Formation Docker : Manipulations pratiques avec Java

El Hadji Gaye

Auteur	El Hadji Gaye
Pour	Formations
Date	06/11/2024
Objet	Formation Docker : Manipulations pratiques avec Java.

I)	Vocabulaire	3
Í)	Les commandes Docker à connaître	4
1.	Commande docker ps	4
2.	Commande docker images	5
3.	Commande docker network	6
4.	Commande docker de runtime	7
5.	Commandes docker de suppression	8
6.	Commandes docker de logs	9
III)	Déployer un Micro Service Spring Boot avec Docker	10
1.	Version 1 : Micro Service Spring Boot sans Docker	10
2.	Version 2 : Micro Service Spring Boot avec une image docker MySql 8	14
	a. Le fichier init_my_data_base.sql	15
	b. Le fichier docker-compose.yaml pour une image MYSQL 8	16
	c. Le fichier docker-compose.yaml pour une image MYSQL XXX	23
	d. Le fichier docker-compose.yaml pour une image MYSQL 8 avec phpmyadmin	27
	e. Lancement de l'application avec Eclipse	32
3.	Version 3 : Image Docker contenant une image JDK 8 alpine (avec Jar du Micro Service	
Sp	pring Boot) + une autre image Docker MySQL 8	36
4.	Version 4 : Image Docker contenant une image JDK 8 alpine (avec Maven 3.5.2 + le Jar du	
Μ	icro Service Spring Boot) + image Docker MySQL 8	43

I) Vocabulaire

Conteneurisation: En informatique, un conteneur est une structure de données, une classe, ou un type de données abstrait, dont les instances représentent des collections d'autres objets. Autrement dit, les conteneurs sont utilisés pour stocker des objets sous une forme organisée qui suit des règles d'accès spécifiques. On peut implémenter un conteneur de différentes façons, qui conduisent à des complexités en temps et en espace différentes. On choisira donc l'implémentation selon les besoins.

Un conteneur est une enveloppe virtuelle qui permet de distribuer une application avec tous les éléments dont elle a besoin pour fonctionner : fichiers source, environnement d'exécution, librairies, outils et fichiers. Ils sont assemblés en un ensemble cohérent et prêt à être déployé sur un serveur et son système d'exploitation (OS). Contrairement à la virtualisation de serveurs et à une machine virtuelle, le conteneur n'intègre pas de noyau, il s'appuie directement sur le noyau de l'ordinateur sur lequel il est déployé.

Virtualisation : La virtualisation consiste, en informatique, à exécuter sur une machine hôte, dans un environnement isolé, des systèmes d'exploitation — on parle alors de virtualisation système ou des applications — on parle alors de virtualisation applicative. Ces ordinateurs virtuels sont appelés serveur privé virtuel (Virtual Private Server ou VPS) ou encore environnement virtuel (Virtual Environment ou VE).

II) Les commandes Docker à connaître

1. Commande docker ps

docker ps vous affiche toutes les instances de docker qui tournent actuellement sur votre environnement. Si vous ajoutez l'option *-a,* alors vous verrez mêmes les containers stoppés.

docker ps -a

2. Commande docker images

docker images est une commande qui vous montre les images que vous avez construites, et le -a vous montre les images intermédiaires.

docker images -a

3. Commande docker network

docker network ls est la commande docker qui liste les différents réseaux.

docker network ls

4. Commande docker de runtime docker-compose up (-d) (--build) docker-compose stop

La docker-compose est la plus simple car vous n'avez besoin que de 2 commandes : up et stop. stop est assez explicite et stop (mais ne supprime pas) vos conteneurs, mais up nécessite plus d'explications : cela va construire vos images si elles ne le sont pas déjà, et va démarrer vos dockers.

docker build (-t NAME) PATH/URL

Si vous voulez re-build vos images, utilisez l'option --build (vous pouvez aussi utiliser la commande docker-compose build pour uniquement construire des images). L'option -d, qui signifie "detach" fait tourner les conteneurs en tâche de fond.

Avec Docker, vous avez besoin d'une commande séparée pour construire votre image, où vous pouvez spécifier le nom de votre image et vous devez spécifier le PATH ou URL selon votre contexte (cela peut être un repo git).

docker run (-d) (-p hostPort :containerPort) (--name NAME)

run crée le conteneur en utilisant l'image que vous indiquez. Vous pouvez spécifier de nombreux paramètres. Nous vous recommandons d'ajouter un nom à votre conteneur et vous pourriez avoir besoin de spécifier quelques ports à exposer. Comme pour docker-compose, le -d lance le conteneur en tâche de fond.

docker start ID /NAME

docker stop ID/NAME

Le start and stop ne devraient pas être trop compliqués à comprendre, mais il faut noter que vous pouvez "start" uniquement des conteneurs qui sont déjà arrêtés, donc déjà build avec la commande run.

docker exec -it NAME /ID "sh" /"/bin/bash"

Cette commande vous permet de lancer un shell sur votre container. Je préfère utiliser "/bin/bash" mais votre conteneur peut ne pas avoir bash d'installé, et seulement "sh" qui est plus courant (surtout sur les alpines). Si vous avez des configurations spéciales dans votre conteneur, vous aurez peut-être besoin d'utiliser des arguments supplémentaires pour vous y connecter.

5. Commandes docker de suppression

Ces commandes permettent de supprimer vos conteneurs et vos images. Vous en aurez probablement besoin pour libérer de l'espace disque.

docker rm ID/NAME

docker-compose rm

Le docker rm supprime seulement un conteneur alors que docker-compose rm supprime tous les conteneurs démarrés avec une commande docker-compose.

docker rmi ID/NAME

Docker rmi supprime l'image que vous passez en paramètre et récursivement toutes les images intermédiaires utilisées pour la construire.

6. Commandes docker de logs

Les commandes suivantes sont utiles quand vous devez débugger certains de vos conteneurs (ou, plus souvent, l'application que vous déployez à l'intérieur).

docker logs ID /NAME (-f --tail NBLINE)

Cette commande affiche les logs du container passé en paramètre. Si vous utilisez l'option -f --tail NBLINE vous pouvez suivre en live le flux de vos logs (NBLINE est le nombre de lignes que vous souhaitez afficher). Gardez à l'esprit de choisir un nombre de lignes que vous serez capable de gérer, pour ne pas être dépassé par vos logs.

docker-compose logs (ID/NAME)

L'option (ID /NAME) avec docker-compose logs vous permet de voir les logs d'un conteneur uniquement, au lieu de voir tous les logs. L'astuce ici est que si vous n'utilisez pas l'option -d quand vous utilisez docker run ou docker-compose up vous verrez vos logs directement (mais vous aurez besoin d'arrêter le conteneur pour quitter la vue). Cela peut toujours être utile pour débugger des applications au démarrage.

III) Déployer un Micro Service Spring Boot avec Docker

1. Version 1 : Micro Service Spring Boot sans Docker

Recupérer le projet **maven-first-app-spring-boot**. Ce projet était développé sous Java 8 avec une base de donnée MySQL 8 qui était installé sur votre poste en local, l'architecture de l'application sera :



 maven-first-app-spring-boot main/java com.cours 		Copy Copy Qualified Name	Ctrl+C
 MainApp.java MainApp Com.cours.controller com.cours.entities com.cours.repository com.cours.repository src/main/resources src/test/java ARE System Library [JavaSE- 	2 2 2	Paste Delete Remove from Context Build Path Refactor Import	Ctrl+V Delete Ctrl+Alt+Shift+Down > Alt+Shift+T >
 Maven Dependencies bin br src br target HELP.md mvnw 	\$	Refresh Close Project Close Unrelated Projects Show in Remote Systems view	F5
Solution in the second mean of the second mean	Q. O	Coverage As Run As	>

0	Run As	>	J	1 Java Application	Alt+Shift+X, J
茶	Debug As	>	Ju	2 JUnit Test	Alt+Shift+X, T
	Profile As	>	m2	3 Maven build	Alt+Shift+X, M
	Restore from Local History		m2	4 Maven build	
	Checkstyle	>	m2	5 Maven clean	
P	PMD	>	m2	6 Maven generate-sources	
	Maven	>	m2	7 Maven install	
	Team	>	m2	8 Maven test	
	Compare With	>		Run Configurations	
	Configure	>	_		

0	Run As		>	J	1 Java A	Applicatio	n	Alt+Shift+X, J
参	Debug As		>	Jυ	2 JUnit	Test		Alt+Shift+X, T
	Profile As		>	m2	3 Mave	n build		Alt+Shift+X, M
	Restore from Local Hist	tory		m2	4 Maye	n build		
	Charlistella				E Maria			
-	Checkstyle			mz	Diviave	n clean		
P	PMD		>	m2	6 Mave	n generat	e-sources	
	Maven		>	m2	7 Mave	n install		、
	Team		>	m2	8 Mave	n test	hs	2
	Compare With		>		Run Co	nfiguratio	ons	
	Configure		`	_		1		-
× 🖻	maven-first-app-sprin	Сору			Ctrl+C			
~	src/main/java	Copy Qualified Name						
	v III Main∆nn iav:	Paste			Ctrl+V			
	> @ MainApp X	Delete			Delete			
	> 🆶 com.cours.conti 🔍	Remove from Context	Ctrl+Al	t+Shi	ft+Down			
	> 🌐 com.cours.entiti	Build Path			>			
	> 🖶 com.cours.repo:	Source		Alt	+Shift+S >			
>	src/main/resources	Refactor		Alt	+Shift+T>			
>	Src/test/java							
	Mayen Dependenci	Import						
5	bin	Export						
>	🕞 src 👔	Refresh			F5			
>	target	References			>			
		Declarations			>			
	mvnw.cmd	Coverage As			>	-		
	script my data has	Run As			>	🗐 1 Ru	in on Server	Alt+Shift+X, R
> 🎥	maven-gestion-persor	Debug As			>	🖵 2 Jav	va Application	N Alt+Shift+X, J
> 🍋	maven-gestion-persor	Profile As			>	_	0.0.0	3
- 🗡 🎥	maven-personnes-dac	Apply Checkstyle fixes				Kun	Configurations.	" WProgram Files\Java\idki
0	Run As			>	J 1 Ja	va Applio	ation N	Alt+Shift+X, J
*	Debug As			>	Run	Configu	wations	3

Drofile As

Run Configurations...

