

Documentation Utilisateur

Projet de Java Avancé
Master 1 - 2020

Local-Kube

SOMMAIRE

Préambule.....	3
Lancer l'application Local-kube.....	3
Requêtes.....	4
Erreurs possibles.....	9

PRÉAMBULE

Ce document est la documentation utilisateur du projet de Java M1 informatique : Local-kube. Si une information n'est pas claire, vous pouvez vous référer à la documentation technique du code : [API doc](#) (si le lien ne fonctionne pas : allez dans 'portefaix-redon/docs/apidocs/index.html'). Vous pouvez aussi vous référer à la documentation technique (allez dans 'portefaix-redon/docs/documentation technique.pdf') afin de voir les détails d'implémentation.

LANCER L'APPLICATION LOCAL-KUBE

Pour lancer l'application local-kube, vous pouvez lancer la commande à la racine du projet :

```
`$ java -jar --enable-preview local-kube.jar`
```

Notes pour la compilation des sources, il faut se placer dans le répertoire apps/local-kube/ et lancer la commande :

```
`$ mvn install -DskipTests -Drun.jvmArguments="-Xlint:all",`
```

puis il faut déplacer le fichier .jar vers la racine avec :

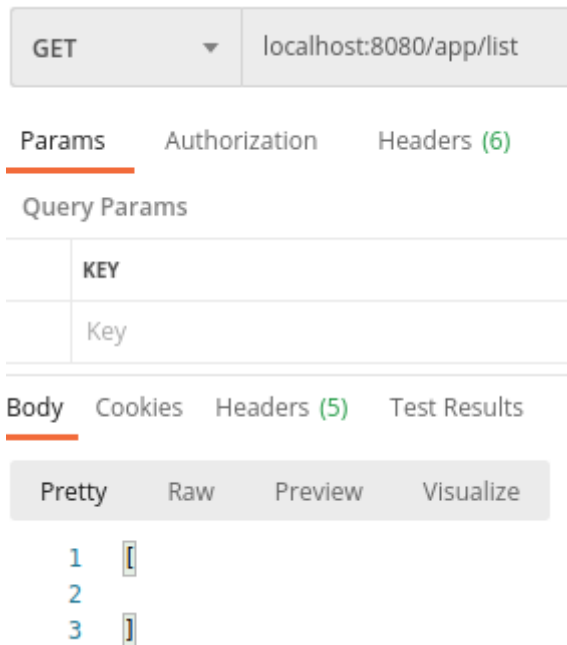
```
`$ mv target/local-kube-0.0.1-SNAPSHOT.jar ../../local-kube.jar`
```

REQUÊTES

L'application de test pour illustrer ici est helloworld.jar situé dans le répertoire apps. C'est dans ce même répertoire que d'autres applications '.jar' peuvent être ajoutées pour plus de test.

- List : `GET localhost:8080/apps/list` : liste les applications lancées.

Attendu sur Postman sans application start :

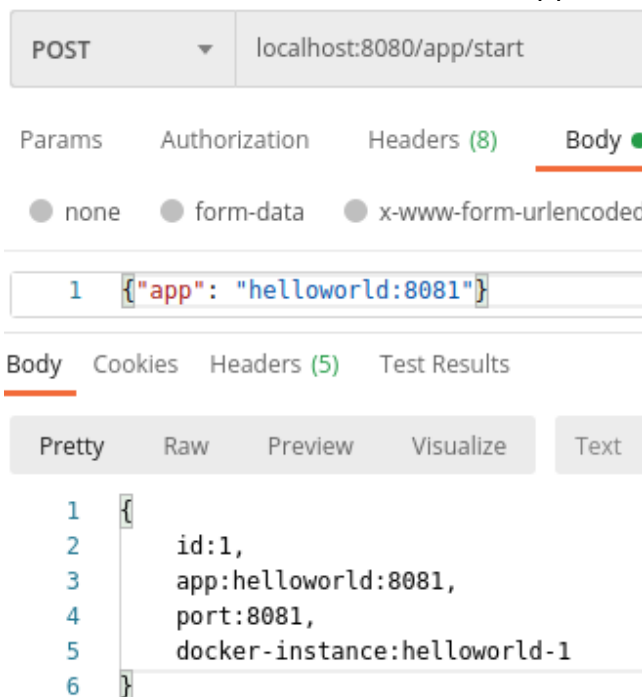


Attendu sur Postman avec 2 applications helloworld start :

