

DISKUSIKAN MATERI

LAPORKAN MATERI

## Daftar Modul

Masukkan kata kunci

Persetujuan Hak Cipta

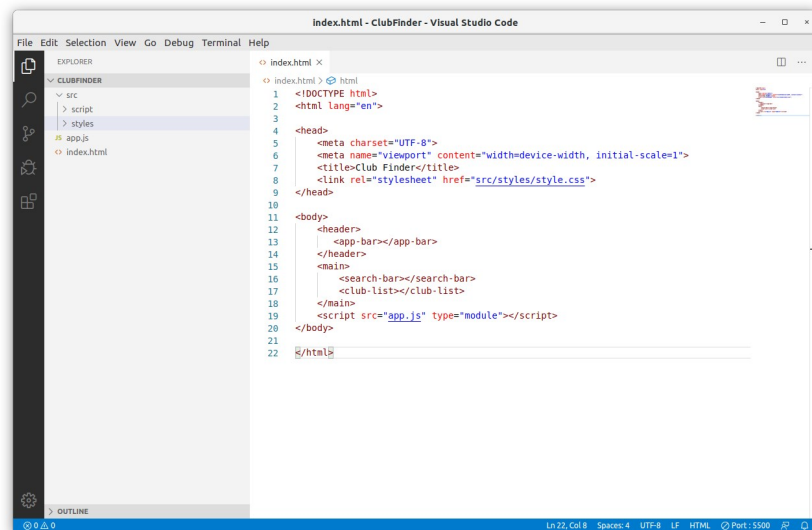
Modul 1:  
Introduction  
Course (Gratis) Prasyarat Kelas  
(Gratis) Apa yang Akan  
Kita Pelajari?  
(Gratis) Tools  
Requirement Modul 2:  
ECMAScript 6  
(Gratis) 

## Solution: Menerapkan Shadow DOM pada Proyek Club Finder

Apakah Anda berhasil menerapkan shadow DOM pada custom element di proyek Club Finder? Jika belum, mari kita lakukan bersama-sama.

### Menerapkan Shadow DOM pada App Bar

Kita mulai dari `<app-bar>` component yuk. Pertama kita buka dulu proyek club finder dengan text editor yang kita gunakan.



Kemudian buka berkas `script -> component -> app-bar.js`, buat constructor dari class tersebut dan di dalamnya kita tetapkan shadow root seperti ini:

```
1 class AppBar extends HTMLElement {
2
3   constructor() {
4     super();
5     this.shadowDOM = this.attachShadow({mode: "open"});
6   }
7
8   connectedCallback(){
9     this.render();
10  }
11
12  render() {
13    this.innerHTML = `<h2>Club Finder</h2>`;
14  }
15 }
16
17 customElements.define("app-bar", AppBar);
```

Karena kita sudah menerapkan Shadow DOM pada AppBar, jangan lupa pada fungsi `render()`, kita ubah `this.innerHTML` menjadi `this.shadowDOM.innerHTML`.

```
1 class AppBar extends HTMLElement {
2
3   constructor() {
4     super();
```

```

5.     this.shadowDOM = this.attachShadow({mode: "open"});
6.   }
7.
8.   connectedCallback(){
9.     this.render();
10.  }
11.
12.  render() {
13.    this.shadowDOM.innerHTML = `<h2>Club Finder</h2>`;
14.  }
15. }
16.
17. customElements.define("app-bar", AppBar);

```

Kemudian buka berkas *style -> appbar.css* dan pindahkan (*cut*) seluruh kode yang ada pada berkas tersebut.

```

1. app-bar {
2.   display: block;
3.   padding: 16px;
4.   width: 100%;
5.   background-color: cornflowerblue;
6.   color: white;
7.   box-shadow: 0 4px 8px 0 rgba(0, 0, 0, 0.2);
8. }

