|  |  |
| --- | --- |
| **Materi** | **Nilai** |
| Java Script 21-30 | 95 |

**Saya Sudah Belajar dan Mengerti dan Saya BISA**

1. Array methods 3

* let namaVariabel = NamaArray.map(function (a) { return a.namaElemen;}); digunakan untuk mengambil isi elemen didalam objek yang berada didalam array, dan mengembalikannya dalam bentuk array
* let siswa = nilai.map(a =>[a.nama,a.ipa,a.bahasa]) ; untuk mengeluarkan lebih dari satu elemen
* namaArray.sort(); digunakan untuk mengurutkan isi array
* let NamaVariabel = NamaArray.reduce(function (a sebagai penampung,b sebagai kolom yang ingin ditambahkan) { return a + b.ipa; },0 sebagai index); digunakan untuk menjumlahkan isi dari elemen ipa, jika menggunakan arrow function let NamaVariabel = nilai.reduce((a,b)=> (a+=b.bahasa),0);

2. Shopping cart

* Membuat array berisi object
* Gunakan map untuk mengambil isi objek kemudian return gunakan ` isi html `
* `<tagHTML>${parameter.namaElemen}</tagHTML>` digunakan untuk mengambil isi elemen didalam objek yang berada didalam array, dan mengembalikannya dalam bentuk array disertai tag html
* Gunakan querySelector seleksi class produk dan gunakan inner html untuk menampilkan isi function map

3. Shopping cart 2

* Gunakan wrap digunakan agar tidak melebihi lebar web
* Tambahkan button beli
* Didalam button beri data-id = “isi dengan idmenu”
* Seleksi class produk gunakan querySelector masukkan kedalam
* For sebanyak jumlah button
* Jika button di klik maka jalankan function
* Deklarikan variabel cart dengan array kosong
* cart.push(btnbeli[index].dataset["idmenu"]) untuk memasukkan Array yang diklik maka array yang mempunyai idmenu masuk kedalam array cart

4. Shopping cart filter

* tblmenu.filter(function (a) {
* if (a.idmenu==btnbeli[index].dataset["idmenu"]) {
* cart.push(a);
* console.log(cart);
* }
* }) digunakan untuk memfilter apakah idmenu sama dengan button dengan dataset idmenu jika ya maka masukkan kedalam array cart

5. Local Storage

* localStorage.setItem("key",value); untuk mengeset item ke localstorage
* localStorage.getItem("key") mengambil item dari localstorage
* localStorage.removeItem("key") untuk menghapus item di localstorage
* localstorage berfungsi untuk penyimpanan dan selalu ada di browser walaupun di restart kecuali local storage nya di hapus
* localStorage.clear(); untuk menghapus semua isi localstorage
* localStorage.key(1); digunakan untuk menampilkan key dari localstorage dengan index 1
* localStorage.length; digunakan untuk melihat jumlah localstorage dibuat

6. Session storage

* session storage itu ada hanya jika browser itu nyala
* sessionStorage.setItem("coba",100); untuk mengeset item ke sessionstorage
* sessionStorage.getItem(“key”); digunakan untuk menampilkan session
* sessionStorage.removeItem("coba"); untuk menghapus session
* sessionStorage.clear(); untuk menghapus semua session
* sessionStorage.length; untuk mengetahui jumlah session
* sessionStorage.key(2); digunakan untuk menampilkan key dari sessionstorage dengan index 2

7. JSON

* JSON adalah singakatan dari JavaScript Object Notation
* JSON digunakan untuk format data yang digunakan untuk pertukaran antar aplikasi
* Penulisan JSON mengikuti format javascript
* JSON mudah dibaca dan ditulis
* JSON sering digunakan pada AJAX
* JSON bisa digunakan pada berbagai Bahasa pemograman
* Tipe data pada JSON adalah string,number,Boolean,array,object,null
* Aturan penulisan JSON : Harus ditulis berpasangan antara KEY dan VALUE, Key harus diapit tanda petik uda (“KEY”),VALUE harus menunjukkan tipe data yang digunakan,Nama file mempunyai extensi .json
* Json item terakhir tidak boleh ada koma
* Jika berisi object maka kurung kurawa harus diapit dengan []
* tblmenu[0].menu untuk menampilkan value dengan key menu dan index 0

8. Fetch Data

* fetch('https://jsonplaceholder.typicode.com/todos/1') fetch akan mengambil url dan dibentuk menjadi json
* ganti url dengan tempat file json yang telah dibuat sendiri
* .then(response => response.json()) mengubah bentuk
* .then(json => console.log(json)) tampilkan
* Gunakan foreach dan innerHTML untuk menampilkan isi json ke html

9. Fetch Data API

* Mengambil data API menggunakan url

10. Async await

* Penggunaan async adalah menambahkan async didepan function
* Kegunaan async adalah menjalankan function lainya lebih dahulu walaupun ada dibawah
* await fetch(url) kegunaannya sama dengan fetch(url)
* await res.json kegunaannya sama dengan then(res=>res.json())
* network status digunakan apabila url salah maka ada pemberitahuan

**Saya Belum Mengerti**