

# Docker container-Port mapping

---

## Port Mapping in Docker

In Docker, port mapping allows you to expose a port from a running container to the host machine's network stack. This enables communication between the container and external services or applications.

## Why Port Mapping is necessary?

When you run a Docker container, it has its own isolated network namespace. By default, this means that any incoming requests on the container's ports are not visible to the outside world (i.e., the host machine). Port mapping solves this problem by creating a temporary binding between the host machine's port and the container's port.

## Example:

Let's consider an example where you have a web server running in a Docker container. The container listens on port 80 (HTTP) internally, but we want to access it from our host machine at port 8080.

## docker run command

```
docker run -p 8080:80 mywebserver:latest
```

Here's what this command does:

- `-p 8080:80` maps the host machine's port 8080 (on the left) to the container's port 80 (on the right). This is called a "port mapping" or "expose port".
- `mywebserver:latest` specifies the Docker image name and tag.

## How it works

When you run this command, Docker:

1. Creates a new container from the specified image (`mywebserver:latest`).
2. Maps port 8080 on the host machine to port 80 in the container.
3. Starts the container and exposes it to the host machine's network stack.

Now, when you access `http://localhost:8080` on your host machine, Docker will forward requests to the container running on port 80, allowing you to interact with the web server inside the container.

## Port mapping syntax

The basic syntax for port mapping is:

```
docker run -p <host_port>:<container_port> <image_name>
```

Here:

- `<host_port>` is the host machine's port.
- `<container_port>` is the container's port (inside the container).
- `<image_name>` is the Docker image name and tag.

You can also specify multiple port mappings using commas, like so:

```
docker run -p 8080:80 -p 443:443 mywebserver:latest
```

This maps both port 8080 to port 80 and port 443 to port 443 inside the container.

---

*Curated by Brajesh Kumar*