

Nom : AZZAIM Nisrine

Classe : 5IIR8

Etude de cas 1 :

```
2  /**
3  15 usages
4  @Entity
5  @Data
6  public class Commande {
7  no usages
8  @Id
9  @GeneratedValue
10 private Integer id ;
11 no usages
12 private String description ;
13 no usages
14 private Integer quantite ;
15 no usages
16 private LocalDate date ;
17 no usages
18 private Long montant ;
19 }
20
21
22
23
24
```

```
2  /**
3  2 usages
4  @Repository
5  public interface CommandeDao
6  extends JpaRepository<Commande, Integer> {
7  }
8
9
10
11
12
```

```

9      * @author azzai
10     **/
11     2 usages
12     @Component
13     @ConfigurationProperties("mes-configs-ms")
14     @RefreshScope
15     @Data
16     public class ApplicationPropertiesConfiguration {
17         no usages
18         private int commandes_last ;
19     }

```

```

@RestController
public class CommandeController implements HealthIndicator {

    8 usages
    @Autowired
    CommandeDao commandeDao ;
    1 usage
    @Autowired
    ApplicationPropertiesConfiguration appProps ;

    no usages
    @GetMapping("/Commandes")
    public List<Commande> getAllCommandes(){
        List<Commande> commandes = commandeDao.findAll();

        if(commandes.isEmpty())
            throw new RuntimeException("Aucune commande n'est disponible");

        List<Commande> liste_commandes_limitee = commandes.stream().filter(
            commande -> {
                LocalDate tenDaysAgo = LocalDate.now().minusDays(appProps.getCommandes_last());
                LocalDate commandeDate = commande.getDate();
                return !commandeDate.isBefore(tenDaysAgo) && !commandeDate.isAfter(LocalDate.now());
            }
        ).collect(Collectors.toList());
        return liste_commandes_limitee;
    }
}

```


```
43     no usages
44     @GetMapping("/{id}")
45     public Commande getCommandeById(@PathVariable Integer id){
46         Commande commande = commandeDao.findById(id).orElse( other: null);
47         if(commande==null) throw new RuntimeException("La commande correspondante à l'id "+ id + " n'existe pas :(");
48         return commande;
49     }
50     no usages
51     @DeleteMapping("/{id}")
52     public void deleteCommande(@PathVariable Integer id){
53         Commande commande = commandeDao.findById(id).orElse( other: null);
54         if(commande==null) throw new RuntimeException("La commande correspondante à l'id "+ id + " n'existe pas :(");
55         commandeDao.delete(commande);
56     }
57     no usages
58     @PutMapping("/{id}")
59     public Commande updateCommande(@RequestBody Commande commande ,
60                                   @PathVariable Integer id){
61         Commande old_commande = commandeDao.findById(id).orElse( other: null);
62         if(commande==null) throw new RuntimeException("La commande correspondante à l'id "+ id + " n'existe pas :(");
63         return commandeDao.save(commande);
64     }
65     no usages
66     @PostMapping("/{id}")
67     public Commande createCommande(@RequestBody Commande commande) { return commandeDao.save(commande); }
68
69     @Override
70     public Health health() {
71         List<Commande> commandes = commandeDao.findAll();
72         if(commandes.isEmpty())
73             return Health.down().build();
74     }
```

```
57     no usages
58     @PutMapping("/{id}")
59     public Commande updateCommande(@RequestBody Commande commande ,
60                                   @PathVariable Integer id){
61         Commande old_commande = commandeDao.findById(id).orElse( other: null);
62         if(commande==null) throw new RuntimeException("La commande correspondante à l'id "+ id + " n'existe pas :(");
63         return commandeDao.save(commande);
64     }
65     no usages
66     @PostMapping("/{id}")
67     public Commande createCommande(@RequestBody Commande commande) { return commandeDao.save(commande); }
68
69     @Override
70     public Health health() {
71         List<Commande> commandes = commandeDao.findAll();
72         if(commandes.isEmpty())
73             return Health.down().build();
74         return Health.up().build();
75     }
```

```
1  spring.application.name = microservice-commandes
2  spring.cloud.config.uri = http://localhost:9101
3
```

```
server.port = 9001
eureka.client.serverUrl.defaultZone=http://localhost:8761/eureka/
spring.jpa.show-sql = true
spring.h2.console.enabled = true
spring.datasource.url=jdbc:h2:mem:testdb
spring.datasource.driverClassName = org.h2.Driver
spring.datasource.username = sa
spring.datasource.password =
spring.jpa.database-platform = org.hibernate.dialect.H2Dialect
spring.jpa.hibernate.ddl-auto = update
mes-configs-ms.commandes_last = 20
management.endpoints.web.exposure.include=*
#management.endpoints.web.exposure.include=refresh
```

