We have found that using a traditional conda environment which uses the command “conda run” causes problems with command line inputs and SMILES strings. We have found it to be much easier to install the packages into the BASE environment rather than its own environment.

Please use the following instructions to create the environment.yml for your container. It is easiest to build the environment interactively inside the miniconda3 container because you can get immediate feedback as to whether the packages are compatible. Building your own container over and over to create the environment is much more time consuming.

Creating a conda environment inside a container

1. Start up a miniconda container
   - `docker run -it --rm continuumio/miniconda3`
   - Following running this command, the CLI command prompt should change to something like:

   ```
   (base) root@b986b748b578:/#
   ```

2. Create a conda environment
   - `conda create --name <env_name> python=<version>`
3. Activate the environment
   - `conda activate <env_name>`
4. Install the required packages
   - `conda install -c <channel> package`
5. Test your environment to make sure all your imports work correctly
   - `python`

   ```
   (env_name) root@b986b748b578:/# python
   Python 3.7.10 (default, Jun 4 2021, 14:48:32) [GCC 7.5.0]
   :: Anaconda, Inc. on linux Type "help", "copyright",
   "credits" or "license" for more information.
   >>> from <package> import <module>
   >>>
   >>> quit()
   ```

6. Once you are sure all the packages have been correctly installed, export your environment to a yaml file
   - With your environment activated, run:
     - `conda env export > environment.yml`
7. `cat` the yaml file and copy and paste its contents into an environment.yml file outside the container
   - `cat environment.yml`
8. Edit your environment.yml file so the "`name:`" is "`base`" on line 1 and remove the last line "`prefix: /opt/conda/envs/<env_name>`" (see below)

`environment.yml` before Step 8

```
1   name: oepy37
2   channels:
3     - openeye
4     - omnia
5     - conda-forge
6     - defaults
7   dependencies:
8     - _libgcc_mutex=0.1=main
9     - _openmp_mutex=4.5=1_gnu
10    - alabaster=0.7.12=py37_0
11    - ambertools=21.3=py37h13c71af_0
12    - arpack=3.7.0=hdefa2d7_2
13    - astunparse=1.6.3=pyhd8ed1ab_0
14    - attrs=21.2.0=pyhd3eb1b0_0
15    - babel=2.9.1=pyhd3eb1b0_0
16    - blas=1.0=mkl
17  prefix: /opt/conda/envs/oepy37
```

→

`environment.yml` after Step 8

```
1   name: base
2   channels:
3     - openeye
4     - omnia
5     - conda-forge
6     - defaults
7   dependencies:
8     - _libgcc_mutex=0.1=main
9     - _openmp_mutex=4.5=1_gnu
10    - alabaster=0.7.12=py37_0
11    - ambertools=21.3=py37h13c71af_0
12    - arpack=3.7.0=hdefa2d7_2
13    - astunparse=1.6.3=pyhd8ed1ab_0
14    - attrs=21.2.0=pyhd3eb1b0_0
15    - babel=2.9.1=pyhd3eb1b0_0
16    - blas=1.0=mkl
17
```

9. In your `Dockerfile`, add the following

```
FROM continuumio/miniconda3:4.9.2-alpine
WORKDIR /opt/app
COPY . ./
RUN conda env update -f environment.yml && \\
    conda clean --all --yes
ENTRYPOINT ["<entrypoint_from_setup.py>"]
```