



KUBERNETES DESIGN PATTERNS EĞİTİMİ

3 GÜN



Digital Vizyon
Akademi

www.digitalvizyon.net



İçindekiler

Eğitim Hakkında.....	3
Neler Öğreneceksiniz?	3
Ön Koşullar	4
Kimler Katılmalı.....	4
Outline	5
Introduction.....	5
Foundational Patterns:	5
Behavioral Patterns:.....	5
Structural Patterns:.....	5
Configuration Patterns:.....	5
Summary and Conclusion	5

Eđitim Hakkında

“Kubernetes Design Patterns Eđitimi”, Kubernetes tasarım kalıpları üzerinde kapsamlı bir eđitim sunar. Bu eđitim, Kubernetes tabanlı uygulamaların tasarımı ve geliřtirilmesinde kullanılan tasarım kalıplarının temel konularını ele alır.

Eđitim, Kubernetes tasarım prensiplerini, veri yönetimi, hata toleransı, ölçeklendirme ve diđer kilit kavramları öğretir. Katılımcılar, gerçek hayattan örnekler ve projeler aracılıđıyla Kubernetes tasarım kalıpları konusundaki bilgilerini artırır.

Eđitim ayrıca, uygulamaların Kubernetes tabanlı olarak tasarlanması ve geliřtirilmesi sürecinde hangi tasarım kalıplarının ve teknolojilerin nasıl kullanılabileceđini de öğretir. Katılımcılar, büyük veri işleme, real-time analiz, hata toleransı ve diđer özelliklerin nasıl yönetileceđini öğrenirler. Ayrıca, Kubernetes tasarım kalıplarıyla oluşturulan uygulamaların nasıl tasarlanıp yönetileceđini de anlarlar.

“Kubernetes Design Patterns Eđitimi”, Kubernetes tabanlı uygulama tasarımı ve geliřtirme sürecinin tümünü kapsar. Katılımcılar, modern uygulamalar tasarlamaya ve geliřtirmeye başlamadan önce ihtiyaç duyacakları temel becerileri kazanırlar. Eđitim, örnekler, pratik uygulamalar ve projeler yoluyla öğrenmeyi sağlar ve katılımcıların Kubernetes tasarım kalıplarıyla uygulama tasarlamaya ve geliřtirmeye başlamalarına yardımcı olur.

Eđitim programı, Kubernetes tasarım kalıplarının temelleriyle başlar. Katılımcılar, veri yönetimi, ölçeklendirme ve hata toleransı gibi temel kavramları öğrenirler. Ayrıca, Kubernetes tabanlı modern uygulamaların tasarlanması ve yönetilmesi konusunda bu kalıpların nasıl bir rol oynadıđına dair bilgi sahibi olurlar. Bu bilgiler, katılımcıların uygulama tasarımı ve geliřtirme sürecinde ihtiyaç duyacakları temel yapı taşlarını oluşturur.

Eđitimde, Kubernetes tasarım kalıpları ve bununla iliřkili temel özellikler ve bileşenler üzerinde duruyoruz. Bu, katılımcılara real-time veri işleme, büyük veri akışları ve hata toleransı gibi yetenekleri kazandırır. Konu akışları, veri yönetimi ve ölçeklendirilebilir sistemlerin yönetimi gibi konular işlenir.

Son olarak, bir uygulamanın nasıl Kubernetes tasarım kalıpları tabanlı geliřtirileceđi hakkında bilgi veriyoruz. Bu süreç, uygulamanın son testlerini yapmayı, veri yönetimini, ve en sonunda uygulamanın Kubernetes tasarım kalıpları tabanlı geliřtirilmesini içerir. Bu bilgiler, katılımcıların uygulamalarını başarılı bir şekilde Kubernetes tasarım kalıplarıyla tasarlamalarına ve geliřtirmelerine yardımcı olur.

Neler Öğreneceksiniz?

Kubernetes Design Patterns Eđitiminde, řunları öğreneceksiniz:

- Kubernetes çalışma prensipleri ve API objeleri: Kubernetes’in nasıl çalıştıđı, API objeleri hakkında bilgi ve bu objeleri nasıl kullanacađınızı öğreneceksiniz
- Deployment stratejileri: Uygulama ve hizmetlerin nasıl dağıtılaçađı, deployment stratejileri hakkında bilgi ve bu stratejilerin nasıl uygulanacađını öğreneceksiniz.



