

# MODUL

## BAHASA PEMROGRAMAN JAVASCRIPT

### A. Tujuan

1. Dapat menjelaskan konsep *client side scripting* dalam pemrograman aplikasi web
2. Dapat membedakan pemrograman *client side scripting* dan pemrograman *server side scripting*
3. Dapat menjelaskan berbagai bahasa pemrograman web yang termasuk dalam *client side scripting*
4. Dapat mengenal pemrograman *Javascript*
5. Dapat menjelaskan struktur penulisan program *client side*

### B. Dasar Teori

#### **Perbedaan *Client Side Scripting* dan *Server Side Scripting***

Dalam dunia pemrograman *web*, dikenal adanya istilah pemrograman sisi klien (*client side programming / client side scripting*) dan pemrograman sisi server (*server side programming / server side scripting*).

Pemrograman sisi klien adalah proses pembuatan kode-kode program (dikenal dengan istilah skrip) yang nantinya akan dieksekusi di lingkungan klien (oleh *web browser*). Skrip-skrip yang dibuat di sisi klien ini pada umumnya disisipkan di dalam kode HTML. Para pengembang *web* membuat skrip-skrip di sisi klien menggunakan bahasa pemrograman seperti *Javascript* atau *VBScript* (bentuk singkatan dari: *Visual Basic Script*). Karena skrip-skrip dalam pemrograman sisi klien ini akan dieksekusi oleh *web browser*, maka peran *web browser* sangatlah penting. Artinya, *web browser* yang digunakan harus mendukung bahasa pemrograman yang digunakan untuk menulis skrip. Jalannya program di lingkungan *web browser* yang satu dengan yang lainnya 'mungkin' saja agak berbeda karena perilaku dari masing-masing *web browser* bersangkutan. Karena alasan tersebut, sebagai pengembang *web* di sisi klien, sebaiknya proses pengujian program dilakukan menggunakan beberapa *web browser* yang berbeda (misal: *Internet Explorer*, *Mozilla Firefox*, *Opera*, dan sebagainya).

Berbeda dengan pemrograman sisi klien yang kodenya dieksekusi oleh *web browser*, dalam pemrograman sisi server kode-kode program (yang juga sering disebut *skrip*) akan dieksekusi oleh *interpreter* yang terpasang di dalam *web server* (tergantung dari bahasa apa yang digunakan untuk menulis skripnya). Skrip di sisi *server* baru akan dieksekusi jika ada permintaan dokumen dari *user* ke *web server*. Selanjutnya skrip tersebut akan diterjemahkan ke dalam format bahasa yang dimengerti oleh *web browser* (bahasa HTML) dan dikirimkan lagi ke *user*. Dengan demikian, dokumen yang diterima oleh *user* bukan lagi berupa skrip, melainkan dokumen yang sudah berupa HTML sehingga dapat ditampilkan di layar *web browser* sebagai halaman *web*. Skrip di sisi *server* dapat dikembangkan menggunakan bahasa-bahasa seperti: PHP, Perl, ASP, dan sebagainya.

Meskipun pemrograman sisi-klien dan sisi-server masing-masing memiliki fungsi yang berbeda, namun pada kenyataannya dalam pengembangan suatu sistem berbasis *web* keduanya tetap dipakai untuk dapat saling bekerja sama. Sebagai contoh sederhana: skrip yang berada di sisi klien digunakan untuk melakukan validasi input dari *user* sedangkan skrip sisi-*server* yang akan memproses data tersebut dan menyimpannya ke dalam suatu *flat file* maupun RDBMS tertentu.

### **Pengenalan Javascript**

*Javascript* adalah bahasa yang berbentuk kumpulan skrip yang berjalan pada suatu dokumen HTML. Sepanjang sejarah internet bahasa ini adalah bahasa skrip pertama untuk *web*. Bahasa ini adalah bahasa pemrograman untuk memberikan kemampuan tambahan terhadap bahasa HTML dengan mengizinkan pengekseskuan perintah-perintah di sisi *user*, yang artinya di sisi *browser* bukan di sisi *server web*.

*Javascript* diperkenalkan pertama kali oleh Brendan Eich yang bekerja di *Netscape* pada tahun 1995. Pada awalnya bahasa ini dinamakan "*LiveScript*" yang berfungsi sebagai bahasa sederhana untuk *browser Netscape Navigator 2*. Pada masa itu bahasa ini banyak dikritik karena kurang aman, pengembangannya yang terkesan buru-buru dan tidak ada pesan kesalahan yang ditampilkan setiap kali membuat kesalahan pada saat menyusun suatu program. Kemudian sejalan dengan sedang giatnya kerjasama antara *Netscape* dan Sun (pengembang bahasa pemrograman "Java") pada masa itu, maka *Netscape* memberikan nama "*Javascript*" kepada bahasa tersebut pada tanggal 4 Desember 1995. Pada saat

yang bersamaan *Microsoft* sendiri mencoba untuk mengadaptasikan teknologi ini yang mereka sebut sebagai "*Jscript*" di *browser Internet Explorer 3*.

