Docker Cheatsheet

Docker Concepts

- **docker**: Creates, deploys and manages containers on different operating environmens (win/mac/linux/kubernetes)

- **image**: A filesystem containing the application docker will run.

- **layer**: The image storage format is composed of read-only file system layers; the container creates writable layers.

- **container**: Isolated environment within a kernel for an application to run. Has it's own root fs, networking, ...

- **Dockerfile**: A file detailing the steps to build a Docker image.

- **docker engine**: Docker installation running on a given host that can run containers, build images, ...

- **docker client**: A client that talks to local or remote Docker daemon which is used to manage containers, images, ...

- **docker daemon**: Background service managing images and containers.

- **docker registry**: Stores versioned docker images, associated with a tag, a string used to mark versions or variants of an image.

- **volume**: Directory shared between container and host.

Docker cli

`docker run [OPTIONS] IMAGE[:TAG] [COMMAND]`

- `--name=CONTAINER_NAME`: Assign name to container.

- `--label NAME=VALUE`: Set metadata on container.

- `-d, --detach`: Run in the background.

- `-i, --interactive`: Interactive.

- `-t, --tty`: Allocate a pseudo-TTY.

- `--rm`: Automatically remove container when entrypoint process exits in the container.

- `-u, --user=[user[:group]]`: Run as specified user/uid, and group/gid.

- `--privileged`: Allow access to all host devices.

- `-v, --volume`: Bind mount a volume from the host.

- `--volume-from`: Mount volumes from another container.

- `-e, --env`: Set environment variables.

- `--env-file`: Read environment variables from a file.

- `-p, --publish [host_port]:[container_port]`: Publish container's ports to the host.

- `--expose`: Expose a port or a range of ports.

- `-h, --hostname`: Set container's hostname.

- `--network`: Connect container to a network.

- `--add-host`: Add a custom host-to-IP mapping.

- `--read-only`: Mount root read-only.

Dockerfile Instructions

Instructions:

- **FROM**: Set the base image to build from.

- **RUN**: Execute a command in the shell or exec form.

- **MAINTAINER**: Set the author field.

- **LABEL**: Add metadata.

- **ARG**: Define build-time variables.

- **ENV**: Set environment variables.

- **ADD**: Copy files from source (can be remote or archive) and add them to the image. *discouraged*

- **COPY**: Copy files + directories from host to image.

- **VOLUME**: specify mount point for a volume.

- **EXPOSE**: Expose a port or ports.

- **WORKDIR**: Set the working directory for any `RUN`, `CMD`, `ENTRYPOINT`, `COPY`, and `ADD` instructions that follow it.

- **USER**: Set the user name or UID to use when running the image.

- **CMD**: Provide defaults for an executing container.

- **ENTRYPOINT**: Configure a container that will run as an executable.