```

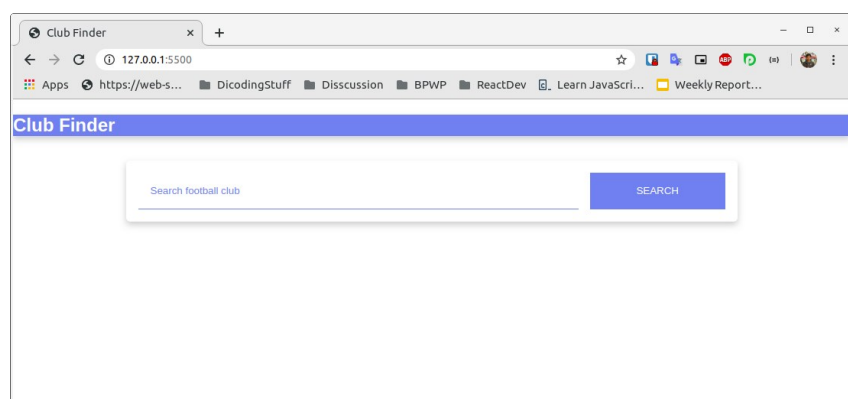
Lalu tempel (paste) pada nilai `this.shadowDOM.innerHTML` dengan dibungkus oleh element `<style>` tepat sebelum element `<h2>` pada fungsi `render()` di berkas *app-bar.js* seperti ini:

```

1. class AppBar extends HTMLElement {
2.
3.   constructor() {
4.     super();
5.     this.shadowDOM = this.attachShadow({mode: "open"});
6.   }
7.
8.   connectedCallback(){
9.     this.render();
10.  }
11.
12.  render() {
13.    this.shadowDOM.innerHTML = `
14.      <style>
15.        app-bar {
16.          display: block;
17.          padding: 16px;
18.          width: 100%;
19.          background-color: cornflowerblue;
20.          color: white;
21.          box-shadow: 0 4px 8px 0 rgba(0, 0, 0, 0.2);
22.        }
23.      </style>

```

Coba kita simpan perubahan yang diterapkan kemudian lihat perubahannya pada browser.



Ups, pada browser kita dapat melihat title yang ditampilkan pada `<app-bar>` tampak berantakan. Untuk menanganinya, kita perlu menyesuaikan kembali style yang diterapkan pada custom element menjadi seperti ini:

```
1. class AppBar extends HTMLElement {
2.
3.   constructor() {
4.     super();
5.     this.shadowDOM = this.attachShadow({mode: "open"});
6.   }
7.
8.   connectedCallback(){
9.     this.render();
10.  }
11.
12.   render() {
13.     this.shadowDOM.innerHTML = `
14.       <style>
15.         * {
16.           margin: 0;
17.           padding: 0;
18.           box-sizing: border-box;
19.         }
20.         :host {
21.           display: block;
22.           width: 100%;
23.           background-color: cornflowerblue;
```

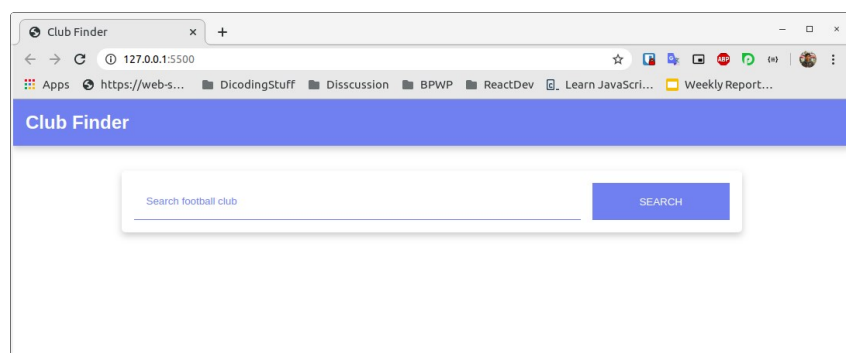
Pada perubahan styling tersebut kita menambahkan

```
1. * {
2.   margin: 0;
3.   padding: 0;
4.   box-sizing: border-box;
5. }
```

