# ExpressJS Exercises: ExpressJS and View Engines

## Project Sculp

Refactor the 'MemeDB' project that is provided, so that your app meets the following requirements:

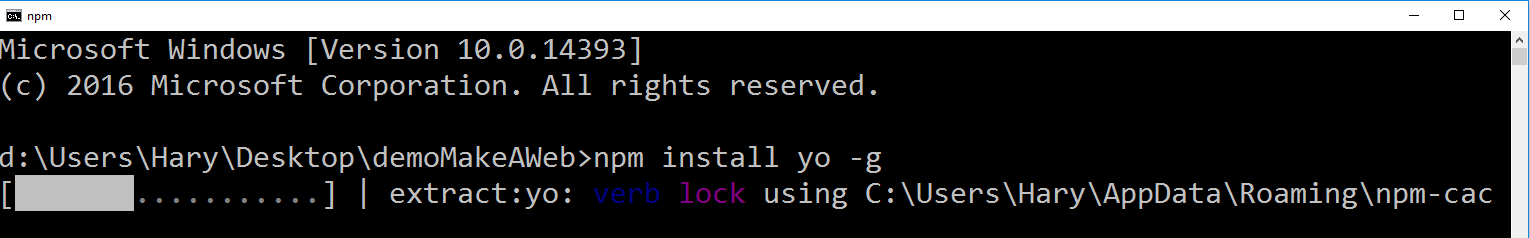
**Minimum requirements:**

* Have at least **6** **pages**
* Use **MongoDB**
* Have at least **2 Schemas**
* Use **Express**
* Use **View Engine of your choice**
* Use **File Upload**

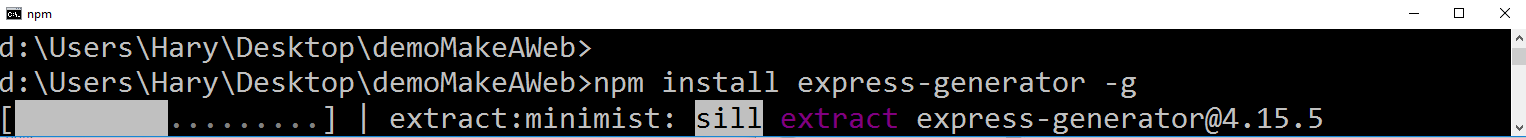
## Create new project

Create new express project using express and view engine of your choice

#### Install YO man

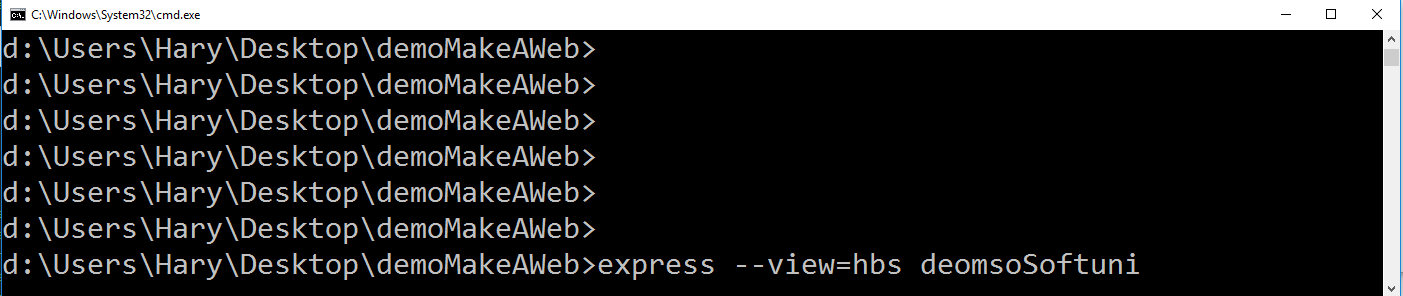
Using the node package manager install [yeoman](http://yeoman.io).

#### 1.2 Install express

Using the node package manager install [express-generator](https://expressjs.com/en/starter/generator.html)

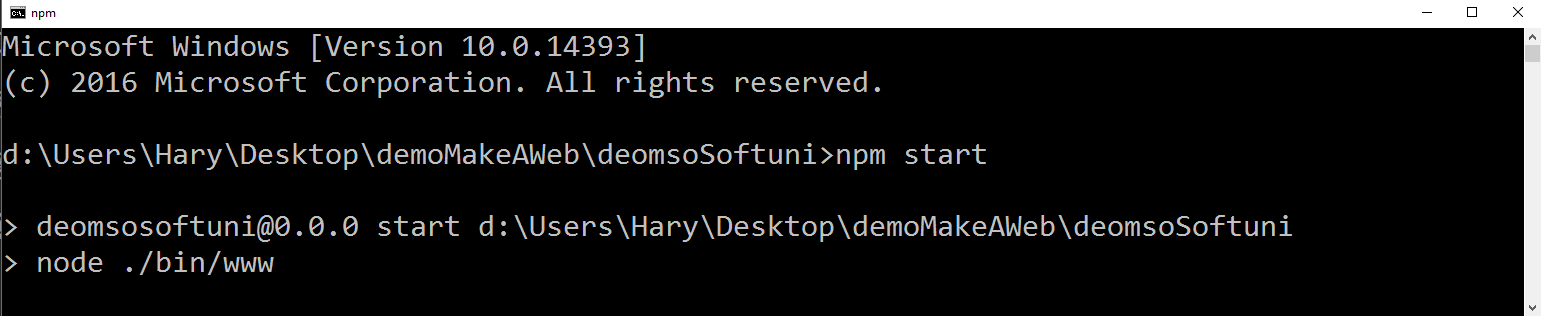
#### 1.3 Run the express generator

Run the express generator, set the view engine that you will use and the name of the project



\*\* Don’t forget to run **npm install,** after that you will be able to run the app

#### Config MongoDB

Create **config** folder, inside of it create **dbConfig.js** module, which should be holding all your MongoDB settings. Connect the module at the appropriate place, so that it establishes connection when the app is started, also try to log a friendly message if the connection is successful.

## Refactor the app

#### 2.1 Refactor the html

Refactor all needed html from the provided task to the format, that is supported by your view engine of choice.

#### 2.2 Implement the routing

Create routes, that lead to the appropriate handlers, when a request is send.

#### 2.3 Create Handlers

For each route, create a handler, that executes the desired back-end logic, when called.

\***Before** implementing any **upload** logic create all needed **Schemas**

## Create all needed Schemas

#### 3.1 Create genre Schema

Create genre schema, that will hold **genreName,** array that holds all memes with the current genre **memeList.**

#### 3.2 Create meme Schema

Create meme Schema, that will hold **memeName, dateOfCreation, votes** which will hold all generated votes**, description, title.**

## Implement creation logic

#### 4.1 Implement logic for creating genre

Implement handler that, will save all send data in the DB, upon creation of a new genre.

#### 4.2 Implement upload logic

Implement upload logic, that will save all send data (it should be a **real** **image**, not a url), upon creation you should fill the new entry’s id in the appropriate genre.

\* You must restrict the creation of new memes in such way, that the user can only create memes, with valid genre properties (only if the genre exists)