**JAVASCRIPT LANJUTAN 4.3 Filter, Map & Reduce (Latihan)**

Kita akan menerapkan ke3 fungsi itu ke dalam kasus yg lebih real lagi. Sekarang saya sudah mempunyai halaman html, ceritanya saya punya sebuah halaman HTML yang didalam nya terdapat list nah list item nya ada banyak, ceritanya itu saya ngmbil data dari youtube padika jadi anggap aja itu data nya diambil menggunakan API gitu lah :

 <ul>

        <li *data-duration*="15:27">Teknik Pomodoro</li>

        <li *data-duration*="11:18">JAVASCRIPT LANJUTAN | Higher Order Functions</li>

        <li *data-duration*="21:40">JAVASCRIPT LANJUTAN | This pada Arrow Function</li>

        <li *data-duration*="19:38">Websaite Penipu</li>

        <li *data-duration*="12:10">JAVASCRIPT LANJUTAN | Arrow Function</li>

        <li *data-duration*="20:43">JAVASCRIPT LANJUTAN | Closure</li>

        <li *data-duration*="14:30">#TANYAPADIKA EP005</li>

        <li *data-duration*="26:38">JAVASCRIPT LANJUTAN | Execution Context</li>

        <li *data-duration*="17:33">JAVASCRIPT LANJUTAN | Prototype</li>

        <li *data-duration*="10:39">JAVASCRIPT LANJUTAN | Object.create()</li>

        <li *data-duration*="17:31">JAVASCRIPT LANJUTAN | Object (Revisited)</li>

        <li *data-duration*="14:25">5 Tips Bertanya Ketika Error</li>

    </ul>

Jadi saya udah dapet data nya dari API terus saya ingin tampil kan ke websaite saya, jadi ceritanya saya ngambil 10 video terakhir dari yt pakdika terus saya simpan kedalam web. Nah didalam list nya itu saya udah simpan durasi dari masing-masing video.

Nah sekarang saya ingin dibawah nya setelah saya tau 10 video terakhir saya ingin menyeleksi video yg merupakan bagian dari Playlist javascript lanjutan ceritanya gitu :

    <ol>

        <li>

            <h4>Javascript Lanjutan</h4>

            <p>Jumlah video: <span *class*="jumlah-video"></span></p>

            <p>Total Durasi: <span *class*="total-durasi"></span></p>

        </li>

    </ol>

Nah saya mau memilih video yg merupakan bagian javascript lanjutan terus saya mau tulis berapa jumlah video nya dan total durasi 1 playlist itu, kalo jumlah video an gampang ya tinggal itung aja. Tapi kalo yg durasi ini agak tricki nih kita harus pilih dulu mana video yg termasuk javascript lanjutan baru jumlahkan tiap-tiap menit nya sehingga nanti muncul total durasinya di HTML nya.

Nah yg akan kita fokuskan adalah kita bikin dulu yg total durasi karena nanti kalo jumalh video kaya nya ga terlalu susah, nah tapi kalian liat dulu bahwa durasi tiap-tiap video itu disimpan kedalam attribute data-duration, Jadi kita bisa bikin attribute sendiri di dalam HTML yg namanya data lalu strip lanjutin nama data nya apa misal yg kita itu duration ya atau bebas bisa. Terus nanti hasil pengolahan nya itu akan saya simpan kedalam element span.

**// MULAI DARI JAVASCRIPT**

Pertama kita tuliskan algoritmanya dulu atau langkah-langkah yg akan kita lakukan untuk program ini. Pertama ambil semua elemen video nya list nya itu lalu kita pilih hanya yg javascript lanjutan saja terus kita ambil durasi masing-masing video nah tapi yang kita ambil itu bentuknya string nah yg namanya string itu tidak bisa kita olah secara matematik ya.