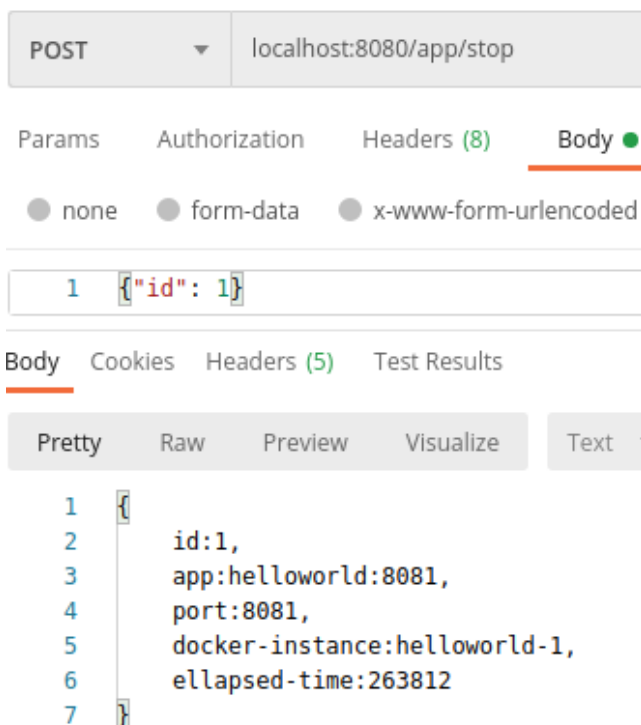
```
1  [
2  {
3      id:1,
4      app:helloworld:8081,
5      port:8081,
6      docker-instance:helloworld-1
7  },
8  {
9      id:2,
10     app:helloworld:8082,
11     port:8082,
12     docker-instance:helloworld-2
13  }
14 ]
```

- Start : `POST localhost:8080/apps/start` (with body : ...) : lance un conteneur docker d'une application.

Attendu sur Postman en startant une application helloworld :



- Stop : `POST localhost:8080/apps/stop` (with body : ...) : stop un conteneur
- Attendu sur Postman en stoppant une application :



- StopAll : `POST localhost:8080/apps/stopall` : stop tout les conteneurs
Attendu sur Postman en stoppant toutes les applications :

POST localhost:8080/app/stopall

Params Authorization Headers (7) Body

Query Params

KEY
Key

Body Cookies Headers (5) Test Results

Pretty Raw Preview Visualize Text

```

1 [{
2   id:2,
3   app:helloworld:8082,
4   port:8082,
5   docker-instance:helloworld-2
6 }]

```

- Kill : `POST localhost:8080/apps/kill` (with body : ...) : détruit un conteneur
Attendu sur Postman en killant une application

POST localhost:8080/app/kill

Params Authorization Headers (8) Body ●

● none ● form-data ● x-www-form-urlencoded

1 [{"id": 1}]

