

Pada gambar tersebut tampak tidak ada hitungan ke-5 dan pada hitungan ke-8 perulangan dihentikan.

## 6.3 Function

Function merupakan blok program yang dapat digunakan secara berulang-ulang dengan memanggil nama function. Function dapat mengembalikan sebuah nilai dengan perintah **return** atau tanpa mengembalikan nilai. Pemanggilan function yang mengembalikan nilai dapat digunakan sebagai nilai suatu variabel. Cara menulis function sebagai berikut:

```
function nama_fungsi(parameter1, parameter 2){  
    pernyataan;  
    return variabel;  
}
```

Berikut ini contoh dari penggunaan function.

**File 6\_3\_function.html**

```
<!DOCTYPE HTML>  
<html>  
  <head>  
    <script>  
      function hitung_luas(p, l){  
        var L = p * l;  
        return L;  
      }  
      function tampilkan_hasil(){  
        var teks = "Luas tanah " + hitung_luas(200, 150) +  
"m2";  
        document.write(teks);  
      }  
      tampilkan_hasil();  
    </script>  
  </head>
```

```
<body>
</body>
</html>
```

Pada skrip tersebut, function **hitung\_luas()** mengembalikan nilai berupa hasil perhitungan luas. Karena mengembalikan nilai, cara penggunaannya dapat diletakkan pada variabel, pada skrip tersebut pada variabel teks. Sedangkan function **tampilkan\_hasil()** tidak mengembalikan nilai, sehingga untuk menggunakannya langsung pada baris tersendiri. Hasilnya seperti gambar berikut:



*Contoh Penggunaan Function*

## 6.4 Event

Event merupakan suatu perlakuan yang dilakukan terhadap suatu elemen HTML. Ada berbagai macam event yang dapat digunakan, sebagai berikut:

- **onblur**, ketika suatu elemen tidak focus.
- **onchange**, ketika nilai suatu elemen diubah.
- **onclick**, ketika suatu elemen diklik.
- **ondblclick**, ketika suatu elemen didobel klik.
- **onerror**, ketika error.
- **onfocus**, ketika suatu elemen sedang fokus.
- **onkeydown**, ketika tombol huruf pada keyboard ditekan.

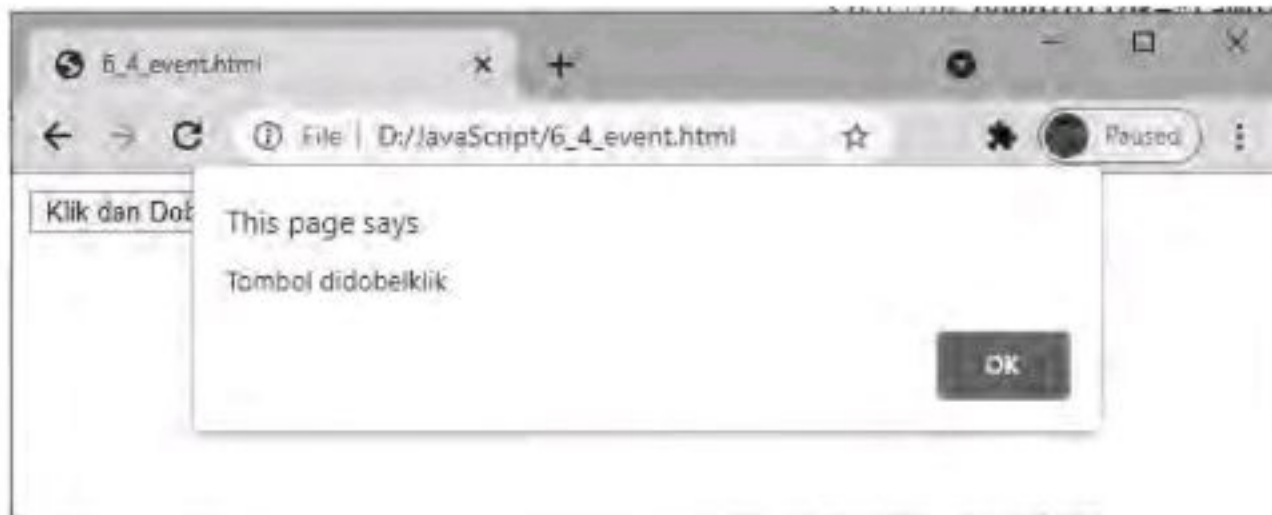
- **onkeypress**, ketika tombol keyboard ditekan.
- **onkeyup**, ketika tombol keyboard di lepas.
- **onload**, ketika dokumen di-load.
- **onmousedown**, ketika tombol mouse ditekan.
- **onmousemove**, ketika mouse digerakkan.
- **onmouseout**, ketika mouse keluar dari area suatu elemen.
- **onmouseover**, ketika mouse melewati area suatu elemen.
- **onmouseup**, ketika tombol mouse dilepas setelah diklik.
- **onreset**, ketika form direset.
- **onresize**, ketika elemen diubah ukuranya.
- **onselect**, ketika suatu elemen dipilih.
- **onsubmit**, ketika data form dikirim.
- **onunload**, ketika keluar dari suatu dokumen.

