

Oefeningen: MongoDB & Mongoose

Oefening 1

Schrijf een NodeJS Webserver met Express & Mongoose. Voor deze applicatie maak je gebruik van een database met de naam "H13oef". Hierin gebruik je een collection "cars" waarin je car objecten aanmaakt. Deze objecten hebben volgende structuur:

```
> db.cars.find().pretty()
{
  "_id" : ObjectId("5a646b5cadda6c304551f67c"),
  "manufacturer" : "Porsche",
  "model" : "911",
  "price" : 135000,
  "wiki" : "http://en.wikipedia.org/wiki/Porsche_997"
}
{
  "_id" : ObjectId("5a646b7aadda6c304551f67d"),
  "manufacturer" : "Nissan",
  "model" : "GT-R",
  "price" : 80000,
  "wiki" : "http://en.wikipedia.org/wiki/Nissan_Gt-r"
}
{
  "_id" : ObjectId("5a646ba5adda6c304551f67e"),
  "manufacturer" : "BMW",
  "model" : "M3",
  "price" : 60500,
  "wiki" : "http://en.wikipedia.org/wiki/Bmw_m3"
}
```

Maak het nodige model in Mongoose en voorzie volgende routes in je NodeJS Webserver:

- `app.get('/cars')`: Geeft alle car objecten terug
- `app.get('/cars/:id')`: Krijg een specifiek car object terug

Zorg ervoor dat de body-parser JSON formaat gebruikt. Test je routes uit met een tool zoals postman.

Oefening 2

Maak een eenvoudige Angular applicatie die de data uit de vorige oefening ophaalt en toont in de console.

Oefening 3

Vertrek vanuit de NodeJS webserver van oefening 1. Voorzie volgende routes bovenop de reeds bestaande routes:

- `app.post('/cars')`: Voegt een nieuw car-object toe aan de databank
- `app.delete('/cars/:id')`: Verwijder een car-object op basis van de id.
- `app.put('/cars')`: Update car-object op basis van de id.

Test de routes uit met een tool zoals postman.

Oefening 4

Vertrek vanuit oefening 2. Zorg voor een angular applicatie waarbij je basis CRUD functionaliteit voorziet.

- Voor het tonen van de data maak je gebruik van een car-detail component. Voor elk car object wordt deze component geladen. Voorzie ergens een link naar het wikipedia artikel.
- Voor het toevoegen van data maak je gebruik van een model driven form.

Uitbreiding: Voorzie een bank-account Service die start met een saldo van € 120 000,00. Voorzie in de car-detail component een knop “buy” waarmee je een auto kan kopen. Het bedrag van de auto wordt dan afgetrokken van je bank saldo.

Probeer gebruik te maken van de 3rd party module toastr (`npm install ng2-toastr`) om een melding te geven “auto gekocht”. Indien je niet genoeg geld hebt, toon je de melding “Niet voldoende geld!” va diezelfde module.