Body Cookies Headers (5) Test Results

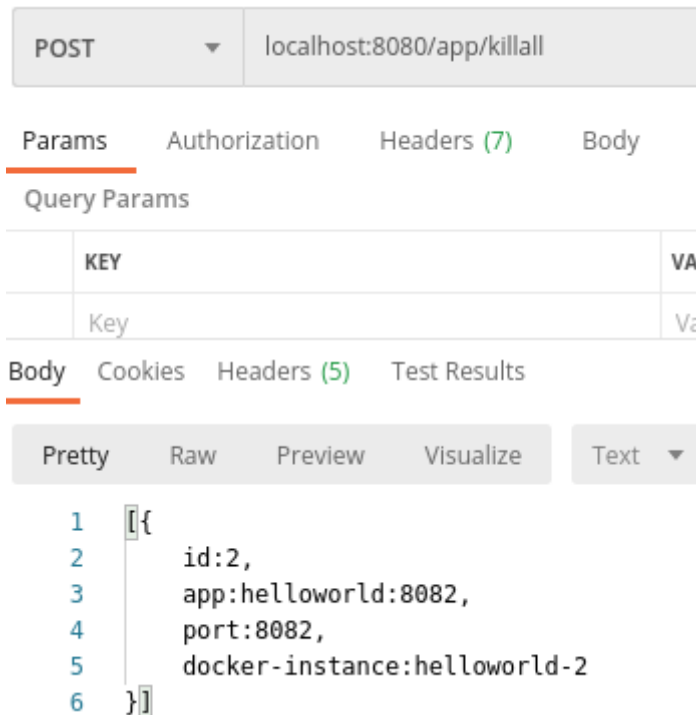
Pretty Raw Preview Visualize Text

```

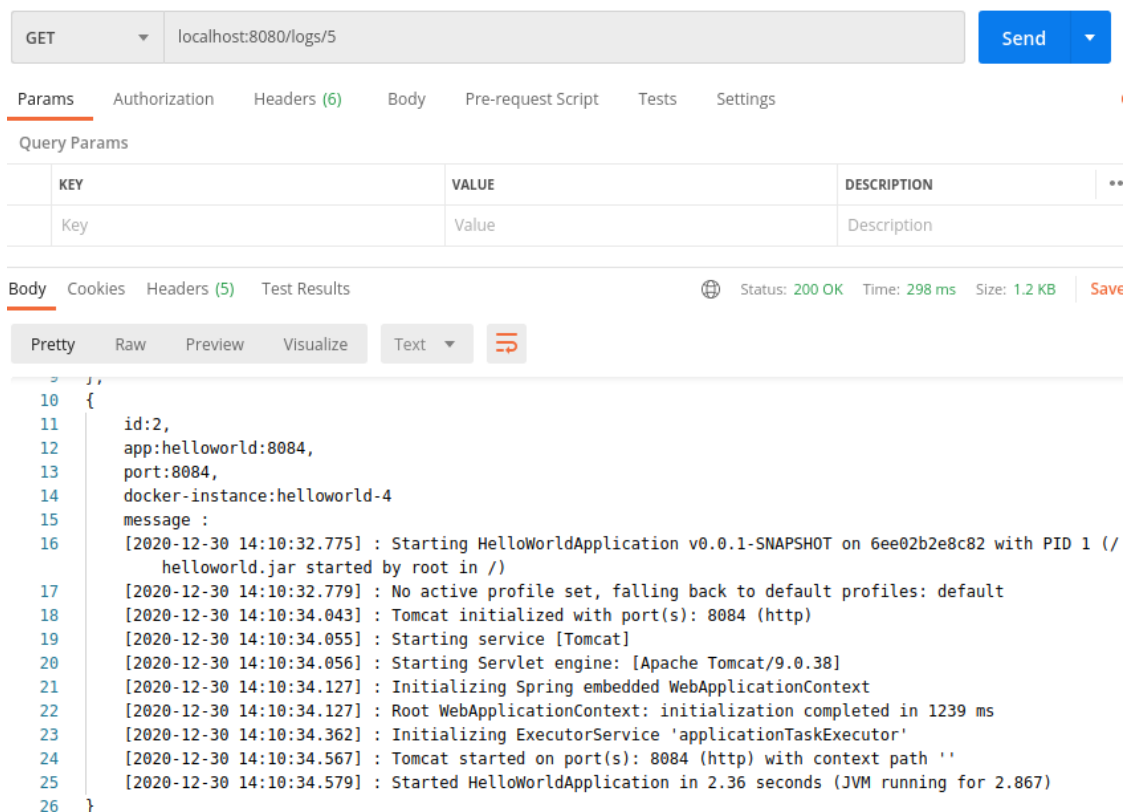
1 {
2   id:1,
3   app:helloworld:8081,
4   port:8081,
5   docker-instance:helloworld-1
6 }