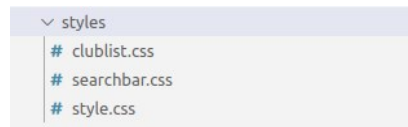
Yang digunakan untuk menghilangkan seluruh margin dan padding standar yang diterapkan pada element html. Dan kita juga mengubah pengaturan `box-sizing` menjadi `border-box`.

Lalu kode pada kode styling lainnya juga kita melihat bahwa selector `app-bar` digantikan dengan `:host`. Apa itu `:host`? Selector `:host` merupakan selector yang digunakan untuk menunjuk element `:host` yang menerapkan Shadow DOM. Pada host kita tidak dapat mengatur padding sehingga kita perlu memindahkannya pada elemen `<h2>`.

Setelah melakukan perubahan tersebut simpan (save) kembali perubahannya dan lihat hasilnya pada browser, seharusnya `<app-bar>` sudah ditampilkan dengan baik.



Karena kita sudah tidak membutuhkan lagi berkas *src -> styles -> appBar.css*, kita dapat menghapus berkas tersebut.



Jangan lupa untuk menghapus import css tersebut pada *src -> styles -> style.css*.

```
1. @import "clublist.css";
2. @import "searchbar.css";
3.
4. * {
5.   padding: 0;
6.   margin: 0;
7.   box-sizing: border-box;
8. }
9.
10. body {
11.   font-family: sans-serif;
12. }
13.
14. main {
15.   width: 90%;
16.   max-width: 800px;
17.   margin: 32px auto;
18. }
```

## Menerapkan Shadow DOM pada Search Bar

Setelah berhasil menerapkan Shadow DOM pada App Bar, selanjutnya kita terapkan Shadow DOM pada search bar. Silakan buka berkas *src -> script -> component -> search-bar.js*, kemudian buat constructor dan terapkan Shadow DOM di dalamnya.

```
1. class SearchBar extends HTMLElement {
2.
3.   constructor() {
4.     super();
5.     this.shadowDOM = this.attachShadow({mode: "open"});
6.   }
7.
8.   connectedCallback(){
9.     this.render();
10.  }
11.
12.   set clickEvent(event) {
13.     this._clickEvent = event;
14.     this.render();
15.   }
16.
17.   get value() {
18.     return this.querySelector("#searchElement").value;
19.   }
20.
21.   render() {
```

```
22.     this.innerHTML = `  
23.     <div id="search-container" class="search-container">
```

Sama seperti yang kita lakukan pada component App Bar, kita ubah `this.innerHTML` menjadi `this.shadowDOM.innerHTML` pada fungsi `render()`.

```
1.  class SearchBar extends HTMLElement {  
2.  
3.      constructor() {  
4.          super();  
5.          this.shadowDOM = this.attachShadow({mode: "open"});  
6.      }  
7.  
8.      connectedCallback(){  
9.          this.render();  
10.     }  
11.  
12.     set clickEvent(event) {  
13.         this._clickEvent = event;  
14.         this.render();  
15.     }  
16.  
17.     get value() {  
18.         return this.querySelector("#searchElement").value;  
19.     }  
20.  
21.     render() {  
22.         this.shadowDOM.innerHTML = `  
23.         <div id="search-container" class="search-container">
```

Selain itu juga kita ubah pemanggilan `this.querySelector` menjadi `this.shadowDOM.querySelector` pada fungsi `render()` dan `get value()`.

```
1.  class SearchBar extends HTMLElement {  
2.  
3.      .....  
4.  
5.     get value() {  
6.         return this.shadowDOM.querySelector("#searchElement").value;  
7.     }  
8.  
9.     render() {  
10.         .....  
11.         this.shadowDOM.querySelector("#searchButtonElement").addEventListener("click", this._clickEvent);  
12.     }  
13. }  
14.  
15.  
16. ....
```