MainApp (2) [Java Application] C:\Program Files\Jav	a\jdk1.8.0_131\bin\jav	aw.exe (27 sept. 2020 à 15:47:02)	
() () () () () () () () () ()			
2020-09-27 15:47:03.214 INFO 8684	[main] com.cours.MainApp	: Starting MainApp on DESKTOP-04L83GP with PID 8684 (C:\Users\elhad\Desktop\AutoEntrepreneur\Java\Projects\NetBeansPr
2020-09-27 15:47:03.218 INFO 8684	[main	com.cours.MainApp	: No active profile set, falling back to default profiles: default
2020-09-27 15:47:04.297 INFO 8684	[main	.s.d.r.c.RepositoryConfigurationDelegate	: Bootstrapping Spring Data JPA repositories in DEFERRED mode.
2020-09-27 15:47:04.382 INFO 8684	[main	.s.d.r.c.RepositoryConfigurationDelegate	: Finished Spring Data repository scanning in 71ms. Found 1 JPA repository interfaces.
2020-09-27 15:47:05.747 INFO 8684	[main] o.s.b.w.embedded.tomcat.TomcatWebServer	: Tomcat initialized with port(s): 8080 (http)
2020-09-27 15:47:05.763 INFO 8684	[main] o.apache.catalina.core.StandardService	: Starting service [Tomcat]
2020-09-27 15:47:05.764 INFO 8684	[main] org.apache.catalina.core.StandardEngine	: Starting Servlet engine: [Apache Tomcat/9.0.38]
2020-09-27 15:47:05.922 INFO 8684	[main] o.a.c.c.C.[Tomcat].[localhost].[/]	: Initializing Spring embedded WebApplicationContext
2020-09-27 15:47:05.923 INFO 8684	[main] w.s.c.ServletWebServerApplicationContext	: Root WebApplicationContext: initialization completed in 2624 ms
2020-09-27 15:47:06.260 INFO 8684	[main] o.s.s.concurrent.ThreadPoolTaskExecutor	: Initializing ExecutorService 'applicationTaskExecutor'
2020-09-27 15:47:06.332 INFO 8684	[task-1] o.hibernate.jpa.internal.util.LogHelper	: HHH000204: Processing PersistenceUnitInfo [name: default]
2020-09-27 15:47:06.448 INFO 8684	[task-1] org.hibernate.Version	: HHH000412: Hibernate ORM core version 5.4.21.Final
2020-09-27 15:47:06.516 WARN 8684	[main] JpaBaseConfiguration\$JpaWebConfiguration	: spring.jpa.open-in-view is enabled by default. Therefore, database queries may be performed during view rendering.
2020-09-27 15:47:06.814 INFO 8684	[task-1] o.hibernate.annotations.common.Version	: HCANN000001: Hibernate Commons Annotations {5.1.0.Final}
2020-09-27 15:47:07.027 INFO 8684	[task-1] com.zaxxer.hikari.HikariDataSource	: HikariPool-1 - Starting
2020-09-27 15:47:07.558 INFO 8684	[task-1] com.zaxxer.hikari.HikariDataSource	: HikariPool-1 - Start completed.
2020-09-27 15:47:07.610 INFO 8684	[task-1] org.hibernate.dialect.Dialect	: HHH000400: Using dialect: org.hibernate.dialect.MySQL5InnoDBDialect
2020-09-27 15:47:08.116 INFO 8684	[main] o.s.b.w.embedded.tomcat.TomcatWebServer	: Tomcat started on port(s): 8080 (http) with context path ''
2020-09-27 15:47:08.120 INFO 8684	[main] DeferredRepositoryInitializationListener	: Triggering deferred initialization of Spring Data repositories
2020-09-27 15:47:08.760 INFO 8684	[task-1] o.h.e.t.j.p.i.JtaPlatformInitiator	: HHH000490: Using JtaPlatform implementation: [org.hibernate.engine.transaction.jta.platform.internal.NoJtaPlatform]
2020-09-27 15:47:08.771 INFO 8684	[task-1] j.LocalContainerEntityManagerFactoryBean	: Initialized JPA EntityManagerFactory for persistence unit 'default'
2020-09-27 15:47:09.161 INFO 8684	[main] DeferredRepositoryInitializationListener	: Spring Data repositories initialized!
2020-09-27 15:47:09.172 INFO 8684	[main	com.cours.MainApp	: Started MainApp in 6.485 seconds (JVM running for 7.148)

Lancer l'URL http://localhost:8080/swagger-ui.html

C () localhost:8080/swagger-ui/index.html?configUrl=/v3/api-docs/swagger-config	<u>B</u> 2	☆	θ
Swagger. /v3/api-docs /v3/api-docs	Explore		
OpenAPI definition ^{CD} CASS V3/api-docs			
Servers http://localhost:8080 - Generated server url v			
my-entity-controller	~		
GET /api/findById/{id}			
GET /api			
GET /api/findByIdBis/{id}			
GET /api/findByField1/{field1}			
GET /api/findByField2/{field2}			
Schemas		\sim	
MyEntity v { id integer(Sint32) field1 string field2 string version integer(Sint32) }			

En cliquant sur GET/api on obtient :

C	Iocalhost:8080/swagger-ui/index.html?configUrl=/v3/api-docs/swagger-config#/my-entity-controller/findAll	Se 🗘
Se	nvers http://localhost:8080 - Generated server url 💙	
	my-entity-controller	~
	GET /api/findById/{id}	
	Parameters	
	No parameters	

C 💿 localhost:8080/swagger-ui/index.html?configUrl=/v3/api-docs/swagger-config#/my-entity-controller/findAll	<u>6</u> 2 5
my-entity-controller	~
GET /api/findById/{id}	
GET /api	
Parameters	Cancel
No parameters	
Exempte	

С 🛈 localhost:8080/swagger-ui/index.html?configUrl=/v3/api-docs/swagger-config#/my-entity-controller/findAll Curl curl -X GET "http://localhost:8080/api" -H "accept: */* Request URL http://localhost:8080/api Server response Code Details 200 Response body ł "id": 1, "field1": "field1-1", "field2": "field2-1", "version": 0 "id": 2, "field1": "field1-2", "field2": "field2-2", "version": 0 "id": 3, "field1": "field1-3", "field2": "field2-3", "version": 0 "id": 4, "field1": "field1-4", "field2": "field2-4", "version": 0 "id": 5, "field1": "field1-5", 'id": Response headers connection: keep-alive
content-type: application/json

Nous allons faire une première amélioration dans cette application en utilisant une image docker de MySQL 8.

2. Version 2 : Micro Service Spring Boot avec une image docker MySql 8

Nous allons améliorer notre architecture micro service en utilisant une image Docker MySQL 8 à la place d'une base de données installé phisiquement dans la machine local.

L'achitecture de l'application resemblera à :



Image Docker MySQL 8

a. Le fichier init_my_data_base.sql

Créer le fichier **maven-first-app-spring-boot/init/init_my_data_base.sql** dont le contenu sera :

DROP DATABASE IF EXISTS my_data_base; /*DROP USER IF EXISTS 'application'@'localhost';*/ CREATE DATABASE my_data_base DEFAULT CHARACTER SET utf8 COLLATE utf8_general_ci; CREATE USER IF NOT EXISTS'application'@'localhost' IDENTIFIED BY 'passw0rd'; /* For MYSQL 8 */ GRANT ALL PRIVILEGES ON my_data_base.* TO'application'@'localhost'; /* For MYSQL 5 */ /*GRANT ALL ON my_data_base.* TO 'application'@'localhost' IDENTIFIED BY 'passw0rd';*/ USE my_data_base; **SET** FOREIGN_KEY_CHECKS = 0; DROP TABLE IF EXISTS MyEntity; CREATE TABLE MyEntity (id INTEGER PRIMARY KEY AUTO_INCREMENT, field1 VARCHAR(100), field2 VARCHAR(100), version int(15))ENGINE=InnoDB DEFAULT CHARSET=utf8; INSERT INTO MyEntity(field1,field2,version) VALUES ('field1-1', 'field2-1',0); INSERT INTO MyEntity(field1,field2,version) VALUES ('field1-2', 'field2-2',0); INSERT INTO MyEntity(field1,field2,version) VALUES ('field1-3', 'field2-3',0); INSERT INTO MyEntity(field1,field2,version) VALUES ('field1-4', 'field2-4',0); INSERT INTO MyEntity(field1,field2,version) VALUES ('field1-5', 'field2-5',0); INSERT INTO MyEntity(field1,field2,version) VALUES ('field1-6', 'field2-6',0); INSERT INTO MyEntity(field1,field2,version) VALUES ('field1-7', 'field2-7',0); INSERT INTO MyEntity(field1,field2,version) VALUES ('field1-8', 'field2-8',0); INSERT INTO MyEntity(field1,field2,version) VALUES ('field1-9', 'field2-9',0);

INSERT INTO MyEntity(field1,field2,version) VALUES ('field1-10', 'field2-10',0);

b. Le fichier docker-compose.yaml pour une image MYSQL 8

Nous allons créer le fichier **maven-gestion-personnes-spring-boot/docker-compose.yaml** qui nous permettra d'initialiser une base de donnée MySQL 8.

Ce fichier aura pour contenu :



En version copiable :

version: "3.7"
services:
db:
hostname: db
image: mysql
For MySQL 8
ports:
- "3308:3306"
For MySQL 5
ports:
- "3306:3306"
environment:
MYSQL_ROOT_PASSWORD: root
MYSQL_DATABASE: "my_data_base"
MYSQL_USER: "application"
MYSQL_PASSWORD: "passw0rd"
command:default-authentication-plugin=mysql_native_password
volumes:
/init:/docker-entrypoint-initdb.d
healthcheck:
test: ["CMD", "mysqladmin" ,"ping", "-h", "localhost"]
timeout: 20s
retries: 3

Ligne 9 : on redirige les données de l'image Docker MySQL 8 du port 3306 de Docker vers le port 3308 de notre machine local.

Attention : ceci est un fichier yaml donc il faut respecter les tabulations.

Lancer votre **Docker Desktop** :

Tout Applications Documents	Web	Plus 🔻	10 😨 & …
Meilleur résultat			
Docker Desktop Application			
Rechercher sur le Web			Docker Desktop
Docker - Afficher les résultats Web		>	Application
Commande			
 Docker 		>	🖵 Ouvrir

Après avoir lancer Docker laisser lui 30 secondes pour qu'il puisse terminer de démarrer. Lancer la commande : **docker version**

Verifier avec votre invité de commande que votre client et votre serveur Docker sont bien démarés :

version	
Client: Docker Engi	ne - Community
Version:	19.03.12
API version:	1.40
Go version:	go1.13.10
Git commit:	48a66213fe
Built:	Mon Jun 22 15:43:18 2020
OS/Arch:	windows/amd64
Experimental:	false
Server: Docker Engi	ne - Community
Engine:	
Version:	19.03.12
API version:	1.40 (minimum version 1.12)
Go version:	go1.13.10
Git commit:	48a66213te
Built:	Mon Jun 22 15:49:27 2020
OS/Arch:	linux/amd64
Experimental:	†alse
containerd:	
Version:	V1.2.13
GITCOMMIT:	/ad184331ta3e55e520890ea95e650a581ae3429
runc:	1.0.0
version:	1.0.0-PC10
Gittommit:	0C920883303TeeT5D3839T432309DeD300T08900
Voncion:	0.18.0
GitCommit:	
GITCOMMITT:	1905085

Lancer l'interface Docker qui permet de voir la liste des images Docker.



