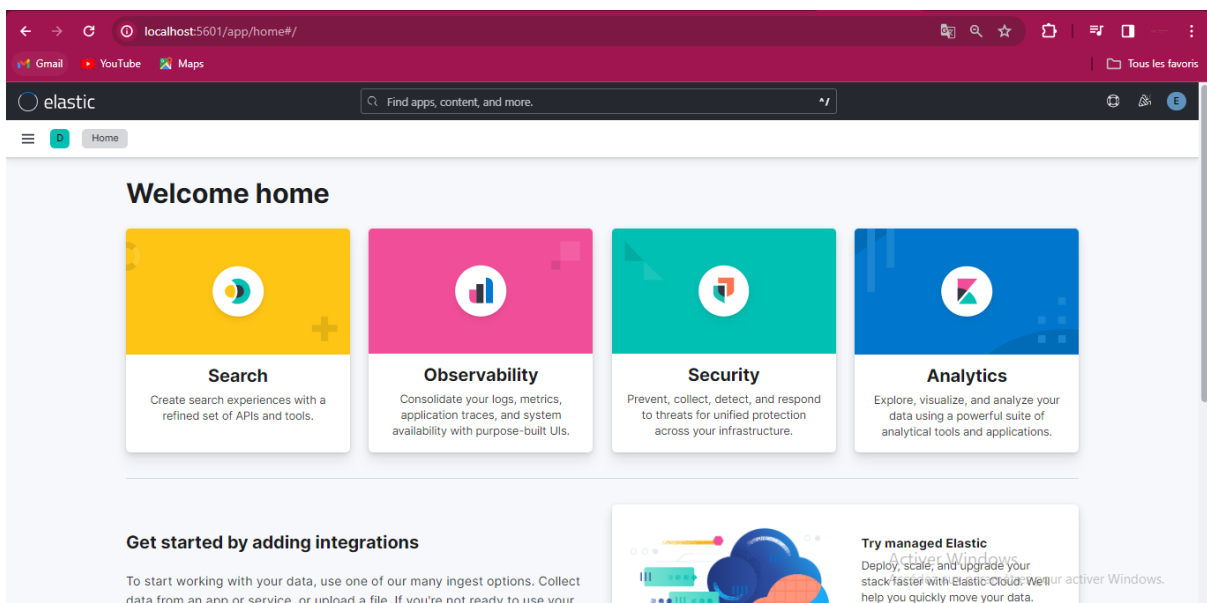
Let's Configure Kibana Now : first start the web server kibana.bat

```
[2024-01-02T15:04:58.536+01:00] [INFO] [[plugins.observability]] Installing SLO ingest pipeline [slo-observability.slo.pipeline]
[2024-01-02T15:04:58.990+01:00] [INFO] [[plugins.alerting]] Installing component template .alerts-stack.alerts-mappings
[2024-01-02T15:04:58.993+01:00] [INFO] [[plugins.alerting]] Installing component template .alerts-observability.slo.alerts-mappings
[2024-01-02T15:04:58.996+01:00] [INFO] [[plugins.alerting]] Installing component template .alerts-observability.threshold.alerts-mappings
[2024-01-02T15:04:58.999+01:00] [INFO] [[plugins.alerting]] Installing component template .alerts-ml.anomaly-detection.alerts-mappings
[2024-01-02T15:04:59.003+01:00] [INFO] [[plugins.alerting]] Installing component template .alerts-observability.uptime.alerts-mappings
[2024-01-02T15:04:59.006+01:00] [INFO] [[plugins.alerting]] Installing component template .alerts-observability.logs.alerts-mappings
[2024-01-02T15:04:59.009+01:00] [INFO] [[plugins.alerting]] Installing component template .alerts-observability.metrics.alerts-mappings
[2024-01-02T15:04:59.012+01:00] [INFO] [[plugins.alerting]] Installing component template .alerts-security.alerts-mappings
[2024-01-02T15:04:59.015+01:00] [INFO] [[plugins.alerting]] Installing component template .alerts-observability.apm.alerts-mappings
[2024-01-02T15:04:59.649+01:00] [INFO] [[plugins.alerting]] Installing index template .alerts-stack.alerts-default-index-template
[2024-01-02T15:04:59.652+01:00] [INFO] [[plugins.alerting]] Installing index template .alerts-observability.slo.alerts-default-index-template
