Docker-Compose Basics

It helps automating the docker image build process

# For a Pre-created Image:

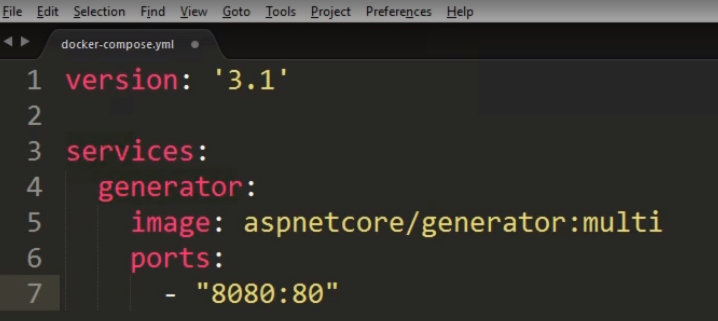
## Building a Basic Docker-compose document:

Name the file as *docker-compose.yml*. This file shall remain in same folder as *Dockerfile*.

Mention the *version* and then start listing the s*ervices*.

For correct version compatibility with docker version, take a look at following page for updated information: <https://docs.docker.com/compose/compose-file/compose-versioning/#compatibility-matrix>

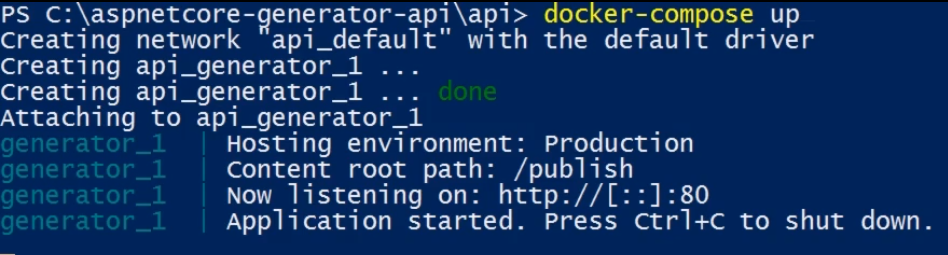
Each service can have any name specified. In following example, *generator* is a service name given for sample project



## Running the Docker-compose:

Go to same directory on PS and run the command *docker-compose up*

By default, it takes docker-compose.yml as compose file name and current directory as default directory.



To run the container in detached mode(to free up your PS), use –d flag

*docker-compose up –d*

* to see the logs

*docker-compose –logs*

* to keep following the logs

*docker-compose logs –f*

To completely clean the container footprint:

docker-compose down

For one-by-one cleanup of specific containers, use following mechanism:

docker-compose stop <container id>

docker-compose rm <container id>

# For an Image to be built using Docker-compose:

Let’s take an example of command line for a typical build process as in below picture

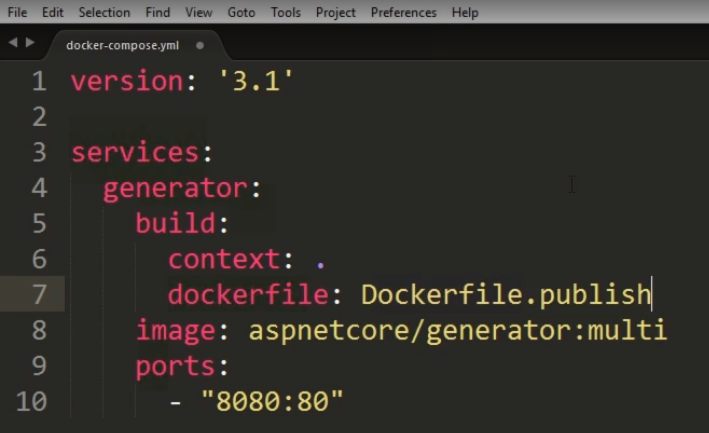


Red marks the context of the build process (relative to *dockerfile*). We are for this example, assuming the *docker-compose.yml* too being in same folder as *dockerfile*.

Aspnetcore/generator – base image name

multi – image tag name

## Docker-compose equivalent for the command line:



In the above Docker-compose.yml,

build – specifies the section for build process of a potential image

context – specifies the docker context relative to docker-compose file

dockerfile – required only in case when Dockerfile is named something different than default *Dockerfile*

args – if there are any arguments to build command

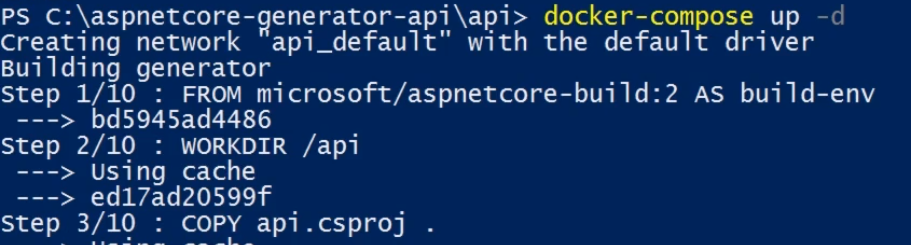
image – tag name of potential image to be built(based on dockerfile)

ports – port forwarding to host computer’s port 8080 from container’s port 80

***Important to note: always ensure that Docker-compose.yml is added as an entry in .dockerignore file, so that any future changes in tis file shall not ignore the caching optimization***

## Runing the Docker-compose with build image process:

Use the docker-compose up -d

****

That would build and run the container.

If image already exists and changes are there in docker-compose file regarding build process, it’s advised to just use –build parameter as following:

Docker-compose build

OR

Docker-compose up –d –build