

VALIDASI FORM

Data dari form isian di halaman web akan dikirim ke database di server untuk diproses lebih lanjut. Guna menghindari adanya kesalahan informasi maka data harus divalidasi terlebih dahulu. Contohnya isian data field tidak boleh kosong, memastikan data yang diisi adalah angka untuk isian jumlah belanja

Contoh pada kasus nyata , website pendaftaran sebuah kegiatan seminar karena tidak ada validasi terjadi kekurangan data no Hp dan email tidak dicantumkan. Lalu terjadi masalah dalam hal pembayaran maka pihak panitia akan kesulitan menghubungi peserta. Jadi proses validasi merupakan proses yang sangat penting untuk menghindari kekacauan di proses berikutnya.

Proses validasi Form web dapat dilakukan disisi server ataupun client(browser). Pembahasan disini difokuskan pada sisi client. Untuk validasi sisi client dapat dilakukan melalui 3 cara, yaitu

1. Validasi melalui elemen input html
2. Validasi melalui CSS
3. Validasi melalui html DOM (Document Object Model)

Berikut ini akan diberikan contoh untuk masing – masing cara validasi di atas.

1. Validasi Melalui Elemen Input Html

Elemen input misalnya adalah text, option, checkbox, radio. Setiap elemen tersebut mempunyai atribut yang dapat diset nilainya untuk validasi. Berikut ini beberapa nilai atribut yang bisa diset untuk validasi

Constraint Validation HTML Input Attributes

Attribute	Description
disabled	Specifies that the input element should be disabled
max	Specifies the maximum value of an input element
min	Specifies the minimum value of an input element
pattern	Specifies the value pattern of an input element
required	Specifies that the input field requires an element
type	Specifies the type of an input element

Referensi lebih lengkap tentang atribut input dapat diakses di:

https://www.w3schools.com/html/html_form_attributes.asp

contoh isian field harus diisi, dengan atribut required

```
<!DOCTYPE html>
<html>
<body>
<form>
  Nama: <input type="text" required> <br>
  <input type="submit" value="Submit">
</form>
</body>
</html>
```

Silahkan dicoba klik tombol submit tanpa mengisi field nama

Contoh penggunaan pattern, isian harus huruf abjad

```
<!DOCTYPE html>
<html>
<body>
<form>
  Nama: <input type="text" pattern="[A-Za-z]+" title="harus huruf"> <br>
  <input type="submit" value="Submit">
</form>
</body>
</html>
```

Contoh penggunaan pattern, isian harus angka

```
<!DOCTYPE html>
<html>
<body>
<form>
  Nomor urut: <input type="text" pattern="[0-9]+" title="harus angka"> <br>
  <input type="submit" value="Submit">
</form>
</body>
</html>
```

Contoh penggunaan validasi pattern email

```
<!DOCTYPE html>
<html>
<body>
<form>
  email <input type="text" pattern="[a-z0-9._%+-]+@[a-z0-9.-]+\.[a-z]{2,4}$"
  title="xyz@something.com" > <br>
  <input type="submit" value="Submit">
</form>
</body>
</html>
```

Email juga bisa divalidasi dengan nilai atribut type=email.

```
<!DOCTYPE html>
<html>
<body>
<form>
  email <input type="email" title="xyz@something.com" > <br>
  <input type="submit" value="Submit">
</form>
</body>
</html>
```

2. Validasi melalui CSS

Validasi melalui CSS dilakukan lewat selector. Beberapa nilai CSS selector untuk validasi

Selector	Description
:disabled	Selects input elements with the "disabled" attribute specified
:invalid	Selects input elements with invalid values
:optional	Selects input elements with no "required" attribute specified
:required	Selects input elements with the "required" attribute specified
:valid	Selects input elements with valid values

[For a full list, go to CSS Pseudo Classes](#)

Berikut ini contoh penggunaan CSS invalid value

```
<!DOCTYPE html>
<html>
<head>
<style>
input:invalid {
  background-color: red;
}
</style>
</head>
<body>
email: <input type="email" value="support@example.com">
<p>jika format email belum tepat maka
  background-color akan berwarna merah</p>
</body>
</html>
```

Silahkan dicoba nilai "invalid" diganti menjadi "valid" dan background-color:yellow maka warna kuning akan muncul saat format email benar/valid

3. Validasi melalui html DOM (Document Object Model)

Validasi melalui DOM dapat dilakukan melalui method dan properti. Validasi yang melalui method dilakukan dengan memanggil method

Constraint Validation DOM Methods

Property	Description
<code>checkValidity()</code>	Returns true if an input element contains valid data.
<code>setCustomValidity()</code>	Sets the <code>validationMessage</code> property of an input element.

Contoh penggunaan method `checkValidity()`

```
<!DOCTYPE html>
<html>
<body>
Masukkan angka 100-400: <br>
<input id="id1" type="number" min="100" max="400" required>
<button onclick="myFunction()">OK</button>
<p id="demo"></p>

<script>
function myFunction() {
  var inpObj = document.getElementById("id1");
  if (!inpObj.checkValidity()) {
    document.getElementById("demo").innerHTML = inpObj.validationMessage;
  } else {
    document.getElementById("demo").innerHTML = "Input OK";
  }
}
</script>

</body>
</html>
```

Berikut ini adalah validasi DOM melalui property, nilai property yang dapat digunakan adalah

Constraint Validation DOM Properties

Property	Description
<code>validity</code>	Contains boolean properties related to the validity of an input element.
<code>validationMessage</code>	Contains the message a browser will display when the validity is false.
<code>willValidate</code>	Indicates if an input element will be validated.

Nilai validity mempunyai beberapa nilai detail, yaitu

The **validity property** of an input element contains a number of properties related to the validity of data:

Property	Description
customError	Set to true, if a custom validity message is set.
patternMismatch	Set to true, if an element's value does not match its pattern attribute.
rangeOverflow	Set to true, if an element's value is greater than its max attribute.
rangeUnderflow	Set to true, if an element's value is less than its min attribute.
stepMismatch	Set to true, if an element's value is invalid per its step attribute.
tooLong	Set to true, if an element's value exceeds its maxLength attribute.
typeMismatch	Set to true, if an element's value is invalid per its type attribute.
valueMissing	Set to true, if an element (with a required attribute) has no value.
valid	Set to true, if an element's value is valid.

Contoh penggunaan nilai *validity rangeOverflow* untuk memvalidasi nilai yang melebihi nilai maksimal yang ditentukan

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

<p>Masukkan nilai UTS (0-100):</p>

<input id="id1" type="number" max="100">
<button onclick="myFunction()" ">OK</button>

<p id="demo"></p>
<script>
function myFunction() {
  var txt = "";
  if (document.getElementById("id1").validity.rangeOverflow) {
    txt = "Value too large";
  } else {
    txt = "Input OK";
  }
  document.getElementById("demo").innerHTML = txt;
}
</script>

</body>
</html>
```