Simple Blog

Design: <https://projects.invisionapp.com/freehand/document/MV9ZD7c0i>

## 1.nézet - Főoldal - **Blog posts**

## **A főoldal betöltődéskor lekéri az adatokat a szerverről egy GET kéréssel, majd a dizájnnak megfelelően 3 oszlopban kiiratja őket. Kritériumok:**

- Kilistázza a blog postokat (a fő oldalon, de maximum 30 darabat )

- a cikk tartalmából maximum 100 akrakter jelenik meg a kártyákon.

- fájlok index.html, index.js  
- A blog bejegyzések API forrása: <https://jsonplaceholder.typicode.com/posts>

-a http kérés Promise-t használjon!  
Interakció:

-a cikk kártya tartalmaz egy linket, ami átnavigál egy blog cikkre (második nézetre), ahol a kiválasztott egy darab blog cikk

megjelnik részletesen.   
A post oldal ugyanazon a lapon jelenik meg!  
  
A linkben megjelenik a post id-ja: post.html#{postId}  
Példa:  
 <a href=“post.html#1”> link to post</a>

## nézet: Blog post **Egy külön html oldal, ahol az egyes cikkek rézletesen megjelennek.** Kritériumok:

* post.html és post.js fájlban valósul meg.
* - a poszt cikket külön GET kéréssel kér lei! Tehát csak azt az egyet, amit éppen meg kell jeleníteni!   
  Az adat forrása mindehhez: https://jsonplaceholder.typicode.com/posts/{id}   
  Példa:<https://jsonplaceholder.typicode.com/posts/1>
* megjelenítendő adat: post title, post body (cikk szövege)

### Post cikk adatai JSON object-ként:

|  |
| --- |
| {  "userId": 1, // cikk szerzőjének egyedi azonosítója  "id": 1, // cikk egyedi azonosíója  "title": " excepturi optio reprehenderit", // cikk címe  "body": "stiae ut ut..." // cikk címe } |

## API források:

* Összes post:   
  <https://jsonplaceholder.typicode.com/posts>
* Egy darab post :  
  [https://jsonplaceholder.typicode.com/posts/{id](https://jsonplaceholder.typicode.com/posts/%7Bid)}   
  //{id} - post id-ja  
    
  példa: <https://jsonplaceholder.typicode.com/posts/1>

Update v2:

## A post.html oldalon a blog cikk alatt szeretnénk megjeleníteni kommenteket!

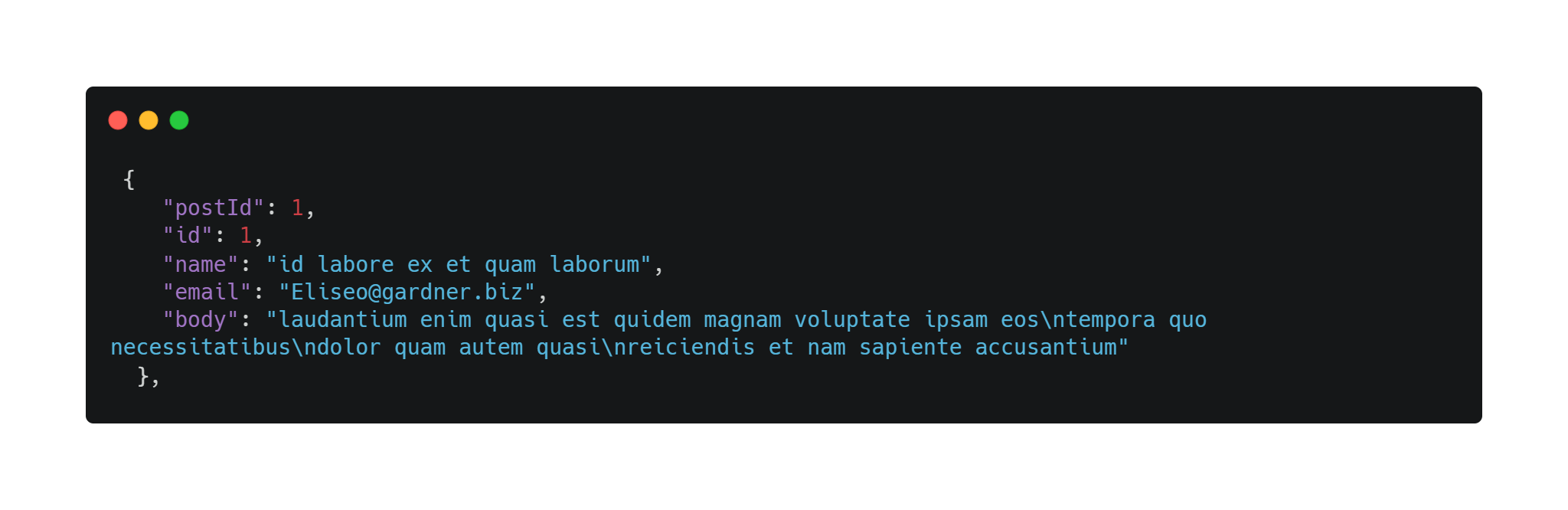
Az adott blog cikkhez tartozó kommenteket az alábbiak szerint lehet elérni:

