Apache Derby Image

These images support the operational use of Apache Derby Docker containers.

1. Creating a new Apache Derby container

1.1 Getting started

```
> REM Assumptions:
> REM   - you want to map the container port 1527 to the host port 8000
> REM   - the name of the Docker container should be: my_derby_db
> REM   - the path the host repository is: //C/projects/my_database
> REM   - you want to use the latest version of the Apache Derby image
> docker run -it -p 8000:1527 \
             --name my_derby_db \
             -v //C/projects/my_database:/dbs \
             konnexionsgmbh/apache_derby:latest

> REM Stopping the container
> docker stop my_derby_db

> REM Restarting the container
> docker start my_derby_db

> REM Entering a running container
> docker exec -it my_derby_db bash
```

1.2 Detailed Syntax

A new container can be created with the `docker run` command.

Syntax:

```
docker run -it
           [-p <port>:1527] \
           [--name <container_name>] \
           [-v <directory_repository>:/dbs] \
           konnexionsgmbh/apache_derby[:<version>]
           [<cmd>]
```

Parameters:

- **port** - an optional listener port
- **container_name** - an optional container identification

- **directory_repository** - an optional host database directory - the default value is expecting the database inside the container
- **version** - an optional version number of the image or the constant `latest`
- **cmd** - an optional command to be executed in the container, default is `bash` for running the `bash` shell

Detailed documentation for the command `docker run` can be found here.

Examples:

1. Creating a new Docker container named `my_derby_db` using a database inside the Docker container:

   ```
   docker run -it --name my_derby_db konnexionsgmbh/apache_derby:latest
   ```

2. Creating a new Docker container named `my_derby_db` using the database of a Windows directory `D:\projects\my_database`:

   ```
   docker run -it --name dderl_dev -v //D/projects/my_database:/dbs konnexionsgmbh/apache_derby:latest
   ```

3. Creating a new Docker container named `my_derby_db` using the host database of a Linux directory `/my_database` and mapping port `8000` to port `1527`:

   ```
   docker run -it --name my_derby_db -p 8000:1527 -v /my_database:/dbs konnexionsgmbh/apache_derby:latest
   ```

2 Working with an existing Apache Derby container

2.1 Starting a stopped container

A previously stopped container can be started with the `docker start` command.

Syntax:

```
docker start <container_name>
```

Parameter:

- **container_name** - the mandatory container identification, that is an UUID long identifier, an UUID short identifier or a previously given name

Detailed documentation for the command `docker start` can be found here.

2.2 Entering a running container

A running container can be entered with the `docker exec` command.

Syntax:

```
docker exec -it <container_name> <cmd>
```

Parameter:

- **container_name** - the mandatory container identification, that is an UUID long identifier, an UUID short identifier or a previously given name
- **cmd** - the command to be executed in the container, e.g. `bash` for running the `bash` shell

Detailed documentation for the command `docker exec` can be found here.

3 Installed packages

With the following command you can check in detail which package versions are included in the Docker image:

```
apk info --vv
```