Kemudian buka berkas `src -> styles -> searchbar.css`, pindahkan (*cut*) seluruh kode yang terdapat pada berkas tersebut.

```
1.  .search-container {  
2.      max-width: 800px;  
3.      box-shadow: 0 4px 8px 0 rgba(0, 0, 0, 0.2);  
4.      padding: 16px;  
5.      border-radius: 5px;  
6.      display: flex;  
7.      position: sticky;  
8.      top: 10px;  
9.      background-color: white;  
10. }  
11.  
12. .search-container > input {  
13.     width: 75%;  
14.     padding: 16px;  
15.     border: 0;
```

```

16.   border-bottom: 1px solid cornflowerblue;
17.   font-weight: bold;
18. }
19.
20. .search-container > input:focus {
21.   outline: 0;
22.   border-bottom: 2px solid cornflowerblue;
23. }

```

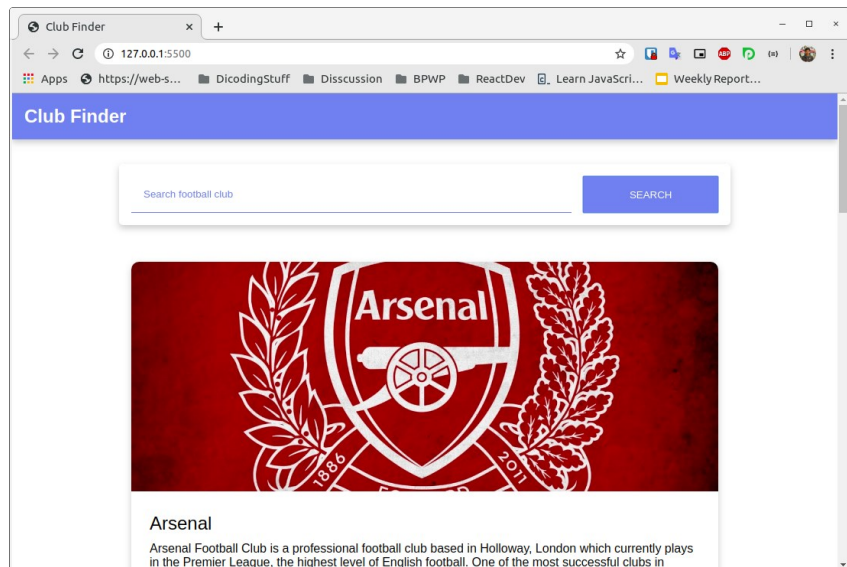
Lalu tempel (paste) pada nilai `this.shadowDOM.innerHTML` dengan dibungkus oleh element `<style>` tepat sebelum element `<div>` pada fungsi `render()` di berkas `search-bar.js` seperti ini:

```

1.  class SearchBar extends HTMLElement {
2.
3.      .....
4.
5.      render() {
6.          this.shadowDOM.innerHTML = `
7.          <style>
8.          .search-container {
9.              max-width: 800px;
10.             box-shadow: 0 4px 8px 0 rgba(0, 0, 0, 0.2);
11.             padding: 16px;
12.             border-radius: 5px;
13.             display: flex;
14.             position: sticky;
15.             top: 10px;
16.             background-color: white;
17.         }
18.
19.         .search-container > input {
20.             width: 75%;
21.             padding: 16px;
22.             border: 0;
23.             border-bottom: 1px solid cornflowerblue;

```

Simpan perubahan yang dilakukan kemudian lihat hasilnya pada browser.



Komponen Search Bar tampak normal dan berfungsi dengan baik sehingga kita tidak perlu menyesuaikan lagi styling-nya.