	About Docker Desktop
	Settings
	Check for Updates
	Troubleshoot
	Switch to Windows containers
	Documentation
	Learn
	Docker Hub 😡
	Dashboard
÷	elhadjigaye : Sign out
	Repositories
	Kubernetes •
	Restart
	Quit Docker Desktop



Lancer ensuite les commandes suivantes :

cd maven-first-app-spring-boot docker-compose build docker-compose up

En spécifiant le fichier docker compose à utiliser, on obtient :

docker-compose –f docker-compose-mysql-8.yaml build docker-compose –f docker-compose-mysql-8.yaml up

Invite de commandes - docker-compose up :\Users\elhad\Desktop\AutoEntrepreneur\Java\Projects\NetBeansProjects\Elements-Cours\Micro-Services\maven-first-app-spring-boot>docke compose up -compose up Creating network "maven-first-app-spring-boot_default" with the default driver Creating maven-first-app-spring-boot_db_1 ... done Attaching to maven-first-app-spring-boot_db_1 db_1 | 2020-09-27 19:21:14+00:00 [Note] [Entrypoint]: Entrypoint script for MySQL Server 8.0.21-1debian10 started. db_1 | 2020-09-27 19:21:14+00:00 [Note] [Entrypoint]: Switching to dedicated user 'mysql' db_1 | 2020-09-27 19:21:14+00:00 [Note] [Entrypoint]: Entrypoint script for MySQL Server 8.0.21-1debian10 started. db_1 | 2020-09-27 19:21:14+00:00 [Note] [Entrypoint]: Entrypoint script for MySQL Server 8.0.21-1debian10 started. db_1 | 2020-09-27 19:21:15+00:00 [Note] [Entrypoint]: Initializing database files db_1 | 2020-09-27T19:21:15.0668822 0 [System] [MY-013169] [Server] /usr/sbin/mysqld (mysqld 8.0.21) initializing of server in progress as process 42 as process 42 as process 42 db_1 | 2020-09-27T19:21:15.086406Z 1 [System] [MY-013576] [InnoDB] InnoDB initialization has started. db_1 | 2020-09-27T19:21:16.416142Z 1 [System] [MY-013577] [InnoDB] InnoDB initialization has ended. db_1 | 2020-09-27T19:21:10.528241Z 6 [Warning] [MY-010453] [Server] root@localhost is created with an empty password ! Please consider switching off the --initialize-insecure option. db_1 | 2020-09-27 19:21:25+00:00 [Note] [Entrypoint]: Database files initialized db_1 | 2020-09-27 19:21:25+00:00 [Note] [Entrypoint]: Starting temporary server db_1 | 2020-09-27T19:21:25.939997Z 0 [System] [MY-010161] [Server] /usr/sbin/mysqld (mysqld 8.0.21) starting as process 89 db_1 | 2020-09-27T19:21:26.036837Z 1 [System] [MY-013576] [InnoDB] InnoDB initialization has started. db_1 | 2020-09-27T19:21:27.159223Z 1 [System] [MY-013577] [InnoDB] InnoDB initialization has ended. db_1 | 2020-09-27T19:21:27.300292Z 0 [System] [MY-011323] [Server] X Plugin ready for connections. Socket: /var/run/mysqld/mysqlx.sock 2020-09-27T19:21:27.460360Z 0 [Warning] [MY-010068] [Server] CA certificate ca.pem is self signed. 2020-09-27T19:21:27.460594Z 0 [System] [MY-013602] [Server] Channel mysql_main configured to support TLS. Encrypted connections b 1 db_1 | 2020-09-27119:21:27.4003942 0 [System] [HI 013002] [Server] ended a supported for this channel. db_1 | 2020-09-27119:21:27.4663002 0 [Warning] [MY-011810] [Server] Insecure configuration for --pid-file: Location '/var/run/mysqld' in the path is accessible to all OS users. Consider choosing a different directory. db_1 | 2020-09-27119:21:27.489031Z 0 [System] [MY-010931] [Server] /usr/sbin/mysqld: ready for connections. Version: '8.0.21' socket: '/var/run/mysqld/mysqld.sock' port: 0 MySQL Community Server - GPL. run/mysqld/mysqld.sock' port: 0 MySQL Community Server - GPL. 2020-09-27 19:21:27+00:00 [Note] [Entrypoint]: Temporary server started. Warning: Unable to load '/usr/share/zoneinfo/iso3166.tab' as time zone. Skipping it. Warning: Unable to load '/usr/share/zoneinfo/leap-seconds.list' as time zone. Skipping it. Warning: Unable to load '/usr/share/zoneinfo/zone.tab' as time zone. Skipping it. Warning: Unable to load '/usr/share/zoneinfo/zone1970.tab' as time zone. Skipping it. 2020-09-27 19:21:31+00:00 [Note] [Entrypoint]: Creating database my_data_base 2020-09-27 19:21:31+00:00 [Note] [Entrypoint]: Creating user application 2020-09-27 19:21:31+00:00 [Note] [Entrypoint]: Giving user application access to schema my_data_base b 1 b 1 2020-09-27 19:21:31+00:00 [Note] [Entrypoint]: /usr/local/bin/docker-entrypoint.sh: running /docker-entrypoint-initdb.d/datas_m _data_base.sql b 1 2020-09-27 19:21:31+00:00 [Note] [Entrypoint]: Stopping temporary server 2020-09-27T19:21:31.996290Z 15 [System] [MY-013172] [Server] Received SHUTDOWN from user root. Shutting down mysqld (Version: & 0.21). | 2020-09-27T19:21:34.982408Z 0 [System] [MY-010910] [Server] /usr/sbin/mysqld: Shutdown complete (mysqld 8.0.21) MySQL Communit Server 2020-09-27 19:21:34+00:00 [Note] [Entrypoint]: Temporary server stopped b 1 2020-09-27 19:21:35+00:00 [Note] [Entrypoint]: MySQL init process done. Ready for start up. 2020-09-27T19:21:35.261396Z 0 [System] [MY-010116] [Server] /usr/sbin/mysqld (mysqld 8.0.21) starting as process 1 2020-09-27T19:21:35.279706Z 1 [System] [MY-013576] [InnoDB] InnoDB initialization has started. 2020-09-27T19:21:35.571157Z 1 [System] [MY-013577] [InnoDB] InnoDB initialization has ended. 2020-09-27T19:21:35.705462Z 0 [System] [MY-011323] [Server] X Plugin ready for connections. Bind-address: '::' port: 33060, soc b 1 /var/run/mysqld/mysqlx.sock | 2020-09-27T19:21:35.807915Z 0 [Warning] [MY-010068] [Server] CA certificate ca.pem is self signed.



Nous voyons d'après les logs que l'image Docker de MySQL a été démaré sur le port 3306. Dans l'application nous écoutons sur le port 3308 c'est pour cela que nous avons dans notre fichier de configuration Docker :



En allant voir sur le Docker Desktop on voit l'image qu'on vient de créer.

	 🔅 🐐 😫 elhadjigaye
Q Search	Sort by V
 maven-first-app-spring-boot RUNNING maven-first-app-spring-boot_db_1 mysql RUNNING PORT: 3308 	
maven-first-app-spring-boot_db_1 mysql RUNNING PORT: 3308	

Il est possible d'arreter le conteneur ou même de le supprimer.





c. Le fichier docker-compose.yaml pour une image MYSQL XXX

Arreter et supprimer le précédent conteneur :

		۲	×.	elhadjigaye	- 🗆 ×
Q Search					Sort by \checkmark
maven-first-app-spring-boot RUNNING		Solution Open	in Visual	Studio Code	
maven-first-app-spring-boot_db_1 mysql RUNNING PORT: 3308					STOP
					- n x
		٠	*	elhadjigaye	
Q Search					Sort by \checkmark
maven-first-app-spring-boot EXITED (UNDEFINED)		🗙 Open i	n Visual S	itudio Code	
					DELETE
	# docker	•	×.	elhadjigaye	- 🗆 ×
Q Search					Sort by 🗸
maven-first-app-spring-boot EXITED (UNDEFINED)					
	_	_		_	
Remove application					
You are about to remove the ap containers.	op 'maven-first-app-spri	ng-boot' and all its			
Do you want to continue?					
		Cancel	Repp	ve	



Modifier le fichier **maven-gestion-personnes-spring-boot/docker-compose.yaml** pour modifier la version du MySQL pour une version **5.7**.

Le fichier docker-compose.yaml aura donc pour contenu :



En version copiable :

version: "3.7"
services:
db:
hostname: db
image: mysql:5.7
For MySQL 8
ports:
- "3308:3306"
For MySQL 5
ports:
- "3306:3306"
environment:
MYSQL_ROOT_PASSWORD: root
MYSQL_DATABASE: "my_data_base"
MYSQL_USER: "application"
MYSQL_PASSWORD: "passw0rd"
command:default-authentication-plugin=mysql_native_password
volumes:
/init:/docker-entrypoint-initdb.d
healthcheck:
test: ["CMD", "mysqladmin" ,"ping", "-h", "localhost"]
timeout: 20s
retries: 3

Ligne 12 : on redirige les données de l'image Docker MySQL 5.7 du port 3306 de Docker vers le port 3306 de notre machine local.

Le fichier **pom.xml** aura la dépendance MySQL suivante :



Le fichier **application.properties** devient :

For MYSQL 8
#spring.datasource.url=jdbc:mysql://localhost:3308/my_data_base?serverTimezone=UTC
#spring.datasource.driver-class-name=com.mysql.cj.jdbc.Driver
For MYSQL 5
spring.datasource.url=jdbc:mysql://localhost:3306/my_data_base?useSSL=false
spring.datasource.driver-class-name=com.mysql.jdbc.Driver
spring.datasource.username=application
spring.datasource.password=passw0rd
For Hibernate version < 5.3.1.Final
$spring. jp a. properties. hibernate. dialect \verb= org. hibernate. dialect. MySQL5InnoDBD ialect$
For Hibernate version > 5.3.1.Final
#spring.jpa.properties.hibernate.dialect=org.hibernate.dialect.MySQL8Dialect
spring.jpa.hibernate.naming.physical-
strategy=org.hibernate.boot.model.naming.PhysicalNamingStrategyStandardImpl
spring.jpa.hibernate.ddl-auto=update
springdoc.swagger-ui.path=/swagger-ui.html
Manage Error Message
server.error.include-message= always

d. Le fichier docker-compose.yaml pour une image MYSQL 8 avec phpmyadmin

Arreter et supprimer le précédent conteneur :

		🗢 🍕	elhadjigaye	- 🗆 ×
Q Search				Sort by 🗸
waven-first-app-spring-boot		Open in Visua	al Studio Code	۵
maven-first-app-spring-boot_db_1 mysql RUNNING PORT: 3308				STOP
		¢ %	elhadjigaye	- 🗆 X
Q Search				Sort by \checkmark
Maven-first-app-spring-boot		Open in Visual	I Studio Code	
				DELETE
				- n x
		. (*)	elhadjigaye	
Q Search				Sort by 🗸
maven-first-app-spring-boot exited (UNDEFINED)				
Remove application				
You are about to remove th containers.	e app 'maven-first-app-sprii	ng-boot' and all its		
Do you want to continue?				
		Cancel Rep	nove	



Modifier le fichier **maven-gestion-personnes-spring-boot/docker-compose.yaml** pour ajouter phpmyadmin à votre base de donnée MySQL 8.