devoir-configs-repo / microservice-commandes.properties


 **nisrinazz** Update microservice-commandes.properties

63c6bc5 · 4

Code

Blame

13 lines (13 loc) · 529 Bytes

 Code 55% faster with GitHub Copilot

Raw

```
1  server.port = 9001
2  eureka.client.serverUrl.defaultZone=http://localhost:8761/eureka/
3  spring.jpa.show-sql = true
4  spring.h2.console.enabled = true
5  spring.datasource.url=jdbc:h2:mem:testdb
6  spring.datasource.driverClassName = org.h2.Driver
7  spring.datasource.username = sa
8  spring.datasource.password =
9  spring.jpa.database-platform = org.hibernate.dialect.H2Dialect
10 spring.jpa.hibernate.ddl-auto = update
11 mes-configs-ms.commandes_last = 20
12 management.endpoints.web.exposure.include=*
13 #management.endpoints.web.exposure.include=refresh
```

```
ConfigServerApplication.java ×
1 package com.devoir.configserver;
2
3 import ...
4
5 1 usage
6
7 @SpringBootApplication
8 @EnableConfigServer
9 public class ConfigServerApplication {
10
11     no usages
12     public static void main(String[] args) { SpringApplication.run(ConfigServerApplication.class, args); }
13
14 }
15
16
```

```
ConfigServerApplication.java × application.properties ×
1 spring.cloud.compatibility-verifier.enabled=false
2 spring.application.name=config-server
3 server.port=9101
4 spring.cloud.config.server.git.uri=https://github.com/nisrinazz/devoir-configs-repo.git
5 spring.cloud.config.server.git.default-label=main
6 logging.level.root=INFO
7 logging.level.com.mcommerce.config.server=INFO
8 logging.level.org.springframework.boot.web.embedded.tomcat=INFO
```

```

1  [
2    {
3      "id": 1,
4      "description": "desc2",
5      "quantite": 200,
6      "date": "2023-12-29",
7      "montant": 23
8    },
9    {
10     "id": 2,
11     "description": "desc2",
12     "quantite": 200,
13     "date": "2023-12-31",
14     "montant": 100
15   },
16   {
17     "id": 3,
18     "description": "desc3",
19     "quantite": 100,
20     "date": "2024-01-01",
21     "montant": 2200
22   },
23   {
24     "id": 6,
25     "description": "desc1",
26     "quantite": 100,
27     "date": "2024-01-01",
28     "montant": 22
29   },
30   {
31     "id": 7,
32     "description": "desc7",
33     "quantite": 300,
34     "date": "2023-12-27",
35     "montant": 22
36   },
37   {
38     "id": 8,
39     "description": "desc8",
40     "quantite": 100,
41     "date": "2024-01-05",
42     "montant": 22
43   },
44   {
45     "id": 9,
46     "description": "desc9",
47     "quantite": 200,
48     "date": "2024-01-04",
49     "montant": 22
50   },
51   {
52     "id": 10,
53     "description": "desc10",
54     "quantite": 100,
55     "date": "2024-01-01",
56     "montant": 22
57   },
58   {
59     "id": 11,
60     "description": "desc11",
61     "quantite": 200,
62     "date": "2024-01-02",
63     "montant": 22