[2024-01-02T15:04:59.655+01:00] [INFO] [[plugins.alerting]] Installing index template .alerts-observability.threshold.alerts-default-index-template
[2024-01-02T15:04:59.658+01:00] [INFO] [[plugins.alerting]] Installing index template .alerts-ml.anomaly-detection.alerts-default-index-template
[2024-01-02T15:04:59.661+01:00] [INFO] [[plugins.alerting]] Installing index template .alerts-observability.uptime.alerts-default-index-template
[2024-01-02T15:04:59.664+01:00] [INFO] [[plugins.alerting]] Installing index template .alerts-observability.metrics.alerts-default-index-template
[2024-01-02T15:04:59.667+01:00] [INFO] [[plugins.alerting]] Installing index template .alerts-observability.logs.alerts-default-index-template
[2024-01-02T15:04:59.670+01:00] [INFO] [[plugins.alerting]] Installing index template .alerts-security.alerts-default-index-template
[2024-01-02T15:04:59.673+01:00] [INFO] [[plugins.alerting]] Installing index template .alerts-observability.apm.alerts-default-index-template
[2024-01-02T15:05:00.628+01:00] [INFO] [[plugins.ruleregistry]] Installing component template .preview.alerts-security.alerts-mappings
[2024-01-02T15:05:01.892+01:00] [INFO] [[plugins.alerting]] Creating concrete write index - .internal.alerts-observability.slo.alerts-default-000001
[2024-01-02T15:05:01.895+01:00] [INFO] [[plugins.alerting]] Creating concrete write index - .internal.alerts-ml.anomaly-detection.alerts-default-000001
[2024-01-02T15:05:01.900+01:00] [INFO] [[plugins.alerting]] Creating concrete write index - .internal.alerts-observability.logs.alerts-default-000001
[2024-01-02T15:05:01.904+01:00] [INFO] [[plugins.alerting]] Creating concrete write index - .internal.alerts-stack.alerts-default-000001
