

## HARJOITUS 1.1: MINIMAALINEN EXPRESS-APPLIKAATIO /-REITILLÄ

- 1) Luo jonnekin täysin uusi nodeprojektikansio ja aja siellä komento:

```
npm init (tämän jälkeen voit käyttää esim. npm install <paketin nimi> )
```

- 2) Asenna express-moduuli
  - 3) Luo app.js -tiedosto, ja luo siitä express-applikaatio
  - 4) Aseta ohjelma kuuntelemaan porttia 3000
  - 5) Luo GET-reitti / -reittiin. Reitin tulee tulostaa sivulle "Tervetuloa" käyttäen res.send -metodia.
  - 6) Käynnistä ohjelma node app.js -komennolla ja avaa selain /-reittiin valitussa portissa. Mikäli teit kaiken oikein, tulostuu sivulle "Tervetuloa". Voit sulkea ohjelman (CTRL-C)
- 

## HARJOITUS 1.2: EJS LISÄYS

- 1) Asenna ejs-moduuli
  - 2) Aseta 1.tehtävän applikaatio käyttämään ejs template engineä (käytä app.set kuten tunneilla, katso aiemmilta tunnelista apua)
  - 3) Luo muuttuja welcomeText, ja aseta siihen haluamasi merkkijono.
  - 4) Luo views-kansio, ja luo sinne index.ejs -tiedosto.
  - 5) Lisää tiedostoon jonkinlainen html-rakenne, jossa on body-osa.
  - 6) Lisää index.ejs -tiedoston body-osaan ejs-muuttuja nimeltään welcome.
  - 7) Muuta GET / reittiä niin, että renderöit index.ejs -sivun ja viettävä sivun welcome-muuttujaan welcomeText-muuttujan.  
( jolloin welcomeText-muuttujan sisältö tulostuu sivulle kun ohjelma laitetaan käyntiin ja / reittiin mennään)
  - 8) Käynnistä ohjelma node app.js -komennolla ja avaa selain /-reittiin valitussa portissa. Mikäli teit kaiken oikein, tulostuu sivulle welcomeText-muuttujaan määrittämäsi merkkijono. Voit sulkea ohjelman (CTRL-C)
- 

## HARJOITUS 2: NODEMON LISÄYS (Tämän voi tehdä erikseen tai yhdistää aiempaan tehtäviin)

- 1) Asenna nodemon-moduuli
  - 2) Aseta ohjelma käynnistymään niin, että npm start -komento suorittaa komennon: nodemon app.js
  - 3) Käynnistä ohjelma npm start -komennolla.
  - 4) Mikäli virheitä ei tule, tehtävä ok, voit sulkea ohjelman (CTRL-C)
- 

## HARJOITUS 3: MODUULIN LUONTI JA KÄYTÖ (Tämän voi tehdä erikseen tai yhdistää aiempaan tehtäviin)

- 1) Luo config.js -niminen tiedosto ja luo siihen port-muuttuja, jonka arvoksi asetat portin numeron. (portti 3000 toimii hyvin Taitotalon luokissa)
  - 2) Määritä port-muuttuja module.exports:in avulla niin, että moduulin port-muuttuja voidaan käyttää muissa moduuleissa.
  - 3) Hae pääohjelman kuuntelema portti luomastasi config-moduulista ja käytä sitä portin määritykseen.
-

#### HARJOITUS 4: TOISEN REITIN LUOMINEN, UUDELLEENOJHAUS (tehtävä 1.1 pitää olla tehty)

- 1) Luo reitti GET /google.
  - 2) Kun käyttäjä menee reittiin, uudelleenohjataan käyttäjä <https://www.google.com> -osoitteeseen.
- 

#### HARJOITUS 5: (edistyneemmille)

- 1) Käytä tunneilla käytettyä users.json -tiedostoa, ja hae päähjelmaan siitä käyttäjät users-nimiseen muuttujaan.
  - 2) Kun käyttäjä yrittää kirjautua, tarkistetaan, löytyikö käyttäjä users.json:ista (salasanaa ei vielä tässä tarkisteta)
  - 3) Luo myös reitti GET /users, joka näyttää kaikki käyttäjänimet
- 

#### JATKOHARJOITUKSIA (edistyneemmille/jos sait tehtyä)

- Lisää myös salasanan tarkistus kirjautumiseen
- Luo toiminnalisuudet, joilla voit päivittää ja poistaa käyttäjiä. Voit käyttää luotua yksinkertaista formia pohjana ja luoda reitit sekä päivittämiseen (jossa haetaan tietyn käyttäjän tiedot) sekä poistamiseen (käyttäjä syöttää esim. poistettavan käyttäjän nimen, ja mikäli käyttäjä löytyy, poistetaan se).
- Sovella käyttäjä asioita johonkin omaan projektiisi