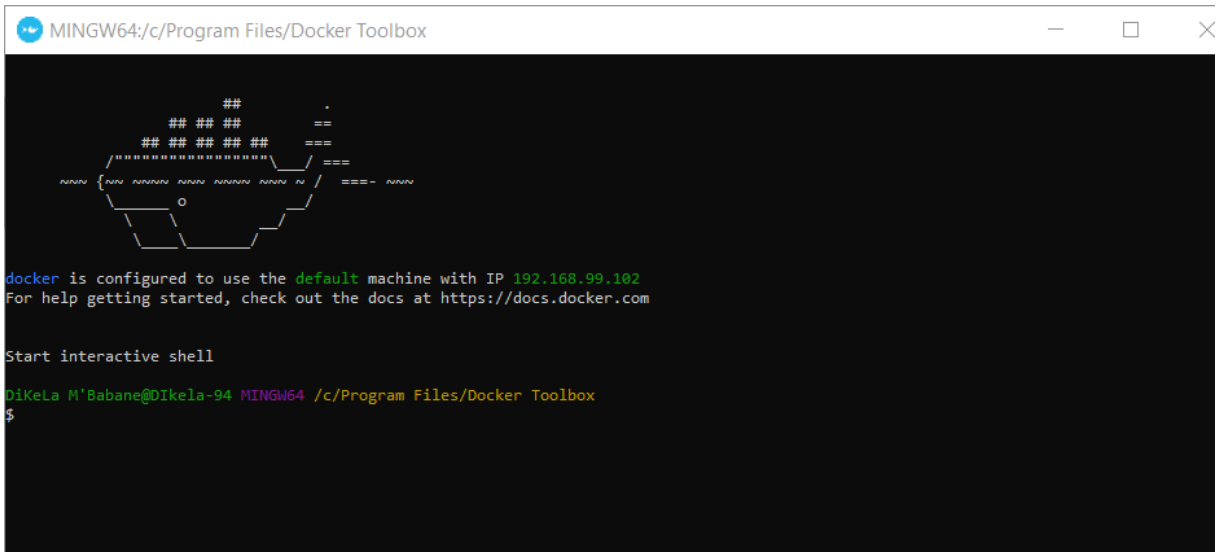
Exécution sous docker et déploiements

I- Docker

Avant l'utilisation de docker, il faut d'abord générer les différents jars pour chaque microservice en faisant « mvn install » en ligne de commande ou directement dans IntelliJ dans le panneau latéral droit .

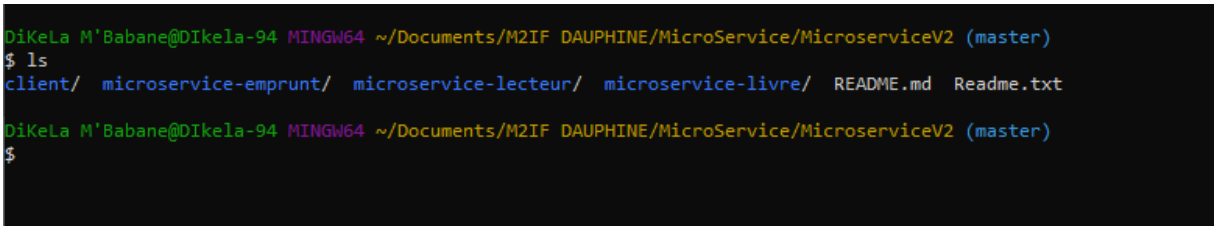
1 – Après installation de Docker toolbox sous Windows Home.

a) Lancer Docker Quickstart Terminal



Recupérer et noté quelque part l'adresse Ip de la machine : sur ma machine c'est le 192.168.99.102

b) Placer vous dans le répertoire du projet.



Vous pouvez voir les différents services de notre application. Il y a également le fichier readme.txt qui détail aussi les différentes instructions pour les exécutions.

c) Création de l'image du service sur docker

Place vous dans le service que vous désirai créer une image docker :

Exemple pour le microservice-lecteur :

> Cd microservice-lecteur

Construction de l’image docker

>docker build -t microservice-lecteur.jar .

Le point à la fin est à ne pas oublié

```
$ docker build -t microservice-lecteur.jar .
Sending build context to Docker daemon  40.67MB
Step 1/4 : FROM openjdk:14
 ---> 72467478c013
Step 2/4 : EXPOSE 8002
 ---> Running in d0691836cfc8
Removing intermediate container d0691836cfc8
 ---> b97c4ac7d3bc
Step 3/4 : ADD target/microservice-lecteur.jar microservice-lecteur.jar
 ---> 57c211e629bc
Step 4/4 : ENTRYPOINT ["java", "-jar", "/microservice-lecteur.jar"]
 ---> Running in 773649954e09
Removing intermediate container 773649954e09
 ---> 798ed0149539
Successfully built 798ed0149539
Successfully tagged microservice-lecteur.jar:latest
SECURITY WARNING: You are building a Docker image from Windows against a non-Windows Docker host. All files and directories add
build context will have '-rwxr-xr-x' permissions. It is recommended to double check and reset permissions for sensitive files a
ectories.

DiKeLa M'Babane@DIkela-94 MINGW64 ~/Documents/M2IF DAUPHINE/MicroService/MicroserviceV2/microservice-lecteur (master)
$ docker images
REPOSITORY                TAG                 IMAGE ID            CREATED             SIZE
microservice-lecteur.jar  latest              798ed0149539        12 seconds ago      537MB
microservice-livre.jar    latest              205e6f963b2f        18 hours ago        537MB
openjdk                   14                  72467478c013        9 days ago          497MB

DiKeLa M'Babane@DIkela-94 MINGW64 ~/Documents/M2IF DAUPHINE/MicroService/MicroserviceV2/microservice-lecteur (master)
$
```

On voit la dernière image créée il y a 12 secondes.

d) Exécution de l’image

> docker run -p 9090:8002 microservice-lecteur.jar

9090 est le port du conteneur
8002 est le port de la configuration dans le Dokerfile
Microservice-lecteur.jar est l’image docker créée

