Teoría

* Body Parser
  + <https://www.npmjs.com/package/body-parser>
  + Instalar:
    - npm install body-parser --save
* Plantillas para Node.js y Express
  + <https://expressjs.com/en/resources/template-engines.html>
* Nunjucks
  + <https://mozilla.github.io/nunjucks/>
  + <https://mozilla.github.io/nunjucks/templating.html>
  + npm install nunjucks --save

Proyecto

* <https://appclima.juliobarbagallo.vercel.app/>
* <https://github.com/juliobarbagallo/appclima/tree/jv2>

Ejercicios

* **Con node y body-parser (sin nunjucks):** 
  + Crear un formulario en / que pida nombre de la persona, destino (select) y cantidad de pasajes.
  + En /cotizacion recibe los parámetros x post y muestra el valor del viaje.
  + Los datos se obtienen de const destinos = [{"destino":"Brasil","precio":"500"},{"destino":"Francia","precio":"1200"},{"destino":"Alemania",precio:"1350"},{"destino":"Noruega","precio":"1500"}]
* **Con node, express y Nunjucks + node-fetch (sin body-parser)**
  + De <https://raw.githubusercontent.com/algolia/datasets/master/movies/actors.json>
  + En la home / mostrar:
    - El listado de nombres (ordenados por nombre)
    - Cada nombre tendrá un link a /actor/id
    - Mostrar un párrafo al final con la cantidad de actores
  + En /actor/id mostrar:
    - Nombre del actor
    - Rating
    - Nombre alternativo si lo tiene
* **Con node, express y Nunjucks (sin body-parser)**
  + Mostrar en la home todos los títulos de los libros de:
    - <https://www.googleapis.com/books/v1/volumes?q=node.js>
    - Los títulos deben tener un link a la ruta: /libro/:id
  + En la ruta libro/id mostrar:
    - Título
    - Subtítulo
    - Imagen (si tiene)
    - Autores
    - Descripción
    - Cantidad de páginas
    - Publisher
    - NOTA: para acceder al libro individual sería:
      * fetch(`https://www.googleapis.com/books/v1/volumes/${req.params.id}`)
      * Ejemplo: https://www.googleapis.com/books/v1/volumes/GOfqDwAAQBAJ