

MÜLAKAT GÖREVLERİ VE SORULARI

Görev (1)

Facebook, Twitter, vb. sosyal ağ platformlarına benzer, yüksek hit alması hedeflenen bir projeyi geliştirmek amacıyla bir proje mimarisi hazırlamanız gerekmektedir. Bu projede, kullanıcılar içerik paylaşmalı, etkileşimde bulunmalı ve platformu büyütmeli. Aşağıda, bu projenin sunucu mimarisi ve performans yönetimi için kullanılacak bileşenler hakkında ayrıntılar yer almaktadır.

Aşağıdaki görevleri yerine getirerek projenin sunucu mimarisi ve performans yönetimi ile ilgili bir tasarım ve açıklama sunmanız beklenmektedir:

Sunucu Mimarisi Tasarımı: Projenin temel sunucu mimarisi hakkında bir açıklama sunun. Sunucular arasındaki rol ve sorumlulukları açıklayan bir diyagram oluşturun.

Veritabanı Yönetimi: Veri tabanı seçiminizi ve bu seçimin nedenlerini açıklayın. Veritabanı sunucusu ve veri tabanı kümesi (cluster) kurulumunu açıklayın.

Ölçeklenebilirlik Stratejileri: Projenin ölçeklenebilirliği nasıl yönetilecek? Yük arttıkça nasıl ölçeklendirilecek? Yük dengeleme stratejilerini açıklayın.

Önbellekleme ve Hızlandırma: Hangi önbellek (cache) mekanizmalarını kullanmayı düşünüyorsunuz? Sunucu ve veri tabanı erişimini hızlandırmak için hangi yöntemleri düşünüyorsunuz?

Güvenlik Stratejileri: Kullanıcı kimlik doğrulama ve yetkilendirme nasıl yönetilecek? Proje güvenliği için hangi önlemleri almayı düşünüyorsunuz?

Performans İzleme ve Analiz: Sunucu performansını izlemek ve sorunları tanımlamak için kullanılacak araçları ve yöntemleri açıklayın.

Görev (2)

Fibonacci Dizisi ve Asal Sayılar ile ilgili bir görev aşağıda verilmiştir. Bu görev, algoritmaları anlama ve uygulama yeteneğini değerlendirmeyi amaçlamaktadır.

Görev Açıklaması:

İlk olarak, 500.000'den küçük en büyük asal sayıyı ve 500.000'den büyük en küçük asal sayıyı belirlemek ve bulunan iki asal sayı arasındaki farkı hesaplamak için bir Java programı yazınız. Bu program, asal sayıları belirlerken etkili bir algoritma kullanmalıdır.

Notlar:

- İşlem süresini minimize etmek için verimli bir algoritma kullanmanız beklenmektedir.
- İşlem sonucunu kullanıcıya açık ve anlaşılır bir şekilde sunmanız gerekmektedir.

Görev (3)

Basit Bir Web Uygulaması Geliştirme: Bu görevde, bir web uygulaması geliştirmeniz beklenmektedir. Bu uygulama, kullanıcıların çeşitli bilgileri girip bu verilerin bir veritabanına kaydedilmesini ve aynı sayfa üzerinde listelenmesini içermelidir. Görev, kullanıcı arayüzü (UI) tasarımı, web servis entegrasyonu ve veritabanı işlemleri gibi farklı beceri alanlarını içerir.

Görev Açıklaması:

UI Tasarımı ve Form Oluşturma: Bootstrap kullanarak basit bir web sayfası tasarlayın. Sayfa, kullanıcıların verileri girebileceği bir form içermelidir.

jQuery ve Ajax Kullanarak Web Servis İsteği: Kullanıcının formdaki verileri girdikten sonra, jQuery ve Ajax kullanarak bu verileri bir web servisine gönderin. Web servisi, verileri kabul edip bir RESTful API aracılığıyla sunmalıdır.

Veri Tabanına Kaydetme: Web servis, aldığı verileri bir veritabanına kaydetmelidir. Veritabanı işlemleri için gerekli SQL sorgularını oluşturun.

Verilerin Listelenmesi: Web servis, verileri alabilir ve aynı sayfa üzerinde bir tablo şeklinde listelenebilir. jQuery ve Ajax kullanarak verileri alın ve tablo içinde görüntüleyin.

Hata İşleme ve Geri Bildirim: Kullanıcı dostu bir şekilde hataları işleyin ve kullanıcıya geri bildirim sağlayın.

