**JAVASCRIPT LANJUTAN | 5.1 Template Literals**

Template Literal atau Template String, divideo sebelum-sebelum nya kita juga udah pernah menggunakan konsep itu walaupun disebelum nya pak dhika salah menyebutkan malah nyebut nya string Literal ya.

Nah padahal yg kita lakukan adalah Template Literal atau Template String nah untuk lebih jelasnya kita liat definisinya dulu ya kalo penjelasan nya di MDN Web Docs *“Template literal adalah string literal yang memungkinkan adanya expression di dalamnya.”* Nah ada string literal jadi apa itu string literal dan maksudnya expression itu apa.

// STRING LITERAL

String literal adalah string yg biasa kita buat baik itu kita simpan kedalam sebuah variabel atau kita cetak misalnya menggunakan console.log(), jadi string literal biasanya itu kita buat menggunakan beberapa cara.

        let nama = 'Sandhika';

        let nrp = "043040023";

        let email = `sandhikagalih@unpas.ac.id`;

yg pertama bisa menggunakan kutip satu atau kita juga bisa menggunakan kutip 2 dan fungsi kutip 1 dan kutip 2 itu sama persis tidak ada bedanya, beda dengan bahasa PHP misalnya string 2 itu sedikit lebih sakti kalo di PHP karena dia bisa melakukan interpolasi, nah kalo di javascript kutip satu dan kutip 2 sama saja.

Nah ada cara ke-3 untuk membuat string Literal itu dengan menggunakan BACK TICK ( `` ) kharakter yg ada di sebelah angka satu sebelah kiri nya, nah dengan membuat string literal dengan back tick itu memungkin kan kita untuk membuat Template Literal.

**// TEMPLATE LITERAL**

Jadi Template Literal itu hanya bisa dibuat menggunakan back tick ( `` ) nah dengan menggunakan back tick itu kita jadi bisa melakukan banyak hal seperti melakukan Multi-line String yg tidak bisa dilakukan kalo kita pake kutip satu atau kutip dua.

Terus kita juga bisa melakukan yg disebut Embedded Expression, bisa juga membuat HTML Fragmenst nah itu bisa digunakan pada saat kalian menggunakan front end framwork misalnya, selanjutnya memungkinkan kita melakukan Expression Interpolation nah itu sama seperti kutip 2 pada bahasa PHP, lalu kita juga bisa melakukan Tagged Template, contoh penggunaan Template Literal :

`string text`

        `string text baris 1

         string text baris 2 *<- (Multi-line string)*

         string text baris 3`

         `string text ${expression} string text` *<- embedded expression*

         tag `string text ${expression} string text` *<- Tagged template*

Jadi kita bisa menuliskan string biasa seperti yg di contoh, lalu yg ke-2 kita bisa melakukan Multi-line string jadi kita bisa menambahkan enter atau new line atau baris baru tanap harus menutup dulu string nyya terus melakukan konkatenasi dengan menambahkan tanda plus ( + ) itu gaperlu kalo pake back tick.

Terus kita juga bisa memasukan expression kedalam string nya atau istilahnya embedded expression dengan cara menambahkan tanda dolar ${variabel}, dan yg terakhir kita juga bsia membuat tagged template dengan menuliskan tag nya didepan template literal.

**// Multi-line string**

Misalnya kalo kalian menggunakan kutip satu atau kutip 2, Terus misalkan kita mau mencetak string menggunakan console.log lalu string nya akan di tampilkan dibaris terpisah didalam console nya itu kalian harus menggunakan Escape kharakter ( \n ) untuk menambahkan baris baru seperti ini :

        console.log('string 1\nstring 2');

Sedang kan jika menggunakan back tick atau template literal itu cukup tambahkan enter aja seperti menulis di notpad maka hasil nya akan sama saja :

console.log(`string 1

string 2`);

Nah itu disesuaikan dengan kebutuhan misalkan satu enter atau 2 enter maka nanti otomatis string nya tampilnya sama seperti kalian menambahkan ( \n ) jadi gampang ga perlu nambahin escape kharakter lagi.

**// Multi-line String (HTML Fragments) (1)**

Nah multi-line string ini bisa kita gunakan juga ketika kita mau bikin HTML Fragments, misalnya saya punya object mhs yg didalam nya terdapat beberapa property ada nama, umur, nrp, dan email. Gini :

const mhs = {

            nama: 'Sandhika Galih',

            umur: 33,

            nrp: '043040023',

            email: 'sandhikagalih@unpas.ac.id'

        };

Lalu kita ingin menampilkan data nya ke halaman html kita, Nah kalo sebelum nya itu kita rangkai element html nya seperti ini :

let el = '';

        el += '<div class="mhs">';

        el += '<h2>' + mhs.nama + '</h2>';

        el += '<span class="nrp">' + mhs.nrp + '</span>';

        el += '</div>';

        console.log(el);

Misalnya saya punya variabel namanya el awal nya string kosong lalu kita rangkai kita kotkatkanasi dengan element yg kita butuhin ya. Lalu element nya bisa kita simpan kedalam DOM misalkan menggunakan innerHTML.

**// Multi-line String (HTML Fragments) (2)**

Nah etika kalian ingin menggunakan Template literal kalian cukup bikin nya variabel nya satu seperti ini :

let el = `<div class="mhs">

                    <h2>${mhs.nama}</h2>

                    <span class="nrp">${mhs.nrp}</span>

                  </div>`;

        console.log(el);

Jadi kalian hanya cukup satu kali tulis saja jadi gausah di gabung’ kan gitu. Hasil nya sama walaupun nanti didalam html nya terdapat waits space ya karena waktu kalian bikin element nya pake tab begitu ya tapi itu tidak akan dipedulikan oleh browser nya.