Ce fichier aura pour contenu :

1	version: "3.7"
2	
З	services:
4	
5	hostname: db
6	image: mysql
7	# For MySQL 8
8	ports:
9	- "3308:3306"
10	# For MySQL 5
11	# ports:
12	# - "3306:3306"
13	environment:
14	MYSQL_ROOT_PASSWORD: root
15	MYSQL_DATABASE: "my_data_base"
16	MYSQL_USER: "application"
17	MYSQL_PASSWORD: "passw0rd"
18	<pre># command:default-authentication-plugin=mysql_native_password</pre>
19	volumes:
20	/init:/docker-entrypoint-initdb.d
21	healthcheck:
22	test: ["CMD", "mysqladmin" ,"ping", "-h", "localhost"]
23	timeout: 20s
24	retries: 3
25	phpmyadmin:
26	image: phpmyadmin/phpmyadmin
27	restart: always
28	ports:
29	
30	environment:
31	PMA_USER: application
32	PPM_PASSWORD: password
33	

En version copiable :

version: "3.7"
services.
dh
hostnamo: dh
image: mysgl
Hildge. Hysqi # Ear MusQL 8
FOF MySQL 8
ports:
- "3308:3306"
For MySQL 5
ports:
- "3306:3306"
environment:
MYSQL_ROOT_PASSWORD: root
MYSQL_DATABASE: "my_data_base"
MYSQL_USER: "application"
MYSQL_PASSWORD: "passw0rd"
command:default-authentication-plugin=mysql_native_password
volumes:
/init:/docker-entrypoint-initdb.d
healthcheck:
test: ["CMD", "mysqladmin" ,"ping", "-h", "localhost"]
timeout: 20s
retries: 3
phpmyadmin:
image: phpmyadmin/phpmyadmin
restart: always
ports:
- 8086:80
environment:
PMA_USER: "application"
PMA_PASSWORD: "passw0rd"

Ligne 9 : on redirige les données de l'image Docker MySQL 8 du port 3306 de Docker vers le port 3308 de notre machine local.

Ligne 29 : on redirige les données de l'image Docker **phpmyadmin** du port 80 de Docker vers le port 8086 de notre machine local.

Lancer ensuite les commandes suivantes :

cd maven-first-app-spring-boot docker-compose build docker-compose up

En spécifiant le fichier docker compose à utiliser, on obtient :

docker-compose –f docker-compose-with-phpmyadmin.yaml build docker-compose –f docker-compose-with-phpmyadmin.yaml up



Lancer l'URL de phpmyadmin sur <u>http://localhost:8086/</u>



\leftrightarrow \rightarrow C (i) localhost:8086		
Image: Constraint of the second state of the second sta	 Serveur: db Bases de données SQL L État Exporter Importer Paramètres Variables Exporter Paramètres généraux Interclassement pour la connexion au serveur : Utf8mb4_unicode_ci Plus de paramètres Paramètres d'affichage Langue - Language Français - French Thème : pmahomme	Jeux

 $\leftarrow \ \ \, \rightarrow \ \ C \quad \textcircled{0} \quad \ \ \, localhost: 8086/sql.php?server=1\&db=my_data_base&table=MyEntity&pos=0\\ \label{eq:constraint}$

phpMyAdmin	🗕 📑 Serveur	: db » 🍵 Bas	se de données:	my_	data_base	» 🔝 Table	MyEntity	
a 🗐 🕢 🗊 🖉	Parcourin	🖌 Stru	icture 📗 🧐	SQL	🔍 Re	chercher	👫 Insé	rer 🖶 Exporter
Récentes Préférées								
	🛹 Affichage	des lignes 0	- 9 (total de 10	, trait	ement en	0.0062 sec	onde(s).)	
⊕-	SELECT * FROM	`MyEntity`						
— my_data_base								
Nouvelle table								
🕀 🛃 MyEntity		Cohor I M		Г	05			0
			ombre de lignes		25 🗸	Filtrer	es lignes:	Chercher dans cette ta
	+ Options							
	⊢T→		\bigtriangledown	id	field1	field2	version	
	🗌 🥔 Éditer	👍 Copier	Supprimer	1	field1-1	field2-1	0	
	🗌 🥜 Éditer	Copier	Supprimer	2	field1-2	field2-2	0	
	🗌 🥜 Éditer	Copier	Supprimer	3	field1-3	field2-3	0	
	🗌 🥜 Éditer	Copier	Supprimer	4	field1-4	field2-4	0	
	🗌 🥜 Éditer	Copier	Supprimer	5	field1-5	field2-5	0	
	🗌 🥜 Éditer	3 Copier	Supprimer	6	field1-6	field2-6	0	
	🗌 🥜 Éditer	Copier	Supprimer	7	field1-7	field2-7	0	
	🗌 🥜 Éditer	Copier	Supprimer	8	field1-8	field2-8	0	
	🗌 🥜 Éditer	Copier	Supprimer	9	field1-9	field2-9	0	
	🗌 🥜 Éditer	≩∎ Copier	Supprimer	10	field1-10	field2-10	0	

e. Lancement de l'application avec Eclipse

Une fois que votre image Docker de base de données est lancée, lancer votre Micro Service.

	 maven-first-app-spring-boot src/main/java src/main/java Com.cours MainApp.java MainApp com.cours.controller com.cours.entities com.cours.repository src/main/resources src/test/java JRE System Library [JavaSE- Maven Dependencies bin src target HELP.md mvnw 		Copy Copy Qualified Na Paste Delete Remove from Con Build Path Refactor Import Export Refresh Close Project Close Unrelated Pr	me text	Ctrl+Alt+Shi Alt	Ctrl+C Ctrl+V Delete ft+Down > +Shift+T >	
[mvnw.cmd	0	Coverage As	yster	ns view	>	
	pom.xml	0	Run As			>	
	script_my_data_basesqr				N		
D R	lun As		>		1 Java Application	Alt+S	hift+X, J
🎋 D	ebug As		>	Ju	2 JUnit Test	Alt+S	hift+X, T
P	rofile As		>	m2	3 Maven build	Alt+Sh	ift+X, M
R	lestore from Local History			m2	4 Maven build		
C	Checkstyle		>	m2	5 Maven clean		
🏴 P	MD		>	m2	6 Maven generate-sources	5	
N	/laven		>	m2	7 Maven install		
Т	eam		>	m2	8 Maven test		
C	Compare With		>		Run Configurations		
C	Configure		>	-			
R R	un As		>		1 Java Application	Alt+S	Shift+X_J
参 D)ebua As		>	Ju	2 JUnit Test	Alt+S	hift+X.T
P	rofile As		>	m2	3 Mayen build	Alt+Sk	nift+X_M
R	estore from Local History				4 Mayen build	710.01	
	heckstyle		>		5 Mayen clean		
P	MD		>	m2	6 Mayen generate-source	<.	
	laven		>	m2	7 Maven install		
Т	eam		>	m2	8 Maven test	3	
C	Compare With		>		D 0 0 1		
0	onfigure		`		Kun Configurations		

 ✓ and maxen-first-app-sprin ✓ and src/main/java ✓ com.cours ✓ MainApp.java > ④ MainApp > ⊕ com.cours.conti > ⊕ com.cours.entiti > ⊕ com.cours.entiti 		Copy Copy Qualified Name Paste Delete Remove from Context Build Path	Ctrl+Alt+Sł	Ctrl+C Ctrl+V Delete hift+Down			
 > # scormeours.epo. > # src/main/resources > # src/test/java > A JRE System Library > A Maven Dependenci 	24 24	Source Refactor Import Export	A	lt+Shift+S > lt+Shift+T >			
> 🔑 bin > 🔑 src > 🔑 target 🐨 HELP.md	Ł	Refresh References		F5 >	-		
mvnw mvnw.cmd pom.xml	Q. ()	Coverage As Run As		>		1 Run on Server	Alt+Shift+X, R
 > > maven-gestion-persor > > maven-gestion-persor > maven-personnes-dac 	*	Debug As Profile As Apply Checkstyle fixes		>		2 Java Application Run Configurations	Alt+Shift+X, J
Dun Ar				. 14			Alta, Chiffe, V. I
The bug As			>	Run		nfigurations	Wit+Shirt+A, J
Drafila Ar			\				
Markers Properties #% Servers	Data So es\Java \\\\ \\\\ //// RELEAS	ource Explorer È Snippets D Console ∞ d J. \jdk1.8.0_131\bin\javaw.exe (27 sept. 2020 à 21:56:59)	Jnit				

Lancer l'URL <u>http://localhost:8080/swagger-ui.html</u>

C () localhost8080/swagger-ui/index.html?configUrl=/v3/api-docs/swagger-config	6 ₂	☆	0
Wagger. /V3/api-docs	Explore		
OpenAPI definition ^{CO} OASS N3/apr-docs			
Servers http://localhost:8080 - Generated server url v			
my-entity-controller	~		
GET /api/findById/{id}			
CET /api			
GET /api/findByIdBis/{id}			
GET /api/findByField1/{field1}			
GET /api/findByField2/{field2}			
Schemas		\sim	
<pre>MyEntity ~ { id integer(Sint32) field1 string field2 string version integer(Sint32) }</pre>			

En cliquant sur **GET/api** on obtient :

G	localhost:8080/swagger-ui/index.html?configUrl=/v3/api-docs/swagger-config#/my-entity-controller/findAll	<u> </u>	
Sei P	rvers http://localhost:8080 - Generated server url 🗸		
r	my-entity-controller	\sim	
	GET /api/findById/{id}		
	GET /api		ן
	Parameters	i out	
	No parameters		

C 🛈 localhost8080/swagger-ui/index.html?configUrl=/v3/api-docs/swagger-config#/my-entity-controller/findAll	<u>a</u> s 2
my-entity-controller	\checkmark
GET /api/findById/{id}	
GET /api	
Parameters	Cancel
No parameters	
Exempte	

G (i) I	ocalhost:8080/swagger-ui/index.html?configUrl=/v3/api-docs/swagger-config#/my-entity-controller/findAll
Curl	
curl -	X GET "http://localhost:8080/api" -H "accept: */*"
Reques	t URL
http:	//localhost:8080/api
Server i	response
Code	Details
200	Response body
	<pre></pre>

3. Version 3 : Image Docker contenant une image JDK 8 alpine (avec Jar du Micro Service Spring Boot) + une autre image Docker MySQL 8

L'architecture de l'application ressemblera cette fois ci à :

Container Docker avec
Image JAVA 8
Jar du Micro Service
Image MySQL 8

Nous allons introduire la notion de l'état ou santé du serveur à travers **actuator/metrics**, **actuator/health et actuator/beans**.