Notlar:

1. Proje için Core Java kullanmalısınız.
2. Kullanıcı deneyimini düşünerek kullanıcı dostu bir arayüz tasarımı yapmalısınız.
3. Bootstrap, jQuery ve Ajax gibi önemli teknolojileri etkili bir şekilde kullanmalısınız.
4. Proje, veritabanıyla etkileşimli bir web uygulaması olmalıdır.

Görev (4)

Veri Normalizasyonu ve Algoritma Tasarımı: Bu görevde, rasgele sayılar içeren bir dizinin verilerini normalleştiren ve her yeni eleman eklediğinde dizinin minimum ve maksimum değerlerini güncelleyerek normalizasyon hesaplayan bir algoritma tasarlamanız ve kodlamayı sunmanız beklenmektedir.

Görev Açıklaması:

1 ile 10.000 arası sayılardan rastgele sayı üreten ve farklı elemanlardan oluşan 500 elemanlı bir dizi oluşturun. Diziye her yeni eleman eklediğinizde, minimum ve maksimum değerleri güncelleyen ve verileri aşağıda verilen normalizasyon formülüne göre normalleştiren bir algoritma tasarlayın ve uygulayın.

$$N = (x_i - x_{\min}) / (x_{\max} - x_{\min})$$

Örnek Çıktı: sadece 3 veri için

Veri seti (tek elemanlı) : {110}

Normalizasyon : [0]
Veri Seti (iki elemanlı) : {1382, 110}
Normalizasyon : [1.0, 0.0]
Veri Seti (üç elemanlı) : {1382, 7357, 110}
Normalizasyon : [0.1755209052021526, 1.0, 0.0]

Notlar:

Diziye her eleman eklediğinizde minimum ve maksimum değerlerini güncellemelisiniz.
Normalizasyon hesaplama adımlarını anlaşılır bir şekilde ekrana yazdırmalısınız.
Dizinin başlangıçta nasıl oluşturulacağını ve nasıl güncelleneceğini düşünmelisiniz.

Görev (5)

Aşağıda verilen veritabanı tabloları kullanılarak, her departmandaki çalışan sayısını hesaplayan ve sonucu departman adıyla birlikte listeleyen bir SQL sorgusu yazmanız gerekmektedir.

Veritabanı Tabloları:

"Emp" tablosu (Alanlar: Id, Name, DeptId) – 10 kayıt içerir.

"Dept" tablosu (Alanlar: Id, Name) – 5 kayıt içerir.

Görev Açıklaması:

1. SQL sorgusu yazarak, her departmandaki çalışan sayısını hesaplayın.
 2. Sonucu departman adıyla birlikte listeleyin.
 3. Sonuç olarak kaç satır listeleneceğini açıklayın.
-

Soru (1)

Uygulama güvenliği konusunda Java projelerinizde aldığınız önlemleri açıklayabilir misiniz?
Cross-Site Scripting (XSS) veya SQL enjeksiyonu gibi güvenlik açıkları hakkında bilgi verebilir misiniz?

Soru (2)

Java uygulamalarının performansını nasıl optimize edersiniz? JVM ayarları, veritabanı endekslemesi, kod incelemesi gibi optimizasyon stratejilerini açıklayın.

Soru (3)

Java projelerinizde hangi tasarım şablonlarını sıkça kullanıyorsunuz? Örnek vererek, belirli bir senaryoda hangi tasarım şablonunu seçtiğinizi açıklayın.

Soru (4)

RESTful veya SOAP tabanlı web servislerini geliştirme deneyiminizi anlatın. Veri formatları (JSON vs. XML) ve güvenlik önlemleri hakkında bilgi verin.

Soru (5)

Aşağıdaki değerlere göre `SELECT count(*) AS cust_not_123_total FROM orders WHERE customer_id <> '123'` sorgu sonucu ne olacaktır?

```
SELECT count(*) AS total FROM orders;
```

```
+-----+
```

```
| total |
```

```
+-----+
```

```
| 100 |
```

```
+-----+
```

```
SELECT count(*) AS cust_123_total FROM orders WHERE customer_id = '123';
```

```
+-----+
```

```
| cust_123_total |
```

```
+-----+
```

```
| 15 |
```

```
+-----+
```

BONUS GÖREV

Mobildev XML SMS API metotlarından SMSMultisender metodunu kendi geliştirmekte olduğunuz bir uygulama içerisine kullanmak için bir kütüphane oluşturunuz.

Not: user / pass / alfanumerik gibi bilgiler, bir ayar dosyasından alınacak şekilde ayarlanmalıdır.

<https://postman.mobildev.com/?version=latest#e7eeba09-7b99-466a-8fb2-95972d5c2386>