**// Embedded Expression (1)**

Jadi kita bisa menyimpan expression kedalam template literal jadi misalnya tadi kita coba hanya mengisikan variabel kalian cukup tambahkan dolar ${variabel nya}, gini :

const nama = 'Sandhika Galih';

        let umur = 33;

        console.log(`Halo, perkenalkan nama saya ${nama}, saya ${umur}

tahun.`);

Nah tapi expresi itu juga bisa kita langsung interpolasi misalnya didalam expresi nya terdapat penjumlahan, sebelum nya kalo kalian pake kutip satu atau kutip dua itu kita hrus buka tutup kutipnya gini :

let a = 10;

        let b = 15;

        console.log('Jika a = 10 dan b = 15, maka hasil penjumlahannya adalah

: ' + (a + b) + ', bukan ' + (2 \* a + b));

Nah kalo kita tulis tanda tambah didalam kutip nya itu akan di cetak sebagai string bukan operasi matematik jadi kita tutup dulu string nya baru kasih tanda tambah dan gitu lagi, ingat peran tanda tamabah disitu peran nya ada yg beda ada konket ada operasi matematik tergantung operatornya.

**// Embedded Expression (2)**

Tapi jika kalian menggunakan template literal cukup gunakan back tick lalu didalam dolar kurung kurawal nya itu sudah melakukan interpolasi expressi nya jadi langsung dikerjakan, gini :

        let a = 10;

        let b = 15;

        console.log(`Jika a = 10 dan b = 15, maka hasil penjumlahannya adalah:

${a + b}, bukan ${2 \* a + b}`);

Atau kalo kalian mau lebih jelas nyobain nya kita langsung coding aja ya. Coba kalian implementasikan kode yg tadi ke codingan, karena bagian template Literals itu gini aja :

**// Template Literals / Template String**

const nama = 'Sandhika Galih';

        let umur = 33;

        console.log(`Halo, perkenalkan nama saya ${nama}, saya ${umur}

tahun.`);

**// Embedded Expressions**

Kalian itu bisa langsung menggunakan dolar tadi itu kita bisa langsung melakukan expressi, expressi itu bisa sesimpel penjumlahan seperti ini :

console.log(`${1 + 1}`);

Nah kalo gitu jadi nya string hasil nya bukan integer. Atau kalian bisa juga memanggil function misalkan alert() itukan termasuk function ya gini :

console.log(`${alert('hallo!')}`);

Nah itu bisa juga karena function merupakan expression juga, atau selanjutnya kalian bisa menyimpan ternary operator, misalnya sebelum di console saya punya variabel x saya isi 10, misalkan saya mau mencetak apakah bilangan x itu genap atau ganjil, kita bisa lakukan pengkondisian sesimpel nulis (if) atau kita bisa simpan ternary operator didalam expression.

Karena ternary pun adalah expression jadi saya gunakan sintaks expression didalam template literal nya, gini :

const x = 10;

console.log(`${(x % 2 == 0) ? 'genap' : 'ganjil'}`);

jika hasilnya true maka tampilkan genap jika false tampilkan ganjil gitu, tapi ternary operator ini true nya itu di kiri jadi kalo tulisan ‘genap’ saya ganti ‘number’ pun yg tampil akan ‘number’ gitu selama hasilnya true yg ditampilkan string yg kiri.

**// HTML Fragments**

Misalnya saya punya sebuah object tadi ya, simpel aja gini :

const mhs = {

            nama: 'Sandhika Galih',

            umur: 33,

            nrp: '043040023',

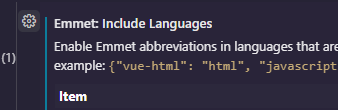
            email: 'sandhikagalih@unpas.ac.id'

        };

Nah simpel aja gitu, nah itu kita bisa simpan kedalam html, kalo pake back tick itu kalian cukup tuliskan aja html nya didalam back tick nya gaperlu di rangkai menggunakan tanda plus (+) nah tapi masalahnya kita gabisa menuliskan HTML menggunakan emmet kalo kalian pake Vscode.

Misalnya mau bikin div class nya mahasiswa itu kalo di HTML nya langsung bisa tulis ( div.mhs ) terus tab jadi deh element nya, jadi kalian harus tulis manual kalo simpel sih gamasalah tapi kalo html nya kompleks ribet juga ya.

Nah itu kita harus jalankan sebuah fitur di Vscode ya, kalian bisa pencet ctrl+koma ( , ) nah kita pengen buat supaya tag html itu emmet nya jalan di javascript, kalian cari saja includelang, terus itu nanti kalian copy kedalam JSON nya. Cara gampang nya klik tanda stting disitinya deket nama includelang nya lalu klik copy setting as JSON disitu setting nya ga keliatan ada jadi harus di hover gitu.



Nah setelah di klik kalian cari bacaan edit in setting json lalu klik itu nah nanti akan diarahkan settingan json kalian, pokoknya kalian tinggal tambahkan saja tanda koma karena disitu object kan harus pake koma ya kalo mau membuat property baru nah baru kalian copy yg tadi tambahkan ini pas bagain edit json nya :

"emmet.includeLanguages": {

    "javascript": "html"

   }

Nah sekarang kalo didalam tempalte literal nya atau di javascript nya saya tulis div.mhs lalu tab maka dia ngaruh otomatis, kita tulis gini :

const el = `<div class="mhs">

<h2>${mhs.nama}</h2>

<span class="nrp">${mhs.nrp}</span>

</div>`;

Nah lalu cerita nya itu tinggal kalian simpan ke DOM atau kita console aja, nah nanti disitu akan ada waitspace tapi kalian gausah khawatirin karena akan di abaikan oleh browsernya.