[2024-01-02T15:05:01.908+01:00] [INFO] [[plugins.alerting]] Creating concrete write index - .internal.alerts-observability.metrics.alerts-default-000001
[2024-01-02T15:05:01.912+01:00] [INFO] [[plugins.alerting]] Creating concrete write index - .internal.alerts-observability.threshold.alerts-default-000001
[2024-01-02T15:05:01.916+01:00] [INFO] [[plugins.alerting]] Creating concrete write index - .internal.alerts-observability.uptime.alerts-default-000001
[2024-01-02T15:05:01.919+01:00] [INFO] [[plugins.alerting]] Creating concrete write index - .internal.alerts-security.alerts-default-000001
[2024-01-02T15:05:01.922+01:00] [INFO] [[plugins.alerting]] Creating concrete write index - .internal.alerts-observability.apm.alerts-default-000001
[2024-01-02T15:05:06.438+01:00] [INFO] [[plugins.observability]] Installing SLO ingest pipeline [slo-observability.summary.pipeline]
[2024-01-02T15:05:06.700+01:00] [WARN] [[plugins.fleet]] spack.encryptedSavedObjects.encryptedKey is not configured, private key passphrase is being stored in memory
[2024-01-02T15:05:07.527+01:00] [WARN] [[plugins.fleet]] spack.encryptedSavedObjects.encryptedKey is not configured, agent uninstall tokens are being stored in memory
[2024-01-02T15:05:07.620+01:00] [INFO] [[plugins.observabilityYuiAssistant.service]] Creating concrete write index - .kibana-observability-ai-assistant-kb-000001
[2024-01-02T15:05:07.668+01:00] [INFO] [[plugins.fleet]] Fleet setup completed
[2024-01-02T15:05:07.671+01:00] [INFO] [[plugins.securitySolution]] Dependent plugin setup complete - Starting ManifestTask
[2024-01-02T15:05:07.675+01:00] [INFO] [[plugins.securitySolution.endpoint.policyProtection]] App feature [endpoint_policy_protections] is enabled. Nothing to do