Karena kita sudah tidak membutuhkan lagi berkas `src -> styles -> searchbar.css`, kita dapat menghapus berkas tersebut.

```

v styles
# clublist.css
# style.css

```

Jangan lupa untuk menghapus import css tersebut pada `src -> styles -> style.css`.

```

1. @import "clublist.css";
2.
3. * {
4.     padding: 0;
5.     margin: 0;
6.     box-sizing: border-box;
7. }
8.
9. body {
10.     font-family: sans-serif;
11. }
12.
13. main {
14.     width: 90%;
15.     max-width: 800px;
16.     margin: 32px auto;
17. }

```

## Menerapkan Shadow DOM pada Club List dan Club Item

Terakhir kita terapkan Shadow DOM pada komponen club list dan club item. Silakan buka berkas *src -> script -> component -> club-list.js*, kemudian buat constructor dan terapkan Shadow DOM di dalamnya.

```

1. import './club-item.js';
2.
3. class ClubList extends HTMLElement {
4.
5.     constructor() {
6.         super();
7.         this.shadowDOM = this.attachShadow({mode: "open"});
8.     }
9.
10.     set clubs(clubs) {
11.         this._clubs = clubs;
12.         this.render();
13.     }
14.
15.     renderError(message) {
16.         this.innerHTML = "";
17.         this.innerHTML += `<h2 class="placeholder">${message}</h2>`;
18.     }
19.
20.     render() {
21.         this.innerHTML = "";
22.         this._clubs.forEach(club => {
23.             const clubItemElement = document.createElement("club-item");

```

Kemudian ubah seluruh kode `this.innerHTML` menjadi `this.shadowDOM.innerHTML` dan `this.appendChild` menjadi `this.shadowDOM.appendChild`.

```

1. import './club-item.js';
2.
3. class ClubList extends HTMLElement {
4.
5.     constructor() {
6.         super();
7.         this.shadowDOM = this.attachShadow({mode: "open"});
8.     }
9.
10.     set clubs(clubs) {
11.         this._clubs = clubs;
12.         this.render();
13.     }

```



```
14.
15.   renderError(message) {
16.     this.shadowDOM.innerHTML = "";
17.     this.shadowDOM.innerHTML += `

## 


```

Kemudian buka berkas *src -> styles -> clublist.css* dan pindahkan (*cut*) kode styling dengan selector *club-list > .placeholder*

```
1. club-list > .placeholder {
2.   font-weight: lighter;
3.   color: rgba(0,0,0,0.5);
4.   -webkit-user-select: none;
5.   -moz-user-select: none;
6.   -ms-user-select: none;
7.   user-select: none;
8. }
```

Lalu tempel (*paste*) pada nilai `this.shadowDOM.innerHTML` dengan dibungkus oleh element `<style>` tepat sebelum element `<h2>` fungsi `renderError()` di berkas *club-list.js* seperti ini:

```
1. import './club-item.js';
2.
3. class ClubList extends HTMLElement {
4.
5.   .....
6.
7.   renderError(message) {
8.     this.shadowDOM.innerHTML = `
9.     <style>
10.       club-list > .placeholder {
11.         font-weight: lighter;
12.         color: rgba(0,0,0,0.5);
13.         -webkit-user-select: none;
14.         -moz-user-select: none;
15.         -ms-user-select: none;
16.         user-select: none;
17.       }
18.     </style>
19.     `;
20.     this.shadowDOM.innerHTML += `

## 


```

Hapus *child selector* (*>*) beserta kombinatornya, sisakan `.placeholder` sebagai selector dari *styling* tersebut. Sehingga kode pada berkas ini seluruhnya tampak seperti:



```
1. import './club-item.js';
2.
3. class ClubList extends HTMLElement {
4.
5.   constructor() {
6.     super();
7.     this.shadowDOM = this.attachShadow({mode: "open"});
8.   }
9.
10.   set clubs(clubs) {
11.     this._clubs = clubs;
12.     this.render();
13.   }
```