Jadi kita harus ubah durasi menjadi integer lalu ktia ubah menit menjadi detik supaya nanti kita gampang kalo mau ngejumlahin nya, misalkan 2 menit 10 detik berartikan 2 dikali 60 ditambah 10 biar jadi detik semua. Nah baru kita jumlahkan semua detik nya nah nanti kita dapet tuh berapa total nya misal sampe ribuan detik nah baru setelah dapet detik ubah jadi menit atau ke jam kalo perlu gitu. Nah baru kalo udah dapet tinggal kita simpan di elemen HTML nya atau di DOM.

**// MENGERJAKAN NYA**

Yang pertama kita ambil DOM nya kita ambil semua elemen li itu jadi pake querySelectorAll, kalian bisa kasih selector didalam kurung nya (‘ul li’) gitu tapi nanti mungkin aja ga spesfik selector nya ya. Nah gimana kalo kita manfaatkan attribute data-duration yg ada di li nya ya jadi kalian bisa tulis kan jadi gini ;

const videos = document.querySelectorAll('[data-duration]');

Jadi cariin elemen yg didalam nya ada attribute yg namanya data-duration jadi kalo mau ngambil attribute buat di DOM nya harus pake kurung yg [] gitu, coba kita liat harusnya itu dapet semua li nya kita console aja :

const videos = document.querySelectorAll('[data-duration]');

console.log(videos);

Nah tapi kalian liat tu dia bentuk nya adalah NodeList bukan array jadi nantinya gabisa kita kasih map(), filter() dan reduce() karena 3 itu kan array ya. Nah kita harus mengubah dulu hasil dari document.querySelectorAll itu biar hasilnya ARRAY nah caranya kita tambahkan sebuah fungsi Array.from() kita bungkus menggunakan itu :

const videos = Array.from(document.querySelectorAll('[data-duration]'));

console.log(videos);

Nah itu artinya ubah array dari itu si document nya, sekarang bentuknya udah arrray kalo udah array aman sekarang kita bisa kasih Heghir Order Function. Nah sekaran kita akan pilih li hanya yg javascript lanjutan aja, berarti kita bikin variabel baru dulu.

Lalu kita akan fillter semua isi dari array videos itu dan kita cari yg didalam li nya ada tulisan javacript lanjutan, cara nya sama saja seperti filter() sebelum nya kali ini nama parameter nya saja yg berbeda kita sebut aja parameter nya video ya.

Nah untuk mengambil tulisan JAVASCRIPT LANJUTAN kita bisa pake beberapa cara bisa pake innerHTML, innerText atau textContent, gini :

const jsLanjut = videos.filter(*video* => *video*.innerHTML.includes('JAVASCRIPT LANJUTAN'));

console.log(jsLanjut);

Jadi kita bisa meletakan innerHTML seperti itu ataupun kita ganti jadi textContent seperti ini :

const jsLanjut = videos.filter(*video* => *video*.textContent.includes('JAVASCRIPT LANJUTAN'));

console.log(jsLanjut);

Nah itu sama saja tapi kita harus tambahkan satu method lagi yaitu includes jadi maksud includes itu terdapat apa didalam nya jadi bukan sama dengan kalo kita pake sama dengan itu gabisa, jadi jika didalam nya terdapat tulisan JAVASCRIPT LANJUTAN maka filter si parameter video itu dan masukan ke array baru.

Nah jadi hasilnya video JAVASCRIPT LANJUTAN itu ada 8 video, terus selanjutnya kita akan ambil masing-masing durasi dari video, kita akan ambil angka yg ada di attribute data-duration di HTML nya. Nah dijavascript itu kalian bisa panggil method yg nama nya dataset untuk mengambil attribute data di HTML itu, kan data-namabebas nah kita mau ambil data yg namanya duration yg data-duration gitu diantara nama data yg lain misalnya.

Nah biar gampang kita chaining aja berarti kita apus titik koma nya ( ; ) kita lanjutkan pake titik buat map() nya, kita pettakan tiap-tiap element javascript tadi hanya mengembalikan duration nya saja jadi return nya bisa gini :

const jsLanjut = videos.filter(*video* => *video*.textContent.includes('JAVASCRIPT LANJUTAN'))

*// ambil durasi masing" video*

    .map(*item* => *item*.dataset.duration);

console.log(jsLanjut);

Nah jadi return nya itu item.dataset nah duration itu nama dari si attribute data nya, kan di HTML nya itu namanya data-duration nah itu bisa ganti misalnya jadi data-statick kalo diganti jadi statick kita ganti di map() nya juga jangan duration ganti jadi statick gitu.