[https://jsonplaceholder.typicode.com/comments?postId={id](https://jsonplaceholder.typicode.com/comments?postId=%7Bid)}

Példa: Az “1”-es id-jű post cikk kommentjeihez a kérés:

<https://jsonplaceholder.typicode.com/comments?postId=1>  
  
vagy

<https://jsonplaceholder.typicode.com/posts/1/comments>  
  
Egy komment json-ben:



Kritériumok:

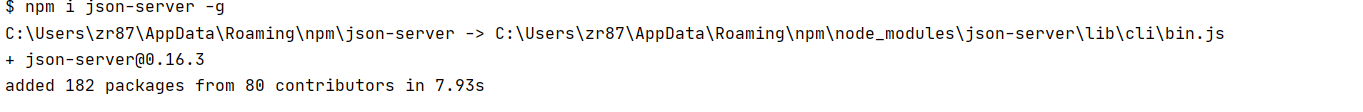
* A post kérés menjen elsőnek, a kommentek lekérése másodiknak!
* legyenek láncolva a Http kérések → ha az első lekérés nem sikerült, akkor a kommenteket már ne kérje le! (promise chaining)
* használj **Fetch API-t**!
* Megjelenítendő adatok:  
  **name (komment szerzőjének neve),   
  body (komment szövege)**
* kommentek lap tetejétől id csökkenő sorrendben kerüljenek be! (1-es id-jű komment a alap alján legyen)

Design update:

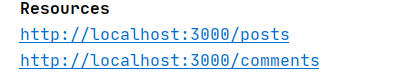
<https://projects.invisionapp.com/freehand/document/MV9ZD7c0i>

Update v3

A blog fejlesztése során problémák merültek fel!  
Kiderült, hogy a backend fejlesztők csak a GET kéréseket fejlesztették le a Simple blog adatait szolgálattó API-nál (https://jsonplaceholder.typicode.com/), de nem implementálták a POST kéréseket, ezért adatot nem tudunk küldeni.  
  
Hogy a fejlesztés ne álljon le, egy saját lokális webszerverrel létrehozunk egy API endpointot!

1. telepítsd a node.js legújabb LTS verzióját! Ez egy konzol program, ezért a windosws alatt cmd, linux és mac alatt a Terminal programból tudjuk tesztelni, hogy sikeresen feltelepült. Esetleg a “git bash” terminált is használhatjátok!
2. Teszteld, hogy működik a node-js:   
   parancssorba írd be a node --version (röviden node -v) parancsot és nyomj ENTER-t  
   Ha valami verziószámot látsz, akkor sikeres a telepítés, működik:  
   
3. A node.js-sel feltelepül egy NPM nevű alkalmazás is, amely js csomagokat kezel.  
   Teszteld, hogy működik az npm a gépeden:  
     
   Parancssorba írd be a npm--version (röviden npm -v) parancsot és nyomj ENTER-t  
   Ha valami verziószámot látsz, akkor sikeres a telepítés, működik:  
   
4. Egy mini server csomagot telepítünk - neve “JSON-server”, ami létrehoz egy lokális API endpointot és egy helyi db.json fájból olvas fel JSON adatokat.  
     
