

Ejemplo tesseract OCR js sencillo

Documentación: <https://github.com/naptha/tesseract.js#tesseractjs>

El código html sería:

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Document</title>
</head>
<body>
  <!-- v4 -->
  <script src='https://unpkg.com/tesseract.js@4.0.2/dist/tesseract.min.js'></script>
  <script src="index.js"></script>
</body>
</html>
```

Mientras el código javascript:

```
// import Tesseract from 'tesseract.js';

Tesseract.recognize(
  'https://tesseract.projectnaptha.com/img/eng_bw.png',
  'eng',
  { logger: m => console.log(m) }
).then(({ data: { text } }) => {
  console.log(text);
})
```

Observaciones

no es necesario usar `import Tesseract from 'tesseract.js';` porque eso es cuando tienes instalado tesseract con node js.

como estoy importando con CDN `<script src='https://unpkg.com/tesseract.js@4.0.2/dist/tesseract.min.js'></script>` no es necesario.

La documentación recomienda los siguientes códigos:

```
import Tesseract from 'tesseract.js';

Tesseract.recognize(
```

```
'https://tesseract.projectnaptha.com/img/eng_bw.png',  
'eng',  
{ logger: m => console.log(m) }  
) .then(({ data: { text } }) => {  
  console.log(text);  
})
```

```
import { createWorker } from 'tesseract.js';  
  
const worker = await createWorker({  
  logger: m => console.log(m)  
});  
  
(async () => {  
  await worker.loadLanguage('eng');  
  await worker.initialize('eng');  
  const { data: { text } } = await  
worker.recognize('https://tesseract.projectnaptha.com/img/eng_bw.png');  
  console.log(text);  
  await worker.terminate();  
})();
```

Instalación

Como te decía, hay 2 formas: con CDN

```
<!-- v4 -->  
<script src='https://unpkg.com/tesseract.js@4.0.2/dist/tesseract.min.js'></script>
```

O con node js

```
# For latest version  
npm install tesseract.js  
yarn add tesseract.js  
  
# For old versions  
npm install tesseract.js@3.0.3  
yarn add tesseract.js@3.0.3
```

Otras ideas, próximamente:

- sería bueno usar git para guardar el historial de modificaciones
- ya tengo un index que carga imagenes, sería unir ese con uno que muestre el texto.

Es algo sencillo, tardaría menos de 1h, no quiero agregar más ya que solo 1 paso a la vez...