twan van maastricht

[ajmaastr@avans.nl](mailto:ajmaastr@avans.nl)

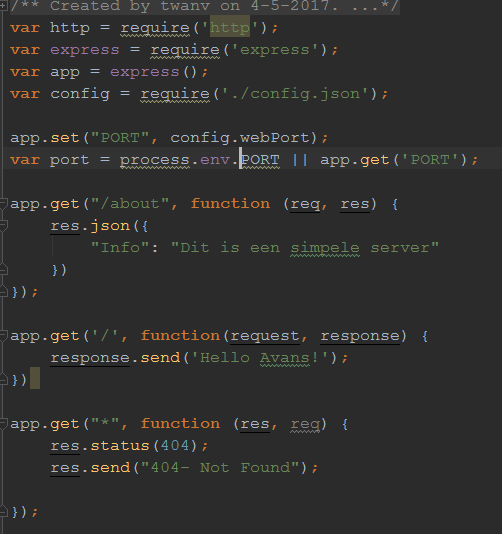
2113135

Practicum 2

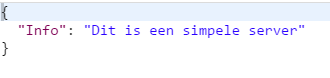
# Opdracht 1

**Maak in je projectdirectory een server.js file met een heel eenvoudige REST server die luistert op poort 8080. Kijk naar de practicumopgaven van practicum 1 en naar de presentatie van training 2 hoe dat ook alweer moest.**

**Test je REST server met een paar requests in Postman.**

**Server:** 

**Input:** http://localhost:3000/about

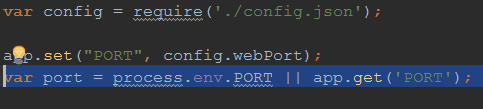
**JSON fragment:** 

**Input:** http://localhost:8080/

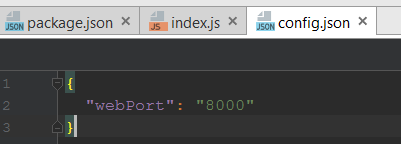
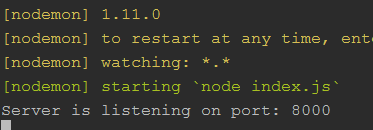
**JSON fragment:** 

# Opdracht 2

**Pas je server.js file aan om de instellingen uit config.json in te lezen en zorg dat je server het poortnummer uit deze config.json file gebruikt.**

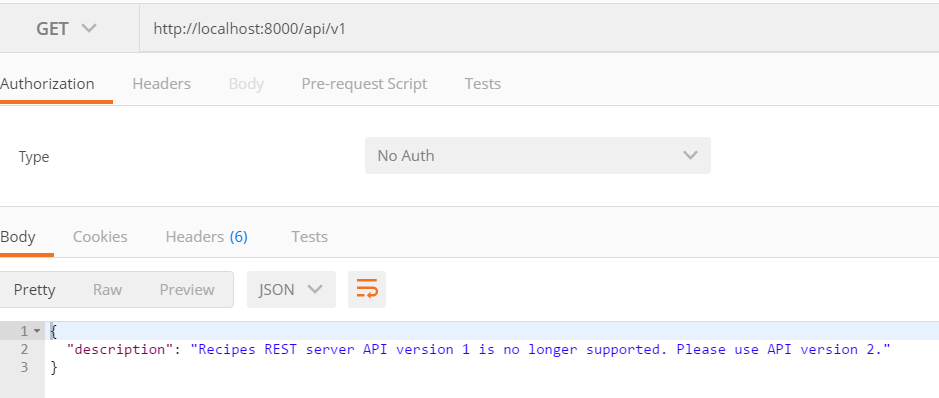
**Webport:** 

**Test je server opnieuw, verifieer dat de wijziging werkt. Zet bijvoorbeeld tijdelijk een ander poortnummer in je config.json en test daarmee.**