Oleh karena ada banyak perusahaan yang mengembangkan bahasa ini kemudian distandarkan dengan nama ECMAScript oleh *Netscape* melalui Organisasi Internasional ECMA. Standar ini dipublikasikan pertama kali pada bulan Juni 1997 dengan nama dokumen spesifikasi ECMA-262. Melalui publikasi tersebut, semua implementasi *Javascript* pada tiap *browser* akan memiliki standar penerapan (pengkodean) yang sama. Saat ini standar ini telah mencapai rilis Edisi ke-5.1, yang dipublikasikan pada bulan Juni 2011. Setiap *browser* saat ini memiliki implementasi sendiri-sendiri untuk ECMAScript ini, diantaranya *Internet Explorer* dengan *Jscript*, *Opera* dengan ECMAScript, dan *Mozilla Firefox*, *Google Chrome* termasuk juga *Safari* dengan nama *Javascript*.

Selain di *browser*, sekarang *Javascript* juga sudah diterapkan pada banyak aplikasi lainnya seperti *Windows 8 Apps (.Net Framework)*, *Adobe Flash ActionScript*, *KDE Desktop Environment*, *Node.js*, *Qt QML*, *jQuery Mobile*, *Firefox OS*, *Ubuntu Touch*, dan masih banyak lagi kemungkinan implementasi lainnya.

*Javascript* bergantung kepada *browser (navigator)* yang memanggil dan menampilkan halaman *web* yang tidak hanya berisi HTML tapi juga dapat berisi skrip-skrip *Javascript*. *Javascript* juga tidak memerlukan penterjemah khusus untuk menjalankannya di sisi *user/klien*. Berikut ini satu tabel yang berisi beberapa perbandingan mendasar antara *Java* dan *Javascript*.

<i>Javascript</i>	<i>Java</i>
Bahasa yang dijalankan langsung oleh <i>browser</i>	Bahasa yang dijalankan oleh program eksternal ( <i>Java Virtual Machine</i> ) diintegrasikan ke dalam <i>browser</i>
Didefinisikan dengan tag <code>&lt;script&gt;</code> pada halaman HTML	Didefinisikan dengan tag <code>&lt;applet&gt;</code>
Kode program bersifat terbuka dan dapat dibaca melalui <i>browser</i>	Kode program dapat bersifat tertutup
Tidak memerlukan proses kompilasi menjadi file lainnya	Perlu kompilasi untuk menghasilkan file <code>.class</code>

*Javascript* adalah bahasa yang “*case sensitive*” artinya membedakan penamaan variabel dan fungsi yang menggunakan huruf besar dan huruf kecil, contoh variabel atau fungsi dengan nama TEST berbeda dengan variabel dengan nama test. Dan yang terakhir seperti bahasa Java ataupun C, walaupun hal ini tidaklah wajib, setiap instruksi dapat diakhiri dengan karakter titik koma (;).

### **Penulisan Kode Javascript**

Ada beberapa cara untuk menuliskan kode *Javascript* di dalam dokumen / halaman HTML, yakni menggunakan *tag* `<script>`, menggunakan *file eksternal* dan melalui *event* tertentu.

Menggunakan tag `<script>`

Kode *javascript* dalam HTML dituliskan dengan menggunakan *tag* `<script>` dengan format sebagai berikut:

```
<script language="javascript">
.....
</script>
```

Atribut *language* di dalam *tag* `<script>` menunjukkan jenis bahasa yang digunakan dan versinya, contohnya “*Javascript*”, “*Javascript1.1*”, “*Javascript1.2*” untuk bahasa *Javascript* atau bahasa lainnya, contohnya “*VBScript*”. Jika kita ingin menggunakan beberapa versi *Javascript* di dalam satu halaman HTML (untuk menyesuaikan dengan kompatibilitas navigator), maka kita hanya perlu meletakkan kode-kode *Javascript* tersebut (berdasarkan versinya) ke dalam beberapa tag `<script>` dengan mencantumkan versi *Javascript*-nya.

### **Menggunakan File Eksternal**

Cara berikutnya adalah menuliskan kode program *Javascript* dalam suatu *file* teks dan kemudian *file* teks yang berisi kode *Javascript* dipanggil dari dalam dokumen HTML (khusus *Netscape* mulai versi 3 ke atas). Kode yang kita sisipkan ke dalam dokumen HTML adalah sebagai berikut:

```
<script language="javascript" src="lokasi/file.js"></script>
```

Dimana lokasi/file.js adalah lokasi dan nama *file* yang berisi kode *Javascript*, jika diperintah tambahan SRC tidak disertakan maka *tag* Script akan mencari kode yang terletak di dalam tag Script.