- Hizmet tipleri: Hizmet tipleri hakkında bilgi, uygulama ve hizmetlerin nasıl erişebileceğini ve nasıl yapılandırılacağını öğreneceksiniz
- Stok yapıları: Kubernetes stok yapıları hakkında bilgi ve bu yapıların nasıl kullanılacağını öğreneceksiniz
- Kaynak yönetimi: Kubernetes ortamında kaynak yönetimi hakkında bilgi ve kaynakların nasıl yönetileceğini öğreneceksiniz
- Güvenlik: Kubernetes ortamında güvenliğin nasıl sağlanacağı hakkında bilgi öğreneceksiniz.

Bu eğitim sırasında, teorik bilgi ve uygulamalı çalışmalarla, Kubernetes Design Patterns'lerin nasıl uygulanacağını ve bu kalıpları nasıl kullanarak uygulamalarınızı ve hizmetlerinizi daha verimli, skalenebilir ve kararlı hale getirebileceğinizi öğrenebilirsiniz.

Ön Koşullar

Kubernetes Design Patterns Eğitimi için şu ön koşullar bulunabilir:

- Programlama Dilleri: Kubernetes Design Patterns Eğitimi için bir programlama dili bilmek gerekli değildir, ancak genel olarak programlama dillerinden birine (örneğin, Python, Java, Go vb.) hakim olmanız faydalı olabilir.
- Cloud Computing: Bulut bilişim ve bulut dağıtım konuları hakkında temel bilgi sahibi olmanız faydalı olabilir.
- Docker: Docker ve container teknolojileri hakkında temel bilgi sahibi olmanız faydalı olabilir.
- Kubernetes: Kubernetes hakkında temel bilgi sahibi olmanız Kubernetes Design Patterns Eğitimi için gereklidir.

Bu ön koşullar, eğitimi daha verimli ve etkili bir şekilde almanıza yardımcı olabilir, ancak eğitimi almak için bu koşulların tamamına sahip olmanız zorunlu değildir. Eğitim verilen kurumların ve eğitmenlerin ön koşullar hakkında ayrıntılı bilgi vermelerine başvurabilirsiniz.

Kimler Katılmalı

Kubernetes Design Patterns Eğitimi, aşağıdaki kişiler için uygun olabilir:

- DevOps ve Sistem Yöneticileri: Kubernetes Design Patterns'leri kullanarak, uygulamalarınızı ve hizmetlerinizi daha verimli, skalenebilir ve kararlı hale getirebilmeniz için gereken bilgi ve becerileri edinebilirsiniz.
- Geliştiriciler: Kubernetes Design Patterns'leri kullanarak, uygulamalarınızı ve hizmetlerinizi daha verimli, skalenebilir ve kararlı hale getirebilmeniz için gereken bilgi ve becerileri edinebilirsiniz.
- Büyük Ölçekli Uygulama ve Hizmetler Geliştiricileri: Kubernetes Design Patterns'leri kullanarak, büyük ölçekli uygulama ve hizmetlerinizi daha verimli, skalenebilir ve kararlı hale getirebilmeniz için gereken bilgi ve becerileri edinebilirsiniz.
- Bulut Bilişim ve Bulut Dağıtım Uzmanları: Kubernetes Design Patterns'leri kullanarak, bulut bilişim ve bulut dağıtım alanında daha verimli ve kararlı çalışma becerileri edinebilirsiniz.



Bu eğitim, Kubernetes ve bulut bilişim ile ilgilenen herkes katılabilir. Ancak, eğitimin verimli ve etkili bir şekilde alınması için temel bilgi ve becerilere sahip olmanız faydalı olabilir.

Outline

Introduction

- Recap about Linux containers technology
- Recap about Kubernetes basic concepts
- Microservice based applications
- The 12 Factors App Principles

Foundational Patterns:

Distributed Primitives, Predictable Demands, Dynamic Placement, Declarative Deployment, Observable Interior, and Life Cycle Conformance.

Behavioral Patterns:

Batch Jobs, Scheduled Jobs, Daemon Services, Singleton Services and Self Awareness.

Structural Patterns:

Sidecar, Initializer, Ambassador, and Adapter.

Configuration Patterns:

Env Var Configuration, Configuration Resources, Configuration Templates, and Immutable Configuration.

Summary and Conclusion