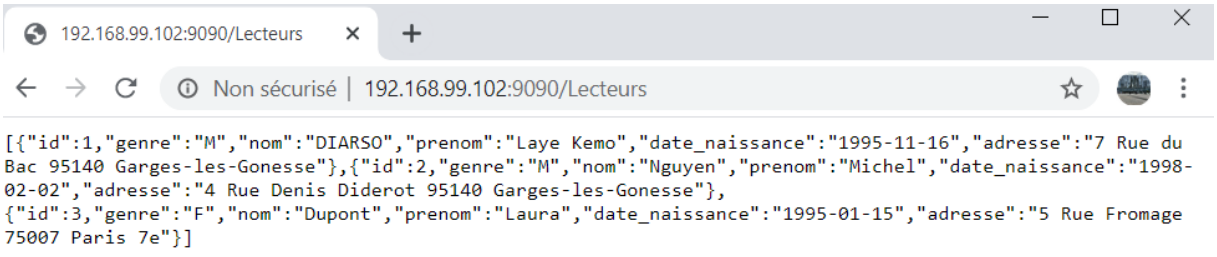
```
DiKeLa M'Babane@DIkela-94 MINGW64 ~/Documents/M2IF DAUPHINE/MicroService/MicroserviceV2/microservice-lecteur (master)
$ docker run -p 9090:8002 microservice-lecteur.jar

  .   ____          _            __ _ _
 /\\ / ___'_ __ _ _(_)_ __  __ _ \ \ \ \
( ( )\___ | '_ | '_| | '_ \/ _` | \ \ \ \
 \\/  ___)| |_)| | | | | || (_| |  ) ) ) )
  '  |____| .__|_| |_|_| |_\__, | / / / /
 =========|_|==============|___/=/_/_/_/
 :: Spring Boot ::        (v2.2.7.RELEASE)

2020-05-31 16:24:39.482  INFO 1 --- [           main] f.m.p.l.m.MicroserviceLecteurApplication : Starting MicroserviceLecteurApplicat
ion v0.0.1-SNAPSHOT on ff6703b335a6 with PID 1 (/microservice-lecteur.jar started by root in /)
2020-05-31 16:24:39.491  INFO 1 --- [           main] f.m.p.l.m.MicroserviceLecteurApplication : No active profile set, falling back
to default profiles: default
2020-05-31 16:24:42.921  INFO 1 --- [           main] .s.d.r.c.RepositoryConfigurationDelegate : Bootstrapping Spring Data JPA reposi
tories in DEFAULT mode.
2020-05-31 16:24:43.135  INFO 1 --- [           main] .s.d.r.c.RepositoryConfigurationDelegate : Finished Spring Data repository scan
ning in 181ms. Found 1 JPA repository interfaces.
2020-05-31 16:24:45.541  INFO 1 --- [           main] o.s.b.w.embedded.tomcat.TomcatWebServer  : Tomcat initialized with port(s): 800
2 (http)
2020-05-31 16:24:45.584  INFO 1 --- [           main] o.apache.catalina.core.StandardService   : Starting service [Tomcat]
2020-05-31 16:24:45.588  INFO 1 --- [           main] org.apache.catalina.core.StandardEngine  : Starting Servlet engine: [Apache Tom
cat/9.0.34]
2020-05-31 16:24:45.875  INFO 1 --- [           main] o.a.c.c.C.[Tomcat].[localhost].[/]       : Initializing Spring embedded WebAppl
icationContext
2020-05-31 16:24:45.879  INFO 1 --- [           main] o.s.web.context.ContextLoader            : Root WebApplicationContext: initiali
zation completed in 6160 ms
2020-05-31 16:24:46.418  INFO 1 --- [           main] com.zaxxer.hikari.HikariDataSource       : HikariPool-1 - Starting...
2020-05-31 16:24:47.244  INFO 1 --- [           main] com.zaxxer.hikari.HikariDataSource       : HikariPool-1 - Start completed.
2020-05-31 16:24:47.271  INFO 1 --- [           main] o.s.b.a.h2.H2ConsoleAutoConfiguration    : H2 console available at '/h2-console
'. Database available at 'jdbc:h2:mem:testdb'
2020-05-31 16:24:48.774  INFO 1 --- [           main] o.hibernate.jpa.internal.util.LogHelper  : HHH000204: Processing PersistenceUni
tInfo [name: default]
2020-05-31 16:24:48.999  INFO 1 --- [           main] org.hibernate.Version                    : HHH000412: Hibernate ORM core versio
n 5.4.15.Final
2020-05-31 16:24:49.497  INFO 1 --- [           main] o.hibernate.annotations.common.Version   : HCANN000001: Hibernate Commons Annot
ations {5.1.0.Final}
2020-05-31 16:24:49.883  INFO 1 --- [           main] org.hibernate.dialect.Dialect            : HHH000400: Using dialect: org.hibern
ate.dialect.H2Dialect
Hibernate: drop table if exists lecteur CASCADE
Hibernate: create table lecteur (id integer not null, adresse varchar(255), date_naissance timestamp, genre char(255), nom varchar(25
5), prenom varchar(255), primary key (id))
2020-05-31 16:24:52.268  INFO 1 --- [           main] o.h.e.t.j.p.i.JtaPlatformInitiator       : HHH000490: Using JtaPlatform impleme
ntation: [org.hibernate.engine.transaction.jta.platform.internal.NoJtaPlatform]
2020-05-31 16:24:52.280  INFO 1 --- [           main] j.LocalContainerEntityManagerFactoryBean : Initialized JPA EntityManagerFactory
 for persistence unit 'default'
2020-05-31 16:24:53.515  WARN 1 --- [           main] JpaBaseConfiguration$JpaWebConfiguration : spring.jpa.open-in-view is enabled b
y default. Therefore, database queries may be performed during view rendering. Explicitly configure spring.jpa.open-in-view to disabl
e this warning
2020-05-31 16:24:53.991  INFO 1 --- [           main] o.s.s.concurrent.ThreadPoolTaskExecutor  : Initializing ExecutorService 'applic
ationTaskExecutor'
2020-05-31 16:24:54.816  INFO 1 --- [           main] o.s.b.a.e.web.EndpointLinksResolver      : Exposing 2 endpoint(s) beneath base
path '/actuator'
2020-05-31 16:24:55.034  INFO 1 --- [           main] o.s.b.w.embedded.tomcat.TomcatWebServer  : Tomcat started on port(s): 8002 (htt
p) with context path ''
2020-05-31 16:24:55.051  INFO 1 --- [           main] f.m.p.l.m.MicroserviceLecteurApplication : Started MicroserviceLecteurApplicati
on in 17.117 seconds (JVM running for 19.17)
```