```

```

{
  "id": 12,
  "description": "desc12",
  "quantite": 100,
  "date": "2024-01-01",
  "montant": 22
},
{
  "id": 13,
  "description": "desc13",
  "quantite": 200,
  "date": "2023-12-29",
  "montant": 22
},
{
  "id": 14,
  "description": "desc14",
  "quantite": 200,
  "date": "2023-12-29",
  "montant": 22
},
{
  "id": 15,
  "description": "desc15",
  "quantite": 200,
  "date": "2023-12-29",
  "montant": 22
},
{
  "id": 16,
  "description": "desc16",
  "quantite": 200,
  "date": "2023-12-29",
  "montant": 22
},
{
  "id": 17,
  "description": "desc17",
  "quantite": 200,
  "date": "2024-01-01",
  "montant": 22
},
{
  "id": 18,
  "description": "desc18",
  "quantite": 200,
  "date": "2023-12-29",
  "montant": 22
},
{
  "id": 19,
  "description": "desc19",
  "quantite": 200,
  "date": "2023-12-29",
  "montant": 22
},
{
  "id": 20,
  "description": "desc20",
  "quantite": 200,
  "date": "2023-12-29",
  "montant": 22
}

```

```
    },
    {
      "id": 18,
      "description": "desc18",
      "quantite": 200,
      "date": "2023-12-29",
      "montant": 22
    },
    {
      "id": 19,
      "description": "desc19",
      "quantite": 200,
      "date": "2023-12-29",
      "montant": 22
    },
    {
      "id": 20,
      "description": "desc20",
      "quantite": 200,
      "date": "2023-12-29",
      "montant": 22
    },
    {
      "id": 21,
      "description": "desc21",
      "quantite": 200,
      "date": "2023-12-29",
      "montant": 22
    }
  ]
}
```

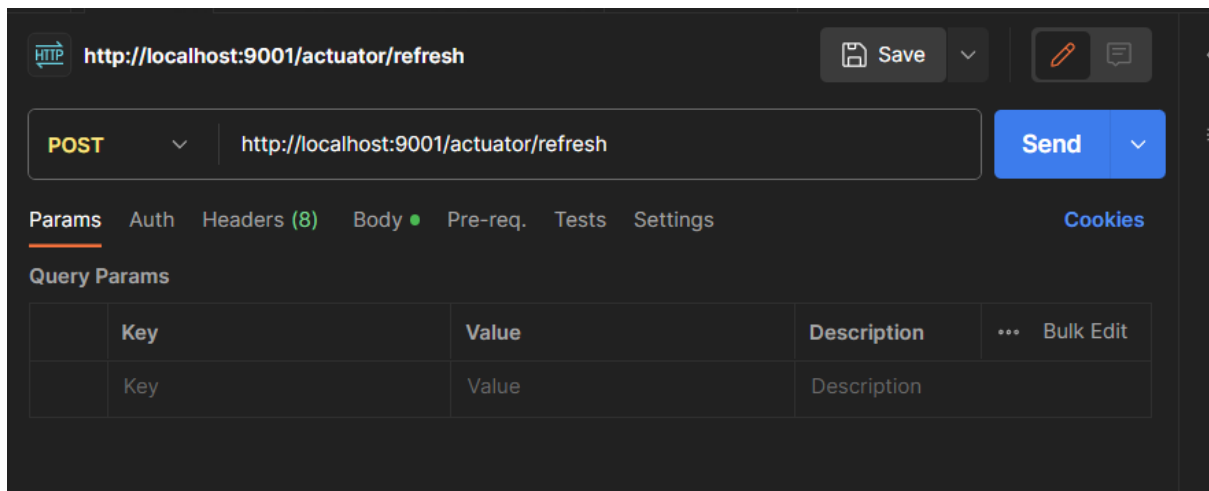
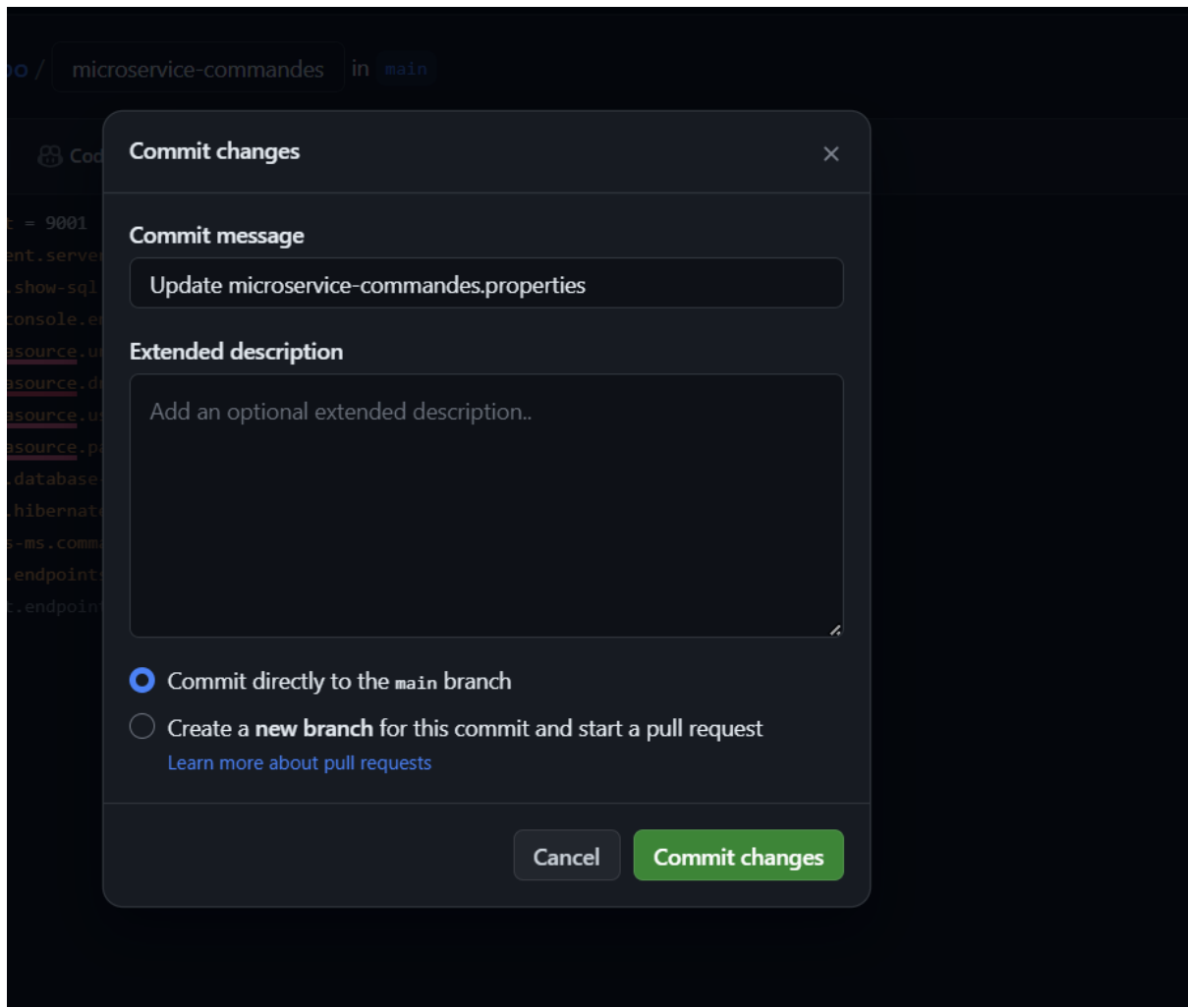
Edit