Contoh penggunaanya dapat diperhatikan pada skrip berikut:

**File 6\_4\_event.html**

```
<!DOCTYPE HTML>
<html>
  <head>
    <script>
      function dobelklik(){
        alert("Tombol didobelklik");
      }
    </script>
  </head>
  <body>
    <button ondblclick="dobelklik()">
      Klik dan Dobelklik</button>
  </body>
</html>
```

Coba jalankan, lalu double klik pada tombol, maka akan tampil alert seperti gambar berikut:



*Contoh Penggunaan Event*

Silakan untuk bereksperimen sendiri dengan mengubah **ondblclick** dengan event yang lain, lalu lakukan event yang sesuai.

## 6.5 Kotak Dialog

Ada 3 macam kotak dialog pada JavaScript yaitu dengan function **alert()**, **confirm()**, dan **prompt()**. **Alert()** digunakan untuk menampilkan pesan tertentu kepada pengguna. **Confirm()** hampir sama dengan alert, tetapi terdapat 2 tombol yaitu Oke dan Batal, fungsinya untuk meminta konfirmasi pada pengguna. Sedangkan **prompt()** digunakan untuk meminta masukan data dari pengguna.

Contoh penggunaanya sebagai berikut:

**File 6\_5\_kotak\_dialog.html**

```
<!DOCTYPE HTML>
<html>
  <head>
    <title>Continue dan break</title>
    <script>
      alert("Selamat Datang");
      confirm("Akan melanjutkan?");
```

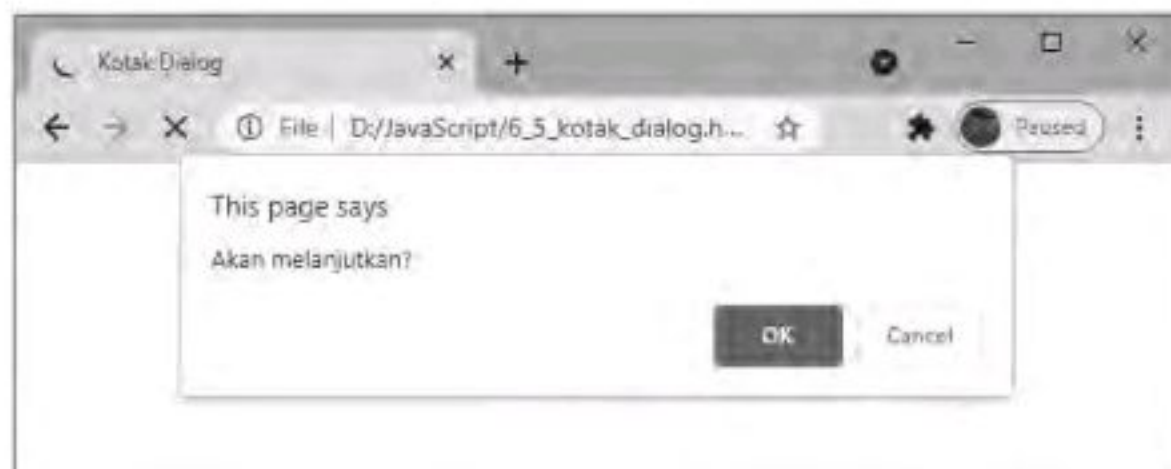


```

        prompt("Masukan Nama Kamu:", "Nama kamu di sini");
    </script>
</head>
<body>
</body>
</html>

