docker-compose cheatsheet

Basic example

```
# docker-compose.yml
version: '2'

services:
  web:
    build: .
    # build from Dockerfile
    context: ./Path
    dockerfile: Dockerfile
    ports:
     - "5000:5000"
    volumes:
     - .:/code
  redis:
    image: redis
```

Commands

```
docker-compose start
docker-compose stop
```

```
docker-compose pause
docker-compose unpause
```

```
docker-compose ps
docker-compose up
docker-compose down
```

Reference

Building

```
web:
  # build from Dockerfile
  build: .
  args:     # Add build arguments
    APP_HOME: app
```

```
  # build from custom Dockerfile
  build:
    context: ./dir
    dockerfile: Dockerfile.dev
```

```
  # build from image
  image: ubuntu
  image: ubuntu:14.04
  image: tutum/influxdb
  image: example-registry:4000/postgresql
  image: a4bc65fd
```

Ports

```
  ports:
    - "3000"
    - "8000:80"  # host:container
```

```
  # expose ports to linked services (not to host)
  expose: ["3000"]
```

Commands

```
  # command to execute
  command: bundle exec thin -p 3000
  command: [bundle, exec, thin, -p, 3000]
```

```
  # override the entrypoint
  entrypoint: /app/start.sh
  entrypoint: [php, -d, vendor/bin/phpunit]
```

Environment variables

```
  # environment vars
  environment:
    RACK_ENV: development
  environment:
    - RACK_ENV=development
  # environment vars from file
  env_file: .env
  env_file: [.env, .development.env]
```

Dependencies

```
  # makes the `db` service available as the hostname `database`
  # (implies depends_on)
  links:
    - db:database
    - redis
```

```
  # make sure `db` is alive before starting
  depends_on:
    - db
```

Other options

```
  # make this service extend another
  extends:
    file: common.yml  # optional
    service: webapp
```

```
  volumes:
    - /var/lib/mysql
    - ./_data:/var/lib/mysql
```

Advanced features

Labels

```
services:
  web:
    labels:
      com.example.description: "Accounting web app"
```

DNS servers

```
services:
  web:
    dns: 8.8.8.8
    dns:
      - 8.8.8.8
      - 8.8.4.4
```

Devices

```
services:
  web:
    devices:
    - "/dev/ttyUSB0:/dev/ttyUSB0"
```

External links

```
services:
  web:
    external_links:
      - redis_1
      - project_db_1:mysql
```

Hosts

```
services:
  web:
    extra_hosts:
      - "somehost:192.168.1.100"
```

Network

```
# creates a custom network called `frontend`
networks:
  frontend:
```

External network

```
# join a pre-existing network
networks:
  default:
    external:
      name: frontend
```

Volume

```
# Mount host paths or named volumes, specified as sub-options to a service
  db:
    image: postgres:latest
    volumes:
      - "/var/run/postgres/postgres.sock:/var/run/postgres/postgres.sock"
      - "dbdata:/var/lib/postgresql/data"

volumes:
  dbdata:
```