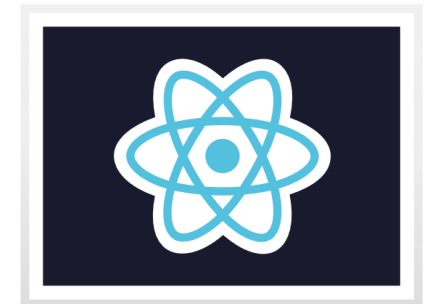
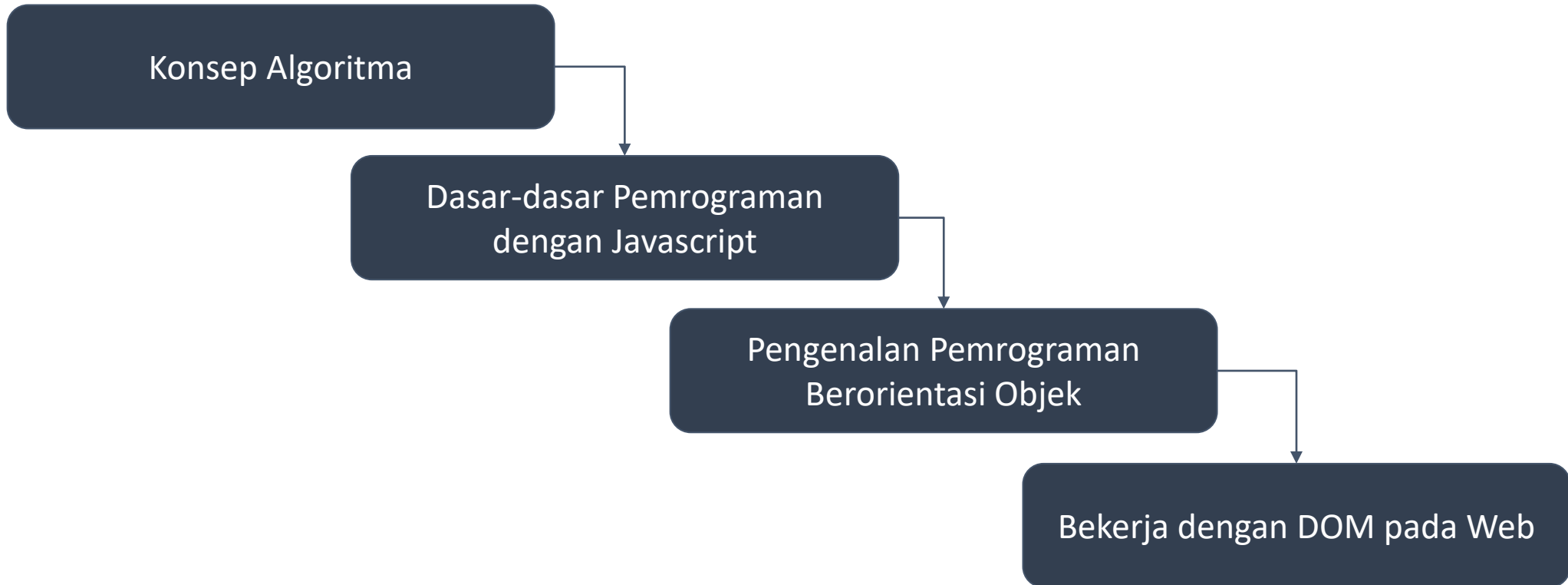


PEMROGRAMAN JAVASCRIPT



- **KODE** : MK03
- **MATAKULIAH** : PEMROGRAMAN JAVASCRIPT
- **SKS** : 4 SKS
- **SEMESTER** : 1 (SATU)



Tidak ada gunanya IQ Anda tinggi namun malas, tidak memiliki disiplin. yang penting adalah Anda sehat dan mau berkorban untuk masa depan yang cerah.”

- BJ. Habibie -

FORM JAVASCRIPT

FORM OBJECT

Form Object

Objek form digunakan untuk membuat form HTML untuk user input.

Form dan elemen control (seperti `<input>`) memiliki banyak properti dan event khusus.

Form adalah bagian dari sebuah koleksi khusus (named collection)

`document.forms`

First name:

Last name:

Submit

Type

`<input type="text">`

`<input type="radio">`

`<input type="checkbox">`

`<input type="submit">`

`<input type="button">`

Form Object

☐ Radio Button



☐ Checkbox



NAMED COLLECTION

```
form name="myform">  
  <input name="elem_one" value="1">  
  <input name="elem_two" value="2">  
</form>
```

Form dengan nama myform

Elemen form dengan nama elem_one dan elem_two

```
<script>  
  let form = document.forms.myform;  
  let elem = form.elements.elem_one;  
  alert(elem.value);  
</script>
```

Mengakses form dengan nama myform

Mengakses elemen form
Alternatif : form.<namaform>

Hasil cetak dari valuenya
akan menampilkan 1
(nilai elem_one)

CONTOH FORM

```
<!DOCTYPE html>
<html>
<body>

<p>Enter names in the fields, then click "Submit" to submit the form:</p>

<form id="frm1">
  First name: <input type="text" name="fname"><br>
  Last name: <input type="text" name="lname"><br><br>
  <input type="button" onclick="myFunction()" value="Submit">
</form>

<script>
function myFunction() {
  document.getElementById("frm1").submit();
}
</script>

</body>
</html>
```

Enter names in the fields, then click "Submit" to submit the form:

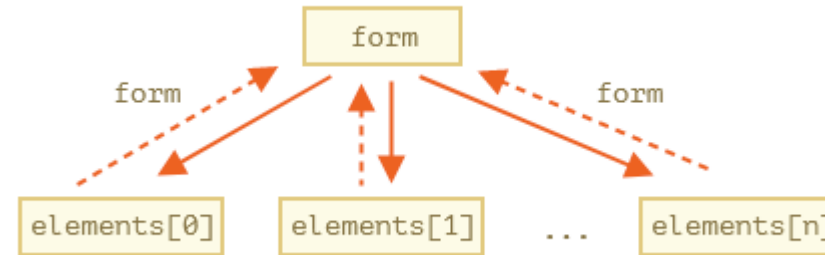
First name:
Last name:

Submit

BACK REFERENCE

Kita bisa mengakses form dari elemen di dalamnya dengan menggunakan atribut form pada elemen

`element.form`



Enter names in the fields, then click "Submit" to submit the form:

First name:

Last name:

Elements Console Sources Network

top Filter

```
> form = document.forms[0]
< ▶ <form id="frm1">...</form>

> firstname = form.elements[0]
< <input type="text" name="fname">

> lastname = form.lname
< <input type="text" name="lname">

> firstname.form
< ▶ <form id="frm1">...</form>

>
```

FORM CONTROL

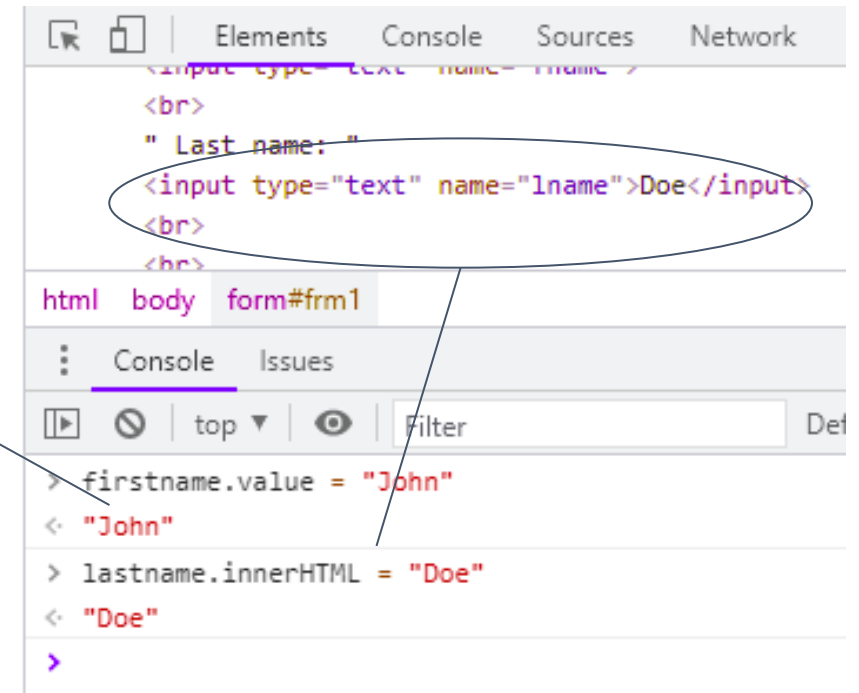
Untuk mengakses text area
dari form, gunakan value,
bukan innerHTML

Enter names in the fields, then click
"Submit" to submit the form:

First name: John

Last name:

Submit



The screenshot shows the browser's developer tools. The 'Elements' panel displays the HTML structure of the form, with the text input field for the last name highlighted. The 'Console' panel shows the output of JavaScript code, indicating that the first name is 'John' and the last name's innerHTML is 'Doe'.

```
<input type="text" name="lname">Doe</input>
```

```
> firstname.value = "John"
< "John"
> lastname.innerHTML = "Doe"
< "Doe"
>
```

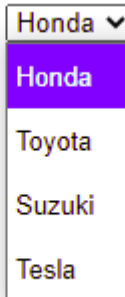
Document Object Model

DOM level 1 mendefinisikan 11 objek HTML, koleksi, dan properti (tahun 1998). Masih valid sampai HTML5 yang digunakan sekarang. Pada DOM level 3, ada beberapa objek, koleksi, dan properti yang ditambahkan

```
<label for="cars">Pilih sebuah mobil:</label>
```

```
<select name="cars" id="cars">  
  <option value="honda">Honda</option>  
  <option value="toyota">Toyota</option>  
  <option value="suzuki">Suzuki</option>  
  <option value="tesla">Tesla</option>  
</select>
```

```
<script>  
  select.options[3].selected = true;  
  select.selectedIndex = 3;  
  select.value = 'tesla';  
</script>
```

Pilih sebuah mobil: 

Memilih Tesla

Event Focus

- Elemen mendapat focus pada saat user mengklik elemen tersebut atau menggunakan tombol tab di keyboard. Focus pada form dapat diterjemahkan sebagai “elemen telah siap menerima inputan data”.
- Ada juga atribut autofocus pada HTML yang menyebabkan focus diberikan pada elemen tersebut setelah halaman selesai dibuka.
- Kehilangan focus (blur) terjadi saat user mengklik elemen lain atau menekan tab. Hal ini bisa diterjemahkan sebagai “data sudah dimasukkan”.

Example

Misalnya form kita ingin mengecek apakah user sudah memasukkan email dengan format yang tepat atau tidak.

blur dapat kita manfaatkan untuk mengecek fieldnya dan menampilkan pesan error jika belum diisi

focus dapat kita manfaatkan untuk menghilangkan pesan errornya

DOM JAVASCRIPT

Masukkan email :

Please enter a correct email.

```
<style>
  .invalid { border-color: red; }
  #error { color: red }
</style>
```

Masukkan email :

```
<div id="error"></div>
```

```
<script>
input.onblur = function() {
  if (!input.value.includes('@')) {
    // jika emailnya tidak sesuai format (harus mengandung @)
    input.classList.add('invalid');
    error.innerHTML = 'Please enter a correct email.'
  }
};

input.onfocus = function() {
  if (this.classList.contains('invalid')) {
    // hapus error agar bisa diinput kembali
    this.classList.remove('invalid');
    error.innerHTML = "";
  }
};
</script>
```

Elemen yang difungsikan untuk berinteraksi dengan user biasanya dapat dikenakan focus/blur, seperti `<button>`, `<input>`, `<select>`, `<a>`, dsb.

Tetapi elemen untuk memformat sesuatu seperti `<div>`, ``, atau `<table>` secara default tidak bisa difokuskan.

Dengan menambahkan atribut `tabindex`, maka elemen tersebut bisa menerima focus.

Tab index akan membuat browser mengarahkan fokus ke elemen yang indexnya di atas fokus saat ini jika tombol tab ditekan.


```
<ul>  
  <li tabindex="0">Elemen pertama</li>  
  <li tabindex="1">Elemen kedua</li>  
  <li tabindex="2">Elemen ketiga</li>  
</ul>
```

-
- Elemen pertama
 - Elemen kedua
 - Elemen ketiga

tabindex 0 akan berada di bagian akhir (setelah yang tabindex>0 selesai). Digunakan untuk membuat sebuah elemen dapat menerima focus tetapi tidak mengubah urutan indeksinya.

Elemen kedua (tab) -> Elemen ketiga (tab) -> Elemen pertama

Tab selanjutnya dapat memindahkan fokus keluar dari halaman (pindah ke browsernya)

Data Update

- Beberapa event terkait data update
- onchange = terpicu saat elemen selesai diubah
- oninput = terpicu setiap kali user memodifikasi nilainya

```
1 <input type="text" onchange="alert(this.value)">
2 <input type="button" value="Button">
```

```
1 <input type="text" id="input" oninput: <span id="result"></span>
2 <script>
3   input.oninput = function() {
4     result.innerHTML = input.value;
5   };
6 </script>
```

 oninput:

Clipboard Event

oncut, oncopy, onpaste dapat digunakan untuk mengatur akses data yang dicut/copy/paste oleh user.

Misalnya saat dicopy, kita ingin mengacak hurufnya agar tidak bisa diplagiat. Atau bahkan dicopy satu elemen, bukan hanya value textnya saja yang kecopy, tapi seluruh strukturnya juga. Atau mengganti respon cut/copy/paste ke fungsi lain seperti contoh berikut.

Clipboard Event

```
1 <input type="text" id="input">
2 <script>
3   input.oncut = input.oncopy = input.onpaste = function(event) {
4     alert(event.type + ' - ' + event.clipboardData.getData('text/plain'));
5     return false;
6   };
7 </script>
```

EVENT METHOD SUBMIT

```
<!DOCTYPE html>
<html>
<body>

<p>Enter names in the fields, then click "Submit" to submit the form:</p>

<form id="frm1">
  First name: <input type="text" name="fname"><br>
  Last name: <input type="text" name="lname"><br><br>
  <input type="button" onclick="myFunction()" value="Submit">
</form>

<script>
function myFunction() {
  document.getElementById("frm1").submit();
}
</script>

</body>
</html>
```

/form.html?fname=&lname=

Inputan disubmit menjadi parameter di urlnya

Enter names in the fields, then click "Submit" to submit the form:

First name:
Last name:

Submit

SIGN FORM

```
<button onclick="myFunction()">Submit</button>
```

```
<p id="demo"></p>
```

```
<script>
```

```
function myFunction() {  
  var x = document.getElementById("frm1");  
  var text = "";  
  var i;  
  for (i = 0; i < x.length ;i++) {  
    text += x.elements[i].value + "<br>";  
  }  
  document.getElementById("demo").innerHTML = text;  
}
```

```
</script>
```

Enter names in the fields, then click "Submit" to submit the form:

First name:

Last name:

Submit

FORM EVENT

Ada 2 cara yang dapat dilakukan untuk mensubmit form :

1. Mengklik `<input type="submit">` atau `<input type="image">`
2. Menekan enter di input field

Keduanya akan memicu submit event.

Untuk mensubmit form secara manual, dapat digunakan fungsi `form.submit()` .

Note : submit event tidak terpicu saat `form.submit()` dipanggil karena diasumsikan semua proses submit sudah terwakili oleh fungsi tersebut.

```
1 let form = document.createElement('form');
2 form.action = 'https://google.com/search';
3 form.method = 'GET';
4
5 form.innerHTML = '<input name="q" value="test">';
6
7 // the form must be in the document to submit it
8 document.body.append(form);
9
10 form.submit();
```


TUGAS JAVASCRIPT

1. Buatlah program Pendaftaran Mahasantri
2. Gunakan Teknik DOM untuk menampilkan hasil inputan
3. Simpan dalam file index.js
4. Upload pada akun github kalian dengan nama repository “Pertemuan-12-JS”



**Terima Kasih
Syukron Katsiron
Arigatou Gozaimasu**