**```dockerfile Instructions and Examples**

This document provides an overview of ```dockerfile instructions, using the Ubuntu base image for each example.

1. FROM

The FROM instruction sets the base image for the Docker container. It is always the first instruction in a ```dockerfile.

FROM ubuntu:latest

This example pulls the latest Ubuntu image from Docker Hub.

Command to Build and Run:

docker build -t app:1.0 .

to run the image:

docker run -it app:1.0

1. COPY The COPY instruction copies files or directories from the host machine to the container.

FROM ubuntu:latest  
COPY ./myfile.txt /app/myfile.txt

This copies myfile.txt from the current directory on the host to the /app/ directory in the container.

Command to Build and Run:

docker build -t copyexample:1.0 .

run it at:

docker run -it copyexample:1.0

1. ADD The ADD instruction works similarly to COPY but can also extract compressed files and fetch files from URLs.

FROM ubuntu:latest  
ADD ./archive.tar.gz /app/

This extracts archive.tar.gz into the /app/ directory in the container.

Command to Build and Run:

docker build -t addexample:1.0 .

Run the container

docker run -it addexample:1.0

1. RUN The RUN instruction executes commands inside the container during the build process.

FROM ubuntu:latest  
RUN apt-get update && apt-get install -y curl

This updates the package list and installs curl.

Command to Build and Run:

docker build -t runexample:1.0 .

Run the container

docker run -it runexample:1.0

1. ENV The ENV instruction sets environment variables inside the container.

FROM ubuntu:latest  
ENV MY\_ENV\_VAR=myvalue

This sets the environment variable MY\_ENV\_VAR to myvalue.

Command to Build and Run:

docker build -t envexample:1.0 .

Run the container

docker run -it envexample:1.0

1. ARG The ARG instruction defines build-time variables that can be passed at build time.

FROM ubuntu:latest  
ARG VERSION=1.0  
RUN echo "Building version $VERSION"

You can pass a different value for VERSION at build time:

docker build --build-arg VERSION=2.0 -t argexample:1.0 .

1. EXPOSE The EXPOSE instruction informs Docker that the container will listen on the specified network ports at runtime.

FROM ubuntu:latest  
EXPOSE 8080

This example exposes port 8080.

Command to Build and Run:

docker build -t exposeexample:1.0 .

Run the container

docker run -it -p 8080:8080 exposeexample:1.0

1. CMD The CMD instruction provides the default command to run when a container is started.

FROM ubuntu:latest  
CMD ["echo", "Hello, World!"]

This will print "Hello, World!" when the container is run.

Command to Build and Run:

docker build -t cmdexample:1.0 .

Run the container

docker run cmdexample:1.0

1. ENTRYPOINT The ENTRYPOINT instruction configures a container to run as an executable.

FROM ubuntu:latest  
ENTRYPOINT ["echo"]  
CMD ["Hello from ENTRYPOINT"]

This makes the container always run echo. The default argument ("Hello from ENTRYPOINT") can be overridden.

Command to Build and Run:

```bash

docker build -t entrypointexample:1.0 .

Run the container

docker run entrypointexample:1.0

Override CMD argument

docker run entrypointexample:1.0 "Overridden!"

1. WORKDIR The WORKDIR instruction sets the working directory for any subsequent RUN, CMD, ENTRYPOINT, COPY, and ADD instructions.

FROM ubuntu:latest  
WORKDIR /app  
RUN touch myfile.txt

This sets /app as the working directory and creates myfile.txt in that directory.

Command to Build and Run:

docker build -t workdirexample:1.0 .

**Run the container**

docker run -it workdirexample:1.0 .dockerignore The .dockerignore file helps reduce the size of the Docker context by excluding files or directories that shouldn’t be copied to the image.

Example .dockerignore file:

node\_modules  
.git  
\*.log

This prevents the node\_modules directory, .git folder, and any .log files from being included in the Docker build context.

Building Docker Images with Tags and Versions To build a Docker image with a specific tag and version, use the -t flag:

docker build -t <image\_name>:<version> .

Example:

docker build -t myapp:1.0 .

Running Docker Containers To run a Docker container from an image, use the docker run command:

docker run -it <image\_name>:<version>

Example:

docker run -it myapp:1.0