

# Play With Docker

# Basic Docker Commands

- docker run **IMAGE\_NAME**
- docker run --name **ContainerName** **IMAGE\_NAME**
- docker run --name **ContainerName** **IMAGE\_NAME** -d
  
- docker image ls
- docker image rm **IMAGE\_NAME**
- docker image rm **IMAGE\_ID**
  
- docker container ls
- docker container ls -a
- docker container logs **ContainerName**
- docker container logs **ContainerID**
  
- docker exec **ContainerName** ls
- docker exec -it **ContainerName** bash
- docker exec -it **ContainerID** bash

# Dockerfile commands

- FROM
- MAINTAINER
- LABEL
- ENV
- EXPOSE
- VOLUME
- WORKDIR
- COPY
- ADD
- CMD
- RUN
- ENTRYPOINT

# Dockerfile commands

- FROM (Base image source)
- MAINTAINER (Author)
- LABEL (Image metadata)
- ENV (Environment variable)
- EXPOSE (Share Ports from inside container to outside)
- VOLUME (Share files from your host inside container)
- WORKDIR (Default working directory that container will use)
- COPY (Add local files to docker image)
- ADD (Add Local/URL/TAR files to docker image)
- CMD (Running command in starting container)
- RUN (Running command in build docker image)
- ENTRYPOINT (Default application that run every time container start)

# Dockerfile example

```
FROM nginx:latest
```

```
MAINTAINER kelany@afaqy.com
```

```
LABEL version="1.0"
```

```
ENV NGINX_HOST kelanyexample.com
```

```
ENV NGINX_PORT 80
```

```
EXPOSE $NGINX_PORT
```

```
COPY website /usr/share/nginx/html
```

```
WORKDIR /usr/share/nginx/html
```

# Dockerfile run example

- docker build -t **IMAGE\_NAME** WorkDirectory
- docker build -t **IMAGE\_NAME:TAG** WorkDirectory

```
JKelanyMBP:Desktop jkelany$ docker build -t afaqy-nginx:1.0.0 afaqy-nginx/  
Sending build context to Docker daemon 8.314MB  
Step 1/8 : FROM nginx:latest  
----> e445ab08b2be  
Step 2/8 : MAINTAINER kelany@afaqy.com  
----> Using cache  
----> 96531125bdf6  
Step 3/8 : LABEL version="1.0"  
----> Using cache  
----> ec8771500289  
Step 4/8 : ENV NGINX_HOST kelanyexample.com  
----> Using cache  
----> 6580f6e5b5e8  
Step 5/8 : ENV NGINX_PORT 80  
----> Using cache  
----> 430ce361e395  
Step 6/8 : EXPOSE $NGINX_PORT  
----> Using cache  
----> 7bd0a73040b0  
Step 7/8 : COPY website /usr/share/nginx/html  
----> Using cache  
----> 7ffef7ad42e6  
Step 8/8 : WORKDIR /usr/share/nginx/html  
----> Using cache  
----> 684be383b65e  
Successfully built 684be383b65e  
Successfully tagged afaqy-nginx:1.0.0
```

# Docker-compose.yml example

```
version: '3'
services:
  web:
    build: .
    ports:
      - "8080:80"
  db:
    image: mysql
    ports:
      - "3306:3306"
    environment:
      - MYSQL_ROOT_PASSWORD=password
      - MYSQL_USER=user
      - MYSQL_PASSWORD=password
      - MYSQL_DATABASE=demodb
```

# Docker-compose.yml run example

- `docker-compose up`
- `docker-compose up -d`
- `docker-compose down`
- `docker-compose down -d`
  
- `docker-compose stop`
- `docker-compose start`



# References

- <https://docs.docker.com/engine/reference/builder>
- <https://docs.docker.com/engine/reference/commandline>
- <https://docs.docker.com/compose/compose-file/>

# Samples

- <https://docs.docker.com/samples/>

# Online lab

- <https://labs.play-with-docker.com/>

# Questions?

## Thank YOU