Nah sekarang kita sudah mendapatkan durasinya, tapi durasinya masih berbentuk string berarti kita akan chaining lagi kita ubah durasinya jadi integer terus kita ubah menit menjadi detik ya. Berarti kita map() lagi ya kita petakkan nah ini mereturn cukup panjang.

Nah kalian inget-inget dulu yg namanya durasi itukan nulis nya gini ( 10:30 ) misal ada ini ( : ) di tengah” nya ya, nah kan kita gabutuh ya, nah gimana kalo kita pecah menjadi array jadinya gini [10, 30] nah itu caranya di **split** namanya.

Caranya jadi kita bisa bikin aja variabel lagi yg isi nya array, nah kita akan ngesplit si waktu itu dengan delimeternya atau pembatas nya si titik dua ( : ), gini :

const videos = Array.from(document.querySelectorAll('[data-duration]'));

*// pilih hanya yang 'JAVASCRIPT LANJUTAN'*

const jsLanjut = videos.filter(*video* => *video*.textContent.includes('JAVASCRIPT LANJUTAN'))

*// ambil durasi masing" video*

    .map(*item* => *item*.dataset.duration)

    .map(*waktu* => {

*// 10:30 -> [10, 30]*

        const parts = *waktu*.split(':');

        return parts;

    });

Nah setelah di console si jsLanjut nya hasilnya itu array yg masih berbentuk string dan sudah hilang titik 2 nya ( : ), jadi tinggal kita kalkulasi element nya kan element pertama nya itu misal tadi [10, 30] nah itu sebelum nya 10 menit 30 detik ya jadinya element pertama array nya kita kalikan 60detik biar jadi detik ya lalu ditambahkan detik nya yg 30 itu.

Berarti gampang nya gini tinggal kalikan parts ke 0 atau array index ke-0 dengan 60 dan ditambah parts ke-1 atau parts yg index nya 1, gini :

const videos = Array.from(document.querySelectorAll('[data-duration]'));

*// pilih hanya yang 'JAVASCRIPT LANJUTAN'*

const jsLanjut = videos.filter(*video* => *video*.textContent.includes('JAVASCRIPT LANJUTAN'))

*// ambil durasi masing" video*

    .map(*item* => *item*.dataset.duration)

    .map(*waktu* => {

*// 10:30 -> [10, 30]*

        const parts = *waktu*.split(':');

        return (parts[0] \* 60) + parts[1];

    });

Nah itu langsung ke total jadi detik, tapi karena itu masih berbentuk string jadi hasil nya masih ngaco belum sesuai, berarti kita harus ubah dulu dia jadi angka jadi integer atau float kalo float kan pecahan, caranya kita map disitunya tiap” element nya anggap aja kita namain variabel didalam map itu part ya.

Jadi tiap-tiap part nya kita ubah jadi float di javascript itu ada yg namanya parseFloat yg di parse si part nya gini :

        const parts = *waktu*.split(':').map(*part* => parseFloat(*part*));

jadi pake map nya disitu ya kan sama-sama array inget. Nah udah jadi sekarang bentuknya udah angka bukan string lagi. Nah tinggal jumlahin semuanya kita chaining lagi, nah jumlahin nya kita pake reduce(), gini :

const videos = Array.from(document.querySelectorAll('[data-duration]'));

const jsLanjut = videos.filter(*video* => *video*.textContent.includes('JAVASCRIPT LANJUTAN')).map(*item* => *item*.dataset.duration)

.map(*waktu* => {

*// 10:30 -> [10, 30]*

        const parts = *waktu*.split(':').map(*part* => parseFloat(*part*));

        return (parts[0] \* 60) + parts[1];

    }).reduce((*total*, *detik*) => *total* + *detik*);

Nah hasil seluruh total detik nya 8292, selanjut nya kita tinggal ubah menjadi jam detik menit, misalkan saya punya variabel jam nah gampang nya gini tadikan kita udah punya totalan detik nah biar jadi nya kita harus bagi ( 60 \* 60 ) artinya pertama dibagi 60 detik hasilnya menit kedua hasil menit dibagi 60menit berarti hasilnya jam jadi kita bisa bagi 8292 dibagi 3600 karena ( 60 dikali 60 = 3600).