```

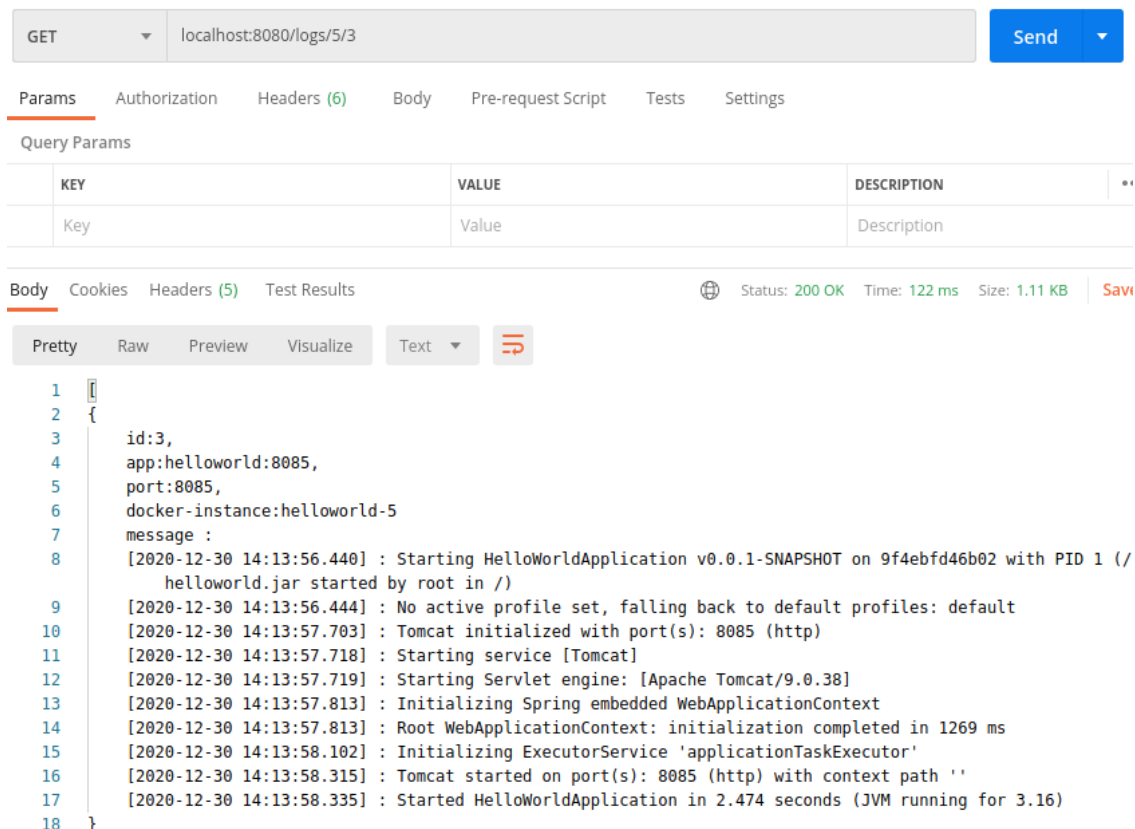
- KillAll : `POST localhost:8080/apps/killall` : détruit tout les conteneurs
Attendu sur Postman en killant toutes les applications :



- Log : `GET localhost:8080/logs/5` : affiche les logs depuis 5 minutes
Attendu sur Postman en regardant les logs sur les 5 dernières minutes :



- Log filter : `GET localhost:8080/logs/5/3` :
Attendu sur Postman en regardant les logs sur les 5 dernières minutes de l'application d'id 3 :



GET localhost:8080/logs/5/3

Params Authorization Headers (6) Body Pre-request Script Tests Settings

Query Params

KEY	VALUE	DESCRIPTION
Key	Value	Description

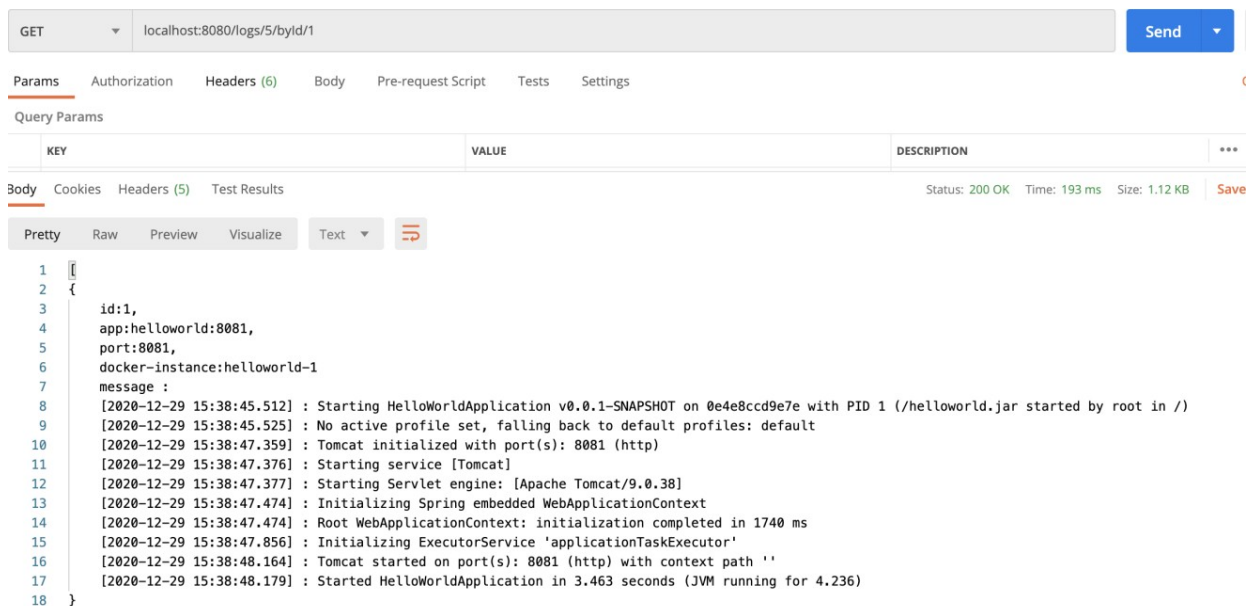
Body Cookies Headers (5) Test Results Status: 200 OK Time: 122 ms Size: 1.11 KB Save