Mettre à jour votre fichier **pom.xml** en ajoutant la dépendance :



Mettre à jour le fichier **application.properties** en ajoutant :

- 1 # Manage actuator
- 2 management.endpoint.health.show-details=ALWAYS
- 3 management.endpoints.web.exposure.include=*
- 4 management.endpoint.beans.enabled=true

En version copiable :

Manage actuator management.endpoint.health.show-details=ALWAYS management.endpoints.web.exposure.include=* management.endpoint.beans.enabled=true

Créer le fichier maven-first-app-spring-boot/Dockerfile dont le contenu sera :



En version copiable :

FROM java:8-jre-alpine
EXPOSE 8080
RUN mkdir /app COPY target/*.jar /app/my-spring-boot-application.jar
ENTRYPOINT ["java","-Djava.security.egd=file:/dev/./urandom","-jar","/app/my-spring-boot-application.jar"]
HEALTHCHECKinterval=3mtimeout=3s CMD curl -f http://localhost:8080/actuator/health/ exit 1

le fichier maven-gestion-personnes-spring-boot/docker-compose.yaml :

	version: "3.7"
	services:
	hostname: db
	image: mysql
	# For MySQL 8
	ports:
	- "3308:3306"
	# For MySQL 5
11	
12	
13	environment:
14	MYSQL_ROOT_PASSWORD: root
	MYSQL DATABASE: "my_data base"
	MYSQL USER: "application"
17	MYSQL PASSWORD: "passw@rd"
18	# command:default-authentication-plugin=mysql native password
19	volumes:
	/init:/docker-entrypoint-initdb.d
21	healthcheck:
22	test: ["CMD", "mysqladmin" ,"ping", "-h", "localhost"]
23	timeout: 20s
	retries: 3
25	
20	piping auman.
27	image. priping admini/priping admini
20	nestart. diways
29	
22	environment.
22	PMA_USER. application
	PPIA_PASSWORD. Password
24	
	app.
50	nostrati en feilure
	ingent our direct on coming heatlatest
	Image: my_app_maven-tirst-app-spring-000t.latest
40	context: ./
41	environment.
42	santa
43	ports:
44	- 8080:8080
	depends_on:
46	
4	- pripmyaamin
47	

En version copiable :

ver	sion: "3.7"
serv	vices:
d	b:
	hostname: db
	image: mysgl
# F	or MySQL 8
	ports:
	- "3308:3306"
# F	or MySQL 5
#	ports:
#	- "3308:3306"
	environment:
	MYSQL_ROOT_PASSWORD: root
	MYSQL_DATABASE: "my_data_base"
	MYSQL_USER: "application"
	MYSQL_PASSWORD: "passw0rd"
#	command:default-authentication-plugin=mysql_native_password
	volumes:
	/ init:/ docker-entrypoint-initdb.d
	test: ["CMD", "mysqladmin", "ping", "-h", "localhost"]
	timeout: 20s
	Tetries. 5
n	hpmyadmin [.]
Р	image: phpmyadmin/phpmyadmin
	restart: always
	ports:
	- 8086:80
	environment:
	PMA_USER: "application"
	PMA_PASSWORD: "passw0rd"
	pp:
	hostname: app
	restart: on-failure
	image: my_app_maven-first-app-spring-boot:latest
	build:
	context: ./
	environment:
	SPRING_DATASOURCE_URL: jdbc:mysql://db:3306/my_data_base
	depends on:
	- dh
	- phpmyadmin
	pripriyaanin

Ligne 4-24 : configuration de l'image Docker MySql 8. Ligne 26-33 : configuration de l'image Docker phpmyadmin. Ligne 35-47 : configuration de l'image Docker du Micro Service Spring Boot.

Générer le jar **maven-first-app-spring-boot/target/maven-first-app-spring-boot-0.0.1**-**SNAPSHOT.jar** avec Eclipse ou un autre programme. Ce jar deviendra **my-spring-bootapplication.jar** dans l'execution du fichier **maven-first-app-spring-boot/Dockerfile**.



0	Run As	>	J	1 Java Application	Alt+Shift+X, J
*	Debug As	>	Ju	2 JUnit Test	Alt+Shift+X, T
	Profile As	>	m2	3 Maven build	Alt+Shift+X, M
	Restore from Local History		m2	4 Maven build	
	Checkstyle	>	m2	5 Maven clean	
P	PMD	>	m2	6 Maven generate-sources	
	Maven	>	m2	7 Maven install	
	Team	>	m2	8 Maven test	
	Compare With	>		Run Configurations	
	Configure	>	_		

0	Run As	>	J	1 Java Application	Alt+Shift+X, J
*	Debug As	>	Ju	2 JUnit Test	Alt+Shift+X, T
	Profile As	>	m2	3 Maven build	Alt+Shift+X, M
	Restore from Local History		m2	4 Maven build	
	Checkstyle	>	m2	5 Maven clean	
P	PMD	>	m2	6 Maven generate-sources	
	Maven	>	m2	7 Maven install	
	Team	>	m2	8 Maven test	2
	Compare With	>		Run Configurations	
	Configure	>	_	turi comiguiutoris	_

Lancer ensuite les commandes suivantes :