Ouvrir votre navigateur

Coller l'adresse IP copié précédemment suivi de : puis le port du conteneur



II- Déploiement des microservices

Les phases sont indiquées dans l'image suivante.
Cet exemple illustre le déploiement de notre service « Client »

MINGW64:/c/Users/DiKeLa M'Babane/Documents/M2IF DAUPHINE/MicroService/MicroserviceV2/client

```
DiKeLa M'Babane@DIkela-94 MINGW64 ~/Documents/M2IF DAUPHINE/MicroService/MicroserviceV2/client (master)
$ minikube start --driver=virtualbox
* minikube v1.10.1 sur Microsoft Windows 10 Home 10.0.18362 Build 18362
* Utilisation du pilote virtualbox basé sur le profil existant
* Démarrage du noeud de plan de contrôle minikube dans le cluster minikube
* minikube 1.11.0 is available! Download it: https://github.com/kubernetes/minikube/releases/tag/v1.11.0
* To disable this notice, run: 'minikube config set WantUpdateNotification false'

* Restarting existing virtualbox VM for "minikube" ...
* Options de réseau trouvées :
  - NO_PROXY=192.168.99.100,192.168.99.102
  - no_proxy=192.168.99.100,192.168.99.102
* Préparation de Kubernetes v1.18.2 sur Docker 19.03.8...
  - env NO_PROXY=192.168.99.100,192.168.99.102
  - env NO_PROXY=192.168.99.100,192.168.99.102
* Terminé ! kubectl est maintenant configuré pour utiliser "minikube".

DiKeLa M'Babane@DIkela-94 MINGW64 ~/Documents/M2IF DAUPHINE/MicroService/MicroserviceV2/client (master)
$ kubectl cluster-info
Kubernetes master is running at https://192.168.99.101:8443
KubeDNS is running at https://192.168.99.101:8443/api/v1/namespaces/kube-system/services/kube-dns:dns/proxy

To further debug and diagnose cluster problems, use 'kubectl cluster-info dump'.

DiKeLa M'Babane@DIkela-94 MINGW64 ~/Documents/M2IF DAUPHINE/MicroService/MicroserviceV2/client (master)
$ kubectl create -f client.yaml
service/client created
deployment.apps/client created

DiKeLa M'Babane@DIkela-94 MINGW64 ~/Documents/M2IF DAUPHINE/MicroService/MicroserviceV2/client (master)
$ kubectl get deployments
NAME     READY   UP-TO-DATE   AVAILABLE   AGE
client   0/3     3            0           14s

DiKeLa M'Babane@DIkela-94 MINGW64 ~/Documents/M2IF DAUPHINE/MicroService/MicroserviceV2/client (master)
$ kubectl get services
NAME         TYPE        CLUSTER-IP     EXTERNAL-IP   PORT(S)          AGE
client       NodePort    10.101.114.8   <none>        8083:30001/TCP   25s
kubernetes   ClusterIP   10.96.0.1      <none>        443/TCP          24h

DiKeLa M'Babane@DIkela-94 MINGW64 ~/Documents/M2IF DAUPHINE/MicroService/MicroserviceV2/client (master)
$ kubectl get services
NAME         TYPE        CLUSTER-IP     EXTERNAL-IP   PORT(S)          AGE
client       NodePort    10.101.114.8   <none>        8083:30001/TCP   2m6s
kubernetes   ClusterIP   10.96.0.1      <none>        443/TCP          24h

DiKeLa M'Babane@DIkela-94 MINGW64 ~/Documents/M2IF DAUPHINE/MicroService/MicroserviceV2/client (master)
$ minikube service client
|-----------|--------|-------------|--------------------------|
| NAMESPACE |  NAME  | TARGET PORT |           URL            |
|-----------|--------|-------------|--------------------------|
| default   | client |        8083 | http://192.168.99.101:30001 |
|-----------|--------|-------------|--------------------------|
* Opening service default/client in default browser...

DiKeLa M'Babane@DIkela-94 MINGW64 ~/Documents/M2IF DAUPHINE/MicroService/MicroserviceV2/client (master)
$
```