# Docker Cheatsheet

## Overview

Docker is a platform for developing, shipping, and running applications in containers.

## Installation

- Docker Desktop for Mac
- Docker Desktop for Windows
- Docker Engine for Linux

## Basic Usage

1. Create a `Dockerfile` in the root directory of your project.
2. Define the base image and any additional dependencies or configuration.
3. Build the Docker image using `docker build`.
4. Run the Docker container using `docker run`.

## Dockerfile Syntax

```
# Use an official Python runtime as a parent image
FROM python:3.7-slim

# Set the working directory to /app
WORKDIR /app

# Copy the current directory contents into the container at /app
COPY . /app

# Install any needed packages specified in requirements.txt
RUN pip install --trusted-host pypi.python.org -r requirements.txt

# Make port 80 available to the world outside this container
EXPOSE 80

# Define environment variable
ENV NAME World

# Run app.py when the container launches
CMD ["python", "app.py"]
```

- The `FROM` keyword specifies the base image to use.
- The `WORKDIR` keyword sets the working directory for subsequent commands.
- The `COPY` keyword copies files from the host to the container.
- The `RUN` keyword executes a command in the container.
- The `EXPOSE` keyword exposes a port for the container.
- The `ENV` keyword sets an environment variable.
- The `CMD` keyword specifies the default command to run when the container starts.

## Commands

- `docker build` : Builds a Docker image from a Dockerfile.
- `docker run` : Runs a Docker container from an image.
- `docker ps` : Lists all running containers.
- `docker stop` : Stops a running container.
- `docker rm` : Removes a stopped container.
- `docker rmi` : Removes an image.
- `docker-compose` : Manages multi-container Docker applications.

## Resources

- Docker website
- Docker documentation
- Docker Hub