cd maven-first-app-spring-boot docker-compose build docker-compose up

<pre>:\Users\elhad\Desktop\AutoEntrepreneur\Java\Projects\WetBeansProjects\Elements-Cours\Wicro-Services\maven-first-app-spring-boot>docker-compose build b uses an image, skipping hpmyadmin uses an image, skipping uilding app tep 1/6 : FROM java:8-jre-alpine> fdc893b19a14 tep 2/6 : EXPOSE 8080>> Using cache>> Using cache>> Using cache>> doc95061588 tep 4/6 : COPY target/*.jar /app/my-spring-boot-application.jar>> energet/setup 4.2000 tep 5/6 : ENTRYPOINT [~java", "-Ojava.security.egd-file:/dev/./urandom", "-jar", "/app/my-spring-boot-application.jar"]>> ae0970cdc409 tep 5/6 : ENTRYPOINT [~java", "-Ojava.security.egd-file:/dev/./urandom", "-jar", "/app/my-spring-boot-application.jar"]>> 0%76693508268 emoving intermediate container 549275983088>> 0%769547cf66 tup 6/6 : HEALTHACHECKinterval=3mtimeout=3s CMD curl -f http://localhost/actuator/health/ exit 1>> 0%7693508368 tuccessfully built d297b6985084 uccessfully built d297b6985084 uccessfully taged mg_app_maven-first-app-spring-boot:latest :\Users\elhad\Desktop\AutoEntrepreneur\Java\Projects\NetBeansProjects\Elements-Cours\Micro-Services\maven-first-app-spring-boot>docker-compose up reating mever-first-app-spring-boot_default" with the default driver reating mever-first-app-spring-boot_dpi_1 done reating mever-first-app-spring-boot_app_1 1 2020-602 20 10:30:450:60 [Mote] [Entrypoint]: Entrypoint script for MySQL Server 8.0.21-1deblan10 started.</pre>
<pre>b uses an image, skipping hpmyadmin uses an image, skipping ulding app tep 1/6 : FROM java:8-jre-alpine> fdc839blaid4 tep 2/6 : EXPOSE 0808> Using Cache> Using Cache> Using Cache> Using Cache> 0406900508 tep 4/6 : COPY target/*.jar /app/my-spring-boot-application.jar> 04069005080560 tep 4/6 : COPY target/*.jar /app/my-spring-boot-application.jar> 04069005080560 tep 4/6 : COPY target/*.jar /app/my-spring-boot-application.jar> 04069005080560 tep 4/6 : COPY target/*.jar /app/my-spring-boot-application.jar> 0406905080560 tep 4/6 : COPY target/*.jar /app/my-spring-boot-application.jar> 0406905080560 tep 4/6 : COPY target/*.jar /app/my-spring-boot-application.jar> 0406905080560 tep 4/6 : CHTRYPOINT ["java", "-Ojava.security.egd=file:/dev/./urandom", "-jar", "/app/my-spring-boot-application.jar"]> 040690506508 tep 4/6 : HEALTHCHECKinterval=3mtimeout=3s CMD curl -f http://localhost/actuator/health/ exit 1> 0406905066 uccessfully built d297b60850af uccessfully built d297b60850af uccessfully tagged my_app_maven-first-app-spring-boot:latest :\Users\elhad\Desktop\AutoEntrepreneur\Java\Projects\WetBeansProjects\Elements-Cours\Wicro-Services\maven-first-app-spring-boot>decfault' with the default driver reating maven-first-app-spring-boot_default' with the default driver reating maven-first-app-spring-boot_default' with the default driver reating maven-first-app-spring-boot_dpmyadmin_1 done reating maven-fir</pre>
<pre>hpmyadmin uses an image, skipping uilding pp tep 1/6 : FROM java:8-jre-alpine > fdc8351ba14 tep 2/6 : EXPOSE 8080 > Using cache > Using cache > Using cache > Using cache > 0 ed6C306b1568 tep 4/6 : COVY target/*.jar /app/my-spring-boot-application.jar > ae0970ccdc49 tep 5/6 : ENTRYPOINT ["java", "-Djava.security.egd=file:/dev/./urandom", "-jar", "/app/my-spring-boot-application.jar"] > Running in 5d92759e3028 emoving intermediate container 5d92759e3028 > 0746934cfc06 tep 6/6 : H&LITHHEKKinterval=amtimeout=3s CMD curl -f http://localhost/actuator/health/ exit 1 > Running in bc2a078356ba emoving intermediate container bc2a078356ba > 0297069850af uccessfully uilt d297b69850af uccessfully tagged my_app_maven-first-app-spring-boot:latest :\Users\elhad\Desktop\AutoEntrepreneur\Java\Projects\NetBeansProjects\Elements-Cours\Micro-Services\maven-first-app-spring-boot>_ :\Users\elhad\Desktop\AutoEntrepreneur\Java\Projects\NetBeansProjects\Elements-Cours\Micro-Services\maven-first-app-spring-boot>_ :\Users\elhad\Desktop\AutoEntrepreneur\Java\Projects\NetBeansProjects\Elements-Cours\Micro-Services\maven-first-app-spring-boot>_ :\Users\elhad\Desktop\AutoEntrepreneur\Java\Projects\NetBeansProjects\Elements-Cours\Micro-Services\maven-first-app-spring-boot>_ :\Users\elhad\Desktop\AutoEntrepreneur\Java\Projects\NetBeansProjects\Elements-Cours\Micro-Services\maven-first-app-spring-boot>_ :\Users\elhad\Desktop\AutoEntrepreneur\Java\Projects\NetBeansProjects\Elements-Cours\Micro-Services\maven-first-app-spring-boot>_ :\Users\elhad\Desktop\AutoEntrepreneur\Java\Projects\NetBeansProjects\Elements-Cours\Micro-Services\maven-first-app-spring-boot>_ :\Users\elhad\Desktop\AutoEntrepreneur\Java\Projects\NetBeansProjects\Elements-Cours\Micro-Services\maven-first-app-spring-boot_dp_1 done reating maven-first-app-spring-boot_dp_1 done reating maven-first-app-spring-boot_dp_1 done reating maven-first-app-spring-boot_dp_1 done ftat\fing maven-first-app-spring-boot_dp_1 done</pre>
<pre>vilding app tep 1/6 : FROM java:8-jre-alpine > tfd:893b19a14 tep 2/6 : FXPOSE 8080 > Using cache > Using cache > Using cache > edecf.80b1568 tep 4/6 : COPY target/*.jar /app/my-spring-boot-application.jar > edecf.80b1568 tep 4/6 : LAITHCHECKinterval=3mtimeout=3s CMD curl -f http://localhost/actuator/health/ exit 1 > Running in bc2a078356ba enoving intermediate container bc2a078356ba > d297b69850af uccessfully built d297b69850af uccessfully tagged my_app_maven-first-app-spring-boot:latest :\Users\elhad\Desktop\AutoEntrepreneur\Java\Projects\Elements-Cours\Micro-Services\maven-first-app-spring-boot>_ :\Users\elhad\Desktop\AutoEntrepreneur\Java\Projects\WetBeansProjects\Elements-Cours\Micro-Services\maven-first-app-spring-boot>_ :\Users\elhad\Desktop\AutoEntrepreneur\Java\Projects\WetBeansProjects\Elements-Cours\Micro-Services\maven-first-app-spring-boot>_ :\Users\elhad\Desktop\AutoEntrepreneur\Java\Projects\WetBeansProjects\Elements-Cours\Micro-Services\maven-first-app-spring-boot>_ :\Users\elhad\Desktop\AutoEntrepreneur\Java\Projects\WetBeansProjects\Elements-Cours\Micro-Services\maven-first-app-spring-boot>_ :\Users\elhad\Desktop\AutoEntrepreneur\Java\Projects\WetBeansProjects\Elements-Cours\Micro-Services\maven-first-app-spring-boot_defaulT with the default driver reating maven-first-app-spring-boot_dphmyadmin_1 done reating maven-first-app-spring-boot_dphmyadmin_1 done reating maven-first-app-spring-boot_app_1 2020-202-28 10:30:41:40:00 [Entrypoint]: Entrypoint script for MySQL Server 8.0.21:1debianl0 started.</pre>
<pre>tep 1/6 : FROM java:8-jre-alpine > frd:893bland tep 2/6 : EXPOSE 8080 >> Using cache >> using cache >>> using cache >>> using cache >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>></pre>
<pre>> thtesp:////ickenses/</pre>
<pre>tep 2/6 : EXPOSE 8088 > Using cache > Using cache > Using cache > Using cache > ed6c38061568 tep 4/6 : COPY target/*.jar /app/my-spring-boot-application.jar > ae0978ctcda9 tep 5/6 : ENTRYPOINT ["java", "-Djava.security.egd=file:/dev/./urandom", "-jar", "/app/my-spring-boot-application.jar"] > Running in 5d92759e3028 emoving intermediate container 5d92759e3028 > 8Pd6934cfc06 tep 6/6 : HEALTHCHECKinterval=3mtimeout=3s CMD curl -f http://localhost/actuator/health/ exit 1 > Running in bc2a078356ba emoving intermediate container bc2a078356ba > 0297b69850af uccessfully built d297b69850af uccessfully tagged my_app_maven-first-app-spring-boot:latest :\Users\elhad\Desktop\AutoEntrepreneur\Java\Projects\WetBeansProjects\Elements-Cours\Micro-Services\maven-first-app-spring-boot>_ :\Users\elhad\Desktop\AutoEntrepreneur\Java\Projects\WetBeansProjects\Elements-Cours\Micro-Services\maven-first-app-spring-boot>_ :\Users\elhad\Desktop\AutoEntrepreneur\Java\Projects\WetBeansProjects\Elements-Cours\Micro-Services\maven-first-app-spring-boot>_ :\Users\elhad\Desktop\AutoEntrepreneur\Java\Projects\WetBeansProjects\Elements-Cours\Micro-Services\maven-first-app-spring-boot>_ :\Users\elhad\Desktop\AutoEntrepreneur\Java\Projects\WetBeansProjects\Elements-Cours\Micro-Services\maven-first-app-spring-boot>_ :\Users\elhad\Desktop\AutoEntrepreneur\Java\Projects\WetBeansProjects\Elements-Cours\Micro-Services\maven-first-app-spring-boot>_ :\Users\elhad\Desktop\SutoEntrepreneur\Java\Projects\WetBeansProjects\Elements-Cours\Micro-Services\maven-first-app-spring-boot>_ :\Users\elhad\Desktop\SutoEntrepreneur\Java\Projects\WetBeansProjects\Elements-Cours\Micro-Services\maven-first-app-spring-boot>docker-compose up reating maven-first-app-spring-boot_default" with the default driver reating maven-first-app-spring-boot_phymyadmin1 done reating maven-first-app-spring-boot_phymyadmin1 done reating maven-first-app-spring-boot_phymyadmin1 done itatAing to maven-first-app-spring-boot_phymyadmin1</pre>
<pre>> Using cache > Using cache > Using cache > e06c306b1568 tep 4/6 : COPY target/*.jar /app/my-spring-boot-application.jar > e06c306b1568 tep 5/6 : ENTRYPOINT ["java", "-Djava.security.egd=file:/dev/./urandom", "-jar", "/app/my-spring-boot-application.jar"] > Running in 50d92759e3028 emoving intermediate container 5d92759e3028 > 07408934cfc06 tep 6/6 : HEALTHCHECKinterval=3mtimeout=3s CMD curl -f http://localhost/actuator/health/ exit 1 > Running in bc2a078350ba emoving intermediate container bc2a078350ba > 0429785050ba emoving intermediate container bc2a078350ba > 0429785050af uccessfully built d297b69850af uccessfully tagged my_app_maven-first-app-spring-boot:latest :\Users\elhad\Desktop\AutoEntrepreneur\Java\Projects\WetBeansProjects\Elements-Cours\Micro-Services\maven-first-app-spring-boot>default" with the default driver reating maven-first-app-spring-boot_default" with the default driver reating maven-first-app-spring-boot_default" with the default driver reating maven-first-app-spring-boot_default" done reating maven-first-app-spring-boot_app_1 done ttaching to maven-first-app-spring-boot_pMinguadmin_1 done reating maven-first-app-spring-boot_dpin_1 done ttaching to maven-first-app-spring-boot_pMinguadmin_1 done ttaching to maven-first-app-spring-boot_dpin_1 done ttaching to maven-first-app-spring-boot_dpin_1 done ttaching to maven-first-app-spring-boot_dpin_1 done ttaching to maven-first-app-spring-boot_dpin_1 done</pre>
<pre>> bb763a17d+71 tep 3/6 : KNN mkdir /app> Using cache> ed6c306b1568 tep 4/6 : COPY target/*.jar /app/my-spring-boot-application.jar> ae0970ccdc49 tep 5/6 : KNRYPOINT ['java', "-Djava.security.egd=file:/dev/./urandom", "-jar", "/app/my-spring-boot-application.jar"]> Running in 5d92759e3028> 0746934cfc06 tep 6/6 : HEALTHCHECKinterval=3mtimeout=3s CMD curl -f http://localhost/actuator/health/ exit 1> Running in bc2a078356ba emoving intermediate container bc2a078356ba> 027b69850af uccessfully built d297b69850af uccessfully tagged my_app_maven-first-app-spring-boot:latest :\Users\elhad\Desktop\AutoEntrepreneur\Java\Projects\NetBeansProjects\Elements-Cours\Micro-Services\maven-first-app-spring-boot/adp11" with the default driver reating maven-first-app-spring-boot_default" with reating maven-first-app-spring-boot_app_1 done ttaching to maven-first-app-spring-boot_default" i: done reating maven-first-app-spring-boot_physdmin_1, done reating maven-first-app-spring-boot_physdmin_1; Entrypoint script for MySQL Server 8.0.21-1deb1an0 started.</pre>
<pre>tep 3/6 : RUW mkdir /app > uedg cache > uedg cache > uedg70ccdc49 tep 5/6 : ENTRYPOINT ["java", "-Djava.security.egd=file:/dev/./urandom", "-jar", "/app/my-spring-boot-application.jar"] > Running in 5020759e3028 emoving intermediate container 5d92759e3028 > Running in bc2a078356ba emoving intermediate container bc2a078356ba > Running in bc2a078356ba emoving intermediate container bc2a078356ba > d297b69850af uccessfully built d297b69850af uccessfully tagged my_app_maven-first-app-spring-boot:latest :\Users\elhad\Desktop\AutoEntrepreneur\Java\Projects\NetBeansProjects\Elements-Cours\Micro-Services\maven-first-app-spring-boot>default" with the default driver reating maven-first-app-spring-boot_default" with the default driver reating maven-first-app-spring-boot_default" with the default driver reating maven-first-app-spring-boot_default" with the default driver reating maven-first-app-spring-boot_app_1 done reating maven-first-app-spring-boot_app_1 done</pre>
