

MongoDB CRUD



Skills
Network

Gerekli tahmini süre: 30 dakika

Hedefler

Bu laboratuvarı tamamladıktan sonra şunları yapabileceksiniz:

- insert yöntemiyle MongoDB’de belgeler oluşturmak
- Belgeleri listeleyerek, sayarak ve bir sorguya eşleştirerek okumak
- Belirli kriterlere dayanarak MongoDB’de belgeleri güncellemek ve silmek

Beceri Ağı Bulut IDE Hakkında

Beceri Ağı Bulut IDE (Theia ve Docker tabanlı) kurs ve proje ile ilgili laboratuvarlar için uygulamalı bir ortam sağlar. Theia, masaüstünde veya bulutta çalıştırılabilen açık kaynaklı bir IDE’dir (Entegre Geliştirme Ortamı). Bu laboratuvarı tamamlamak için, Beceri Ağı tarafından sağlanan Theia ve MongoDB tabanlı Bulut IDE’yi kullanacağız.

Bu laboratuvar ortamı hakkında önemli bir not

Lütfen bu laboratuvar ortamı için oturumların kalıcı olmadığını unutmayın. Bu laboratuvara her bağlandığınızda, sizin için yeni bir ortam oluşturulur. Önceki oturumda kaydettiğiniz veriler kaybolacaktır. Verilerinizi kaybetmemek için bu laboratuvarları tek bir oturumda tamamlamayı planlayın.

Alıştırma 1 - Ortamı Hazırlamak

Aşağıdaki butona tıklayarak MongoDB veritabanı sayfasını açın:

Open MongoDB Page in IDE

O sayfada, bir MongoDB veritabanı oluşturmak için Create butonuna tıklayın.

Oluşturma işlemi tamamlandıktan sonra, veritabanına mongosh CLI ile bağlanmak için MongoDB CLI butonuna (önceki adımdaki Create butonunun altında) tıklayın.

1. Problem:

training veritabanını seçin.

▼ İpucu için buraya tıklayın

[MongoDb Başlarken](#) laboratuvarına bakın.

▼ Çözüm için buraya tıklayın

```
use training
```

2. Problem:

languages adlı bir koleksiyon oluşturun.

▼ İpucu için buraya tıklayın

[MongoDb Başlarken](#) laboratuvarına bakın.

▼ Çözüm için buraya tıklayın

```
db.createCollection("languages")
```

Alıştırma 2 - Belgeleri Ekle

Beş belgeyi **languages** koleksiyonuna ekleyelim.

Mongo istemcisinde aşağıdaki komutları çalıştırın.

```
db.languages.insert({"name":"java","type":"object oriented"})
db.languages.insert({"name":"python","type":"general purpose"})
db.languages.insert({"name":"scala","type":"functional"})
db.languages.insert({"name":"c","type":"procedural"})
db.languages.insert({"name":"c++","type":"object oriented"})
```

Egzersiz 3 - Belgeleri Okuma

Belgeleri sorgulamanın farklı yollarını deneyelim.

Belgelerin sayısını bulun.

```
db.languages.countDocuments()
```

```
\\ List the first document in the collection.
\\
db.languages.findOne()
```

Koleksiyondaki tüm belgeleri listele.

```
db.languages.find()
```

```
\\ List first 3 documents in the collection.
```

```
db.languages.find().limit(3)
```

```
\\ Query for "python" language.
```

```
db.languages.find({"name":"python"})
```

Query for "object oriented" languages.

```
db.languages.find({"type":"nesne yönelimli"})
```

List only specific fields.

Using a projection document you can specify what fields we wish to see or skip in the output.

This command lists all the documents with only name field in the output.

```
db.languages.find({}, {"name":1})
```

Bu komut, çıktıda isim alanı olmadan tüm belgeleri listeler.

```
db.languages.find({}, {"name":0})
```

This command lists all the "object oriented" languages with only "name" field in the output.

```
db.languages.find({"type":"object oriented"}, {"name":1})
```

```
::page{title="Exercise 4 - Update documents"}
```

Update documents based on a criteria.

Add a field to all the documents.

The `\updateMany\` command is used to update documents in a mongodb collection, and it has the following generic syntax.

```
db.collection.updateMany({bulunacak belgeler}, {$set:{ayarlancak alanlar}})
```

Here we are adding a field `**description**` with value `**programming language**` to all the documents.

```
db.languages.updateMany({},{$set:{“description”:”programlama dili”}})
```

```
Set the creator for python language.
```

```
db.languages.updateMany({"name":"python"},{$set:{"creator":"Guido van Rossum"}})
```

Tüm **nesne yönelimli** diller için **compiled** adında bir alanı **true** değeri ile ayarlayın.

```
db.languages.updateMany({"type":"object oriented"},{$set:{"compiled":true}})
```

```
::page{title="Exercise 5 - Delete documents"}
Delete documents based on a criteria.
Delete the scala language document.
```

```
db.languages.remove({"name":"scala"})
```

```
Delete the object oriented languages.
```

```
db.languages.remove({"type":"nesne yönelimli"})
```

```
Delete all the documents in a collection.
```

```
db.languages.remove({})
```

Uygulama Alıştırmaları

Aşağıdaki kodu mongo konsolunda çalıştırın. Bu, sonraki adımlar için örnek veri olarak kullanılacak 5 belge ekleyecektir.

```
use training
db.languages.insert({"name":"java","type":"object oriented"})
db.languages.insert({"name":"python","type":"general purpose"})
db.languages.insert({"name":"scala","type":"functional"})
db.languages.insert({"name":"c","type":"procedural"})
db.languages.insert({"name":"c++","type":"object oriented"})
```

1. Problem:

'Haskell' programlama dili için 'fonksiyonel' türünde bir giriş ekleyin.

▼ İpucu için buraya tıklayın

db.collection.insert() komutunu kullanın.

▼ Çözüm için buraya tıklayın

Mongo istemcisinde aşağıdaki komutları çalıştırın.

```
db.languages.insert({"name":"Haskell","type":"functional"})
```

2. Problem:

Tüm fonksiyonel dilleri sorgulayın.

▼ İpucu için buraya tıklayın

uygun seçeneklerle db.collection.insert() komutunu kullanın

▼ Çözüm için buraya tıklayın

Mongo istemcisinde aşağıdaki komutları çalıştırın.

```
db.languages.find({"type":"functional"})
```

3. Problem:

'Bjarne Stroustrup'u c++ için yaratıcı olarak ekleyin.

▼ İpucu için buraya tıklayın

uygun seçeneklerle db.collection.insert() komutunu kullanın

▼ Çözüm için buraya tıklayın

Mongo istemcisinde aşağıdaki komutları çalıştırın.

```
db.languages.updateMany({"name":"c++"},{$set:{"creator":"Bjarne Stroustrup"}})
```

4. Problem:

Tüm fonksiyonel programlama dillerini silin.

▼ İpucu için buraya tıklayın

uygun seçeneklerle db.collection.remove() komutunu kullanın

▼ Çözüm için buraya tıklayın

Mongo istemcisinde aşağıdaki komutları çalıştırın.

```
db.languages.remove({"type":"functional"})
```

5. Problem:

MongoDB sunucusundan bağlantıyı kes.

▼ İpucu için buraya tıklayın

[MongoDB Başlarken](#) laboratuvarına bakın.

▼ Çözüm için buraya tıklayın

Terminalde aşağıdaki komutu çalıştırın.

```
exit
```

Yazarlar

Ramesh Sannareddy

Diğer Katkıda Bulunanlar

Rav Ahuja

© IBM Corporation. Tüm hakları saklıdır.