[2024-01-02T15:05:08.140+01:00] [INFO] [[plugins.observabilityYuiAssistant.service]] Successfully set up index assets
[2024-01-02T15:05:09.438+01:00] [INFO] [[plugins.observability]] Installing SLO summary transform [slo-summary-occurrences-7d-rolling]
[2024-01-02T15:05:11.865+01:00] [INFO] [[plugins.securitySolution.endpoint.metadataCheck.Transform.task0.0.1]] no endpoint installation found
[2024-01-02T15:05:14.807+01:00] [INFO] [[plugins.observability]] Starting SLO summary transform [slo-summary-occurrences-7d-rolling]
[2024-01-02T15:05:10.749+01:00] [INFO] [[plugins.observability]] Installing SLO summary transform [slo-summary-occurrences-30d-rolling]
[2024-01-02T15:05:18.016+01:00] [INFO] [[plugins.observability]] Starting SLO summary transform [slo-summary-occurrences-30d-rolling]
[2024-01-02T15:05:18.932+01:00] [INFO] [[plugins.observability]] Installing SLO summary transform [slo-summary-occurrences-90d-rolling]
```

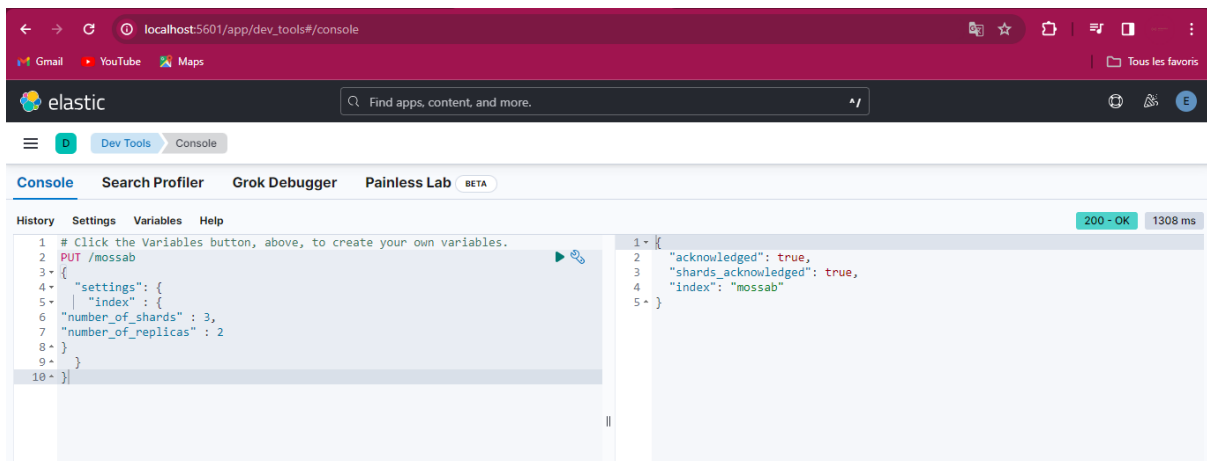
Configure kabana.yml using built in user = **kibana_system**
and go to <http://localhost:5601/> you should see that login interface :



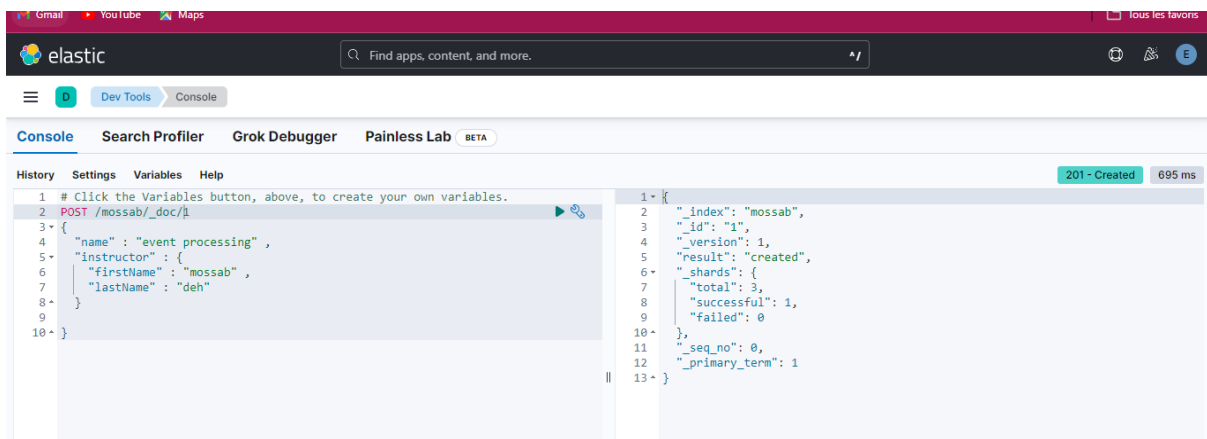
you can't access using custom user , you need to access as root system user which is : **elasticsearch.username: "elastic"**



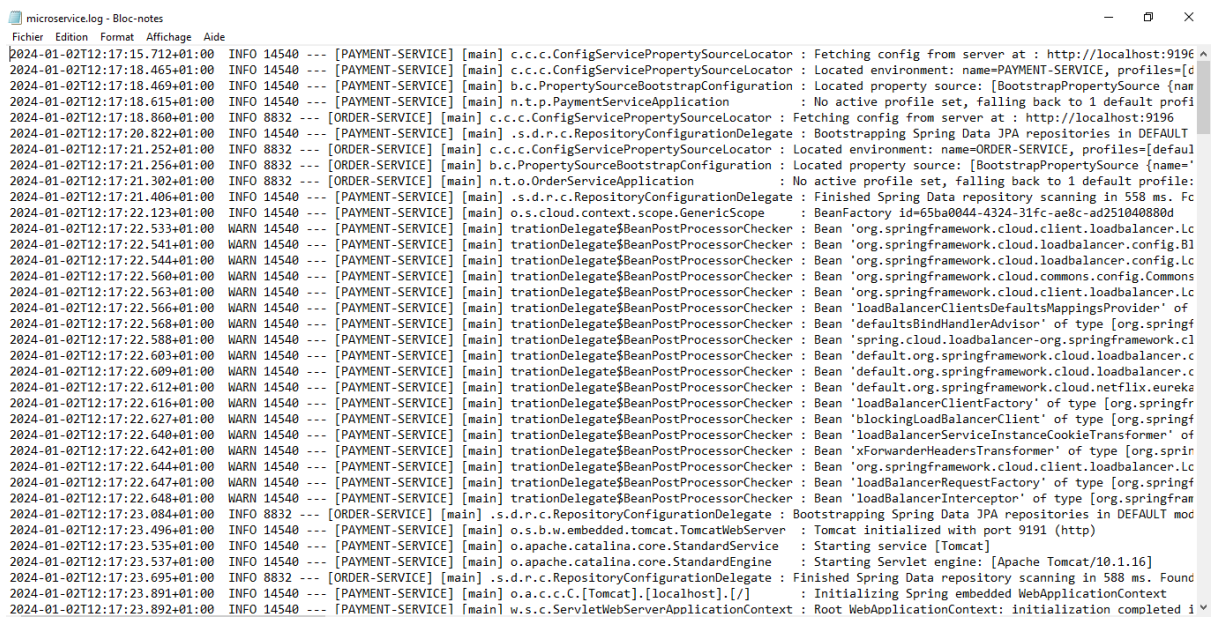
Let's create index : using kibana dev tools



Let's add document to this index :



Let's Configure LogStash Now :
we have already log from our microservices :



Let's start the logstash web server :

```
C:\Windows\System32\cmd.exe - logstash.bat -f C:\Tools\logstash-8.11.3-windows-x86_64\logstash-8.11.3\config\logstash.yml

ists - C:/Users/Mossab/Desktop/log/microservice.log>, :backtrace=>["C:/Tools/logstash-8.11.3-windows-x86_64/logstash-8.