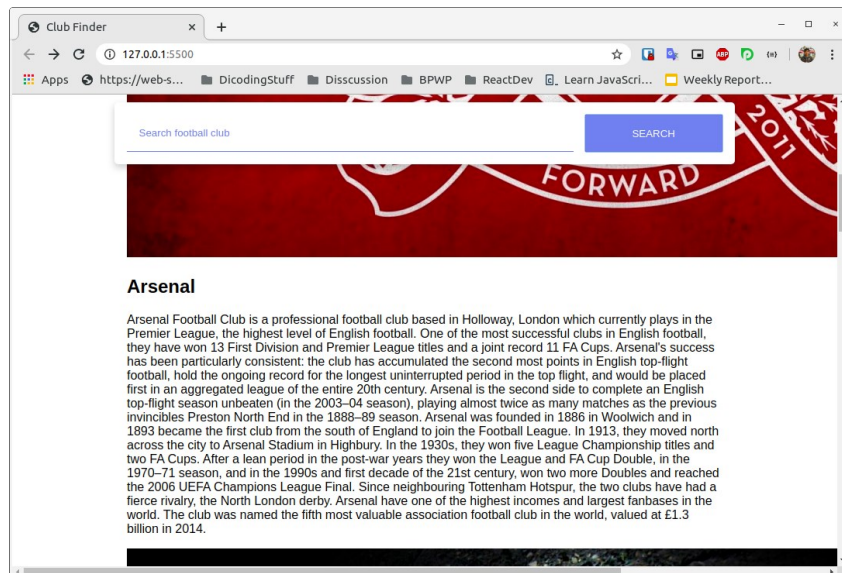


```

14.
15.   renderError(message) {
16.     this.shadowDOM.innerHTML = `
17.       <style>
18.         .placeholder {
19.           font-weight: lighter;
20.           color: rgba(0,0,0,0.5);
21.           -webkit-user-select: none;
22.           -moz-user-select: none;
23.           -ms-user-select: none;

```

Simpan perubahan tersebut dan lihat hasilnya pada browser, tampilan dari daftar club akan sangat berantakan.



Tenang kita akan memperbaikinya dengan beranjak ke berkas *src* -> *script* -> *component* -> *club-item.js*.

Pada berkas tersebut buat sebuah *constructor* dan terapkan Shadow DOM di dalamnya.

```

1.   class ClubItem extends HTMLElement {
2.
3.     constructor() {
4.       super();
5.       this.shadowDOM = this.attachShadow({mode: "open"});
6.     }
7.
8.     set club(club) {
9.       this._club = club;
10.      this.render();
11.    }
12.
13.    render() {
14.      this.innerHTML = `
15.        
17.        <div class="club-info">
18.          <h2>${this._club.name}</h2>
19.          <p>${this._club.description}</p>
20.        </div>`;
21.      }
22.    }

```

Seperti biasa jangan lupa untuk mengubah `this.innerHTML` menjadi `this.shadowDOM.innerHTML` ya.

```

1.   class ClubItem extends HTMLElement {
2.
3.     constructor() {

```

```

4.     super();
5.     this.shadowDOM = this.attachShadow({mode: "open"});
6. }
7.
8. set club(club) {
9.     this._club = club;
10.    this.render();
11. }
12.
13. render() {
14.    this.shadowDOM.innerHTML = `
15.        
17.        <div class="club-info">
18.            <h2>${this._club.name}</h2>
19.            <p>${this._club.description}</p>
20.        </div>`;
21.    }
22. }

```

Selanjutnya buka kembali berkas *src -> styles -> clublist.css* dan pindahkan *styling* berikut:

```

1. club-item {
2.     display: block;
3.     margin-bottom: 18px;
4.     box-shadow: 0 4px 8px 0 rgba(0, 0, 0, 0.2);
5.     border-radius: 10px;
6.     overflow: hidden;
7. }
8.
9. club-item .fan-art-club {
10.    width: 100%;
11.    max-height: 300px;
12.    object-fit: cover;
13.    object-position: center;
14. }
15.
16. .club-info {
17.    padding: 24px;
18. }
19.
20. .club-info > h2 {
21.    font-weight: lighter;
22. }
23.

