# Mongo

## JSON Beispiel

{  
 "name": "Bob",  
 "alter": 25,  
 "schuhgroesse": 42.5,  
 "hobbys": ["Fußball", "Schwimmen"],  
 "adresse": {  
 "straße": "Musterstraße 123",  
 "stadt": "Musterstadt",  
 "plz": "12345"  
 },  
 "verheiratet": false  
}

* Mongo Actions
  + CRUD
    - Create
    - Read
    - Update
    - Delete
  + IFUR
    - Insert
    - Find
    - Update
    - Remove

## Datenbank erstellen

use VIT

## Collection erstellen

db.createCollection("dozenten")

## Datensatz in eine Collection hinzufügen

### Ein Dokument

db.dozenten.insert({  
 "name": "Fiebig",  
 "vorname": "D.",  
 "titel": "Prof. Dr."  
});

### Mehrere Dokumente

insertMany(...) mit einem Array von Dokumenten

db.dozenten.insertMany([  
 {  
 "name": "Austrup",  
 "vorname": "Sascha",  
 "titel": "Prof. Dr."  
 },  
 {  
 "name": "Schmittwilken",  
 "vorname": "Jörg",  
 "titel": "Dr."  
 }  
]);

## Datensatz aktualisieren

* updateOne(...)
  + Das erste gefundene Dokument wird bearbeitet
* update(...)
  + Alle Dokumente, die gefunden werden, werden bearbeitet

Neue JSON Key-Value Paare mit $SET übergeben.

db.dozenten.updateOne(  
 { "name": "Fiebig" },  
 {   
 "$set": {  
 "vorname": "Daniele"  
 }  
 }  
);

### Modifier Operatoren

Modifier Operatoren werden beim update oder updateOne benutzt.

| Operator | Beispiel | Funktion |
| --- | --- | --- |
| $set | { "$set": { "age": 15 } } | Ändert einen Wert |
| $unset | { "$unset": { "age": 1 } } | Löscht das angegebene Feld |
| $rename | { "$rename": { "age": "alter" } } | Benennt ein Feld um |

## Dokumente Ausgeben

* findOne(...)
  + Das erste gefundene Dokument wird ausgegeben
* find(...)
  + Alle Dokumente, die gefunden werden, werden ausgegeben

// Alle Dokumente  
db.dozenten.find();  
// Alle mit bestimmten Filter  
db.dozenten.find({ "titel": "Dr." });  
// Nur ein Dokument  
db.dozenten.findOne({ "name": "Fiebig" });  
// Alle Dozenten aus dem zweiten Teil des Alphabets  
db.dozenten.find({  
 "name": {  
 "$gte": "M",  
 "$lte": "Z"  
 }  
});  
// Dokumente mit $OR  
db.dozenten.find({  
 "$OR": [  
 { "name": "Fiebig" },  
 { "name": "Austrup" }  
 ]  
});

### Query Operatoren

Query Operatoren werden im find oder findOne benutzt.

| Operator | Beispiel | Funktion |
| --- | --- | --- |
| $gt | { "age": { "$gt": 10 } } | größer als (greater than) |
| $gte | { "age": { "$gte": 10 } } | größer gleich (greator than or equal) |
| $lt | { "age": { "$lt": 10 } } | kleiner als (less than) |
| $lte | { "age": { "$lte": 10 } } | kleiner gleich (less than or equal) |
| $ne | { "age": { "$ne": 10 } } | ungleich (not equals) |

### Ausgabe Sortieren

* 1 - ASC
* -1 - DSC

db.dozenten.find(...).sort({ "name": 1 })

### Dokumente Zählen

db.dozenten.find(...).count()

### Ausgabe Begrenzen

db.dozenten.find(...).limit(2);  
db.dozenten.find(...).skip(10).limit(2);

## Dokumente löschen

* deleteOne(...)
  + Das erste gefundene Dokument wird löschen
* deleteMany(...)
  + Alle Dokumente, die gefunden werden, werden löschen

db.deleteOne({  
 "name": "Fiebig"  
});  
db.delete({  
 "titel": "Prof. Dr"  
});