### **Melalui Event tertentu**

*Event* adalah sebutan dari satu aksi yang dilakukan oleh *user*, contohnya seperti klik tombol *mouse*, pembahasan lebih lanjut ada di materi lainnya. Kodenya dapat ditulis sebagai berikut:

```
<tag eventHandler="kode Javascript yang akan dimasukkan">
```

dimana *eventHandler* adalah nama dari *event* tersebut.

### **Dokumentasi Program**

Seperti dalam banyak bahasa pemrograman lainnya, sangat dianjurkan untuk menambahkan komentar-komentar di dalam skrip atau kode program yang dibuat untuk keperluan dokumentasi pengembang. Hal ini berguna antara lain:

- a. Sebagai pengingat bagian-bagian khusus di dalam skrip tersebut, jika akan merubah sesuatu di dalamnya, mungkin beberapa bulan kemudian sudah tidak ingat lagi dengan detail dan alur dari skrip tersebut.
- b. Untuk membuat orang yang tidak tahu program yang dibuat jadi mengerti dengan membaca petunjuk-petunjuk yang dibuat melalui komentar-komentar.

Penulisan komentar di *Javascript*, kita bisa menggunakan cara yang sama dengan aturan yang ada di bahasa C/C++ ataupun Java.

- a. Untuk menulis komentar dalam satu baris kita gunakan karakter garis miring ganda (*//*).  
*//* semua karakter di belakang  
*//* tidak akan dieksekusi
- b. Untuk menulis komentar yang terdiri dari beberapa baris kita gunakan karakter */\** dan *\*/*  
*/\** Semua baris antara 2 tanda tersebut  
Tidak akan di eksekusi oleh  
kompilator *\*/*

### **Contoh Program Javascript**

Kode *Javascript* dapat ditulis dalam dua cara, yaitu:

1. disisipkan langsung ke dalam kode HTML
2. ditulis dalam *file* terpisah (dengan ekstensi \*.js)

Pada contoh berikut ini adalah contoh skrip *Javascript* di dalam suatu dokumen HTML, disini kita akan membuat satu program untuk menampilkan satu kotak dialog (dijelaskan lebih lanjut di BAB lainnya) pada saat kita membuka dokumen HTML.

File: halo.html

```
<html>
<head>
<title>contoh program javascript</title>
</head>
<body>
<script language="javascript">
    alert("Hallo !");
</script>
</body>
</html>
```

Saat program ini dijalankan melalui *browser*, maka akan ditampilkan satu dialog berisikan pesan "Hallo !" tanpa tanda kutip. Selain dengan bentuk lengkap seperti di atas, kita juga dapat menuliskannya seperti berikut dengan hasil yang sama.

File: halo\_alternatif.html

```
<script language="javascript">
    alert("Hallo !");
</script>
```

## C. Latihan

### Latihan 7.1 (Javascript.html)

```
<html>
<head>
  <title>Javascript</title>
</head>
<body>
<h2>Demo Menyisipkan Javascript dalam dokumen HTML</h2>
<script type="text/javascript">
  document.write("Javascript is not Java");
</script>
</body>
</html>
```

### Latihan 7.2

(contoh.js)

```
document.write("<strong>Demo Javascript</strong>: ");
document.write(
  "Teks ini ditulis menggunakan Javascript");
```

(DemoJavascript.html)

```
<html>
<head>
  <title>Javascript</title>
</head>
<body>
<h2>Demo Javascript yang Disimpan dalam File Terpisah</h2>
<!-- memanggil kode dalam file contoh.js -->
<script type="text/javascript" src="contoh.js">
</script>

</body>
</html>
```

### Latihan 7.3

(cobaAlert.js)

```
<html>
<head>
  <title>Javascript</title>
  <script language="javascript">
    function cobaAlert ()
    {
      alert ("Hallo Dunia ... !");
    }
  </script>
</head>
<body>
  <h2>Demo Menyisipkan Javascript dalam Dokumen HTML</h2>
  <input type="button" value="Kirim" onClick="cobaAlert()" />
</body>
</html>
```

#### D. Tugas Mandiri (Ditulis di Buku Tulis per Individu)

- 1) Sebutkan beberapa contoh (minimal 3) penggunaan *Javascript* yang sering dijumpai dalam halaman *web* masa ini !
- 2) Adakah perbedaan dan persamaan antara struktur penulisan *javascript* di bawah ini:

<pre>&lt;script language="javascript"&gt;   ..... &lt;/script&gt;</pre>	<pre>&lt;script type="text/javascript"&gt;   ..... &lt;/script&gt;</pre>
---	--

### Kata Mutiara:

Boleh saja bermimpi setinggi langit, tetapi jangan lupa untuk bangun dari mimpi-mimpi kalian. Saatnya berusaha dan berdoa !!! Kalian BISA !!!