<pre>> using cache > using cache tep 4/6 : COPY target/*.jar /app/my-spring-boot-application.jar tep 5/6 : ENTRYPOINT ["java", "-Djava.security.egd=file:/dev/./urandom", "-jar", "/app/my-spring-boot-application.jar"] > Running in 5d92759e3028 emoving intermediate container 5d92759e3028 > 0746934cfc60 tep 6/6 : HEALTHCHECKinterval=3mtimeout=3s CMD curl -f http://localhost/actuator/health/ exit 1 > Running in bc2a078356ba emoving intermediate container bc2a078356ba > d297b69850af uccessfully built d297b69850af uccessfully built d297b69850af uccessfully tagged my_app_maven-first-app-spring-boot:latest :\Users\elhad\Desktop\AutoEntrepreneur\Java\Projects\NetBeansProjects\Elements-Cours\Micro-Services\maven-first-app-spring-boot>_ :\Users\elhad\Desktop\AutoEntrepreneur\Java\Projects\WetBeansProjects\Elements-Cours\Micro-Services\maven-first-app-spring-boot>default" with the default driver reating maven-first-app-spring-boot_default" with the default driver reating maven-first-app-spring-boot_dp1 done reating maven-first-app-spring-boot_app_1 done reating maven-first-app-spring-boot_app_1 done reating maven-first-app-spring-boot_phmyadmin_1, maven-first-app-spring-boot_db_1, maven-first-app-spring-boot_app_1 </pre>
<pre>> evec.sect.sect.sect.sect.sect.sect.sect.se</pre>
<pre>http://discorr.target/*.jak/pap/my-spring-bout-appirtation.jak > aee09Cott target/*.jak/pap/my-spring-bout-appirtation.jak tep 5/6 : ENTRYPOINT ["java", "-Djava.security.egd=file:/dev/./urandom", "-jar", "/app/my-spring-boot-application.jar"] > Running in 5d92759e3028 emoving intermediate container 5d92759e3028 emoving intermediate container 5d92759e3028 emoving intermediate container bd92759e3028 emoving intermediate container bd92759e3028 emoving in bc2a078356ba emoving in bc2a078356ba emoving in bc2a078356ba emoving intermediate container bc2a078356ba > d297b69850af uccessfully built d297b69850af uccessfully tagged my_app_maven-first-app-spring-boot:latest :\Users\elhad\Desktop\AutoEntrepreneur\Java\Projects\NetBeansProjects\Elements-Cours\Micro-Services\maven-first-app-spring-boot>_ :\Users\elhad\Desktop\AutoEntrepreneur\Java\Projects\NetBeansProjects\Elements-Cours\Micro-Services\maven-first-app-spring-boot>_ :\Users\elhad\Desktop\AutoEntrepreneur\Java\Projects\NetBeansProjects\Elements-Cours\Micro-Services\maven-first-app-spring-boot>_ :\Users\elhad\Desktop\AutoEntrepreneur\Java\Projects\NetBeansProjects\Elements-Cours\Micro-Services\maven-first-app-spring-boot>default" with the default driver reating maven-first-app-spring-boot_db_1 done reating maven-first-app-spring-boot_app_1 done reating maven-f</pre>
<pre>bits provide prov</pre>
<pre>c+c+c+c+c+c+c+c+c+c+c+c+c+c+c+c+c+c+c+</pre>
<pre>winning in Jobs/PSUG0 enving intermediate container 5d92759e3028 > 0746934cfc06 tep 6/6: HEALTHCHECKinterval=3mtimeout=3s CMD curl -f http://localhost/actuator/health/ exit 1 > Running in bc2a078356ba emoving intermediate container bc2a078356ba > d297b69850af uccessfully built d297b69850af uccessfully tagged my_app_maven-first-app-spring-boot:latest :\Users\elhad\Desktop\AutoEntrepreneur\Java\Projects\NetBeansProjects\Elements-Cours\Micro-Services\maven-first-app-spring-boot>_ :\Users\elhad\Desktop\AutoEntrepreneur\Java\Projects\NetBeansProjects\Elements-Cours\Micro-Services\maven-first-app-spring-boot>_ :\Users\elhad\Desktop\AutoEntrepreneur\Java\Projects\NetBeansProjects\Elements-Cours\Micro-Services\maven-first-app-spring-boot>_ :\Users\elhad\Desktop\AutoEntrepreneur\Java\Projects\NetBeansProjects\Elements-Cours\Micro-Services\maven-first-app-spring-boot>default" with the default driver reating maven-first-app-spring-boot_db_1 done reating maven-first-app-spring-boot_app_1 done reating maven-first-app-spring-boot_app_1 done ttaching to maven-first-app-spring-boot_phmyadmin_1, maven-first-app-spring-boot_db_1, maven-first-app-spring-boot_app_1 </pre>
<pre>> 0749934cfc06 tep 6/6 : HEALTHCHECKinterval=3mtimeout=3s CMD curl -f http://localhost/actuator/health/ exit 1 > Running in bc2a078356ba emoving intermediate container bc2a078356ba > d297b69850af uccessfully built d297b69850af uccessfully tagged my_app_maven-first-app-spring-boot:latest :\Users\elhad\Desktop\AutoEntrepreneur\Java\Projects\NetBeansProjects\Elements-Cours\Micro-Services\maven-first-app-spring-boot>_ :\Users\elhad\Desktop\AutoEntrepreneur\Java\Projects\NetBeansProjects\Elements-Cours\Micro-Services\maven-first-app-spring-boot>default with the default driver reating maven-first-app-spring-boot_default with the default driver reating maven-first-app-spring-boot_db_1 done reating maven-first-app-spring-boot_app_1 done ttaching to maven-first-app-spring-boot_app_1 done ttaching to maven-first-app-spring-boot_app_1 done ttaching to maven-first-app-spring-boot_oft_Entrypoint]: Entrypoint script for MySQL Server 8.0.21-1deblan10 started.</pre>
<pre>tep 6/6 : HEALTHCHECKinterval=3mtimeout=3s CMD curl -f http://localhost/actuator/health/ exit 1 > Running in bc2a078356ba emoving intermediate container bc2a078356ba > d297b69850af uccessfully built d297b69850af uccessfully tagged my_app_maven-first-app-spring-boot:latest ::\Users\elhad\Desktop\AutoEntrepreneur\Java\Projects\NetBeansProjects\Elements-Cours\Micro-Services\maven-first-app-spring-boot>_ :\Users\elhad\Desktop\AutoEntrepreneur\Java\Projects\NetBeansProjects\Elements-Cours\Micro-Services\maven-first-app-spring-boot>_ :\Users\elhad\Desktop\AutoEntrepreneur\Java\Projects\NetBeansProjects\Elements-Cours\Micro-Services\maven-first-app-spring-boot>_ :\Users\elhad\Desktop\AutoEntrepreneur\Java\Projects\NetBeansProjects\Elements-Cours\Micro-Services\maven-first-app-spring-boot>docker-compose up reating maven-first-app-spring-boot_default</pre>
> Running in bc2a078356ba emoving intermediate container bc2a078356ba > d297b69850af uccessfully built d297b69850af uccessfully tagged my_app_maven-first-app-spring-boot:latest :\Users\elhad\Desktop\AutoEntrepreneur\Java\Projects\NetBeansProjects\Elements-Cours\Micro-Services\maven-first-app-spring-boot>_ :\Users\elhad\Desktop\AutoEntrepreneur\Java\Projects\NetBeansProjects\Elements-Cours\Micro-Services\maven-first-app-spring-boot>_ :\Users\elhad\Desktop\AutoEntrepreneur\Java\Projects\NetBeansProjects\Elements-Cours\Micro-Services\maven-first-app-spring-boot>default" with the default driver reating maven-first-app-spring-boot_default" with the default driver reating maven-first-app-spring-boot_default" done reating maven-first-app-spring-boot_app_1 done reating maven-first-app-spring-boot_app_1 done ttaching to maven-first-app-spring-boot_app_1 202-09-28 10:30:45+00:00 [Mote] [Entrypoint script for MySQL Server 8.0.21:1deb1an10 started.
<pre>emoving intermediate container bc2a078356ba> d297b69850af uccessfully built d297b69850af uccessfully tagged my_app_maven-first-app-spring-boot:latest :\Users\elhad\Desktop\AutoEntrepreneur\Java\Projects\NetBeansProjects\Elements-Cours\Micro-Services\maven-first-app-spring-boot> :\Users\elhad\Desktop\AutoEntrepreneur\Java\Projects\NetBeansProjects\Elements-Cours\Micro-Services\maven-first-app-spring-boot>default" with the default driver reating maven-first-app-spring-boot_default" with the default driver reating maven-first-app-spring-boot_db_1 done reating maven-first-app-spring-boot_app_1 done reating maven-first-app-spring-boot_app_1 done reating maven-first-app-spring-boot_app_1 done reating maven-first-app-spring-boot_app_1 done reating to maven-first-app-spring-boot_app_1 done ttaching to ma</pre>
> d297b69850af uccessfully built d297b69850af uccessfully tagged my_app_maven-first-app-spring-boot:latest :\Users\elhad\Desktop\AutoEntrepreneur\Java\Projects\NetBeansProjects\Elements-Cours\Micro-Services\maven-first-app-spring-boot>
<pre>uccessfully built d297b69850af uccessfully tagged my_app_maven-first-app-spring-boot:latest :\Users\elhad\Desktop\AutoEntrepreneur\Java\Projects\NetBeansProjects\Elements-Cours\Micro-Services\maven-first-app-spring-boot></pre>
<pre>uccessfully built d297b69850af uccessfully tagged my_app_maven-first-app-spring-boot:latest :\Users\elhad\Desktop\AutoEntrepreneur\Java\Projects\NetBeansProjects\Elements-Cours\Micro-Services\maven-first-app-spring-boot>docker-compose up reating network "maven-first-app-spring-boot_default" with the default driver reating maven-first-app-spring-boot_db_1 done reating maven-first-app-spring-boot_app_1 done reating maven-first-app-spring-boot_app_1 done reating to maven-first-app-spring-boot_app_1 done ttaching to maven-first-app-spring-boot_app_1 done b_1 2020-09-28 10:30:45+00:00 [[Entrypoint]: Entrypoint script for MySQL Server 8.0.21-1debian10 started.</pre>
<pre>uccessfully tagged my_app_maven-first-app-spring-boot:latest :\Users\elhad\Desktop\AutoEntrepreneur\Java\Projects\NetBeansProjects\Elements-Cours\Micro-Services\maven-first-app-spring-boot>docker-compose up reating network "maven-first-app-spring-boot_default" with the default driver reating maven-first-app-spring-boot_dot_1 done reating maven-first-app-spring-boot_app_1 done reating maven-first-app-spring-boot_app_1 done ttaching to maven-first-app-spring-boot_app_1 done ttaching to maven-first-app-spring-boot_app_1 done</pre>
:\Users\elhad\Desktop\AutoEntrepreneur\Java\Projects\NetBeansProjects\Elements-Cours\Micro-Services\maven-first-app-spring-boot> :\Users\elhad\Desktop\AutoEntrepreneur\Java\Projects\NetBeansProjects\Elements-Cours\Micro-Services\maven-first-app-spring-boot>docker-compose up reating network "maven-first-app-spring-boot_default" with the default driver reating maven-first-app-spring-boot_db_1 done reating maven-first-app-spring-boot_app_1 done reating maven-first-app-spring-boot_app_1 done ttaching to maven-first-app-spring-boot_app_1 ttaching to maven-first-app-spring-boot_app_1 2020-09-28 10:30:45+00:00 [Mote] [Entrypoint]: Entrypoint script for MySQL Server 8.0.21-1debtan10 started.
:\Users\elnad\Desktop\AutoEntrepreneur\Java\Projects\WetBeansProjects\Elements-Lours\Micro-Services\maven-first-app-spring-boot>docker-compose up reating network "maven-first-app-spring-boot_default" with the default driver reating maven-first-app-spring-boot_db_1 done reating maven-first-app-spring-boot_app_1 done reating maven-first-app-spring-boot_app_1 done ttaching to maven-first-app-spring-boot_app_1 done b_1 2020-09-28 10:30:45+00:00 [Note] [Entrypoint]: Entrypoint script for MySQL Server 8.0.21-1debian10 started.
:\Users\elhad\Desktop\AutoEntrepreneur\Java\Projects\NetBeansProjects\Elements-Cours\Micro-Services\maven-first-app-spring-boot_default with the default driver reating maven-first-app-spring-boot_db_1 done reating maven-first-app-spring-boot_phpmyadmin_1 done reating maven-first-app-spring-boot_app_1 done ttaching to maven-first-app-spring-boot_app_1 done ttaching to maven-first-app-spring-boot_app_1 done ttaching to maven-first-app-spring-boot_app_1 done
reating maven-first-app-spring-boot_db_1 done reating maven-first-app-spring-boot_phpmyadmin_1 done reating maven-first-app-spring-boot_phpmyadmin_1 done reating maven-first-app-spring-boot_phpmyadmin_1 done reating maven-first-app-spring-boot_phpmyadmin_1, maven-first-app-spring-boot_db_1, maven-first-app-spring-boot_app_1 taching to maven-first-app-spring-boot_phpmyadmin_1, maven-first-app-spring-boot_db_1, maven-first-app-spring-boot_app_1 b_1
reating metwork maven-first-app-spring-boot_dolatic with the default driver reating maven-first-app-spring-boot_dola
reating movem-first-app-spring-boot_dpipmyadmin_1 done reating maven-first-app-spring-boot_app_1 done ttaching to maven-first-app-spring-boot_app_1 done ttaching to maven-first-app-spring-boot_app_1 b_1 2020-09-28 10:30:45+00:00 [Note] [Entrypoint]: Entrypoint script for MySQL Server 8.0.21-1debian10 started.
reating movem-first-app-spring-boot_phymydaman_r done reating maven-first-app-spring-boot_phpmyddmin_1, maven-first-app-spring-boot_db_1, maven-first-app-spring-boot_app_1 b_1
<pre>traching to maven-first-app-spring-boot_phpmyadmin_1, maven-first-app-spring-boot_db_1, maven-first-app-spring-boot_app_1 b_1</pre>
2020-09-28 10:30:45+00:00 [Note] [Entrypoint]: Entrypoint script for MySQL Server 8.0.21-1debian10 started.
b 1 2020-09-28 10:30:46+00:00 [Note] [Entrypoint]: Switching to dedicated user 'mysol'
2020-09-28 10:30:46+00:00 [Note] [Entrypoint]: Entrypoint script for MySOL Server 8.0.21-1debian10 started.
b 1 2020-09-28 10:30:46+00:00 [Note] [Entrypoint]: Initializing database files
b 1 2020-09-28T10:30:46.1816612 0 [System] [MY-013169] [Server] /usr/sbin/mysqld (mysqld 8.0.21) initializing of server in progress as pr
cess 43
b_1 2020-09-28T10:30:46.188463Z 1 [System] [MY-013576] [InnoDB] InnoDB initialization has started.
hpmyadmin_1 AH00558: apache2: Could not reliably determine the server's fully qualified domain name, using 192.168.208.2. Set the 'ServerName' di
ective globally to suppress this message
hpmyadmin_1 AH00558: apache2: Could not reliably determine the server's fully qualified domain name, using 192.168.208.2. Set the 'ServerName' di
ective globally to suppress this message
hpmyadmin_1 📋 [Mon Sep 28 10:30:44.867376 2020] [mpm_pretork:notice] [pid 1] AH00163: Apache/2.4.38 (Debian) PHP/7.4.9 contigured resuming norma
operations
njmysdamin_i _ [PON Sep 28 10:30:44.868610 2020] [COPE:NOTICE] [pid i] AH00094: Command line: apachez -D FOREGROUND.
pp 1 :: Spring Boot :: (v2.3.4.RELEASE)