   A csomag adatlapja: <https://www.npmjs.com/package/json-server>  
     
   Telepítéséhez: futtasd terminálban az alábbi parancsot:  
   **npm install json-server -g**Ha valami ilyesmit olvasol, akkor rendben ment a tepelítés: ****
5. A Simple Blog projekt mappájában hozz létre egy db.json nevű fájlt ez lesz a blog alkalmazás adattárolója,ez tárolja majd a post-okat és a commentekeket is.  
     
   A json fájlban hozd létre az alábbi json struktúrát:

|  |
| --- |
| {  "posts": [ ],  "comments": [ ]  } |

1. A terminál-t a blog mappájéba navigálva futtasd a **json-server --watch db.json** parancsot!  
   - ezzel elindítasz egy lokális web szervert, ami a localhost:3000 címen érhető el.  
   Nyisd meg és olvasd át, amit a <http://localhost:3000/> címen találsz!
2. A webszerveren az alábbi endpointok elérhetőek: posts, comments  
   
3. Használjuk az új API-t, írjuk át a blog meglévő API URL címeket a localhost:3000-re.
4. Teszteld az eddigi GET kéréseket a böngés
5. Tölsd meg a db.json post tömbjét pár post item-mel.

|  |
| --- |
| {  "userId": 1,  "id": 1,  "title": "sunt aut facere repellat provident occaecati excepturi optio reprehenderit",  "body": "quia et suscipit\nsuscipit recusandae consequuntur expedita et cum\nreprehenderit molestiae ut ut quas totam\nnostrum rerum est autem sunt rem eveniet architecto"  },  {  "userId": 1,  "id": 2,  "title": "qui est esse",  "body": "est rerum tempore vitae\nsequi sint nihil reprehenderit dolor beatae ea dolores neque\nfugiat blanditiis voluptate porro vel nihil molestiae ut reiciendis\nqui aperiam non debitis possimus qui neque nisi nulla"  },  {  "userId": 1,  "id": 3,  "title": "ea molestias quasi exercitationem repellat qui ipsa sit aut",  "body": "et iusto sed quo iure\nvoluptatem occaecati omnis eligendi aut ad\nvoluptatem doloribus vel accusantium quis pariatur\nmolestiae porro eius odio et labore et velit aut"  }, |

1. Teszteld a GET kéréseket az új adatokkal! Böngészőben aztán az alkalmazásban!

Megtörtént a váltás, most már egy lokálisan futtatott server szolgálja ki az alkalmazást adatokkal. Ideje új feature fejlesztésébe vágni!

## Komment küldése

A post.html oldalon meg jell oldani, hogy a felhasználók tudjanak kommentelni!  
  
Invision drótváz:

<https://projects.invisionapp.com/freehand/document/MV9ZD7c0i>Kell egy form, aminek minden inputja kötelező!Kritériumok:  
- a fejlesztés kerüljön egy különálló comments.js fájlba   
- a komment POST request a **/comments** API endpoint-ra küldje a kérést.   
(A localhost:3000 cím is kelleni fog !  
- A komment elküldése után (submit)  
 - sikeres POST request esetén (200-299 status code) → komment egyből kerüljön be a html-be a többi komment elé!  
 - sikertelen küldés esetén adjon hibaüzenetet a felhasználónak. (Something happened, please try again later!)  
- böngésző újratöltéskor már látszodjon az uj komment!  
- a submit után a form alapállapotba kerülön vissza!  
- további kritériumok a dizájnban!