Pretty Raw Preview Visualize Text

```

1 {
2   {
3     id:3,
4     app:helloworld:8085,
5     port:8085,
6     docker-instance:helloworld-5
7     message :
8     [2020-12-30 14:13:56.440] : Starting HelloWorldApplication v0.0.1-SNAPSHOT on 9f4ebfd46b02 with PID 1 (/
      helloworld.jar started by root in /)
9     [2020-12-30 14:13:56.444] : No active profile set, falling back to default profiles: default
10    [2020-12-30 14:13:57.703] : Tomcat initialized with port(s): 8085 (http)
11    [2020-12-30 14:13:57.718] : Starting service [Tomcat]
12    [2020-12-30 14:13:57.719] : Starting Servlet engine: [Apache Tomcat/9.0.38]
13    [2020-12-30 14:13:57.813] : Initializing Spring embedded WebApplicationContext
14    [2020-12-30 14:13:57.813] : Root WebApplicationContext: initialization completed in 1269 ms
15    [2020-12-30 14:13:58.102] : Initializing ExecutorService 'applicationTaskExecutor'
16    [2020-12-30 14:13:58.315] : Tomcat started on port(s): 8085 (http) with context path ''
17    [2020-12-30 14:13:58.335] : Started HelloWorldApplication in 2.474 seconds (JVM running for 3.16)
18  }
  }
```

- Log filter by : `GET localhost:8080/logs/5/byId/1` (byId, byInstance, byApp) :
Attendu sur Postman en regardant les logs sur les 5 dernières minutes de l'application d'id 3 :



GET localhost:8080/logs/5/byId/1

Params Authorization Headers (6) Body Pre-request Script Tests Settings

Query Params

KEY	VALUE	DESCRIPTION
Key	Value	Description

Body Cookies Headers (5) Test Results Status: 200 OK Time: 193 ms Size: 1.12 KB Save

Pretty Raw Preview Visualize Text

```

1 {
2   {
3     id:1,
4     app:helloworld:8081,
5     port:8081,
6     docker-instance:helloworld-1
7     message :
8     [2020-12-29 15:38:45.512] : Starting HelloWorldApplication v0.0.1-SNAPSHOT on 0e4e8ccd9e7e with PID 1 (/helloworld.jar started by root in /)
9     [2020-12-29 15:38:45.525] : No active profile set, falling back to default profiles: default
10    [2020-12-29 15:38:47.359] : Tomcat initialized with port(s): 8081 (http)
11    [2020-12-29 15:38:47.376] : Starting service [Tomcat]
12    [2020-12-29 15:38:47.377] : Starting Servlet engine: [Apache Tomcat/9.0.38]
13    [2020-12-29 15:38:47.474] : Initializing Spring embedded WebApplicationContext
14    [2020-12-29 15:38:47.474] : Root WebApplicationContext: initialization completed in 1740 ms
15    [2020-12-29 15:38:47.856] : Initializing ExecutorService 'applicationTaskExecutor'
16    [2020-12-29 15:38:48.164] : Tomcat started on port(s): 8081 (http) with context path ''
17    [2020-12-29 15:38:48.179] : Started HelloWorldApplication in 3.463 seconds (JVM running for 4.236)
18  }
  }
```


ERREURS POSSIBLES

- Start : `The specified application in [...] is not found` → jib n'a pas trouvé le .jar de l'application entre [], vérifiez que l'application jar existe bien dans le dossier 'apps' ou que vous l'ayez bien orthographié dans la requête start,
- Start : `should have a non null json to parse to ApplicationData` → l'application start nécessite obligatoirement un corps (body) en json style,
- Autres erreurs dans start → le body est mal écrit (vérifier l'usage dans ce document),
- Stop : `The requested app to stop with id [...] is not running` → l'application à stopper ne peut être stoppé (déjà fait ou n'existe pas),
- Stop : `Input format for id is Integer, get : [...]` → le body est mal écrit (vérifier l'usage de ce document),
- StopAll : `No app to stop` → aucune application n'est à stopper
- Kill: `The requested app to kill with id [...] is not stopped or does not exist, can't be killed` → l'application à kill ne peut être supprimé (déjà fait ou non stoppée),
- Kill : `Input format for id is Integer, get : [...]` → le body est mal écrit (vérifier l'usage de ce document),
- KillAll : `No app to kill` → aucune application n'est à kill.