U5S8CNULD : Guys, anyone worked with docker?
U1UFZTD4J : lots of us have <@U5S8CNULD>
U5S8CNULD : Could you help me? I can't connect mysql image with my python image;(
U1BP42MRS : I presume you are using compose?
U1BP42MRS : Just post your question :smile:
U5S8CNULD : one moment;)
U5S8CNULD : &gt;I presume you are using compose?no:slightly_smiling_face:

U5S8CNULD : first I running, MySQL image```
docker run --name my-db -e MYSQL_ROOT_PASSWORD=t123 mysql
```

then I build the image for the python app
```
docker build --rm -t myapp:dev -f Dockerfile .
```
Dockerfile body
```
FROM python:2.7
RUN apt-get update &amp;&amp; apt-get install -y python-dev

ENV DEBIAN_FRONTEND noninteractive
RUN echo "en_US.UTF-8 UTF-8" &gt; /etc/locale.gen &amp;&amp; locale-gen

ENV PYTHONPATH /app

RUN mkdir app
WORKDIR /app/

# Ensure that Python outputs everything that's printed inside
# the application rather than buffering it.
ENV PYTHONUNBUFFERED 1

# Defaults to PIP
ENV PIP_NO_CACHE_DIR off
ENV PIP_DISABLE_PIP_VERSION_CHECK on

ENTRYPOINT ["/bin/bash"]
```

then I running my app image
```
docker run --name=test-app -p 8000:8000 -it -v $PWD:/app --link my-db --env-file=.env myapp:dev
```

and tried to run flask app in the docker container, and got error
```
Can't connect to local MySQL server through socket '/var/run/mysqld/mysqld.sock'
```

U5S8CNULD : Spend a lot of time, and can't find solution:disappointed:
U1BP42MRS : Looks like you are trying to access the MySQL server through the socket, I think you need to set it to use the port and the hostname `my-db` (or whatever `link` arg you name)
U5S8CNULD : ```docker run --name -p 3306 -h localhost  -e MYSQL_ROOT_PASSWORD=t123  mysql
```
something like this?

U1BP42MRS : No, it's what you link it as, so the `docker run --name=test-app -p 8000:8000 -it -v $PWD:/app --link my-db --env-file=.env myapp:dev` The `--link my-db` would name the link `my-db`
U1BP42MRS : So I think you need to change your mysql connection string to be looking at the port, not a local socket

U5S8CNULD : okay I'll try.
```
docker run --name=test-app -p 8000:8000 -it -v $PWD:/app --link -p 3306 -h localhost my-db --env-file=.env myapp:dev
```
this one;)? sorry, I :disappointed:confused

U1BP42MRS : that looks invalid, `--link` requires an argument
U1BP42MRS : I am saying, just use what you originally posted, but in your app code where you connect to the database, use the hostname `my-db`
U1BP42MRS : What lib are you using to talk to your DB?
U5S8CNULD : mysql-python
U5S8CNULD : and uri for now
```
SQLALCHEMY_DATABASE_URI = '<mysql://root:t123@localhost/dstore>'
```

U1BP42MRS : So presuming you are using `--link my-db`, you should be able to change `localhost` to `my-db` for the host
U5S8CNULD : One moment, I'll try.
U5NMSURAQ : <@U1BP42MRS> unrelated question, but can you theoretically share the socket between containers? or is this too ridiculous?
U5S8CNULD : <@U1BP42MRS> Thanks! it's working!!! beer from me:beers: