

1) Senkron ve Asenkron iletişim nedir örneklerle açıklayın?

Senkron programlama ile geliştirilen bir yazılımda kodun veya sürecin ilerlemesi için birbirini takip eden işlemlerin bitmesi beklenir. Örneğin Quicksort gibi bir sıralama algoritmasında elimizdeki sayıların önce bir yarısında sonra da diğer yarısında sıralama yaparız. Burada aynı anda sayıların iki yarısını da sıralamayız. Gerçek hayattan bir örnek vermek gerekirse de sıralar ve kuyruklar örnek verilebilir. Önümüzdeki herkesin işlemi bitmeden biz işlem yapamayız.

Asenkron programlama ile geliştirilen kodda ise kod parçacıkları dağıtık yazılır ve birbirlerini beklemeden sadece bir çağrı ile çalışırlar. Bilgisayarımıza aynı anda iki dosya birden indirebilmemiz buna bir örnek olabilir. Gerçek hayatta ise bir lokantada kimse bir başkasının yemeğinin bitmesini beklemeden sipariş verebilir.

2) RabbitMQ ve Kafka arasındaki farkları araştırın?

Kafka, açık kaynaklı platformdur ve ham veri akışını kolaylaştırır. Java ve Scala ile yazılmış olan Kafka, verileri yeniden sunma odaklı bir publisher/subscriber mesaj kuyruğudur. Kafka mesajları belleğine ekler ve orada tutar, subscriber okuyana veya saklama sınırına ulaşana kadar da orada tutar.

Kafka, kullanıcılara belirli limitler dahilinde mesaj paketleri talep etmelerine izin veren bir "pull based" kullanım ortamı sağlar. Kullanıcılar, daha yüksek veri akışı ve etkili mesaj Kafka'yı kullanırlar

RabbitMQ ise daha karmaşık yönlendirme projelerinde verimli mesaj iletimini kolaylaştıran açık kaynaklı bir dağıtık mesaj kuyruğudur. "Dağıtık" olarak adlandırılmasının sebebi, RabbitMQ'nun genellikle node yapısı ile çalışması ve kuyrukların nodelar arasında dağıtıldığı, yüksek erişilebilirlik ve hata toleransı için, bir yapıda çalışmasıdır.

RabbitMQ, bir push modeli kullanır ve tüketiciler için ayarlanmış ön yükleme limitiyle kullanıcıları korur. Bu yapı, düşük gecikmeli mesaj iletişimi için kurulmuştur. Ayrıca, RabbitMQ queue mantığı ile iyi çalışır. RabbitMQ, bir postaneye benzetilebilir, yani posta alır, depolar ve teslim eder, Kafka ise ikili veri mesajlarını kabul eder, depolar ve iletir.

RabbitMQ, AMQP 1.0, HTTP, STOMP ve MQTT gibi ek protokolleri destekler. Kafka ise video yayınları gibi projelerde daha yaygın kullanılır.

3) Docker ve Virtual Machine nedir?

Virtual machine tıpkı gerçek bir bilgisayar gibi giriş/çıkışları olan, ram ve işlemcisi, sanal olarak verilmiş olsa da, olan kendi içinde tamamen bir işletim sistemi çalıştırabilen sanal bir bilgisayardır. Gerçekten de bilgisayarımızın içinde bir bilgisayar daha varmış gibi düşünülebilir.

Docker ise VM'in aksine gerçek bir bilgisayarı taklit etmez. Konuk olduğu işletim sisteminin kernelini kullanarak sadece kendisine atanmış özellikleri gösterir. Örneğin bir java projesi için docker kullanacaksak, ilgili Docker imajında sadece java projemizi çalıştırmak için gerekli özellikleri ve programları tanımlayabiliriz. Docker'ın avantajı ise, projeyi başarılı bir şekilde üzerinde çalıştırdıktan sonra Docker konteyneri oluşturup projemizi Docker kurulu olan her yerde çalıştırabiliriz. Ama benim bilgisayarımda çalışıyordu gibi problemlerden kurtuluruz. Konteynerler tıpkı gerçek konteynerler gibi sadece bizim için gerekli olan programları ve kodları içinde tutan dosyalardır. Onları Docker engine ile çalıştırmak mümkündür.

4) Docker ile RabbitMQ ve PostgreSQL ve ya MySQL kurulumu yapın?

Proje dizininde yer alan docker-compose.yml dosyası ile oluşturduğum konteynerin ekran görüntüsü.

The screenshot displays the Docker Desktop application window. On the left is a sidebar with navigation options: Containers, Images, Volumes, Builds (marked as NEW), Dev Environments (marked as BETA), Docker Scout, Extensions, and Add Extensions. The main panel is titled 'Containers' and shows a summary of container usage: CPU usage at 0.33% (8 CPUs available) and memory usage at 160.45MB (3.74GB). Below this is a search bar and a toggle for 'Only show running containers'. A table lists three running containers:

	Name	Image	Status	CPU (%)	Port(s)	Last started	Actions
<input type="checkbox"/>	week-2-egekkege		Running (2/2)	0.33%		2 minutes ago	[Stop] [Refresh] [Delete]
<input type="checkbox"/>	postgres	postgres	Running	0.05%	5432:5432	2 minutes ago	[Stop] [Refresh] [Delete]
<input type="checkbox"/>	rabbitmq	rabbitmq	Running	0.28%	15672:15672	2 minutes ago	[Stop] [Refresh] [Delete]

Below the table, it says 'Showing 3 items'. There is a 'Walkthroughs' section with two cards: 'Multi-container applications' (8 mins) and 'Containerize your application' (3 mins). The bottom status bar shows 'Engine running', system resources (RAM 1.88 GB, CPU 0.75%, Disk 54.49 GB avail. of 62.67 GB), and the user is 'Not signed in'.