On obtient sur http://localhost:8080/api :

\leftrightarrow \rightarrow C (i) localhost:8080/api

☆ θ :

[{"id":1, "field1:" "field2:" "field2-1", "version":0}, {"id":2, "field1": "field1-2", "field2": "field2-2", "version":0}, {"id":3, "field1": "field1-3", "field2": "field2-3", "version":0}, {"id":4, "field1": "field1-2", "field2": "field2-2", "version":0}, {"id":5, "field1": "field1-5", "field2": "field2-5", "version":0}, {"id":6, "field1": "field1-6", "field2": "field2-6", "version":0}, {"id":7, "field1": "field1-7", "field2": "field2-7", "version":0}, {"id":8, "field1": "field1-8", "field2": "field2-8", "version":0}, {"id":7, "field1": "field1-7", "field2": "field2-7", "version":0}, {"id":8, "field1": "field1-8", "field2": "field2-8", "version":0}, {"id":7, "field2": "field2-7", "field2": "field2-7", "version":0}, {"id":8, "field1": "field1-8", "field2": "field2-8", "version":0}, {"id":7, "field2": "field2-7", "field2": "field2-7", "version":0}, {"id":8, "field1": "field1-8", "field2": "field2-8", "version":0}, {"id":7, "field2": "field2-7", "version":0}, {"id":7, "field2": "field2-7", "field2": "field2-9", "version":0}, {"id":10, "field2": "field2-10", "field2": "field2-10", "version":0}, {"id":7, "field2": "field2-10", "version":0}, {"id":10, "field2": "field2-10", "version":0}, "version":0}, "id":10, "field2": "field2-10", "version":0}, "id":10, "field2": "field2-10", "version":0}, "version":0}, "version":0, "version":0}, "version":0, "version

Verifions l'état de santé de notre Micro Service Spring Boot : http://localhost:8080/actuator/health/

← → C ① localhost:8080/actuator/health/

{"status":"UP","components":{"db":{"status":"UP","details":{"database":"MySQL","validationQuery":"isValid()"}},"diskSpace":{"status":"UP","details": {"total":67371577344,"free":48623411200,"threshold":10485760,"exists":true}},"ping":{"status":"UP"}} 4. Version 4 : Image Docker contenant une image JDK 8 alpine (avec Maven 3.5.2 + le Jar du Micro Service Spring Boot) + image Docker MySQL 8

L'architecture de l'application ressemblera à :



Le fichier **maven-first-app-spring-boot/Dockerfile** devient donc :



En version copiable :

FROM maven:3.5.2-jdk-8-alpine AS MAVEN_TOOL_CHAIN COPY pom.xml /tmp/ RUN mvn -B dependency:go-offline -f /tmp/pom.xml -s /usr/share/maven/ref/settings-docker.xml COPY src /tmp/src/ WORKDIR /tmp/ RUN mvn -B -Dmaven.test.skip=true package FROM java:8-jre-alpine EXPOSE 8080 RUN mkdir /app COPY --from=MAVEN_TOOL_CHAIN /tmp/target/*.jar /app/my-spring-boot-application.jar ENTRYPOINT ["java","-Djava.security.egd=file:/dev/./urandom","-jar","/app/my-spring-boot-application.jar"] HEALTHCHECK --interval=3m --timeout=3s CMD curl -f http://localhost:8080/actuator/health/ || exit 1

Ligne 3 : Utilisation de Maven pour generer le projet jar du Micro Service à partir du **pom.xml**.

Lancer ensuite les commandes suivantes :

cd maven-first-app-spring-boot docker-compose build docker-compose up

Invite de commar	ides - docker-compose build — 🗆 🗙
C:\Users\elhad\	Desktop\AutoEntrepreneur\Java\Projects\NetBeansProjects\Elements-Cours\Micro-Services\maven-first-app-spring-boot>docker-compose build
db uses an imag phpmyadmin uses Puilding onn	e, skipping an image, skipping
Step 1/12 : FRO > 293423a98	1 maven:3.5.2-jdk-8-alpine AS MAVEN_TOOL_CHAIN La7
Step 2/12 : COP > 3d65ddf1c	/ pom.xml /tmp/ 560
Step 3/12 : RUN > Running i	mvn -B dependency:go-offline -f /tmp/pom.xml -s /usr/share/maven/ref/settings-docker.xml n 044c57cc3ffa
[INFO] Scanning [INFO] Download	for projects ing from central: https://repo.maven.apache.org/maven2/org/springframework/boot/spring-boot-starter-parent/2.3.4.RELEASE/spring-boot-s
[INFO] Download	.3.4.RELEASE.pom ed from central: https://repo.maven.apache.org/maven2/org/springframework/boot/spring-boot-starter-parent/2.3.4.RELEASE/spring-boot-st 3.4.RELEASE.pom (8.6 kB at 14 kB/s)
[INFO] Download endencies-2.3.4	ing from central: https://repo.maven.apache.org/maven2/org/springframework/boot/spring-boot-dependencies/2.3.4.RELEASE/spring-boot-dep .RELEASE.com
[INFO] Download ndencies-2.3.4.	ed from central: https://repo.maven.apache.org/maven2/org/springframework/boot/spring-boot-dependencies/2.3.4.RELEASE/spring-boot-depe RELEASE.pom (121 kB at 1.1 MB/s)
[INFO] Download [INFO] Download B/s)	ing from central: https://repo.maven.apache.org/maven2/com/datastax/oss/java-driver-bom/4.6.1/java-driver-bom-4.6.1.pom ad from central: https://repo.maven.apache.org/maven2/com/datastax/oss/java-driver-bom/4.6.1/java-driver-bom-4.6
[INFO] Download [INFO] Download	ing from central: https://repo.maven.apache.org/maven2/io/dropwizard/metrics/metrics-bom/4.1.12.1/metrics-bom-4.1.12.1.pom ed from central: https://repo.maven.apache.org/maven2/io/dropwizard/metrics/metrics-bom/4.1.12.1/metrics-bom-4.1.12.1.pom (5.3 kB at 1
[INFO] Download [INFO] Download	ing from central: https://repo.maven.apache.org/maven2/io/dropwizard/metrics/metrics-parent/4.1.12.1/metrics-parent-4.1.12.1.pom ed from central: https://repo.maven.apache.org/maven2/io/dropwizard/metrics/metrics-parent/4.1.12.1/metrics-parent-4.1.12.1.pom (15 kB
C:\Users\elhad\	Desktop\AutoEntrepreneur\Java\Projects\NetBeansProjects\Elements-Cours\Micro-Services\maven-first-app-spring-boot>docker-compose up
Creating networ Creating mayen-	k "maven-first-app-spring-boot_default" with the default driver First-ann-snring-boot dh 1
Creating maven-	First-app-spring-boot phpmyadmin 1 done
Creating maven-	first-app-spring-boot_app_1 done
Attaching to ma	ven-first-app-spring-boot_phpmyadmin_1, maven-first-app-spring-boot_db_1, maven-first-app-spring-boot_app_1
db_1	2020-09-28 10:5/12+00:00 [Note] [Entrypoint]: Entrypoint script for MySQL Server 8.0.21-10eDian10 started.
db_1	2020-09-28 10:57:12+00:00 [Note] [Entrypoint]: Smitching to deficated user mysdi 2020-09-28 10:57:12+00:00 [Note] [Entrypoint]: Entrypoint script for MySO Server 8.0.21-1debian10 started.
db 1	2020-09-28 10:57:12+00:00 [Note] [Entrypoint]: Initializing database files
db_1	2020-09-28T10:57:12.703795Z 0 [System] [MY-013169] [Server] /usr/sbin/mysqld (mysqld 8.0.21) initializing of server in progress as pr
ocess 42	
db_1 phpmyadmin_1	2020-09-28T10:57:12.711404Z 1 [System] [MY-013576] [InnoDB] InnoDB initialization has started. AH00558: apache2: Could not reliably determine the server's fully qualified domain name, using 192.168.224.2. Set the 'ServerName' di
rective globall	y to suppress this message
phpmyadmin_1	AH00558: apache2: Could not reliably determine the server's fully qualified domain name, using 192.168.224.2. Set the 'ServerName' di
phomyadmin 1	y to suppress this message Mon Sen 28 10:57:10 089612 2020] [mom prefork:notice] [nid 1] AH00163: Apache/2.4.38 (Debian) PHP/7.4.9 configured resuming norma
1 operations	[ion set to reputedoser recol [whwThelevenuerce] [hrs 1] where reputed (sector) / which is constrained and communication of the sector of the
phpmyadmin_1	[Mon Sep 28 10:57:10.089669 2020] [core:notice] [pid 1] AH00094: Command line: 'apache2 -D FOREGROUND'
app_1	
db_1	2020-09-28T10:57:14.563437Z 1 [System] [MV-013577] [InnoDB] InnoDB initialization has ended.
app_1	:: Spring Boot :: (V2.3.4.RELEASE)

On obtient sur http://localhost:8080/api :

\leftrightarrow \rightarrow C (i) localhost:8080/api

☆ \varTheta :

[{"id":1,"field1":"field1-1","field2":"field2-1","version":0},{"id":2,"field1":"field1-2","field2":"field2-2","version":0},{"id":3,"field1":"field1-3","field2":"field2-3","version":0},{"id":4,"field1":"field1-2","field2":"field2-4","version":0},{"id":5,"field1":"field1-5","field2-5","version":0},{"id":5,"field1":"field1-5","field2-5","version":0},{"id":6,"field1":"field1-6","field2":"field2-6","version":0},{"id":7,"field1":"field1-7","field2":"field2-5","version":0}, {"id":8,"field1":"field1-8","field1":"field1-8","field1":"field1-9","field2":"field2-9","version":0},{"id":10,"field1":"field1-10","field2":"field2-10","version":0}]

Verifions l'état de santé de notre Micro Service Spring Boot : <u>http://localhost:8080/actuator/health/</u>

← → C (i) localhost:8080/actuator/health/

{"status":"UP","components":{"db":{"status":"UP","details":{"database":"MySQL","validationQuery":"isValid()"}},"diskSpace":{"status":"UP","details": {"total":67371577344,"free":48623411200,"threshold":10485760,"exists":true}},"ping":{"status":"UP"}}