```

Tempel pada nilai `this.shadowDOM.innerHTML` dengan dibungkus oleh element `<style>` tepat sebelum element `<img>` pada fungsi `render()` di berkas *club-item.js* seperti ini:

```

1. class ClubItem extends HTMLElement {
2.
3.     .....
4.
5.     render() {
6.         this.shadowDOM.innerHTML = `
7.             <style>
8.                 club-item {
9.                     display: block;
10.                    margin-bottom: 18px;
11.                    box-shadow: 0 4px 8px 0 rgba(0, 0, 0, 0.2);
12.                    border-radius: 10px;
13.                    overflow: hidden;
14.                }
15.
16.                club-item .fan-art-club {
17.                    width: 100%;
18.                    max-height: 300px;
19.                    object-fit: cover;

```

```

20.         object-position: center;
21.     }
22.
23.     .club-info {

```

Sesuaikan kembali selector pada *styling* tersebut menjadi seperti ini:

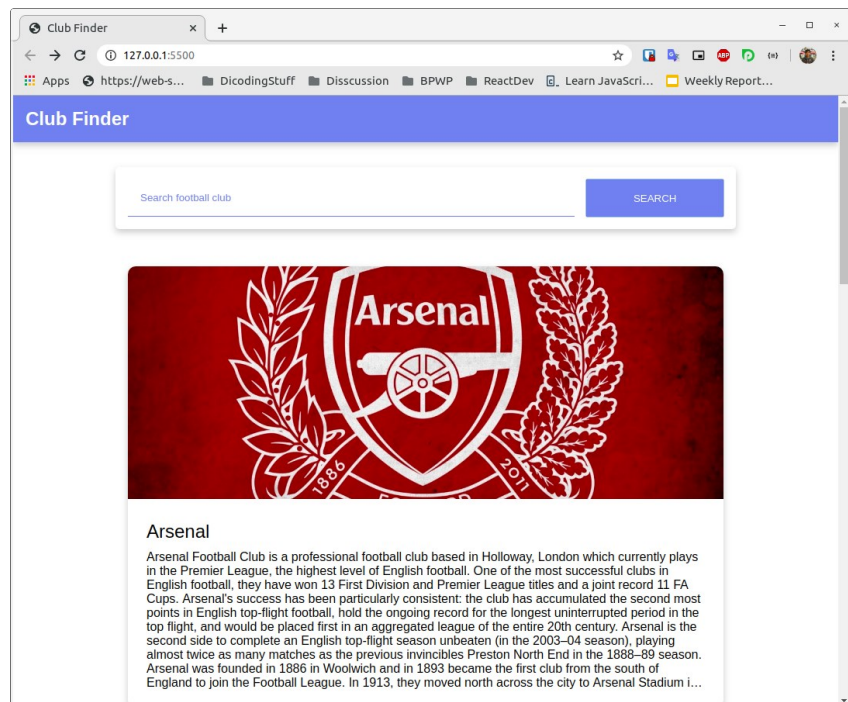


```

1.  class ClubItem extends HTMLElement {
2.
3.      .....
4.
5.      render() {
6.          this.shadowDOM.innerHTML = `
7.              <style>
8.                  * {
9.                      margin: 0;
10.                     padding: 0;
11.                     box-sizing: border-box;
12.                 }
13.                 :host {
14.                     display: block;
15.                     margin-bottom: 18px;
16.                     box-shadow: 0 4px 8px 0 rgba(0, 0, 0, 0.2);
17.                     border-radius: 10px;
18.                     overflow: hidden;
19.                 }
20.
21.                 .fan-art-club {
22.                     width: 100%;
23.                     max-height: 300px;

```

Simpan perubahan tersebut dan lihat pada browser, seharusnya tampilan daftar tim sudah kembali normal.