Preview



Code 55% faster with GitHub Copilot

```
1  server.port = 9001
2  eureka.client.serverUrl.defaultZone=http://localhost:8761/eureka/
3  spring.jpa.show-sql = true
4  spring.h2.console.enabled = true
5  spring.datasource.url=jdbc:h2:mem:testdb
6  spring.datasource.driverClassName = org.h2.Driver
7  spring.datasource.username = sa
8  spring.datasource.password =
9  spring.jpa.database-platform = org.hibernate.dialect.H2Dialect
10 spring.jpa.hibernate.ddl-auto = update
11 mes-configs-ms.commandes_last = 20
12 management.endpoints.web.exposure.include=*
13 #management.endpoints.web.exposure.include=refresh
14
```



POST

http://localhost:9001/actuator/refresh

Send

Params

Auth

Headers (8)

Body

Pre-req.

Tests

Settings

Cookies

Query Params

	Key	Value	Description	...	Bulk Edit
	Key	Value	Description		

Body

200 OK 2.82 s 249 B

Save as example

Pretty

Raw

Preview

Visualize

JSON

1

["mes-configs-ms.commandes_last","config.client.version"]

```
1 [
2   {
3     "id": 1,
4     "description": "desc2",
5     "quantite": 200,
6     "date": "2023-12-29",
7     "montant": 23
8   },
9   {
10    "id": 2,
11    "description": "desc2",
12    "quantite": 200,
13    "date": "2023-12-31",
14    "montant": 100
15  },
16  {
17    "id": 3,
18    "description": "desc3",
19    "quantite": 100,
20    "date": "2024-01-01",
21    "montant": 2200
22  },
23  {
24    "id": 6,
25    "description": "desc1",
26    "quantite": 100,
27    "date": "2024-01-01",
28    "montant": 22
29  },
30  {
31    "id": 7,
32    "description": "desc7",
33    "quantite": 300,
34    "date": "2023-12-27",
35    "montant": 22
36  },
37  {
38    "id": 8,
39    "description": "desc8",
40    "quantite": 100,
41    "date": "2024-01-05",
42    "montant": 22
43  },
44  {
45    "id": 9,
46    "description": "desc9",
47    "quantite": 200,
48    "date": "2024-01-04",
49    "montant": 22
50  },
51  {
52    "id": 10,
53    "description": "desc10",
54    "quantite": 100,
55    "date": "2024-01-01",
56    "montant": 22
57  },
58  {
59    "id": 11,
60    "description": "desc11",
61    "quantite": 200,
62    "date": "2024-01-02",
63    "montant": 22
64  }
65 ]
```

```
58 {
59   "id": 11,
60   "description": "desc11",
61   "quantite": 200,
62   "date": "2024-01-02",
63   "montant": 22
64 }
65
66   "id": 12,
67   "description": "desc12",
68   "quantite": 100,
69   "date": "2024-01-01",
70   "montant": 22
71 },
72 {
73   "id": 13,
74   "description": "desc13",
75   "quantite": 200,
76   "date": "2023-12-29",
77   "montant": 22
78 }
79 {
80   "id": 14,
81   "description": "desc14",
82   "quantite": 200,
83   "date": "2023-12-29",
84   "montant": 22
85 },
86 {
87   "id": 15,
88   "description": "desc15",
89   "quantite": 200,
90   "date": "2023-12-29",
91   "montant": 22
92 },
93 {
94   "id": 16,
95   "description": "desc16",
96   "quantite": 200,
97   "date": "2023-12-29",
98   "montant": 22
99 },
100 {
101   "id": 17,
102   "description": "desc17",
103   "quantite": 200,
104   "date": "2024-01-01",
105   "montant": 22
106 },
107 {
108   "id": 18,
109   "description": "desc18",
110   "quantite": 200,
111   "date": "2023-12-29",
112   "montant": 22
113 },
114 {
115   "id": 19,
116   "description": "desc19",
117   "quantite": 200,
118   "date": "2023-12-29",
119   "montant": 22
120 },
121 }
```

```

122 {
123   "id": 19,
124   "description": "desc19",
125   "quantite": 200,
126   "date": "2023-12-29",
127   "montant": 22
128 }
129
130   "id": 20,
131   "description": "desc20",
132   "quantite": 200,
133   "date": "2023-12-29",
134   "montant": 22
135 },
136 {
137   "id": 21,
138   "description": "desc21",
139   "quantite": 200,
140   "date": "2023-12-29",
141   "montant": 22
142 },
143 {
144   "id": 22,
145   "description": "desc22",
146   "quantite": 200,
147   "date": "2023-12-18",
148   "montant": 22
149 },
150 {
151   "id": 23,
152   "description": "desc23",
153   "quantite": 200,
154   "date": "2023-12-19",
155   "montant": 22
156 },
157 {
158   "id": 24,
159   "description": "desc23",
160   "quantite": 200,
161   "date": "2023-12-17",
162   "montant": 22
163 },
164 {
165   "id": 25,
166   "description": "desc23",
167   "quantite": 200,
168   "date": "2023-12-22",
169   "montant": 22
170 }
171 }
```

localhost:9001/actuator/health

1 {
2 "status": "UP"
3 }

jdbc:h2:mem:testdb

+

COMMANDE

+

INFORMATION_SCHEMA

+

Sequences

+

Users

i

H2 2.2.224 (2023-09-17)

Run

Run Selected

Auto complete

Clear

SQL statement:

delete from commande

delete from commande;
Update count: 28
(4 ms)

localhost:9001/actuator/health

1 {
2 "status": "DOWN"
3 }