11.3/logstash-core/lib/logstash/settings.rb:662:in `block in value'", "org/jruby/RubyKernel.java:2036:in `tap'", "C:/Too
ls/logstash-8.11.3-windows-x86_64/logstash-8.11.3/logstash-core/lib/logstash/settings.rb:654:in `value'", "C:/Tools/logs
tach-8.11.3-windows-x86_64/logstash-8.11.3/logstash-core/lib/logstash/settings.rb:136:in `get_value'", "C:/Tools/logstas
h-8.11.3-windows-x86_64/logstash-8.11.3/logstash-core/lib/logstash/environment.rb:147:in `block in LogStash'", "C:/Tools
/logstash-8.11.3-windows-x86_64/logstash-8.11.3/logstash-core/lib/logstash/settings.rb:197:in `block in post_process'",
"org/jruby/RubyArray.java:1989:in `each'", "C:/Tools/logstash-8.11.3-windows-x86_64/logstash-8.11.3/logstash-core/lib/lo
gstash/settings.rb:196:in `post_process'", "C:/Tools/logstash-8.11.3-windows-x86_64/logstash-8.11.3/logstash-core/lib/lo
gstash/util/settings_helper.rb:42:in `post_process'", "C:/Tools/logstash-8.11.3-windows-x86_64/logstash-8.11.3/logstash-
core/lib/logstash/runner.rb:291:in `execute'", "C:/Tools/logstash-8.11.3-windows-x86_64/logstash-8.11.3/vendor/bundle/jr
uby/3.1.0/gems/clamp-1.0.1/lib/clamp/command.rb:68:in `run'", "C:/Tools/logstash-8.11.3-windows-x86_64/logstash-8.11.3/l
ogstash-core/lib/logstash/runner.rb:287:in `run'", "C:/Tools/logstash-8.11.3-windows-x86_64/logstash-8.11.3/vendor/bundl
e/jruby/3.1.0/gems/clamp-1.0.1/lib/clamp/command.rb:133:in `run'", "C:\\Tools\\logstash-8.11.3-windows-x86_64\\logstash-
8.11.3\\lib\\bootstrap\\environment.rb:89:in `<main>'"]}]
[2024-01-02T18:11:34,650][FATAL][org.logstash.Logstash ] Logstash stopped processing because of an error: (SystemExit
) exit
org.jruby.exceptions.SystemExit: (SystemExit) exit
    at org.jruby.RubyKernel.exit(org/jruby/RubyKernel.java:808) ~[jruby.jar:?]
    at org.jruby.RubyKernel.exit(org/jruby/RubyKernel.java:767) ~[jruby.jar:?]
    at C_3a_Tools.logstash_minus_8_dot_11_dot_3_minus_windows_minus_x86_64.logstash_minus_8_dot_11_dot_3.lib.bootstrap
rap.environment.<main>(C:/Tools/logstash-8.11.3-windows-x86_64/logstash-8.11.3/lib/bootstrap/environment.rb:90) ~[?:?]