Oh ya, sebelum beranjak kita buka kembali berkas *src -> styles -> clublist.css*. Di sana masih terdapat satu rule styling berikut:

```

1.  club-list {
2.      display: block;
3.      margin-top: 32px;
4.      width: 100%;
5.      padding: 16px;
6.  }

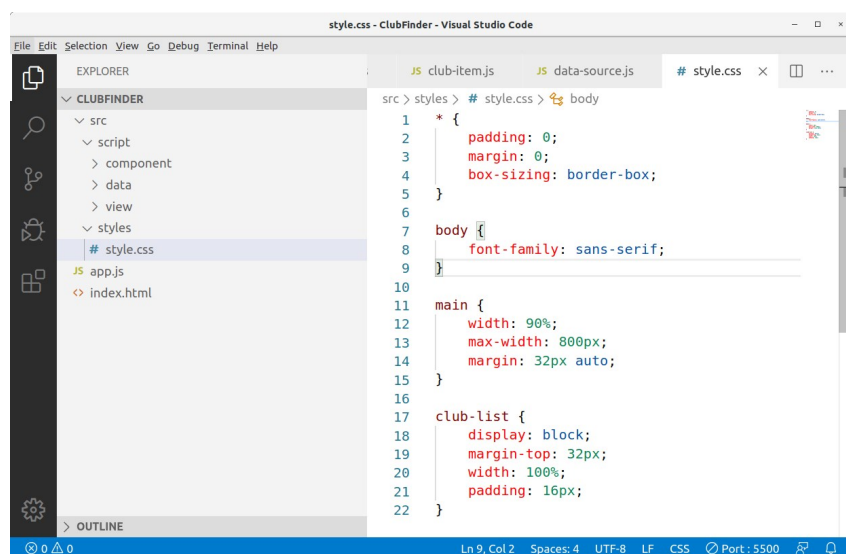
```

Jangan hapus rule styling tersebut karena kita masih menggunakannya untuk

mengatur jarak daftar liga yang ditampilkan. Namun sebaiknya kita pindahkan rule styling tersebut pada berkas *src -> styles -> style.css*.

```
1. @import "clublist.css";
2.
3. * {
4.   padding: 0;
5.   margin: 0;
6.   box-sizing: border-box;
7. }
8.
9. body {
10.  font-family: sans-serif;
11. }
12.
13. main {
14.  width: 90%;
15.  max-width: 800px;
16.  margin: 32px auto;
17. }
18.
19. club-list {
20.  display: block;
21.  margin-top: 32px;
22.  width: 100%;
23.  padding: 16px;
```

Dengan begitu kita dapat leluasa menghapus berkas *clublist.css* dan menghapus `@import` pada berkas *style.css*.



Selamat! Kita sudah berhasil menerapkan Shadow DOM pada seluruh custom element yang digunakan di proyek Club Finder. Sampai ketemu di materi selanjutnya ya!

Langkah dari solution ini bisa Anda temukan juga pada repository berikut:  
<https://github.com/dicodingacademy/a163-bfwd-labs/tree/110-club-finder-shadow-dom-solution>

← KEMBALI KE MATERI SEBELUMNYA

LANJUTKAN KE MATERI BERIKUTNYA →

image  
click bila  
belum muncul

image  
click bila  
belum muncul

#### PERUSAHAAN

[Tentang Kami](#)

[Blog](#)

[Berita Terbaru](#)



#### PROGRAM

[Academy](#)

[Challenge](#)

[Event](#)

[Job](#)

[Rewards](#)

#### SUPPORT

[Bantuan](#)

[FAQ](#)

[Hubungi Kami](#)