```

Hasil untuk kotak dialog dengan confirm() dan prompt() seperti pada gambar berikut:



*Hasil Penggunaan confirm()*



*Hasil Penggunaan prompt()*

## 6.6 Objek String

Seperti telah dijelaskan sebelumnya bahwa setiap variabel akan menjadi objek sehingga akan otomatis memiliki property dan method. Untuk variabel dengan tipe data string memiliki property dan method sebagai berikut:

Property/Method	Keterangan
length	Menghasilkan jumlah karakter dari suatu teks.
anchor(nama)	Menghasilkan link sama seperti menggunakan tag <a name="nama">.
big()	Menjadikan teks menjadi lebih besar karena diapit tag <big>.
blink()	Menjadikan teks berkedip karena diapit tag <blink>.
bold()	Menjadikan teks bercetak tebal karena diapit tag <b>.
fontcolor(warna)	Menjadikan teks berwarna karena diapit tag <font color="warna">.
fontsize(ukuran)	Mengubah ukuran teks karena diapit tag <font size="ukuran">.
italics()	Menjadikan teks miring karena diapit tag <i>.
link(url)	Merubah teks menjadi link karena diapit tag <a href="url">.
small()	Menjadikan teks kecil karena diapit tag <small>.
strike()	Menjadikan teks tercoret karena diapit tag <strike>.
sub()	Menjadikan teks subscript karena diapit tag <sub>.
sup()	Menjadikan teks superscript karena diapit tag <sup>.
toLowerCase()	Mengubah teks menjadi huruf kecil.
toUpperCase()	Mengubah teks menjadi huruf besar.
charAt(x)	Menghasilkan karakter pada posisi x pada string.
charCodeAt(x)	Menghasilkan nilai Unicode dari karakter pada posisi x pada suatu string
concat(teks1, teks2, ...)	Menggabungkan dua atau lebih string menjadi satu.
fromCharCode(c1, c2, ....)	Menghasilkan string yang dibuat menggunakan urutan dari Unicode.
indexOf(substr, mulai)	Mencari dan menghasilkan nomor index.
charCodeAt(x)	Menghasilkan nilai unicode value dari karakter pada posisi x pada suatu string.
concat(string1, string2, ...)	Menggabungkan beberapa string yang ada di dalam kurung.

<code>fromCharCode(c1, c2, ...)</code>	Menghasilkan string yang dibuat menggunakan urutan dari nilai Unicode.
<code>indexOf(substr, start)</code>	Menghasilkan nomor index dari karakter atau substring yang ada di dalam kurung.
<code>match(regex)</code>	Mencocokkan suatu string dengan pola regular expression.
<code>replace(regex, replace_text)</code>	Menukar suatu string berdasarkan pola regular expression.
<code>search(regex)</code>	Mengetes apakah string cocok dengan pola regular expression, jika cocok akan menghasilkan index dari yang cocok.
<code>slice(start, end)</code>	Menghasilkan substring dari suatu string dimulai dari parameter pertama dan diakhiri parameter kedua.
<code>split(delimiter, limit)</code>	Memecah string menjadi array berdasarkan tanda pemisah yang ditentukan.
<code>substr(start, length)</code>	Menghasilkan potongan dari suatu string yang dimulai dari parameter pertama sebanyak parameter kedua.
<code>trim()</code>	Menghilangkan spasi di sebelah kanan dan kiri string.

Berikut contoh penggunaan dari beberapa method dan property tersebut. Untuk contoh method dan property yang lain silakan bereksperimen sendiri.

#### File 6\_6\_string.html

```
<!DOCTYPE HTML>
<html>
  <head>
    <title>Manipulasi string</title>
    <script>
      var teks = "Saya suka belajar JavaScript";
      document.write(teks.fontcolor('red') + '<br>');
      document.write(teks.toUpperCase() + '<br>');
      document.write(teks.substr(10, 18));
    </script>
```

```

</head>
<body>
</body>
</html>

```

Hasilnya seperti gambar berikut:



*Contoh Manipulasi Objek String*

## 6.7 Objek Math

Objek math digunakan untuk melakukan berbagai operasi matematika. Ada beberapa method yang digunakan untuk melakukan operasi matematika, sebagai berikut:

Property/Method	Keterangan
<i>PI</i>	Menghasilkan nilai PI.
<i>abs(x)</i>	Mengkonversi angka x menjadi absolut, sehingga yang negatif menjadi positif.
<i>acos(x)</i>	Mendapatkan nilai acos dari x dalam radian.
<i>asin(x)</i>	Mendapatkan nilai asin dari x dalam radian.
<i>atan(x)</i>	Mendapatkan nilai atan dari x dalam radian.
<i>atan2(y, x)</i>	Mendapatkan nilai atan dari sebuah koordinat.
<i>ceil(x)</i>	Membulatkan bilangan decimal ke angka terdekat di atasnya.



