

MongoDB Übungen

Warming up – Runde 1 mit der Mongo Console

Aufgabe 1: Lege eine Datenbank codingssession an und füge eine Collection persons hinzu.

Aufgabe 2a: Füge verschiedene Personen ein.

```
{ "name" : "Beat", "nationality" : "swiss", "age" : 35 }  
{ "name" : "Peter", "nationality" : "german", "age" : 29 }  
{ "name" : "Heinz", "nationality" : "german", "age" : 55 }  
{ "name" : "Yannis", "nationality" : "german", "age" : 6 }
```

Aufgabe 2b: Experimentiere mit Schemafreiheit und füge folgende Datensätze ein:

```
{ "name" : "Yannis", "nationality" : "german", "livesIn": "Hamburg", "age" : 6 }  
{ "name" : "Mike", "bornIn": "Bremen", "livesIn": "Hamburg", "hobbies" : [ "Java", "Karate" ] }  
{ "name" : "Tim", "bornIn": "Kiel", "livesIn": "Kiel", "hobbies" : [ "Java", "Musik" ] }  
{ "name" : "Tim", "address": { "city" : "Kiel", "PLZ": 24106 } }
```

Aufgabe 3: Erstelle eine Collection employees. Füge verschiedene Mitarbeiter ein, etwa:

```
{ "surname" : "Marius", "name" : "Reusch", "role" : "Junior SW Engineer", "age" : 29 }  
{ "surname" : "Michael", "name" : "Inden", "role" : "Lead Engineer & Trainer", "age" : 44 }  
{ "surname" : "Carlo", "name" : "Giorgetta", "company" : "Swisscom", "age" : 42 }  
{ "surname" : "Hermann", "name" : "Schnyder", "company" : "Swisscom", "age" : 42 }
```

Aufgabe 4: Liste alle Namen von Datenbanken auf.

Aufgabe 5: Liste alle Namen von Collections auf.

Aufgabe 6a: Ermittle alle Personen.

Aufgabe 6b: Finde alle Personen, die jünger als 30 sind.

Aufgabe 6c: Finde alle Mitarbeiter, die älter als 40 sind.

Aufgabe 6d: Finde alle Mitarbeiter, die bei Swisscom arbeiten.

Aufgabe 7: Finde alle Personen, die „Java“ als Hobby haben. Wer mag „Karate“?

Aufgabe 8a: Finde alle Personen, die in Kiel wohnen.

Aufgabe 8b: Finde alle Personen, die in Kiel mit PLZ 24106 wohnen.

Aufgabe 8c: Finde nun wirklich alle Personen, die in Kiel wohnen. Beachte die Schemafreiheit und die Varianz in den Daten, also alle auch diejenigen mit «livesIn».

Programming – Runde 2 Hands on in der IDE

Aufgabe 1: Wiederhole die Aufgaben aus dem „Warming up“ in Form kleiner Java-Programme oder einzelner Methoden, aber in der DB «coding-session-2».

Tipp: Die JSON-Verarbeitung wird einfacher, wenn man folgenden Umwandlungstrick kennt:

```
String jsonString = "{ \"surname\" : \"Hendrik\", \"name\" : \"Schöneberg\", \"role\" : 'Senior Software Engineer', 'age' : 33}";
```

```
Document doc = Document.parse(jsonString);
```

Aufgabe 2: Lege eine Datenbank crud_example und dort einige Collection employees an. Erstelle eine CRUD-Applikation für folgende Klasse Employee.

```
public class Employee {
```

```
    private String surname;  
    private String name;  
    private String company;  
    private String role;  
    private int age;
```

```
    public Employee(String surname, String name, String company, String role, int age) {  
        this.surname = surname;  
        this.name = name;  
        this.company = company;  
        this.role = role;  
        this.age = age;  
    }
```

```
    public String getSurname() {  
        return surname;  
    }
```

```
    public void setSurname(String name) {  
        this.name = surname;  
    }
```

```
    public String getName() {  
        return name;  
    }
```

```
    public void setName(String name) {  
        this.name = name;  
    }
```

```
    public String getCompany() {  
        return company;  
    }
```

```
    public void setCompany(String company) {  
        this.company = company;  
    }  
  
    public String getRole() {  
        return role;  
    }  
  
    public void setRole(String role) {  
        this.role = role;  
    }  
  
    public int getAge() {  
        return age;  
    }  
  
    public void setAge(int age) {  
        this.age = age;  
    }  
}
```