# Docker Dökümantasyonu

## 1. Docker’ın Kurulumu

Docker, çeşitli işletim sistemlerinde farklı kurulum yöntemleri ile yüklenebilir. Örneğin, Windows için Docker Desktop kullanılırken, Linux işletim sistemlerinde genellikle komut satırı üzerinden kurulum yapılır. Bu bölümde, popüler işletim sistemleri için Docker kurulum adımları ele alınacaktır.

## 2. Docker Kullanımı

Docker, uygulamaları izole edilmiş container'lar içerisinde çalıştırmanıza olanak tanır. Bu bölümde, Docker container'larının nasıl oluşturulacağı, çalıştırılacağı ve yönetileceği üzerine temel bilgiler verilecektir. Ayrıca, Docker imajlarının nasıl oluşturulacağı ve Docker Hub üzerinden nasıl paylaşılacağı da anlatılacaktır.

## 3. Containerler Arası Haberleşme

Docker container'ları arasında veri akışı ve haberleşme, Docker'ın sunduğu ağ özellikleri ile mümkündür. Bu bölümde, farklı Docker container'larının birbiriyle nasıl haberleşebileceği ve ağ ayarlarının nasıl yapılandırılacağı üzerine detaylı bilgiler sunulacaktır.

Windows üzerinde Docker Desktop, Docker'ın resmi web sitesinden indirilebilir ve kurulum sihirbazı talimatları takip edilerek yüklenebilir. Linux kullanıcıları için Docker, çoğu dağıtımın paket yöneticisi üzerinden (örneğin Ubuntu'da 'apt-get install docker.io' komutu ile) kurulabilir. macOS kullanıcıları için de Docker Desktop, macOS için özel olarak tasarlanmıştır ve resmi web sitesinden indirilebilir.

Docker container'larını yönetmek için kullanılan temel komutlar şunlardır: 'docker run' bir container başlatmak, 'docker stop' bir container'ı durdurmak, 'docker rm' bir container'ı silmek için kullanılır. Docker imajları, uygulamaların çalıştırılabilir bir versiyonunu temsil eder ve 'docker build' komutu ile oluşturulur. Oluşturulan imajlar 'docker push' komutu ile Docker Hub'a yüklenebilir. Örnek komutlar:   
'docker build -t myimage .' (bir Dockerfile kullanarak 'myimage' adında bir imaj oluşturur),   
'docker run -d --name mycontainer myimage' (myimage imajını kullanarak arkaplanda mycontainer adında bir container başlatır).

Docker container'ları, Docker'ın sağladığı ağ ayarları ile birbirleriyle haberleşebilir. 'docker network create' komutu ile özel ağlar oluşturulabilir ve container'lar bu ağlar üzerinden birbirlerini görebilirler. Örneğin, 'docker network create mynetwork' komutu 'mynetwork' adında bir ağ oluşturur. Container'ları bu ağa bağlamak için, 'docker run' komutuna '--network=mynetwork' argümanı eklenir.

## 4. Docker Network Drivers

Docker, farklı türde ağ sürücüleri (network drivers) sunar ve bu sayede çeşitli ağ gereksinimlerini karşılar. En yaygın kullanılan ağ sürücüleri şunlardır: 'bridge', 'host', 'overlay', 'macvlan'. 'Bridge' driver, Docker'ın varsayılan ağ sürücüsüdür ve container'ların aynı host üzerinde izole edilmiş bir şekilde haberleşmelerini sağlar. 'Host' driver, container'ın ağını doğrudan host'un ağına bağlar, böylece ağ izolasyonu olmaz. 'Overlay' driver, farklı Docker host'ları üzerinde çalışan container'ların birbirleriyle haberleşebilmeleri için kullanılır ve genellikle swarm modunda çalışan Docker ortamlarında tercih edilir. 'Macvlan' driver ise her container'a benzersiz bir MAC adresi atayarak fiziksel ağa doğrudan bağlanmasını sağlar, bu da özellikle bazı geleneksel ağ yapılarında kullanışlıdır.