**Test:** 

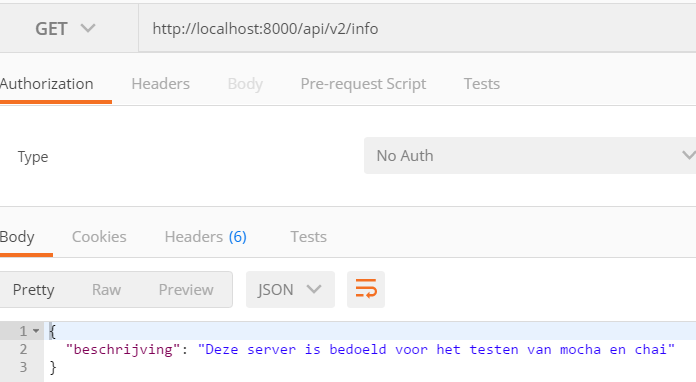
# Opdracht 3

**Start je server en ga na (b.v. met Postman) dat de route naar /api/v1 de verwachte melding geeft:**

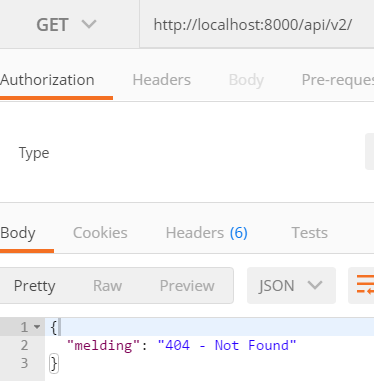


# Opdracht 4

**Vul file ‘routes\_api\_v2.js’ op soortgelijke wijze en neem daarin een route op voor het pad ‘/info’. Laat dat pad een JSON object opleveren met een tekst die je server kort beschrijft. De status code moet 200 worden:**



**Voeg voor API versie 2 ook een route toe voor ‘\*’ die een status code 404 met een JSON object met daarin een not-found melding oplevert:**



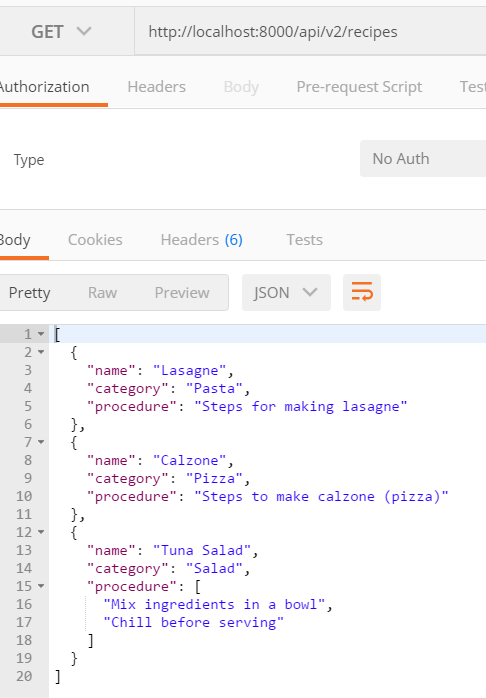
# Opdracht 5

**We moeten in routes\_api\_v2.js nu een route toevoegen die deze recepten oplevert:**

**/api/v2/recipes 🡪 levert de hele array met recepten**

**Pas de code zo aan dat dit werkt zoals gevraagd.**

**Test je code, verifieer dat deze route werkt:**

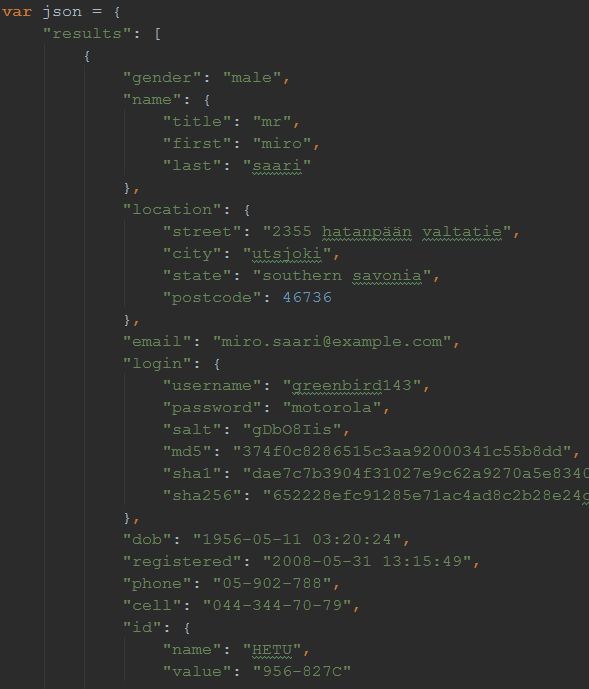


# Opdracht 7

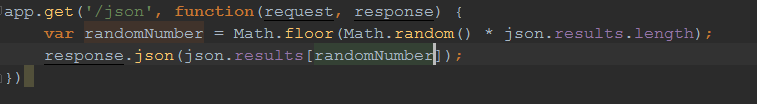
1. **Pas de code zo aan dat telkens de data van één van die gebruikers in JSON formaat wordt teruggestuurd op een GET request naar het pad ‘/api’ van je app, net als bij** [**http://randomuser.me/api**](http://randomuser.me/api)**. Als je Android app kan selecteren op gebruikers (b.v. alleen Nederlandse gebruikers of zoeken op naam), pas dan je REST server code zo aan dat ook die selectie in de API werkt.  
   Neem je gewijzigde code op in je verslag (niet de hele file, alleen het belangrijke stuk).**

**Code in server.js:**

*In var json staan 5 verschillende random users in een json array*



*/json get request geeft vervolgens een random user uit de array mee in json formaat.*



1. **Deploy je wijzigingen op Heroku en test met Postman. Zet een screenshot van Postman in je verslag om te laten zien dat het werkt.**

Resultaat: 