Nah hasilnya 2.3033333333333332 tapi kita kan gamau ambil koma nya ya jadi kita mau ambil pembulatan kebawah pake fungsi Math.floor, gini :

const jam = Math.floor(jsLanjut / 3600);

Nah udah ada hasilnya jadi 2jam ya, selanjutnya kita akan hitung menit tapi kita harus ambil dari sisa jam nya dulu, sisa bagi nya maksud sisa bagi disini itu kita harus berfikir kan jam hasil nya 2jam itu kita anggap ubah 2 jam itu berapa detik sih, nah 7200detik kan, Nah jsLanjut kan 8292 kita kurangin 2jam itu jadinya (8292detik – 7200detik) nah ini yg dimaksud sisa ini nya yg kita ambil ya.

Jadi nya kita harus timpa dulu dengan jsLanjut – jam \* 3600 kan ya, jadi nya artinya 8292 – 7200 ya, Lalu kita bagi hasil dari 8292 – 7200 itu 1092 nah 1092 dibagi 60 jadi 18, nah 18 itu menit nya, Karena 1092detik dibagi 60detik jadinya menit ya gays, gini :

const jam = Math.floor(jsLanjut / 3600);

jsLanjut = jsLanjut - jam \* 3600;

const menit = Math.floor(jsLanjut / 60);

Nah yg tadinya jsLanjut itu const ganti jadi let ya karena jsLanjut nya ditimpa gitu gabisa pake const. tinggal detik nya nih bikin variabel detik lalu isi nya gini :

const jam = Math.floor(jsLanjut / 3600);

jsLanjut = jsLanjut - jam \* 3600;

const menit = Math.floor(jsLanjut / 60);

const detik = jsLanjut - menit \* 60;

Nah artinya itu jsLanjut nya kan isi nya 1092DETIK nah lalu menit dikali 60 itu kan artinya (18 x 60) yaitu 1080 nah berarti 1092detik – 1080detik hasilnya 12detik gitu :

const videos = Array.from(document.querySelectorAll('[data-duration]'));

*// pilih hanya yang 'JAVASCRIPT LANJUTAN'*

let jsLanjut = videos.filter(*video* => *video*.textContent.includes('JAVASCRIPT LANJUTAN'))

*// ambil durasi masing" video*

    .map(*item* => *item*.dataset.duration)

*// ubah durasi menjadi integer, ubah menit menjadi detik*

    .map(*waktu* => {

*// 10:30 -> [10, 30]*

        const parts = *waktu*.split(':').map(*part* => parseFloat(*part*));

        return (parts[0] \* 60) + parts[1];

    })

*// jumlahkan semua detik*

    .reduce((*total*, *detik*) => *total* + *detik*);

*// ubah formatnya jadi jam meut detik*

const jam = Math.floor(jsLanjut / 3600);

jsLanjut = jsLanjut - jam \* 3600;

const menit = Math.floor(jsLanjut / 60);

const detik = jsLanjut - menit \* 60;

Nah tinggal terakhir kita simpan di DOM, gini :

const durasi = document.querySelector('.total-durasi');

durasi.textContent = `${jam} jam ${menit} menit ${detik} detik`;

const jmlVideo = videos.filter(*video* => *video*.textContent.includes('JAVASCRIPT LANJUTAN')).length;

const totalVideo = document.querySelector('.jumlah-video');

totalVideo.innerHTML = `${jmlVideo} Video.`;

Nah itu yg variabel jmlVideo dia mengambil jumlah video playlist javascript lanjutan aja dan itu kita copy dari atas lalu tambahkan titik length supaya mengambil length array nya.