C:\Tools\logstash-8.11.3-windows-x86_64\logstash-8.11.3\bin>logstash.bat -f C:\Tools\logstash-8.11.3-windows-x86_64\logs
tach-8.11.3\config\logstash.yml
Using bundled JDK: C:\Tools\logstash-8.11.3-windows-x86_64\logstash-8.11.3\jdk\bin\java.exe"
C:/Tools/logstash-8.11.3-windows-x86_64/logstash-8.11.3/vendor/bundle/jruby/3.1.0/gems/concurrent-ruby-1.1.9/lib/concurr
ent-ruby/concurrent/executor/java_thread_pool_executor.rb:13: warning: method redefined; discarding old to_int
C:/Tools/logstash-8.11.3-windows-x86_64/logstash-8.11.3/vendor/bundle/jruby/3.1.0/gems/concurrent-ruby-1.1.9/lib/concurr
ent-ruby/concurrent/executor/java_thread_pool_executor.rb:13: warning: method redefined; discarding old to_f
```

7 : Zipkin & Sleuth to centralize tracing in microservice architecture. :

Let's start the zipkin web server using its JAR file :

```
C:\Users\Mossab\Downloads>java -jar zipkin-server-3.0.0-rc0-exec.jar

      OO
     OOOO
    OOOOOO
   OOOOOOOO
  OOOOOOOOOO
 OOOOOOOOOOOO
OOOOOOOOOOOO
OOOOOOO  OOOOOOOO
OOOOOOO  OOOOOOOO
OOOOOOO  OOOOOOOO
OOOOOOO  O  OOOOOO
OOOOOOO  OO OO OOOOOO
OOOOOOOO OO OO OOOOOOOO
OOOOOOO  OOOOO  OOOOOO OOOOOOOO
OOOOOOO  OOOOOO OOOOOO OOOOOOOO
OOOOOOOOO  OO OO OOOOOOOO
OOOOOOOOOOOOOO OO OO OOOOOOOOOOOOOO
OOOOOOOOOOOOOOO OOOOOOOOOOOOOO
OOOOOOOOO  OOOOOOOO
OOOO  OOOO

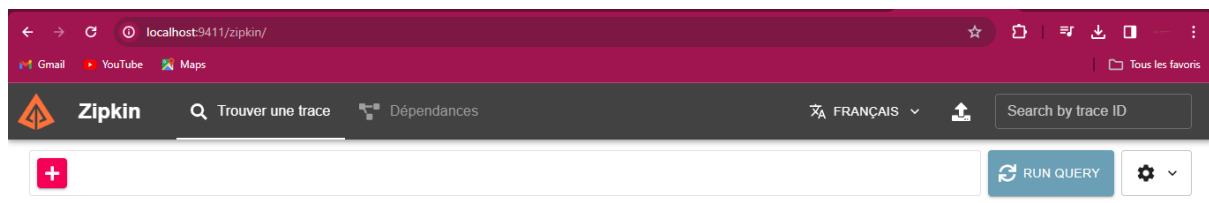
ZIPKIN

:: version 3.0.0-rc0 :: commit 1fc34ec ::

2024-01-02 22:08:37.843 WARN [/] 9996 --- [          main] s.c.a.AnnotationConfigApplicationContext : Exception encountered during co
nfig refresh attempt: org.springframework.context.ApplicationContextException: Failed to start bean 'ArmeriaServerGracefulShutdownLifecycle
concurrent.CompletionException: java.lang.IllegalStateException: Armeria server failed to start
2024-01-02 22:08:37.899 INFO [/] 9996 --- [          main] ConditionEvaluationReportLoggingListener :

Error starting ApplicationContext. To display the conditions report re-run your application with 'debug' enabled.
2024-01-02 22:08:37.990 ERROR [/] 9996 --- [          main] o.s.b.SpringApplication : Application run failed@ccédez au
```


we access to zipkin dashboard using the url : localhost:9411 :



Let's check if our zipkin working fine by accessing our microservices API :

