**M165 Pokémon Quiz Dokumentation**

Inhaltsverzeichnis

[Unser Thema 2](#_Toc154586968)

[DB erstellen 4](#_Toc154586969)

[DB testen 4](#_Toc154586970)

[DB werte lesen in Java 5](#_Toc154586971)

## Unser Thema

Wir haben uns entschieden für das Quiz Pokémon Karten als Vorlage zu haben wo wir als Messbare Werte:

1. HP, Healhpoints
2. DMG, Strength
3. Energy Points
4. Stage, Evolution Stage

Beispielsweisse wird dann für das Quiz die Frage kommen für welche Kategorie man befragt werden will, man gibt die Antwort im Terminal ein und bekommt eine Frage mit 3 Pokémons zur Auswahl.

Wir machen mal ein Beispiel Lauf

Ein Bild, das Text, Schrift, Screenshot enthält.

Automatisch generierte Beschreibung

wir wählen die Kategorie «Strength»

Ein Bild, das Text, Schrift, Screenshot, Algebra enthält.

Automatisch generierte Beschreibung

Ein Bild, das Text, Screenshot, Schrift enthält.

Automatisch generierte BeschreibungWir bekommen jetzt diese 3 zufällige Pokémons zur Auswahl:

  

! im Spiel hat man keine Bilder zu den Pokémons !

Und wir geben unsere Antwort:

Ein Bild, das Text, Screenshot, Schrift, Zahl enthält.

Automatisch generierte Beschreibung

Nachdem wir die Antwort geben, wird es aufgelöst, ob wir richtig oder falsch sind und welches die richtige Antwort ist.

Ein Bild, das Text, Screenshot, Schrift enthält.

Automatisch generierte Beschreibung

## DB erstellen

Mit einem Config File erstellen wir die Anforderungen für unsere DB, welche eine ganz normale MongoDB ist, die schon mit werten voll ist welche es aus einem init.js File die Objekte in die Collection hinzufügt. Wir gehen im Terminal auf das Verzeichnis, wo das File gespeichert ist und geben docker-compose up ein.

*# Monog DB and Mongo-Express and initial pokemondb*version: '3.9'  
  
services:  
  
 mongodb:  
 image: mongo  
 restart: always  
 ports:  
 - 27017:27017  
 environment:  
 MONGO\_INITDB\_ROOT\_USERNAME: root  
 MONGO\_INITDB\_ROOT\_PASSWORD: 1234  
 volumes:  
 - ./init.js:/docker-entrypoint-initdb.d/mongo-init.js:ro  
  
 mongo-express:  
 image: mongo-express  
 restart: always  
 ports:  
 - 8081:8081  
 environment:  
 ME\_CONFIG\_MONGODB\_ADMINUSERNAME: root  
 ME\_CONFIG\_MONGODB\_ADMINPASSWORD: 1234  
 ME\_CONFIG\_MONGODB\_URL: mongodb://root:1234@mongodb:27017/

db = db.getSiblingDB('mongodb');  
  
db.createCollection('pokemons');  
  
db.pokemons.insertMany([  
 {  
 "name": "Pikachu",  
 "health": 80,  
 "strenght": 30,  
 "energy": 2,  
 "stage": 1  
 },  
 {  
 "name": "Karpador",  
 "health": 30,  
 "strenght": 10,  
 "energy": 1,  
 "stage": 1  
 },  
 {  
 "name": "Squirtle",  
 "health": 40,  
 "strenght": 10,  
 "energy": 1,  
 "stage": 1  
 }

### DB testen

Diese Daten der Pokémons sind jetzt auf unserer PokémonDB gespeichert und das können wir über InteliJ oder MongoDB Compass testen.

Ein Bild, das Text, Screenshot, Software, Zahl enthält.

Automatisch generierte Beschreibung

Ein Bild, das Text, Screenshot, Software, Display enthält.

Automatisch generierte Beschreibung

## DB werte lesen in Java

Um jetzt etwas mit diesen Pokémons aus der Datenbank anfangen zu können, müssen die werte gelesen werden können.