<code>cos(x)</code>	Mendapatkan nilai cos dari x dalam radian.
<code>floor(x)</code>	Membulatkan bilangan decimal ke angka terdekat di bawahnya.
<code>log(x)</code>	Mendapatkan nilai log dari x.
<code>max(x, y, z, ..., n)</code>	Mendapatkan angka tertinggi di dalam kurung.
<code>min(x, y, z, ..., n)</code>	Mendapatkan nilai terendah di dalam kurung.
<code>pow(x, y)</code>	Mendapatkan nilai x pangkat y.
<code>random()</code>	Mendapatkan angka acak antara 0 dan 1.
<code>round(x)</code>	Membulatkan bilangan decimal ke angka terdekat.
<code>sin(x)</code>	Mendapatkan nilai sin dari x dalam radian.
<code>sqrt(x)</code>	Mendapatkan nilai akar kuadrat dari x.
<code>tan(x)</code>	Mendapatkan nilai tan dari x.

Setiap penggunaan method dan property tersebut diawali dengan kata **Math**. Berikut ini contoh penggunaan beberapa method tersebut untuk melakukan operasi matematika:

**File 6\_7\_operasi\_matematika.html**

```
<!DOCTYPE HTML>
<html>
  <head>
    <title>Operasi Matematika</title>
    <script>
      document.write("Akar 81 = " + Math.sqrt(81));
      document.write("<br>Nilai 5 pangkat 3 = " + Math.pow(5,
3));
      document.write("<br>5.23 dibulatkan menjadi " + Math.ro
und(5.23));
    </script>
  </head>
  <body>
  </body>
</html>
```

Hasilnya seperti gambar berikut:



*Contoh Operasi Matematika*

## 6.8 Objek Tanggal dan Waktu

Tanggal dan waktu juga merupakan objek yang memiliki property dan method. Ada berbagai method untuk memanipulasi tanggal atau waktu. Untuk menggunakan method tersebut, harus memanggil objek `Date()` terlebih dahulu seperti berikut:

```
var tanggal = new Date();
```

Selanjutnya method tersebut digunakan pada variabel tanggal. Adapun daftar method yang bisa dipakai sebagai berikut:

Property/Method	Keterangan
<code>getDate()</code>	Menghasilkan tanggal dalam satu bulan.
<code>getDay()</code>	Menghasilkan nama hari.
<code>getHours()</code>	Menghasilkan jam dimulai dari 0 hingga 23.
<code>getMinutes()</code>	Menghasilkan menit.
<code>getSeconds()</code>	Menghasilkan detik.
<code>getMonth()</code>	Menghasilkan bulan dimulai dari 0 hingga 11.
<code>getFullYear()</code>	Menghasilkan tahun.
<code>getTime()</code>	Menghasilkan waktu yang lengkap.

Berikut ini contoh penggunaan beberapa method tersebut untuk memanipulasi data tanggal dan waktu:

File 6\_8\_tanggal\_waktu.html

```
<!DOCTYPE HTML>
<html>
  <head>
    <script>
      var waktu = new Date();
      document.write('Hari ini: ' + waktu.getDate() + '/'
        + waktu.getMonth() + '/' + waktu.getFullYear() );
    </script>
  </head>
  <body>
  </body>
</html>
```

Hasilnya seperti gambar berikut:



*Contoh Manipulasi Objek Tanggal*

Pada ECMAScript 5 dikenalkan method *Date.now()* yang menghasilkan angka mili dimulai dari nol pada 1 Januari 1970 pukul 00:00.

## 6.9 Array

Array merupakan sebuah variabel yang dapat menampung beberapa data sekaligus yang terindeks secara teratur. Untuk mendeklarasikan array ada beberapa cara sebagai berikut: