



MODUL PEMROGRAMAN FULL STACK 1

CYNTHIA HAYAT SKOM MMSI



UNIVERSITAS KRISTEN KRIDA WACANA
Program Studi Sistem Informasi
TAHUN AKADEMIK GENAP 2021/2022

Kata Pengantar

Puji Syukur kehadiran Tuhan Yang Maha Esa karena atas berkat dan kasih Karunia-Nya sehingga saya dapat menyelesaikan modul Pemrograman Full Stack 1 ini. Modul ini disusun berdasarkan Kurikulum Program Studi (S1) Sistem Informasi 2017, dibentuk mengacu pada Permenristek Dikti No. 44 Tahun 2015 mengenai Standar Nasional Pendidikan Tinggi (SNPT). Capaian Pembelajaran Lulusan (CPL) yang disusun merujuk pada KKNI (Kualifikasi Kompetensi Nasional Indonesia) level 6, untuk jenjang pendidikan S1.

Modul ini juga dilengkapi dengan latihan soal untuk menguji pemahaman mahasiswa terkait dengan materi yang terdapat pada modul. Terima kasih kepada berbagai pihak yang telah membantu proses penyelesaian modul ini. Semoga modul ini dapat bermanfaat bagi kita semua, khususnya para peserta didik.

Jakarta, 2 Februari 2022

Penyusun

Cynthia Hayat S.Kom., M.MSI



Modul Praktikum CSS

CYNTHIA HAYAT S.KOM., M.MSI

KRIDA WACANA CHRISTIAN UNIVERSITY
Faculty of Engineering and Computer Science
Departement of Information System



UKRIDA
Universitas Kristen Krida Wacana



Belajar CSS Dasar Cara Penulisan CSS

CSS(Cascading Style Sheet) merupakan sebuah **bahasa pemrograman web** yang memiliki fungsi dan tujuan untuk mengatur atau mendesign tiap-tiap komponen dari **HTML** seperti elemen dan tag. dengan menggunakan CSS kita dapat mengatur ukuran, warna dan bentuk dari element HTML. Mengganti font, mengatur margin dan mengatur padding dan banyak lagi yang dapat di lakukan menggunakan CSS untuk mempercantik atau mendesign halaman website. **Belajar CSS Dasar Cara Penulisan CSS**

CSS menggunakan selector(id dan class) untuk menentukan element yang akan di modifikasi atau yang akan di beri sentuhan css. jika di ibaratkan HTML sebagai tiang pada sebuah bangunan rumah, maka CSS berfungsi sebagai cet dan dekorasi pada bangunan rumah tersebut. Ada tiga teknik metode penulisan CSS, yaitu :

- Inline CSS Style
- Internal CSS Style
- External CSS Style

Teknik penulisan CSS Inline Style

Cara pertama **Belajar CSS Dasar Cara Penulisan CSS**. Teknik penulisan syntax css dengan Inline style adalah teknik cara penulisan syntax CSS yang tidak memerlukan selector(id dan class) sehingga syntax CSS di letakkan atau langsung di sisipkan pada element HTML. syntax css di letakkan di dalam atribut style="".


Contoh penulisan inline css style

Contoh **Belajar CSS Dasar Cara Penulisan CSS**. Untuk membuat contoh penulisan CSS dengan gaya inline caranya sediakan sebuah file HTML atau PHP. di sini saya menggunakan HTML. saya membuat sebuah file dengan nama index.html.

index.html

```
1 <!DOCTYPE html>
2 <html>
3 <head>
4   <title>Penulisan CSS dengan Inline Style</title>
5 </head>
6 <body>
7
8   <h1 style="color:blue">Belajar CSS Dasar Di <a href="https://www.malasngoding.co
9
10 </body>
11 </html>
```

perhatikan pada contoh di atas. **syntax css** di letakkan dalam element h1 menggunakan atribut style="" . perintah color adalah perintah css yang berfungsi untuk mengatur warna font. jadi pada contoh inline css style ini kita membuat heading h1 dengan warna font yang kita atur menjadi blue (biru). dan saat di jalankan maka hasilnya sebagai berikut:



Belajar CSS Dasar Di www.malasngoding.com

Teknik penulisan CSS Internal Style

Teknik penulisan **syntax CSS dengan Internal style** adalah teknik cara penulisan **syntax css** yang di letakkan satu file dengan file html atau file php. syntax CSS di letakkan di dalam tag `<style>` dan di akhiri dengan tag `</style>`. biasanya tag `<style>` .. `</style>` di letakkan pada bagian tag `<head>` pada HTML.

Contoh penulisan internal CSS Style

```
1 <!DOCTYPE html>
2 <html>
3 <head>
4   <title>Penulisan CSS dengan Internal Style</title>
5   <style type="text/css">
6     #tulisan{
7       color: 10px;
8     }
9
10    .box{
11      background: red;
12      padding: 10px;
13    }
14  </style>
15 </head>
16 <body>
17
18   <div class="box">
19     <h1 id="tulisan">Belajar CSS Dasar Di <a href="https://www.malasngoding.com">
20   </div>
21
22 </body>
```

Perhatikan contoh **penulisan internal css style** di atas. syntax css di letakkan satu file dengan file html. syntax css di letakkan di dalam tag `<style>`. syntax css padding berfungsi sebagai pengatur jarak pada sisi dalam element. pada contoh ini kita memberikan jarak sebesar 10px atau 10 pixel. css memanggil selector class dengan tanda titik "." dan memanggil selector id dengan tanda pagar "#".

```
1 <style type="text/css">
2   #tulisan{
3     color: 10px;
4   }
5
6   .box{
7     background: red;
8     padding: 10px;
9   }
10 </style>
```

file:///E:/malas%20ngoding/css/index.html

Belajar CSS Dasar Di www.malasngoding.com

Teknik penulisan CSS External Style

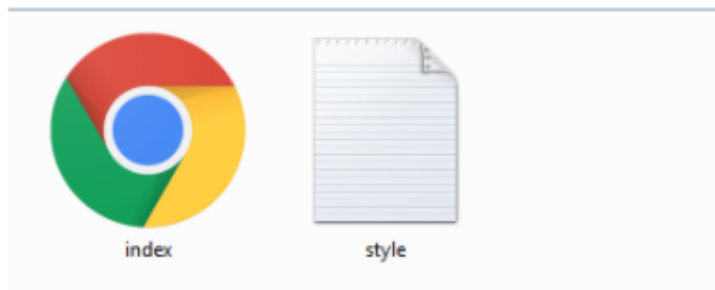
Teknik **Belajar CSS Dasar Cara Penulisan CSS kedua**. Teknik **penulisan Syntax CSS dengan External Style** adalah teknik penulisan yang memisahkan file css dan html. penggunaan css yang baik adalah menggunakan teknik penulisan **CSS** ini karena kode program kita tidak akan berantakan karena syntax css di simpan pada file css. file css dan html di hubungkan menggunakan

```
1 <link rel="stylesheet" type="text/css" href="file css anda">
```

baca tutorial sebelumnya tentang [pengertian](#) dan [pengenalan dasar CSS](#) di www.malasngoding.com.

Contoh penulisan CSS menggunakan External Style

Untuk membuat **contoh penulisan CSS dengan External style** siapkan satu buah file html dan satu buah file css. di sini saya membuat sebuah file dengan nama index.html dan style.css



index.html

```
1 <!DOCTYPE html>
2 <html>
3 <head>
4   <title>Penulisan CSS dengan External Style</title>
5   <link rel="stylesheet" type="text/css" href="style.css">
6 </head>
7 <body>
8
9   <div class="box">
10     <h1 id="tulisan">Belajar CSS Dasar Di <a href="https://www.malasngoding.com">
11   </div>
12
13 </body>
14 </html>
```

style.css

```
1 #tulisan{
2   color: 10px;
3 }
4 .box{
5   background: red;
6   padding: 10px;
7 }
```

dan jika di jalankan maka hasilnya





Belajar CSS Mengubah Background Dengan CSS

Background atau **latar belakang** pada sebuah halaman website merupakan salah satu bagian yang wajib di design untuk menghasilkan sebuah halaman website yang terlihat menarik dan modern. ada beberapa cara mempercantik background atau latar belakang website. bisa mempercantik background website dengan menggunakan gambar. jika tidak ingin menggunakan gambar bisa juga dengan mengubah warna background dengan bantuan CSS. untuk **mengubah background dengan CSS** gunakan property background pada CSS. property background selain di gunakan untuk **mengubah background body website** bisa juga di gunakan untuk mengubah gambar background pada sebuah elemen HTML yang di inginkan.

Mengubah warna background dengan CSS

Perhatikan penjelasan berikut ini untuk cara mengubah warna background dengan menggunakan CSS.

index.html

```
1 <!DOCTYPE html>
2 <html>
3 <head>
4   <title>Merubah warna background dengan CSS</title>
5   <link rel="stylesheet" type="text/css" href="style.css">
6 </head>
7 <body>
8
9   <h1>Tutorial mengubah warna background dengan CSS</h1>
10
11 </body>
12 </html>
```

style.css

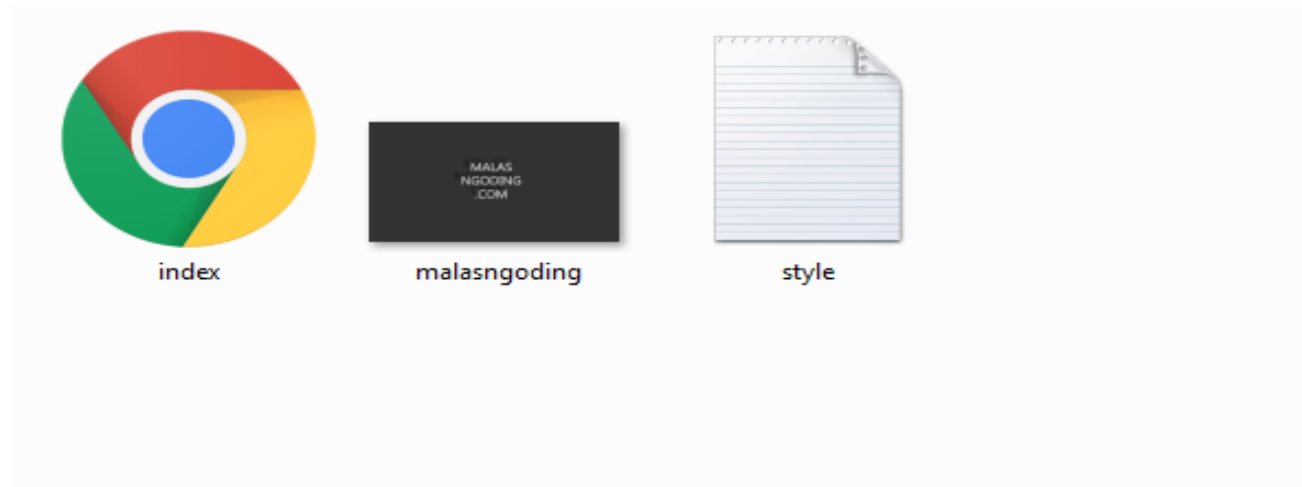
```
1 body{
2   background: blue;
3   color: white;
4 }
```



cara merubah warna dengan css

Mengubah gambar background dengan CSS

Pada sub judul **mengubah gambar background dengan css** ini akan di jelaskan bagaimana cara mengubah background sebuah halaman website dengan menggunakan gambar. oleh sebab itu anda harus menyediakan sebuah file gambar untuk dijadikan background untuk halaman website anda. di sini saya sudah menyediakan sebuah gambar sederhana yang akan saya jadikan gambar pada background. letakkan file gambar satu folder dengan file html dan css agar tidak kebingungan nantinya. nama file gambar saya adalah malasngoding.png



index.html

```
1 <!DOCTYPE html>
2 <html>
3 <head>
4   <title>Membuat background dengan gambar</title>
5   <link rel="stylesheet" type="text/css" href="style.css">
6 </head>
7 <body>
8
9   <h1>Membuat background gambar dengan CSS</h1>
10
11 </body>
12 </html>
```

style.css

```
1 body{
2   background: url('malasngoding.png');
3   color: #fff;
4 }
```

perhatikan pada syntax css di atas. untuk membuat gambar background gunakan syntax



tutorial cara merubah background gambar dengan css

Gambar pun berhasil di pasang sebagai background halaman. sekian **tutorial belajar dasar CSS mengubah background dengan CSS.**



Belajar CSS Margin dan Padding pada CSS

Margin dan padding merupakan yang paling banyak di gunakan dalam mendesain sebuah website dengan menggunakan CSS. karena margin dan padding di gunakan untuk mengatur sisi dalam dari pada sebuah element dan mengatur sisi sebelah luar dari sebuah **element**. untuk itu pada tutorial belajar CSS Dasar Margin dan Padding pada CSS ini akan dijelaskan tentang apa itu margin? apa itu padding? dan bagaimana cara penggunaannya pada CSS.

Mengenal Margin pada CSS

Margin adalah sisi luar dari sebuah element. misalnya anda ingin mengatur jarak antar element. anda bisa menggunakan syntax margin untuk mengaturnya. terdapat beberapa sisi luar margin yaitu, margin atas di tuliskan pada css dengan 'margin-top', margin bawah atau jarak luar bagian bawah di tulis di CSS dengan 'margin-bottom', 'margin-left' sebagai jarak luar sebelah kiri element, dan 'margin-right' adalah sisi luar pada bagian sebelah kanan. tetapi jika anda hanya menggunakan syntax 'margin' saja maka akan secara otomatis mengatur jarak atas,bawah, kiri dan kanan element. berikut ini adalah **Contoh penggunaan margin pada CSS**



index.html

```
2 <html>
3 <head>
4   <title>Margin CSS</title>
5   <link rel="stylesheet" type="text/css" href="style.css">
6 </head>
7 <body>
8
9   <div class="box">
10     <h1>Ini adalah box</h1>
11   </div>
12
13   <div class="box-dua">
14     <h1>Ini adalah box dua</h1>
15   </div>
16
17 </body>
18 </html>
```

style.css

```
1 .box{
2   background: blue;
3   height: 200px;
4   width: 300px;
5   margin: 20px;
6 }
7
8 .box-dua{
9   background: red;
10  height: 100px;
11  width: 200px;
12  margin-left: 70px;
13 }
```




mengenal margin pada css

perhatikan pada contoh di atas. pada kotak dua kita **mengatur margin-left** atau margin kiri sebesar 70px atau 70 pixel. dan pada kotak yang berwarna biru kita memberikan **margin** atau jarak luar segala sisi sebesar 20px.



Mengenal Padding pada CSS

Padding adalah sisi dalam dari sebuah element. kita bisa menggunakan syntax padding untuk mengatur jarak pada sisi dalam sebuah element yang kita tentukan. sama seperti **margin** yang memiliki sisi-sisi . seperti top, left, right, bottom. jenis padding yaitu padding atas di tuliskan pada css dengan 'padding-top' yang berarti mengatur sisi dalam sebelah atas sebuah element, pading bawah atau jarak dalam bagian bawah di tulis di CSS dengan 'padding-bottom', 'padding-left' sebagai jarak dalam sebelah kiri element, dan 'padding-right' adalah sisi luar pada bagian sebelah kanan. jika anda hanya menggunakan syntax 'padding' saja maka akan secara otomatis mengatur jarak atas,bawah, kiri dan kanan element yang bagian dalam. berikut ini adalah **Contoh penggunaan padding pada CSS**



index.html

```
1 <!DOCTYPE html>
2 <html>
3 <head>
4   <title>Padding CSS</title>
5   <link rel="stylesheet" type="text/css" href="style.css">
6 </head>
7 <body>
8   <h1>www.malasngoding.com</h1>
9   <div class="box">
10    <h1>Ini adalah box</h1>
11  </div>
12
13  <div class="box-dua">
14    <h1>Ini adalah box dua</h1>
15  </div>
16
17 </body>
18 </html>
```

style.css

```
1 h1{
2   text-align: center;
3 }
4
5 .box{
6   background: blue;
7   height: 200px;
8   width: 300px;
9   padding: 20px;
10 }
11
12 .box-dua{
13   background: red;
14   height: 100px;
15   width: 600px;
16   padding-left: 70px;
17 }
```



file:///E:/malas%20ngoding/css/index.html

www.malasngoding.com

Ini adalah box

Ini adalah box dua



Belajar CSS Mengatur Font dengan CSS

Pada tutorial belajar css mengatur font dengan css ini akan di jelaskan tentang cara melakukan pengaturan font dengan css. misalnya seperti merubah font, mengatur **ukuran font**, warna font serta jenis bentuk font(garis miring, huruf tebal, tulisan miring).

Beberapa syntax css yang digunakan untuk mengatur font:

- font-size digunakan untuk mengatur ukuran font
- font-weight di gunakan untuk mengatur ketebalan font
- font-family untuk mengubah jenis font
- font-style digunakan untuk merubah gaya pada font.
- font-color digunakan untuk merubah warna font.

berikut ini adalah contoh **cara mengatur font dengan css**.



index.html

```
1 <!DOCTYPE html>
2 <html>
3 <head>
4   <title>Font CSS</title>
5   <link rel="stylesheet" type="text/css" href="style.css">
6 </head>
7 <body>
8   <h1 class="tulisan_satu">Belajar CSS Dasar di www.malasngoding.com</h1>
9   <h1 class="tulisan_dua">Belajar CSS Dasar di www.malasngoding.com</h1>
10  <h1 class="tulisan_tiga">Belajar CSS Dasar di www.malasngoding.com</h1>
11 </body>
12 </html>
```

style.css

```
1 .tulisan_satu{
2   font-color: red;
3   font-family: sans-serif;
4   font-style: normal;
5 }
6
7 .tulisan_dua{
8   font-color: green;
9   font-size: 24pt;
10  font-style: italic;
11 }
12
13 .tulisan_tiga{
14   font-color: red;
15   font-weight: bold;
16   font-style: oblique;
17 }
```

file:///E:/malas%20ngoding/css/index.html

Belajar CSS Dasar di www.malasngoding.com

Belajar CSS Dasar di www.malasngoding.com

Belajar CSS Dasar di www.malasngoding.com



Belajar CSS Mengatur Hyperlink dengan CSS

Hyperlink merupakan link yang dibuat untuk mengalihkan halaman saat di klik. hyperlink atau link dibuat dengan menggunakan tag `<a>` dan di akhiri dengan tag `` di HTML. dan untuk mempercantik, mengubah atau mengatur hyperlink dengan css caranya sangat mudah. pada tutorial belajar css mengatur hyperlink dengan css ini akan di jelaskan tentang cara mengatur hyperlink dengan css. mulai dari membuat warna pada link, mengubah link menjadi bentuk tombol, membuat tampilan hyperlink berubah pada saat di klik dan sebagainya.

Ada empat status yang di miliki oleh hyperlink html dan bisa di manipulasi dengan menggunakan css. yaitu :

- link . merupakan link aktif biasa.
- visited. merupakan status sebuah link yang telah di kunjungi.
- hover. merupakan status sebuah link pada saat diletakkan cursor mouse di atasnya.
- active. merupakan status sebuah link atau hyperlink pada saat sudah di klik.

untuk penggunaan atau pemanggilan hyperlink menurut statusnya dapat di tulis dengan syntax:

- `a:link` = untuk link biasa
- `a:visited` = merupakan status sebuah link yang telah di kunjungi.
- `a:hover` = merupakan status sebuah link pada saat diletakkan cursor mouse di atasnya.
- `a:active` = merupakan status sebuah link atau hyperlink pada saat sudah di klik.



index.html

```
1 <!DOCTYPE html>
2 <html>
3 <head>
4   <title>Hyperlink CSS</title>
5   <link rel="stylesheet" type="text/css" href="style.css">
6 </head>
7 <body>
8   <a class="link" href="https://www.malasngoding.com">Klik di sini</a>
9 </body>
10 </html>
```

style.css

```
1 .link{
2     font-size: 20pt;
3 }
4 .link:hover{
5     color: red;
6 }
7 .link:link{
8     color: blue;
9 }
10 .link:active{
11     color: green;
12 }
13 .link:visited{
14     background: yellow;
15 }
```

← → ↻ 🏠 file:///E:/malas%20ngoding/css/index.html

[Klik di sini](#)



design link dengan css

index.html

```
2 <html>
3 <head>
4   <title>Hyperlink CSS</title>
5   <link rel="stylesheet" type="text/css" href="style.css">
6 </head>
7 <body>
8   <a class="contoh-link" target="_blank" href="https://www.malasngoding.com">Klik
9 </body>
10 </html>
```

style.css

```
1 .contoh-link,
2 .contoh-link:link,
3 .contoh-link:active,
4 .contoh-link:visited{
5   font-size: 20pt;
6   background: #1ABC9C;
7   color: #fff;
8   text-decoration: none;
9   padding: 10px;
10  font-family: sans-serif;
11 }
12
13 .contoh-link:hover{
14   background: #16A085;
15 }
```



Belajar CSS Mengatur Format Text dengan CSS

Pada **tutorial belajar CSS Mengatur Format Text Dengan CSS** ini akan di jelaskan tentang **penggunaan css** dalam **mengatur atau memodifikasi text** dengan tujuan agar format text dapat menjadi seperti yang kita inginkan dan kita butuhkan. contohnya untuk mengatur text menjadi justify atau kita kenal dengan sebutan rata kiri rata kanan, mengatur jarak **indent** text, mengatur huruf menjadi **uppercase** atau huruf besar, huruf kecil, warna text dan juga mengatur dekorasi atau design pada text. silahkan simak penjelasan berikut tentang **mengatur format text** yang dapat di lakukan menggunakan **CSS**.



adapun beberapa syntax CSS yang bisa di gunakan untuk **mengatur format text** diantaranya adalah:

- **color.** di gunakan untuk mengatur warna text, value yang dapat di isi berupa warna atau kode warna.
- **text-align.** di gunakan untuk mengatur posisi align pada text atau rata text, value yang bisa di isi diantaranya adalah center untuk membuat text rata tengah, left untuk membuat text menjadi rata kiri, right untuk membuat text menjadi rata kanan dan justify untuk membuat text menjadi rata kanan dan rata kiri.
- **text-decoration.** di gunakan untuk mengatur dekorasi text, value nya berupa none untuk membuat text tidak memiliki dekorasi, overline untuk membuat text memiliki garis pada bagian atas text, line-through untuk membuat garis yang mencoreng pada text, dan underline untuk membuat garis pada bawah text(garis bawah).
- **text-transform.** digunakan untuk mengatur huruf kapital pada text, value yang bisa di gunakan diantaranya adalah uppercase untuk membuat text menjadi huruf besar, lowercase untuk membuat text menjadi huruf kecil, dan capitalize untuk membuat huruf awal pada tiap kata menjadi huruf besar.
- **text-indent.** digunakan untuk mengatur jarak alinea pada text, value yang bisa digunakan berupa nilai pixel dan lainnya sesuai kebutuhan.
- **text-spacing.** digunakan untuk mengatur jarak antar karakter pada text, value yang di isi berupa nilai pixel dan lain-lain. **Mengatur Format Text Dengan CSS**
- **word-spacing.** digunakan untuk mengatur jarak antar kata pada text, value yang di isi juga berupa nilai pixel.
- **line-height.** digunakan untuk mengatur jarak antar baris pada text value yang di isi berupa nilai.



- `text-shadow`. digunakan untuk mengatur efek bayang pada text, value yang di isikan pertama mengisi nilai untuk jarak kiri kanan, dan kedua mengisi jarak atas bawah dan yang ketiga mengisi warna. untuk contoh penulisanya `2px 5px blue`.
- `vertical-align`. digunakan untuk mengatur align dalam bentuk vertikal pada text value yang digunakan adalah `left` untuk membuat text rata kiri, `right` untuk rata atas dan `center` untuk rata tengah.

Untuk contoh **cara mengatur format text dengan css**. buat sebuah file html dan sebuah file css. di sini saya membuat file `index.html` dan `style.css`



Cara mengubah warna text dengan CSS

untuk mengubah warna text dengan CSS perhatikan contoh berikut ini.

index.html

```
1 <!DOCTYPE html>
2 <html>
3   <head>
4     <title>format text CSS</title>
5     <link rel="stylesheet" type="text/css" href="style.css">
6   </head>
7   <body>
8     <p class="tulisan_warna">Mengatur format text dengan CSS</p>
9   </body>
10 </html>
```

style.css

```
1 .tulisan_warna{
2   color: blue;
3 }
```

dan hasilnya

[Mengatur format text dengan CSS](#)

dapat di lihat pada contoh di atas terdapat sebuah **tag paragraf** yang mempunyai class tulisan_warna. kemudian **mengatur warna biru dengan css**.

Cara menggunakan text align CSS

index.html

```
1 <!DOCTYPE html>
2 <html>
3 <head>
4   <title>Format text CSS</title>
5   <link rel="stylesheet" type="text/css" href="style.css">
6 </head>
7 <body>
8   <p class="tulisan_kiri">Mengatur format text dengan CSS</p>
9   <p class="tulisan_kanan">Mengatur format text dengan CSS</p>
10  <p class="tulisan_tengah">Mengatur format text dengan CSS</p>
11  <p class="tulisan_justify">
12    Belajar tutorial CSS dasar di www.malasngoding.com.
13    Mengatur format text dengan CSS. belajar HTML dasar di www.malasngoding.com.
14    Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod
15    tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam,
16    quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo
17    consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse
18    cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat no
19    proident, sunt in culpa qui officia deserunt mollit anim id est laborum.</p>
20 </body>
21 </html>
```

style.css

```
1 .tulisan_kiri{
2   text-align: left;
3 }
4
5 .tulisan_kanan{
6   text-align: right;
7 }
8
9 .tulisan_tengah{
10  text-align: center;
11 }
12
13 .tulisan_justify{
14   text-align: justify;
15 }
```



file:///E:/malas%20ngoding/css/index.html

Mengatur format text dengan CSS

Mengatur format text dengan CSS

Mengatur format text dengan CSS

Belajar tutorial CSS dasar di www.malasngoding.com. Mengatur format text dengan CSS. belajar HTML dasar di www.malasngoding.com. Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.



Belajar CSS Mengatur format text dengan CSS

untuk contoh mengatur format text dengan CSS lainnya silahkan perhatikan contoh berikut.

index.html

```
1 <!DOCTYPE html>
2 <html>
3 <head>
4   <title>Format text CSS</title>
5   <link rel="stylesheet" type="text/css" href="style.css">
6 </head>
7 <body>
8   <p class="tulisan_satu">Mengatur format text dengan CSS</p>
9   <p class="tulisan_dua">Mengatur format text dengan CSS</p>
10  <p class="tulisan_tiga">Mengatur format text dengan CSS</p>
11  <p class="tulisan_empat">
12    Belajar tutorial CSS dasar di www.malasngoding.com.
13    Mengatur format text dengan CSS. belajar HTML dasar di www.malasngoding.com.
14    Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod
15    tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam,
16    quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo
17    consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse
18    cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat no
19    proident, sunt in culpa qui officia deserunt mollit anim id est laborum.</p>
20 </body>
21 </html>
```

style.css

```
1 .tulisan_satu{
2   color: blue;
3   text-decoration: underline;
4 }
5
6 .tulisan_dua{
7   text-align: right;
8   text-transform: capitalize;
9   text-decoration: overline;
10 }
11
12 .tulisan_tiga{
13   text-align: center;
14   text-transform: lowercase;
15   text-decoration: line-through;
16   word-spacing: 10px;
17 }
18
19 .tulisan_empat{
20   text-transform: uppercase;
21   text-indent: 30px;
22   line-height: 30px;
23   letter-spacing: 5px;
24 }
```



Belajar CSS Mengenal Position CSS

Position pada **CSS** di gunakan untuk **mengatur posisi sebuah element HTML**. terkadang kita ingin membuat atau menetapkan posisi sebuah element dengan element yang lain. maka dengan menggunakan property position CSS ini kita dapat menentukan posisi sebuah element HTML sesuai dengan yang kita inginkan. **belajar css mengenal position css**.

Adapun beberapa property CSS yang dapat digunakan untuk menentukan posisi sebuah element HTML adalah

- static
- relative
- fixed
- absolute

secara umum untuk membuat posisi atau **menetapkan posisi sebuah element** kita pasti menggunakan property css lainnya seperti mengatur top, left, bottom, right untuk menetapkan posisi sebuah element. tetapi property tersebut tidak akan bekerja jika position belum di atur terlebih dahulu. karena property lain tergantung dengan position yang di tetapkan.



Static

position static digunakan untuk **mengatur element** menjadi statis secara default. element akan mengikuti posisi normal secara default. **belajar css mengenal position css**

contoh penggunaan position static

index.html

```
1 <!DOCTYPE html>
2 <html>
3 <head>
4   <title>Mengenal Position CSS</title>
5   <link rel="stylesheet" type="text/css" href="style.css">
6 </head>
7 <body>
8   <div class="kotak">Tutorial mengenal position css di www.malasngoding.com</div>
9 </body>
10 </html>
```

style.css

```
1 .kotak{
2   background: blue;
3   position: static;
4 }
```



Relative

Sebuah element HTML yang menggunakan position relative akan terletak pada posisi normal.

index.html

```
1 <!DOCTYPE html>
2 <html>
3 <head>
4   <title>Position CSS</title>
5   <link rel="stylesheet" type="text/css" href="style.css">
6 </head>
7 <body>
8   <div class="kotak">Tutorial position css di www.malasngoding.com</div>
9 </body>
10 </html>
```

style.css

```
1 .kotak{
2   background: blue;
3   position: relative;
4   left: 40px;
5   padding: 10px;
6 }
```



Fixed

Sebuah **element HTML** yang di setting dengan position fixed akan memiliki sifat tetap. tanpa ada perubahan bahkan jika halaman website di scroll. berlaku pengaturan left, bottom, top dan right pada position fixed.

index.html

```
2 <html>
3 <head>
4   <title>Position CSS</title>
5   <link rel="stylesheet" type="text/css" href="style.css">
6 </head>
7 <body>
8   <div class="kotak">Tutorial position css di www.malasngoding.com</div>
9 </body>
10 </html>
```

style.css

```
1 .kotak{
2   background: blue;
3   position: fixed;
4   left: 40px;
5   padding: 10px;
6 }
```



Absolute

Element HTML yang menggunakan position absolute akan tertimpa dengan element lain. position adalah salah satu property **position css** yang sangat berguna. salah satunya adalah untuk membuat menu dropdown dengan HTML dan CSS.





Belajar CSS Mengubah border dengan CSS

Pada tutorial kali ini di www.malasngoding.com kita akan membahas tentang **manipulasi border dengan menggunakan CSS** . **CSS** memungkinkan kita untuk **memanipulasi atau mengubah ukuran, warna, dan gaya atau bentuk pada border**. Misalnya border yang berbentuk titik-titik, strip dan yang bermodel biasa saja .

untuk **contoh mengubah border dengan CSS** silahkan simak penjelasan berikut ini.

Mengubah Border Style

Pada **CSS** terdapat banyak gaya atau model untuk membuat garis. Ada yang berbentuk titik-titik, berbentuk garis yang putus-putus, garis ganda dan banyak lagi style/gaya yang bisa di gunakan pada garis dengan CSS . untuk membuat garis sekaligus memberikan value style yang kita inginkan kita bisa menggunakan property css border-style. silahkan perhatikan contoh di bawah ini.



index.html

```
1 <!DOCTYPE html>
2 <html>
3 <head>
4   <title>Border style CSS www.malasngoding.com</title>
5   <link rel="stylesheet" type="text/css" href="style.css">
6 </head>
7 <body>
8   <p id="garis_1">Garis dengan gaya solid</p>
9   <p id="garis_2">Garis dengan gaya dotted</p>
10  <p id="garis_3">Garis dengan gaya dashed</p>
11  <p id="garis_4">Garis dengan gaya double</p>
12  <p id="garis_5">Garis dengan gaya groove</p>
13  <p id="garis_6">Garis dengan gaya inset</p>
14  <p id="garis_7">Garis dengan gaya outset</p>
15  <p id="garis_8">Garis dengan gaya ridge</p>
16 </body>
17 </html>
```

style.css

```
1 #garis_1{
2   border-style:solid;
3 }
4 #garis_2{
5   border-style: dotted;
6 }
7 #garis_3{
8   border-style: dashed;
9 }
10 #garis_4{
11   border-style: double;
12 }
13 #garis_5{
14   border-style: groove;
15 }
16 #garis_6{
17   border-style: inset;
18 }
19 #garis_7{
20   border-style: outset;
21 }
22 #garis_8{
23   border-style: ridge;
24 }
```



Mengatur Ukuran Border CSS

Untuk mengatur ukuran pada border kita bisa menggunakan property css yaitu border-width.berikut ini adalah contoh cara penulisan property border-width untuk membuat ukuran garis.

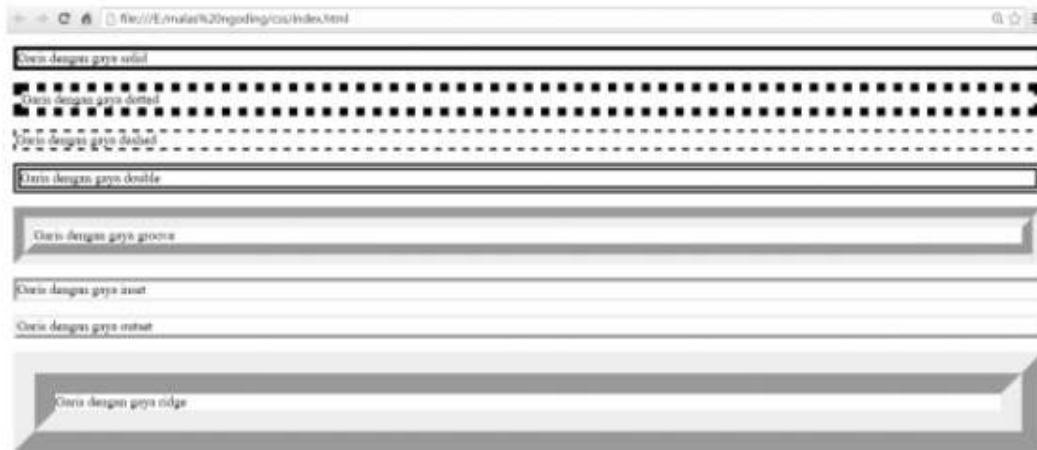
index.html

```
2 <html>
3 <head>
4   <title>Border style CSS www.malasngoding.com</title>
5   <link rel="stylesheet" type="text/css" href="style.css">
6 </head>
7 <body>
8   <p id="garis_1">Garis dengan gaya solid</p>
9   <p id="garis_2">Garis dengan gaya dotted</p>
10  <p id="garis_3">Garis dengan gaya dashed</p>
11  <p id="garis_4">Garis dengan gaya double</p>
12  <p id="garis_5">Garis dengan gaya groove</p>
13  <p id="garis_6">Garis dengan gaya inset</p>
14  <p id="garis_7">Garis dengan gaya outset</p>
15  <p id="garis_8">Garis dengan gaya ridge</p>
16 </body>
17 </html>
```

style.css

```
1 #garis_1{
2   border-style:solid;
3   border-width: 5px;
4 }
5 #garis_2{
6   border-style: dotted;
7   border-width: 10px;
8 }
9 #garis_3{
10  border-style: dashed;
11  border-width: 3px;
12 }
13 #garis_4{
14  border-style: double;
15  border-width: 9px;
16 }
17 #garis_5{
18  border-style: groove;
19  border-width: 25px;
20 }
21 #garis_6{
22  border-style: inset;
23  border-width: 5px;
24 }
25 #garis_7{
26  border-style: outset;
27  border-width: 5px;
28 }
29 #garis_8{
30  border-style: ridge;
31  border-width: 50px;
32 }
```

dan perhatikan ukuran garis setelah kita mengaturnya dengan border-width.



Memberi warna pada border

untuk memberikan warna pada garis gunakan property css yaitu **border-color**. berikut adalah contoh **membuat warna pada garis menggunakan CSS**.

index.html

```
2 <html>
3 <head>
4   <title>Border style CSS www.malasngoding.com</title>
5   <link rel="stylesheet" type="text/css" href="style.css">
6 </head>
7 <body>
8   <p id="garis_1">Garis dengan gaya solid</p>
9   <p id="garis_2">Garis dengan gaya dotted</p>
10  <p id="garis_3">Garis dengan gaya dashed</p>
11  <p id="garis_4">Garis dengan gaya double</p>
12  <p id="garis_5">Garis dengan gaya groove</p>
13  <p id="garis_6">Garis dengan gaya inset</p>
14  <p id="garis_7">Garis dengan gaya outset</p>
15  <p id="garis_8">Garis dengan gaya ridge</p>
16 </body>
17 </html>
```

style.css

```
1 #garis_1{
2   border-style:solid;
3   border-width: 5px;
4 }
5 #garis_2{
6   border-style: dotted;
7   border-width: 10px;
8   border-color:red;
9 }
10 #garis_3{
11   border-style: dashed;
12   border-width: 3px;
13   border-color:blue;
14 }
15 #garis_4{
16   border-style: double;
17   border-width: 9px;
18   border-color:green;
19 }
20 #garis_5{
21   border-style: groove;
22   border-width: 25px;
23   border-color:#12ff00;
24 }
25 #garis_6{
26   border-style: inset;
27   border-width: 5px;
28   border-color:#333333;
29 }
30 #garis_7{
31   border-style: outset;
32   border-width: 5px;
33   border-color:yellow;
34 }
35 #garis_8{
36   border-style: ridge;
37   border-width: 50px;
38   border-color: violet;
39 }
```



Cara cepat memanipulasi border dengan CSS

Cara untuk memanipulasi atau mengubah border dengan cepat menggunakan CSS kita bisa langsung menggunakan property css border. dan mengisi langsung value ukuran, jenis garis dan warna. untuk **cara cepat memanipulasi border dengan css** silahkan perhatikan contoh berikut ini.

index.html

```
2 <html>
3 <head>
4   <title>Border style CSS www.malasngoding.com</title>
5   <link rel="stylesheet" type="text/css" href="style.css">
6 </head>
7 <body>
8   <p id="garis_1">Garis dengan gaya solid</p>
9   <p id="garis_2">Garis dengan gaya dotted</p>
10  <p id="garis_3">Garis dengan gaya dotted</p>
11 </body>
12 </html>
```

style.css

```
1 #garis_1{
2   border:1px solid blue;
3 }
4 #garis_2{
5   border:10px dotted green;
6 }
7 #garis_3{
8   border:30px dashed #12ff00;
9 }
```


perhatikan pada contoh syntax css di atas. untuk mengubah border dengan cepat bisa langsung menyisipkan ukuran, jenis border dan warna yang di inginkan di dalam property border. **Belajar CSS Mengubah border dengan CSS**

```
1 border:1px solid blue;
```

contoh di atas berarti border yang di maksud berukuran 1px, dengan jenis **border solid** dan berwarna biru. maka hasilnya.



Dan berikut adalah beberapa property lain yang di gunakan untuk mengubah tampilan border dengan CSS.

- **border-bottom**, Mengatur garis yang terletak di bawah.
- **border-bottom-color**, Mengatur warna garis yang terletak di bawah.
- **border-bottom-style**, Mengatur style garis yang terletak di bawah.
- **border-bottom-width**, Mengatur ukuran garis yang terletak di bawah.
- **border-color**, Mengatur warna garis.
- **border-left**, Mengatur garis yang terletak di sebelah kiri.
- **border-left-color**, Mengatur warna garis yang terletak di sebelah kiri.
- **border-left-style**, Mengatur style garis yang terletak di sebelah kiri.
- **border-left-width**, Mengatur ukuran garis yang terletak di sebelah kiri.
- **border-right**, Mengatur garis yang terletak di sebelah kanan.
- **border-right-color**, Mengatur warna garis yang terletak di sebelah kanan.
- **border-right-style**, Mengatur style garis yang terletak di sebelah kanan.
- **border-right-width**, Mengatur ukuran garis yang terletak di sebelah kanan.
- **border-style**, Mengatur style garis.
- **border-top**, Mengatur garis yang terletak di sebelah atas.
- **border-top-color**, Mengatur warna garis yang terletak di sebelah atas.
- **border-top-style**, Mengatur style garis yang terletak di sebelah atas.
- **border-top-width**, Mengatur ukuran garis yang terletak di sebelah atas.
- **border-width**, Mengatur ukuran garis.

Belajar CSS Mengubah List Dengan CSS

CSS memiliki kemampuan untuk memanipulasi atau mengubah list HTML. CSS sangat berguna ketika anda ingin membuat daftar list dengan model tertentu misalnya list yang berbentuk angka, titik, angka romawi dan lainnya. Dengan menggunakan CSS kita dapat memanipulasi list HTML dengan mengubah bentuk tanda listnya. Untuk mengubah list HTML dengan CSS kita bisa menggunakan property "list-style-type" yang berarti tipe gaya list.

Untuk mengubah list HTML dengan CSS perhatikan contoh berikut ini.

style.css

```
1 h1{
2   text-align: center;
3 }
4 ul.makanan{
5   list-style-type: square; /* list dengan bentuk square */
6 }
7 ul.minuman{
8   list-style-type: circle; /* list dengan bentuk lingkaran */
9 }
10 ol.alamat{
11   list-style-type: upper-alpha; /* list dengan bentuk alpha */
12 }
13 ol.mobil{
14   list-style-type: upper-roman; /* list dengan bentuk romawi */
15 }
```

```
1 <!DOCTYPE html>
2 <html>
3 <head>
4   <title>Border style CSS www.malasngoding.com</title>
5   <link rel="stylesheet" type="text/css" href="style.css">
6 </head>
7 <body>
8   <h1>Tutorial cara mengubah list dengan CSS di <br/> www.malasngoding.com</h1>
9   <!-- unordered list -->
10  <ul class="makanan">
11    <li>Bakso</li>
12    <li>Mie Goreng</li>
13    <li>Sate Padang</li>
14    <li>Rujak</li>
15  </ul>
16  <ul class="minuman">
17    <li>Soft drink</li>
18    <li>Teh manis</li>
19    <li>Kopi</li>
20    <li>Jus jeruk</li>
21    <li>Susu</li>
22  </ul>
23
24  <!-- ordered list -->
25  <ol class="alamat">
26    <li>Bali</li>
27    <li>Jawa Timur</li>
28    <li>Jawa Barat</li>
29    <li>Jakarta</li>
30  </ol>
31  <ol class="mobil">
32    <li>Sedan</li>
33    <li>Mini Bus</li>
34    <li>Truk</li>
35    <li>Pick Up</li>
36  </ol>
37 </body>
38 </html>
```



Belajar CSS Mengubah List Dengan CSS . dapat di lihat pada contoh di atas bahwa untuk membuat list dengan bentuk square bisa menggunakan property dan value css:

```
1 list-style-type: square; /* list dengan bentuk square */
```

untuk membuat list dengan bentuk lingkaran kecil bisa menggunakan property dan value css:

```
1 list-style-type: circle; /* list dengan bentuk lingkaran */
```

untuk membuat list dengan bentuk alphabet bisa menggunakan property dan value css:

```
1 list-style-type: upper-alpha; /* list dengan bentuk alphabet */
```

untuk membuat list dengan bentuk angka romawi bisa menggunakan property dan value css:

```
1 list-style-type: upper-roman; /* list dengan bentuk romawi */
```

sedikit tambahan untuk menghilangkan tanda list anda bisa menggunakan property dan value sebagai berikut:

```
1 list-style-type: none; /* menghilangkan tanda list */
```

Sekian lah penjelasan singkat tentang cara mengubah list dengan CSS pada tutorial belajar CSS mengubah list dengan CSS di www.malasngoding.com.

Membuat Sudut Melengkung Dengan CSS3

Untuk contoh nya buatlah sebuah file html dan sebuah file css. atau teman-teman bisa juga menyisipkan css nya langsung ke file html nya. baca tentang [cara penulisan css](#). di sini kita akan membuat beberapa element html dengan bentuk kotak. dan nantinya akan kita terapkan perintah border-radius css3 ini untuk membuat sudut dari element nya melengkung.

index.html

```
1 <!DOCTYPE html>
2 <html>
3 <head>
4   <title>Membuat sudut melengkung dengan css3 | www.malasngoding.com</title>
5   <link rel="stylesheet" type="text/css" href="style.css">
6 </head>
7 <body>
8   <h1>Membuat sudut melengkung dengan css3 | www.malasngoding.com</h1>
9
10  <div class="ketengah">
11    <div class="kotak kotak1">kotak 1</div>
12    <div class="kotak kotak2">kotak 2</div>
13    <div class="kotak kotak3">kotak 3</div>
14    <div class="kotak kotak4">kotak 4</div>
15    <div class="kotak kotak5">kotak 5</div>
16    <div class="kotak kotak6">kotak 6</div>
17    <div class="kotak kotak7">kotak 7</div>
18    <div class="kotak kotak8">kotak 8</div>
19  </div>
20 </body>
21 </html>
```

```
1 body{
2   background: #35A9DB;
3   font-family: roboto;
4   text-align: center;
5 }
6
7 h1{
8   color: #fff;
9 }
10
11 .ketengah{
12   margin: 10px auto;
13   width: 1150px;
14 }
15
16 .kotak{
17   background: #fcfcfc;
18   padding: 20px;
19   width: 200px;
20   float: left;
21   margin: 20px;
22   height: 200px;
23 }
24
25 .kotak1{
26   border-radius: 10px 10px 10px 10px;
27 }
28
29
30 .kotak2{
31   border-radius: 40px 20px 60px 90px;
32 }
33
34 .kotak3{
35   border-radius: 2px 140px 20px 60px;
36 }
37
```

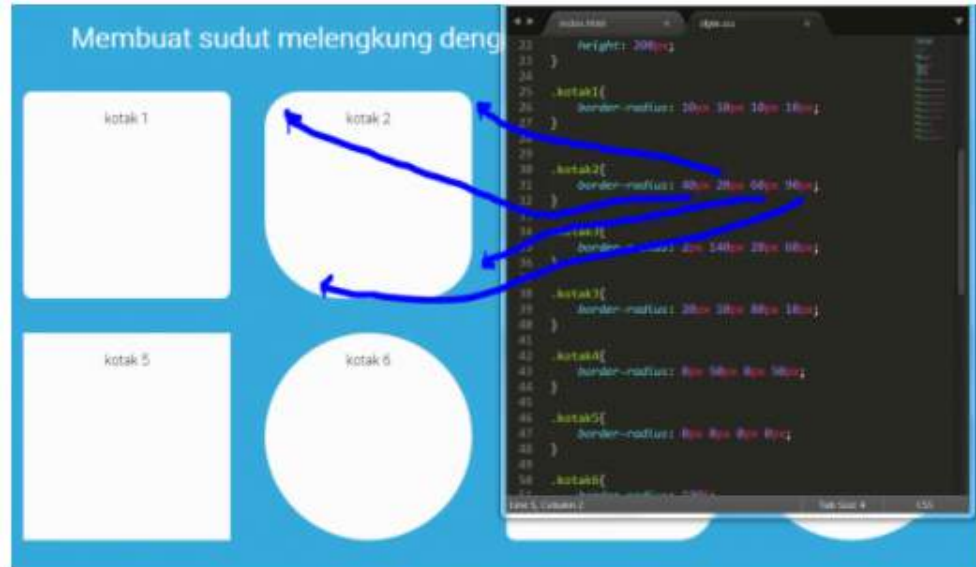
```
38 .kotak3{
39     border-radius: 20px 10px 80px 10px;
40 }
41
42 .kotak4{
43     border-radius: 0px 50px 0px 50px;
44 }
45
46 .kotak5{
47     border-radius: 0px 0px 0px 0px;
48 }
49
50 .kotak6{
51     border-radius: 100%;
52 }
53
54 .kotak7{
55     border-radius: 40px 10px;
56 }
57
58 .kotak8{
59     border-radius: 10px 10px 50% 50%;
60 }
```

dan jalankan



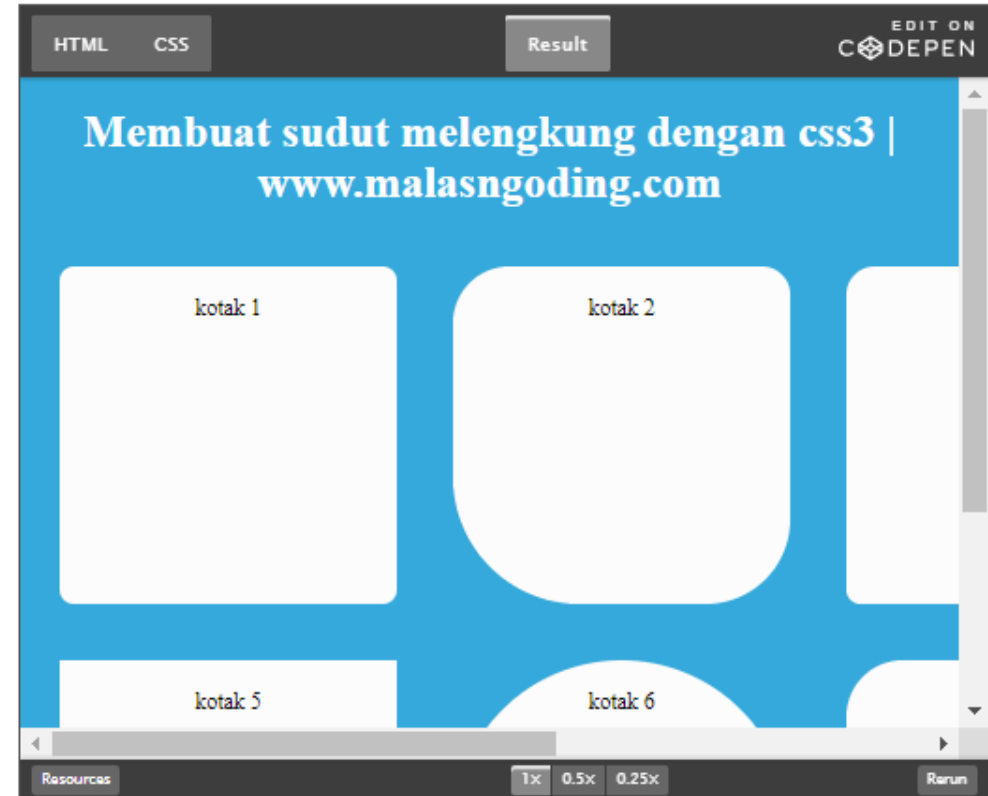
membuat sudut melengkung dengan css3

pertama masukkan nilai untuk sudut kiri atas, yang kedua nilai sudut untuk kanan atas, ketiga nilai untuk sudut kanan bawah dan yang keempat nilai untuk sudut kiri bawah.



membuat sudut melengkung dengan css3

dan teman-teman bisa menggunakan nilai dalam bentuk persen (%) juga. contoh nya seperti **membuat lingkaran** yang bisa teman-teman lihat pada kotak 6 di atas. juga bisa hanya memberikan dua nilai value. jika menggunakan dua buah nilai value maka nilai yang pertama di baca nilai atas dan nilai kedua di baca nilai bawah. teman-teman juga bisa hanya memberikan satu nilai value. maka satu nilai value ini akan di gunakan untuk tiap sudut element.



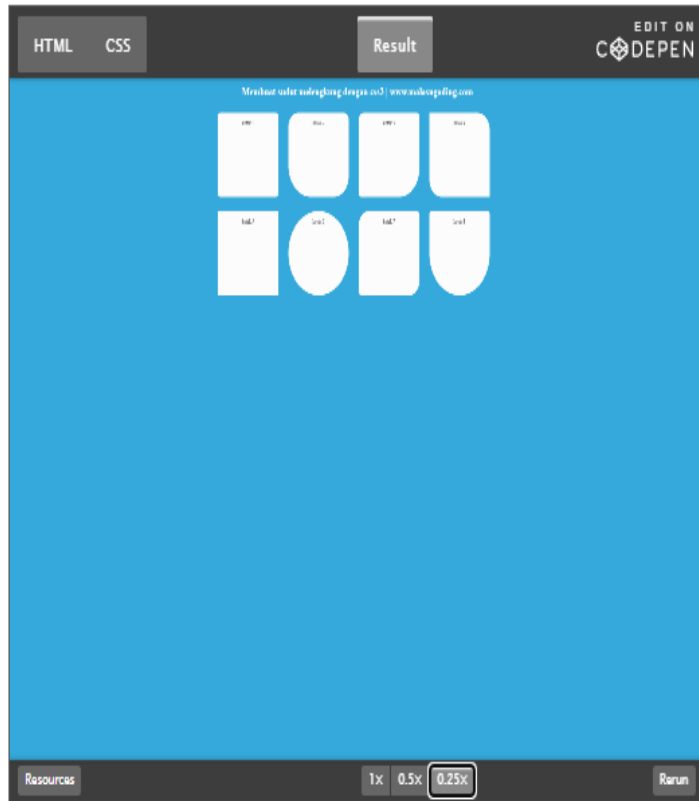


HTML CSS **Result** EDIT ON CODEPEN

Membuat sudut melengkung dengan css3 | www.malasngoding.com

1x 0.5x 0.25x Run

Resources



```
HTML CSS LIVE
<h1>Membuat sudut melengkung dengan css3
| www.malasngoding.com</h1>

<div class="ketengah">
<div class="kotak kotak1">kotak
1</div>
<div class="kotak kotak2">kotak
2</div>
<div class="kotak kotak3">kotak
3</div>
<div class="kotak kotak4">kotak
4</div>
<div class="kotak kotak5">kotak
5</div>
<div class="kotak kotak6">kotak
6</div>
<div class="kotak kotak7">kotak
7</div>
<div class="kotak kotak8">kotak
8</div>
</div>
```

```
HTML CSS LIVE
body {
background: #35A9D8;
font-family: roboto;
text-align: center;
}

h1 {
color: #fff;
}

.ketengah {
margin: 18px auto;
width: 1150px;
}

.kotak {
background: #fcfcfc;
padding: 20px;
width: 200px;
float: left;
margin: 20px;
height: 200px;
}

.kotak1 {
border-radius: 10px 10px 10px 10px;
```

```
HTML CSS LIVE
.kotak2 {
border-radius: 40px 20px 60px 90px;
}

.kotak3 {
border-radius: 2px 140px 20px 60px;
}

.kotak3 {
border-radius: 20px 10px 80px 10px;
}

.kotak4 {
border-radius: 0px 50px 0px 50px;
}

.kotak5 {
border-radius: 0px 0px 0px 0px;
}

.kotak6 {
border-radius: 100%;
}

.kotak7 {
```

Tutorial CSS3 Part 3 – Membuat Warna Gradient CSS3

Ada 2 tipe warna gradient atau gradasi yang dapat di buat dengan css3. yaitu **linear gradient** dan **radial gradient**. linear gradient adalah gradient yang berbentuk warna yang berdampingan tapi efeknya sangat lembut. jadi kedua warna akan terlihat menyatu. baik itu atas ke bawah atau kiri ke kanan. sedangkan radial gradient adalah warna gradient yang memiliki pusat tengah. jadi seperti membentuk lingkaran dalam pewarnaannya. kita lihat langsung ke contoh. **Tutorial CSS3 Part 3 – Membuat Warna Gradient CSS3**

index.html

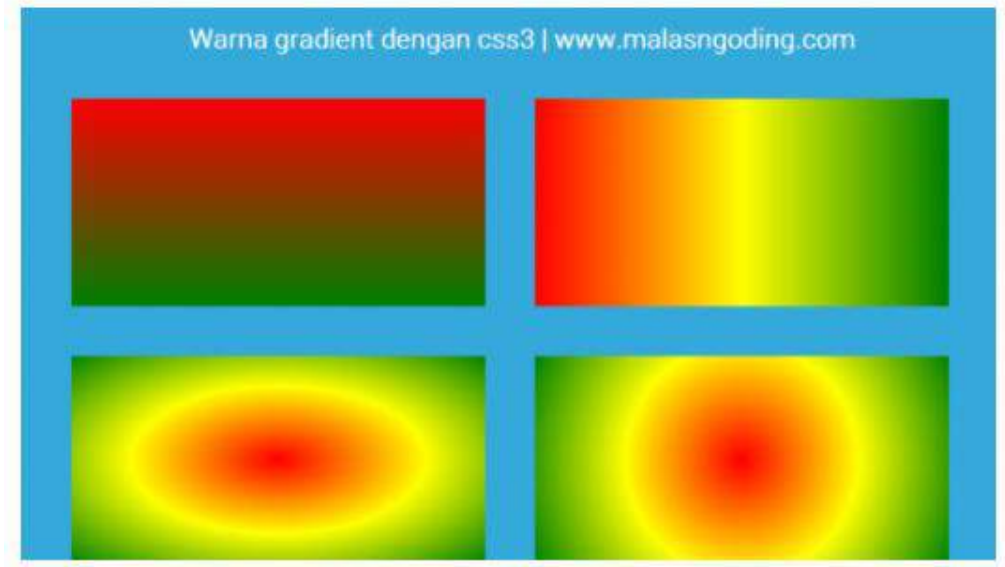
```
1 <!DOCTYPE html>
2 <html>
3 <head>
4   <title>Warna gradient dengan css3 | www.malasngoding.com</title>
5   <link rel="stylesheet" type="text/css" href="style.css">
6 </head>
7 <body>
8   <h1>Warna gradient dengan css3 | www.malasngoding.com</h1>
9
10  <div class="ketengah">
11    <!-- linear gradient -->
12    <div class="kotak gradient1"></div>
13
14    <div class="kotak gradient2"></div>
15
16    <!-- radial gradient -->
17    <div class="kotak gradient3"></div>
18
19    <div class="kotak gradient4"></div>
20  </div>
21 </body>
22 </html>
```

in

style.css

```
2   background: #35A9DB;
3   font-family: roboto;
4   text-align: center;
5 }
6
7 h1{
8   color: #fff;
9 }
10
11 .ketengah{
12   margin: 10px auto;
13   width: 1150px;
14 }
15
16 .kotak{
17   background: blue;
18   width: 500px;
19   float: left;
20   height: 250px;
21   margin: 30px;
22 }
23
24 .gradient1 {
25   background: linear-gradient(red, green); /* Standard syntax */
26   background: -webkit-linear-gradient(red, green); /* For Safari 5.1 to 6.0 */
27   background: -o-linear-gradient(red, green); /* For Opera 11.1 to 12.0 */
28   background: -moz-linear-gradient(red, green); /* For Firefox 3.6 to 15 */
29 }
30
```

```
31 .gradient2 {  
32   background: linear-gradient(to right, red , yellow, green); /* Standard syntax */  
33   background: -webkit-linear-gradient(left, red , yellow, green); /* For Safari 5.1 to 6.0 */  
34   background: -o-linear-gradient(right, red, yellow, green); /* For Opera 11.1 to 12.0 */  
35   background: -moz-linear-gradient(right, red, yellow, green); /* For Firefox 3.6 to 15.0 */  
36 }  
37  
38 .gradient3 {  
39   background: radial-gradient(red, yellow, green); /* Standard syntax */  
40   background: -webkit-radial-gradient(red, yellow, green); /* Safari 5.1 to 6.0 */  
41   background: -o-radial-gradient(red, yellow, green); /* For Opera 11.6 to 12.0 */  
42   background: -moz-radial-gradient(red, yellow, green); /* For Firefox 3.6 to 15.0 */  
43 }  
44  
45 .gradient4 {  
46   background: radial-gradient(circle, red, yellow, green); /* Standard syntax */  
47   background: -webkit-radial-gradient(circle, red, yellow, green); /* Safari */  
48   background: -o-radial-gradient(circle, red, yellow, green); /* Opera 11.6 to 12.0 */  
49   background: -moz-radial-gradient(circle, red, yellow, green); /* Firefox 3.6 to 15.0 */  
50 }
```



Membuat Efek Bayangan Dengan CSS3 (Box Shadow)

untuk membuat efek bayangan pada sebuah element html kita bisa menggunakan perintah **box-shadow**. jadi cara nya sangat mudah. coba teman-teman perhatikan pada contoh penulisannya berikut ini.

```
1 box-shadow:rata_atas_bawah rata_kiri_kanan jarak_bayangan ketebalan_bayangan warna;
```

langsung saja masuk ke contoh penggunaannya. jadi kita bisa menggunakan box shadow atau efek bayangan css3 ini di element html. dan jika kita ingin membuat efek bayangan pada text atau tulisan kita bisa menggunakan perintah text-shadow. silahkan teman-teman perhatikan pada contoh berikut ini.

- text-shadow = untuk membuat bayangan pada text
- box-shadow = untuk membuat bayangan pada element html selain text

index.html

```
2 <html>
3 <head>
4   <title>Membuat Efek Bayangan Dengan CSS3 | www.malasngoding.com</title>
5   <link rel="stylesheet" type="text/css" href="style.css">
6 </head>
7 <body>
8   <h1>Membuat Efek Bayangan Dengan CSS3 | www.malasngoding.com</h1>
9
10  <div class="ketengah">
11
12    <h1 class="tulisan1">MALAS NGODING</h1>
13
14    <h1 class="tulisan2">MALAS NGODING</h1>
15
16    <div class="kotak1">kotak 1</div>
17
18    <div class="kotak2">kotak 2</div>
19
20  </div>
21 </body>
22 </html>
```



style.css

```
1 body{
2   background: #35A9DB;
3   font-family: sans-serif;
4   text-align: center;
5 }
6
7 h1{
8   color: #fff;
9 }
10
11 .ketengah{
12   margin: 10px auto;
13   width: 1150px;
14 }
15
16
17 /*
18 nilai 20 adalah nilai tinggi rendah nya bayangan
19 nilai 10 adalah nilai untuk mengatur rata kiri kanan bayangan
20 nilai 5 adalah nilai untuk mengatur ketebalan bayangan
21 dan yang terakhir adalah mengatur warna bayangan
22 */
23 .tulisan1{
24   text-shadow: 20px 10px 5px #232323;
25 }
26
27 /*
28 nilai 0 pertama adalah nilai tinggi rendah nya bayangan
29 nilai 0 kedua adalah nilai untuk mengatur rata kiri kanan bayangan
30 nilai 20 adalah nilai untuk mengatur ketebalan bayangan
31 dan yang terakhir adalah mengatur warna bayangan
32 */
```

```
33 .tulisan2{
34   text-shadow: 0px 0px 20px #232323;
35 }
36
37
38
39 /*
40 nilai 20 yang pertama adalah nilai tinggi rendah nya bayangan
41 nilai 20 kedua adalah nilai untuk rata kiri kanan bayangan
42 nilai 40 adalah nilai radius bayangan
43 nilai 10 adalah nilai ketebalan bayangan
44 dan yang terakhir adalah menentukan warna bayangan
45 */
46 .kotak1{
47   background: white;
48   width: 300px;
49   height: 300px;
50   margin: 20px;
51   float: left;
52   box-shadow: 20px 20px 40px 10px red;
53 }
54
55
56 /*
57 nilai 0 yang pertama adalah nilai tinggi rendah nya bayangan
58 nilai 0 kedua adalah nilai untuk rata kiri kanan bayangan
59 nilai 10 pertama adalah nilai radius bayangan
60 nilai 10 kedua adalah nilai ketebalan bayangan
61 dan yang terakhir adalah menentukan warna bayangan
62 */
63 .kotak2{
64   background: white;
65   width: 300px;
66   margin: 20px;
67   float: left;
68   height: 300px;
69   box-shadow: 0px 0px 10px 10px #333333;
70 }
```

dan hasilnya



cara membuat efek bayangan dengan css3

Tutorial CSS3 Part 5 – Belajar CSS3 Transform

Tutorial CSS3 Part 5 – Belajar CSS3 Transform – pada kesempatan kali ini kita akan masuk ke tutorial css transform css. css transform adalah salah satu yang terpenting yang harus di pahami bagi teman-teman yang ingin membuat efek animasi dengan css3. teman-teman nantinya bisa membuat efek memutar element, membuat element bergerak dan sebagainya. yang salah satu yang menjadi dasarnya adalah css transform. oleh sebab itu kita akan mempelajari css transform terlebih dulu di tutorial css3 part 5 ini. baru kemudian nya di tutorial selanjutnya kita akan belajar tentang css transition dan css animation.

CSS Transform atau Transformasi css memungkinkan kan kita untuk membuat efek rotate(memutar element), scale,dan skew pada element html. masing-masing dari efek ini akan kita pelajari satu persatu. dengan css transformation ini juga memungkinkan kita dapat mengganti warna element html, mengubah ukuran serta mengubah posisi element [html](#). silahkan teman-teman perhatikan pada penjelasan berikut ini. *Tutorial CSS3 Part 5 – Belajar CSS3 Transform*

Berikut ini adalah beberapa metode transform css:

- **translate()**
translate() adalah fungsi yang bisa di gunakan untuk memindahkan sebuah element. kita bisa menentukan posisi nye dengan memberikan niai pada parameter x untuk jarak samping dan parameter y untuk jarak tingginya.
- **scale()**
fungsi method scale() ini di gunakan untuk melebarkan atau menjauhkan element dengan menambah atau mengurangi ukuran element. seperi effect zoom lah kira-kiranya.
- **skewX()**
Skew() di gunakan untuk memiringkan element dengan mengatur tiap nilai pada sumbu x nya.
- **skewY()**
Skew() di gunakan untuk memiringkan element dengan mengatur tiap nilai pada sumbu y nya.
- **rotate()**
rotate() adalah fungsi yang bisa kita gunakan untuk memutar sebuah element. untuk memutar elemeny nya kita bisa menentukan nilainya di dalam parameter fungsi rotate ini. untuk nilainya sendiri bisa kita tentukan dengan cara memberikan nilai derajat putar nya.
- **matrix()**
fungsi matrix berguna untuk menggabungkan beberapa transform lainnya seperti menggabungkan rotate(), skew() scale() dan lainnya menjadi satu. sehingga transform tersebut akan di lakukan secara sekaligus.

Contoh CSS Transform : translate()

seperti yang sudah di jelaskan pada penjealasan di atas. fungsi translate ini di gunakan untuk memindahkan element html. cara penulisannya adalah

```
1 translate(nilai_jarak_samping, nilai_jarak_atas)
```

pada parameter pertama untuk nilai jarak kiri kanan nya. pada parameter kedua berikan nilai untuk jarak atas bawah nya. baca juga [Membuat efek hover zoom dengan css3](#). [Tutorial CSS3 Part 5 – Belajar CSS3 Transform.](#)

translate.html

```
1 <!DOCTYPE html>
2 <html>
3 <head>
4   <title>Tutorial CSS Transform | www.malasngoding.com</title>
5   <link rel="stylesheet" type="text/css" href="style.css">
6 </head>
7 <body>
8   <h1>Tutorial CSS Transform | www.malasngoding.com</h1>
9   <h2>translate()</h2>
10  <div class="kotak1">
11    Ini adalah kotak 1. kotak ini tidak di berikan method translate
12  </div>
13  <div class="kotak2">
14    Ini adalah kotak 2. ini contoh fungsi translate() CSS Transform.
15  </div>
16 </body>
17 </html>
```



UKRIDA

```
1 body{
2   background: #35A9DB;
3   font-family: roboto;
4   text-align: center;
5 }
6
7 h1,h2{
8   color: #fff;
9 }
10
11 .ketengah{
12   margin: 10px auto;
13   width: 1150px;
14 }
15
16 .kotak1{
17   background: #fcfcfc;
18   padding: 20px;
19   float: left;
20   margin: 20px;
21   width: 150px;
22   height: 150px;
23 }
24
25 .kotak2{
26   background: #fcfcfc;
27   padding: 20px;
28   float: left;
29   margin: 20px;
30   width: 150px;
31   height: 150px;
32   -ms-transform: translate(50px,100px); /* Support IE 9 */
33   -webkit-transform: translate(50px,100px); /* support Safari */
34   transform: translate(50px,100px); /* Standard syntax */
35 }
```

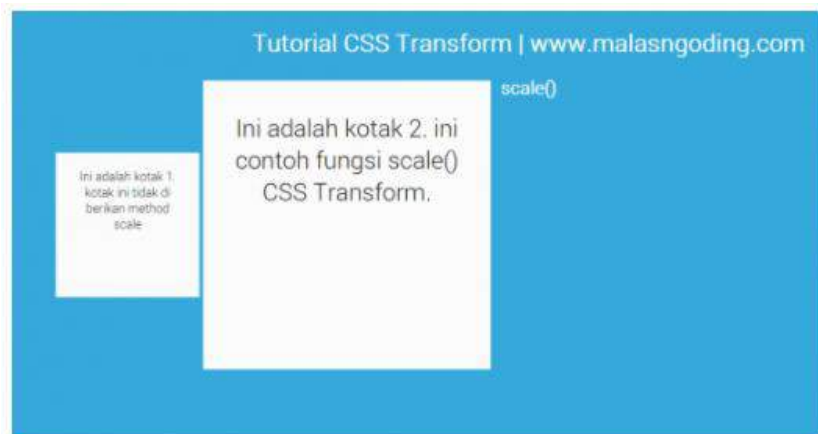

Contoh CSS Transform : scale()

scale.html

```
1 <!DOCTYPE html>
2 <html>
3 <head>
4   <title>Tutorial CSS Transform | www.malasngoding.com</title>
5   <link rel="stylesheet" type="text/css" href="scale.css">
6 </head>
7 <body>
8   <h1>Tutorial CSS Transform | www.malasngoding.com</h1>
9   <h2>scale()</h2>
10  <div class="kotak1">
11    Ini adalah kotak 1. kotak ini tidak di berikan method scale
12  </div>
13  <div class="kotak2">
14    Ini adalah kotak 2. ini contoh fungsi scale() CSS Transform.
15  </div>
16 </body>
17 </html>
```

scale.css

```
2   background: #35A9DB;
3   font-family: roboto;
4   text-align: center;
5 }
6
7 h1,h2{
8   color: #fff;
9 }
10
11 .ketengah{
12   margin: 10px auto;
13   width: 1150px;
14 }
15
16 .kotak1{
17   background: #fcfcfc;
18   padding: 20px;
19   float: left;
20   margin: 50px;
21   width: 150px;
22   height: 150px;
23 }
24
25 .kotak2{
26   background: #fcfcfc;
27   padding: 20px;
28   float: left;
29   margin: 50px;
30   width: 150px;
31   height: 150px;
32   -ms-transform: scale(2,3); /* Support IE 9 */
33   -webkit-transform: scale(2,3); /* support Safari */
34   transform: scale(2,2); /* Standard syntax */
35 }
```



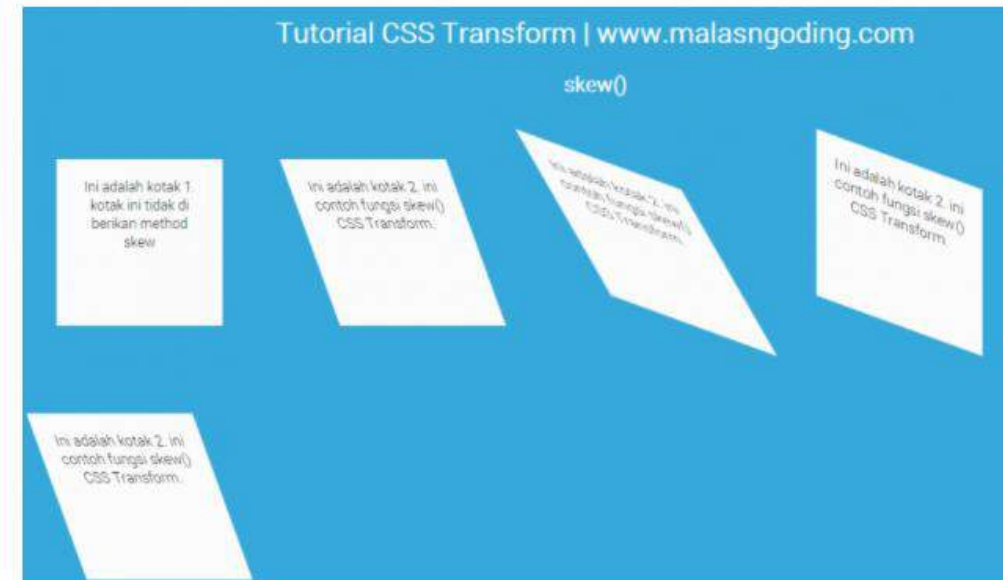
Contoh CSS Transform : skew()

skew.html

```
2 <html>
3 <head>
4   <title>Tutorial CSS Transform | www.malasngoding.com</title>
5   <link rel="stylesheet" type="text/css" href="skew.css">
6 </head>
7 <body>
8   <h1>Tutorial CSS Transform | www.malasngoding.com</h1>
9   <h2>skew()</h2>
10  <div class="kotak1">
11    Ini adalah kotak 1. kotak ini tidak di berikan method skew
12  </div>
13  <div class="kotak2">
14    Ini adalah kotak 2. ini contoh fungsi skew() CSS Transform. skew().
15  </div>
16  <div class="kotak3">
17    Ini adalah kotak 2. ini contoh fungsi skew() CSS Transform. skew() 2 paramete
18  </div>
19  <div class="kotak4">
20    Ini adalah kotak 2. ini contoh fungsi skew() CSS Transform. skewY().
21  </div>
22  <div class="kotak5">
23    Ini adalah kotak 2. ini contoh fungsi skew() CSS Transform. skewX().
24  </div>
25
26 </body>
27 </html>
```

```
1 body{
2   background: #35A9DB;
3   font-family: roboto;
4   text-align: center;
5 }
6
7 h1,h2{
8   color: #fff;
9 }
10
11 .ketengah{
12   margin: 10px auto;
13   width: 1150px;
14 }
15
16 .kotak1{
17   background: #fcfcfc;
18   padding: 20px;
19   float: left;
20   margin: 50px;
21   width: 150px;
22   height: 150px;
23 }
24
25 .kotak2{
26   background: #fcfcfc;
27   padding: 20px;
28   float: left;
29   margin: 50px;
30   width: 150px;
31   height: 150px;
32   -ms-transform: skew(20deg); /* Support IE 9 */
33   -webkit-transform: skew(20deg); /* support Safari */
34   transform: skew(20deg); /* Standard syntax */
35 }
```

```
37 .kotak3{
38   background: #fcfcfc;
39   padding: 20px;
40   float: left;
41   margin: 50px;
42   width: 150px;
43   height: 150px;
44   -ms-transform: skew(30deg,20deg); /* Support IE 9 */
45   -webkit-transform: sken(30deg,20deg); /* support Safari */
46   transform: skew(30deg,20deg); /* Standard syntax */
47 }
48
49 .kotak4{
50   background: #fcfcfc;
51   padding: 20px;
52   float: left;
53   margin: 50px;
54   width: 150px;
55   height: 150px;
56   -ms-transform: skewY(20deg); /* Support IE 9 */
57   -webkit-transform: skewY(20deg); /* support Safari */
58   transform: skewY(20deg); /* Standard syntax */
59 }
60
61 .kotak5{
62   background: #fcfcfc;
63   padding: 20px;
64   float: left;
65   margin: 50px;
66   width: 150px;
67   height: 150px;
68   -ms-transform: skewX(20deg); /* Support IE 9 */
69   -webkit-transform: skewX(20deg); /* support Safari */
70   transform: skewX(20deg); /* Standard syntax */
71 }
```



Contoh CSS Transform : rotate()

rotate.html

```
1 <!DOCTYPE html>
2 <html>
3 <head>
4   <title>Tutorial CSS Transform | www.malasngoding.com</title>
5   <link rel="stylesheet" type="text/css" href="rotate.css">
6 </head>
7 <body>
8   <h1>Tutorial CSS Transform | www.malasngoding.com</h1>
9   <h2>rotate()</h2>
10  <div class="kotak1">
11    Ini adalah kotak 1. kotak ini tidak di berikan method rotate
12  </div>
13  <div class="kotak2">
14    Ini adalah kotak 2. ini contoh fungsi rotate() CSS Transform. rotate(30deg).
15  </div>
16  <div class="kotak3">
17    Ini adalah kotak 2. ini contoh fungsi rotate() CSS Transform. rotate(90deg).
18  </div>
19  <div class="kotak4">
20    Ini adalah kotak 2. ini contoh fungsi rotate() CSS Transform. rotate(-90deg).
21  </div>
22
23 </body>
24 </html>
```

rotate.css

```
1 body{
2   background: #35A9DB;
3   font-family: roboto;
4   text-align: center;
5 }
6
7 h1,h2{
8   color: #fff;
9 }
10
11 .ketengah{
12   margin: 10px auto;
13   width: 1150px;
14 }
15
16 .kotak1{
17   background: #fcfcfc;
18   padding: 20px;
19   float: left;
20   margin: 50px;
21   width: 150px;
22   height: 150px;
23 }
24
```

```
25 .kotak2{
26   background: #fcfcfc;
27   padding: 20px;
28   float: left;
29   margin: 50px;
30   width: 150px;
31   height: 150px;
32   -ms-transform: rotate(30deg); /* Support IE 9 */
33   -webkit-transform: rotate(30deg); /* support Safari */
34   transform: rotate(30deg); /* Standard syntax */
35 }
36
37 .kotak3{
38   background: #fcfcfc;
39   padding: 20px;
40   float: left;
41   margin: 50px;
42   width: 150px;
43   height: 150px;
44   -ms-transform: rotate(90deg); /* Support IE 9 */
45   -webkit-transform: rotate(90deg); /* support Safari */
46   transform: rotate(90deg); /* Standard syntax */
47 }
48
```

```
48
49 .kotak4{
50   background: #fcfcfc;
51   padding: 20px;
52   float: left;
53   margin: 50px;
54   width: 150px;
55   height: 150px;
56   -ms-transform: rotate(-60deg); /* Support IE 9 */
57   -webkit-transform: rotate(-60deg); /* support Safari */
58   transform: rotate(-60deg); /* Standard syntax */
59 }
```

Tutorial CSS Transform | www.malasngoding.com

rotate()



Ini adalah kotak 1
kotak ini tidak di
berikan method
rotate

Ini adalah kotak 2. ini
contoh fungsi
rotate() CSS
Transform
rotate(30deg)

Ini adalah kotak 2. ini
contoh fungsi
rotate() CSS
Transform
rotate(90deg)

Ini adalah kotak 2. ini
contoh fungsi
rotate() CSS
Transform
rotate(-90deg)

Contoh CSS Transform : matrix()

contoh penulisan fungsi matrix adalah sebagai berikut

```
1 matrix(scaleX(),skewY(),skewX(),scaleY(),translateX(),translateY())
```

matrix.html

```
2 <html>
3 <head>
4   <title>Tutorial CSS Transform | www.malasngoding.com</title>
5   <link rel="stylesheet" type="text/css" href="matrix.css">
6 </head>
7 <body>
8   <h1>Tutorial CSS Transform | www.malasngoding.com</h1>
9   <h2>matrix()</h2>
10  <div class="kotak1">
11    Ini adalah kotak 1. kotak ini tidak di berikan method matrix
12  </div>
13  <div class="kotak2">
14    Ini adalah kotak 2. ini contoh fungsi matrix() CSS Transform.
15  </div>
16
17 </body>
18 </html>
```

matrix.css

```
2   background: #35A9DB;
3   font-family: roboto;
4   text-align: center;
5 }
6
7 h1,h2{
8   color: #fff;
9 }
10
11 .ketengah{
12   margin: 10px auto;
13   width: 1150px;
14 }
15
16 .kotak1{
17   background: #fcfcfc;
18   padding: 20px;
19   float: left;
20   margin: 50px;
21   width: 150px;
22   height: 150px;
23 }
24
```



```
25 .kotak2{
26   background: #fcfcfc;
27   padding: 20px;
28   float: left;
29   margin: 50px;
30   width: 150px;
31   height: 150px;
32   matrix(scaleX(),skewY(),skewX(),scaleY(),translateX(),translateY())
33   -ms-transform: matrix(1,2,-1,1,5,4); /* Support IE 9 */
34   -webkit-transform: matrix(1,2,-1,1,5,4); /* support Safari */
35   transform: matrix(1,2,-1,1,5,4); /* Standard syntax */
36 }
```

Tutorial CSS Transform | www.malasngoding.com

matrix()

Ini adalah kotak 1.
kotak ini tidak di berikan method matrix

Ini adalah kotak 2. ini contoh fungsi Transform CSS matrix(30deg).



Tutorial Belajar Transition di CSS3

Dengan menggunakan transition atau transisi di css3 ini dapat memungkinkan kita untuk memanipulasi property atau value sebuah element dengan sangat lembut. memberikan durasi dan banyak lagi lainnya kelebihan dan kemudahan transition css3 yang akan teman-teman rasakan sendiri nantinya setelah menggunakan transisi css3 ini. baiklah langsung saja teman-teman simak penjelasan berikut ini tentang transisi css3. **Tutorial CSS3 Part 6 – Belajar CSS3 Transition**

Ada beberapa perintah css3 transition yang bisa teman-teman coba. di antaranya adalah :

- Transition
perintah standar transisi css3. biasanya kita gunakan untuk memanipulasi property atau nilai value css dengan sangat baik.
- Transition-delay
di gunakan untuk memberikan waktu tunda pada saat peng-eksekusian sebuah efek css3.
- Transition-duration
di gunakan untuuk memberikan waktu durasi terhadap sebuah efek yang kita buat dengan css3.
- Transition-property
untuk menentukan property css yang akan di gunakan,manipulasi atau lainnya(di pilih).
- Transition-timing-function
untuk menentukan kecepatan sebuah efek yang kita buat dengan css3.

dari penjelasan singkat di atas tentang pengertian-pengertian perintah transisi css3, akan langsung kita coba cara penggunaanya masing-masing. **Tutorial CSS3 Part 6 – Belajar CSS3 Transition**

Contoh Penggunaan Dasar Transition css3

Oke sebelum lebih jauh mempelajari css3 transisi, ada baiknya saya berikan contoh dulu tentang penggunaan transition. agar teman-teman bisa lebih mudah mendapatkan bayangan tentang transition css3. cara mikir mudahnya gini teman-teman. misalnya jika transform adalah aksinya, maka transition ini adalah pelembut atau pelancar jalannya transform. seperti pendukung lah bisa di bilang. tapi di sini kita gunakan perintah dasar dulu, jangan transform dulu. **Tutorial CSS3 Part 6 – Belajar CSS3 Transition**

index.html

```
1 <!DOCTYPE html>
2 <html>
3 <head>
4   <title>Tutorial CSS3 Transition | www.malasngoding.com</title>
5   <link rel="stylesheet" type="text/css" href="style.css">
6 </head>
7 <body>
8   <h1>Tutorial CSS3 Transition | www.malasngoding.com</h1>
9   <div class="kotak">ini kotak</div>
10  <div class="kotak2">ini kotak 2</div>
11
12 </body>
13 </html>
```

style.css

```
2   background: #35A9DB;
3   font-family: roboto;
4   text-align: center;
5 }
6
7 h1{
8   color: #fff;
9 }
10
11 div{
12   margin: 10px;
13 }
14
15 .kotak{
16   background: #fff;
17   width: 100px;
18   height: 100px;
19   transition: width 1s;
20   -webkit-transition: width 1s;
21 }
22
23 .kotak:hover{
24   width: 300px;
25 }
26
27 .kotak2{
28   background: #fff;
29   width: 100px;
30   height: 100px;
31   transition: width 1s, height 2s;
32   -webkit-transition: width 1s, height 2s;
33 }
34
35 .kotak2:hover{
36   width: 300px;
37   height: 300px;
38 }
```

dan jika di jalankan contoh nya seperti berikut ini. coba letakkan cursor mouse di atas kotak.



```
1 .kotak{
2   background: #fff;
3   width: 100px;
4   height: 100px;
5   transition: width 1s;
6   -webkit-transition: width 1s;
7 }
8
9 .kotak:hover{
10  width: 300px;
11 }
```

pertama nilai width atau lebar pada kotak adalah 100px. dan kita berikan transition untuk property width dengan speed 1 detik.

```
1 transition: width 1s;
```

dan kita berikan hover pada element kotak dengan mengubah lebar kotak menjadi 300px.

```
1 .kotak:hover{
2   width: 300px;
3 }
```

jadi dapat di simpulkan di sini adalah dengan transition perubahan lebar pada kotak terjadi dengan lembut dan lebih rapi dan bagus. seperti yang teman-teman lihat pada contoh di atas.



Mengatur Speed pada transition

nah mudah-mudah penjelasan di atas tentang penggunaan dan kegunaan dari transition css3 sudah jelas dan mudah di mengerti oleh teman-teman. jika ada yang belum jelas silahkan teman-teman tanyakan saja di komentar di bawah. **Tutorial CSS3 Part 6 – Belajar CSS3 Transition**

sesuai dengan yang sudah di jelaskan di atas. untuk mengatur speed transition, kita bisa menggunakan perintah **transition-timing-function**. dan adapun beberapa value yang bisa di gunakan di antaranya adalah :

- ease – mengatur kecepatan transisi dengan permulaan yang pelan dan di akhirnya juga pelan.
- linear – kecepatan efek transition/transisi dengan kecepatan yang sama dari awal hingga akhir.
- ease-in – kecepatan efek transisi css3 dengan permulaan yang pelan.
- ease-out – kecepatan efek transisi dengan akhir yang pelan.
- ease-in-out – kecepatan efek transisi dengan akhir dan awal pelan.
- cubic-bezier(n,n,n,n) – kita bisa memberikan nilai pada n untuk menentukan nilai speed transition.



index.html

```
1 <!DOCTYPE html>
2 <html>
3 <head>
4   <title>Tutorial CSS3 Transition | www.malasngoding.com</title>
5   <link rel="stylesheet" type="text/css" href="transisi.css">
6 </head>
7 <body>
8   <h1>Tutorial CSS3 Transition | www.malasngoding.com</h1>
9   <div class="ketengah">
10
11     <div class="kotak kotak1">ini kotak 1 - linear</div>
12     <div class="kotak kotak2">ini kotak 2 - ease</div>
13     <div class="kotak kotak3">ini kotak 3 - ease-in</div>
14     <div class="kotak kotak4">ini kotak 4 - ease-out</div>
15     <div class="kotak kotak5">ini kotak 5 - ease-in-out</div>
16     <div class="kotak kotak6">ini kotak 6 - cubic-bezier</div>
17
18   </div>
19 </body>
20 </html>
```

transisi.css

```
1 body{
2   background: #35A9DB;
3   font-family: roboto;
4   text-align: center;
5 }
6
7 h1{
8   color: #fff;
9 }
10
11 .ketengah{
12   width: 800px;
13   margin: 10px auto;
14 }
15
16 .kotak{
17   width: 100px;
18   background: #fff;
19   height: 60px;
20   margin: 10px;
21   -webkit-transition: width 2s;
22   transition: width 2s;
23 }
24
25 .kotak:hover{
26   width: 400px;
27 }
28
```

```
28
29 .kotak1{
30   transition-timing-function: linear;
31 }
32
33 .kotak2{
34   transition-timing-function: ease;
35 }
36
37 .kotak3{
38   transition-timing-function: ease-in;
39 }
40
41 .kotak4{
42   transition-timing-function: ease-out;
43 }
44
45 .kotak5{
46   transition-timing-function: ease-in-out;
47 }
48
49 .kotak5{
50   transition-timing-function: cubic-bezier(2,4,5,6);
51 }
```

Tutorial CSS3 Transition | www.malasngoding.com

ini kotak 1 -
linear

ini kotak 2 -
ease

ini kotak 3 - ease-in

ini kotak 4 -
ease-out

ini kotak 5 -
ease-in-out

ini kotak 6 -
cubic-bezier



Tutorial CSS3 Part 7 – Membuat Efek Animasi CSS3

Tutorial CSS3 Part 7 – Membuat Efek Animasi CSS3 – CSS3 memiliki banyak sekali fitur yang baru. salah satunya adalah animasi. animasi adalah salah satu fitur baru dari css3. dengan css3 animation ini memungkinkan kita untuk membuat berbagai animasi dasar. dan jika di dalam lagi kita bisa membuat berbagai bentuk animasi dengan css. pada tutorial css3 part 7 ini saya ingin menjelaskan dan mengajak teman-teman yang baru belajar css3. untuk ***mengenal animasi menggunakan css3*** ini. di asumsikan untuk teman-teman yang akan mempelajari tutorial ini alangkah lebih baiknya jika memahami css dasar terlebih dulu. teman-teman bisa membaca tutorial saya sebelumnya tentang [css dasar](#). ***Tutorial CSS3 Part 7 – Membuat Efek Animasi CSS3***

Membuat Efek Animasi CSS3

CSS3 menyediakan fitur baru. yaitu animasi. dengan animasi css3 ini kita bisa membuat berbagai animasi. tanpa harus menggunakan javascript dan lainnya. sebelum membuat animasi dengan css. kita harus berkenalan dulu dengan yang namanya `@keyframes`. `@keyframes` adalah fungsi baru dari css3 untuk membuat animasi. ada banyak sekali kelebihan menggunakan css untuk membuat animasi. karena animasi yang di buat tidak berat dan tidak akan memakan bandwidth yang besar.

Mengenal @keyframes CSS3

@keyframes adalah fungsi baru dari css3 untuk mendefinisikan animasi pada sebuah element. lalu bagaimana cara menggunakan @keyframes untuk membuat animasi css3 ? yang harus di lakukan untuk menggunakan @keyframes adalah dengan mendefinisikan dulu nama animasi yang kita buat. lalu buat animasi dengan nama animasi yang di buat. bingung ? langsung aja kita masuk ke contoh membuat animasi sederhana dengan @keyframes css3. **Tutorial CSS3 Part 7 – Membuat Efek Animasi CSS3**

untuk penggunaan @keyframes ada yang namanya kondisi awal dan kondisi akhir. berikut perhatikan contohnya, sekalian memahami **cara penggunaan @keyframes css3 untuk membuat animasi**.

contoh.html

```
1 <!DOCTYPE html>
2 <html>
3 <head>
4   <title>Membuat efek animasi dengan css3 | www.malasngoding.com</title>
5   <link rel="stylesheet" type="text/css" href="contoh.css">
6 </head>
7 <body>
8   <h1>Membuat efek animasi dengan css3 | www.malasngoding.com</h1>
9   <h2>Contoh penggunaan @keyframe</h2>
10
11   <div class="kotak"></div>
12 </body>
13 </html>
```

contoh.css

```
1 body{
2   background: #35A9DB;
3   font-family: roboto;
4   color: #fff;
5 }
6
7 h1{
8   width: 550px;
9 }
10
11 .kotak{
12   background: blue;
13   width: 200px;
14   height: 200px;
15   animation-name: animasi_kotak;
16   animation-duration: 2s;
17 }
18
19 @keyframes animasi_kotak{
20   from{ background: blue;}
21   to{background: red;}
22 }
```


dan hasilnya



perhatikan pada contoh di atas. terdapat sebuah element kotak. dan kita beri nama animasi untuk kotak tersebut dengan nama 'animasi_kotak'. dan memberikan durasi animasinya selama 2 detik.

```
1 animation-name: animasi_kotak;  
2 animation-duration: 2s;
```

dan langsung kita definisikan animasi yang mau kita buat dengan **@keyframes**. dengan animasi yang mengubah warna pada kotak dari warna biru. menjadi warna merah. *Tutorial CSS3 Part 7 – Membuat Efek Animasi CSS3*

```
1 @keyframes animasi_kotak{  
2   from{ background: blue;}  
3   to{background: red;}  
4 }
```

Selain menggunakan kondisi awal dan kondisi akhir kita juga bisa menggunakan persentase untuk membuat animasi dengan **@keyframes** css3.



contoh2.html

```
1 <!DOCTYPE html>
2 <html>
3 <head>
4   <title>Membuat efek animasi dengan css3 | www.malasngoding.com</title>
5   <link rel="stylesheet" type="text/css" href="contoh2.css">
6 </head>
7 <body>
8   <h1>Membuat efek animasi dengan css3 | www.malasngoding.com</h1>
9   <h2>Contoh penggunaan @keyframe</h2>
10
11   <div class="kotak"></div>
12 </body>
13 </html>
```

contoh2.css

```
1 body{
2   background: #35A9DB;
3   font-family: roboto;
4   color: #fff;
5 }
6
7 h1{
8   width: 550px;
9 }
10
11 .kotak{
12   background: blue;
13   width: 200px;
14   height: 200px;
15   position: relative;
16   animation-name: animasi_kotak;
17   animation-duration: 2s;
18 }
19
20 @keyframes animasi_kotak{
21   0%   {background-color: red; left:0px; top:0px;}
22   25%  {background-color: yellow; left:200px; top:0px;}
23   50%  {background-color: blue; left:0px; top:200px;}
24   75%  {background-color: green; left:200px; top:200px;}
25   100% {background-color: red; left:0px; top:0px;}
26 }
```

Dukungan Browser Animasi @keyframes CSS3

agar browser dapat mendukung animasi css3. khususnya @keyframe. kita bisa menambahkan syntax berikut.

- @-moz-keyframes = untuk mendukung animasi berjalan di mozilla
- @-webkit-keyframes = untuk mendukung animasi @keyframes di safari atau google chrome
- @-o-keyframes = untuk mendukung @keyframes berjalan di opera

silahkan teman-teman perhatikan contoh penulisan nya berikut ini.

```
1 @keyframes animasi_kotak{
2   0%   {background-color: red; left:0px; top:0px;}
3   25%  {background-color: yellow; left:200px; top:0px;}
4   50%  {background-color: blue; left:0px; top:200px;}
5   75%  {background-color: green; left:200px; top:200px;}
6   100% {background-color: red; left:0px; top:0px;}
7 }
8
9 @-webkit-keyframes animasi_kotak{
10  0%   {background-color: red; left:0px; top:0px;}
11  25%  {background-color: yellow; left:200px; top:0px;}
12  50%  {background-color: blue; left:0px; top:200px;}
13  75%  {background-color: green; left:200px; top:200px;}
14  100% {background-color: red; left:0px; top:0px;}
15 }
```

```
10
17 @-moz-keyframes animasi_kotak{
18   0%   {background-color: red; left:0px; top:0px;}
19   25%  {background-color: yellow; left:200px; top:0px;}
20   50%  {background-color: blue; left:0px; top:200px;}
21   75%  {background-color: green; left:200px; top:200px;}
22   100% {background-color: red; left:0px; top:0px;}
23 }
24
25 @-o-keyframes animasi_kotak{
26   0%   {background-color: red; left:0px; top:0px;}
27   25%  {background-color: yellow; left:200px; top:0px;}
28   50%  {background-color: blue; left:0px; top:200px;}
29   75%  {background-color: green; left:200px; top:200px;}
30   100% {background-color: red; left:0px; top:0px;}
31 }
```



Membuat Tulisan Koran Atau Multiple Column Dengan CSS3

Membuat Tulisan Koran Atau Multiple Column Dengan CSS3 – Salah satu kelebihan css3 lagi yang akan kita bahas masih di tutorial css3. adalah multiple column. pernah terfikir tidak oleh teman-teman untuk membuat efek atau bentuk tulisan dengan memiliki beberapa column seperti **koran**. mungkin teman-teman akan memerlukan **fungsi multiple column css3** ini untuk **membuat tulisan dengan bentuk koran**. yaitu tulisannya seperti tulisan pada koran yang memiliki beberapa column. mungkin teman-teman akan membutuhkan multiple column css ini dalam membuat sebuah template website. apalagi template website yang teman-teman buat misalnya seperti template magazine atau blog/website yang fokus pada bidang berita atau majalah.

Baca juga :

- [Cara Membuat Pagination Pada CodeIgniter](#)
- [Membuat CRUD Dengan CodeIgniter : Update Data](#)

Pada **Tutorial css3** part 8 ini kita akan belajar tentang **cara membuat tulisan koran dengan css3**. yaitu menggunakan fungsi baru css3 yang di sebut dengan multiple column. di sini kita akan jelaskan tentang cara membuat multiple column pada text atau text seperti koran. apa saja property css yang sudah di sediakan untuk membuat multiple column pada text. dan sebagainya. baiklah silahkan teman-teman perhatikan penjelasan berikut untuk membuat efek koran dengan css. **Tutorial CSS3 Part 8 – Membuat Tulisan Koran Atau Multiple Column Dengan CSS3.**

Membuat Tulisan Koran Atau Multiple Column Dengan CSS3

Ada beberapa property yang di sediakan oleh css3 untuk membuat efek multiple column. yaitu :

- column-count
- column-width
- column-gap
- column-rule
- column-rule-style
- column-rule-color
- column-rule-width
- column-span

Ke-8 property di atas akan saya jelaskan satu persatu fungsi kegunaannya untuk **membuat tulisan koran atau multiple column text dengan css3**. silahkan teman-teman buat sebuah file html atau php. dan buat juga sebuah file css nya. di sini saya membuat sebuah file dengan nama index.html.



```
2 <html>
3 <head>
4   <title>Membuat Tulisan Koran Atau Multiple Column Dengan CSS3 | www.malasngoding.
5   <link rel="stylesheet" type="text/css" href="style.css">
6 </head>
7 <body>
8
9   <div class="kotak">
10    <h1>Membuat Tulisan Koran Atau Multiple Column Dengan CSS3 | www.malasngoding
11    <h2>Contoh Tulisan Koran</h2>
12
13    <br/>
14
15    <p class="koran">
16      Membuat Tulisan Koran Atau Multiple Column Dengan CSS3. Lorem ipsum dolor
17      tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veni
18      quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo
19      consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse
20      cillum dolore eu fugiat nulla pariatur. Membuat Tulisan Koran Atau Multip
21      proident, sunt in culpa qui officia deserunt mollit anim id est laborum.
22      Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod
23      tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veni
24      quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo
25      consequat.Membuat Tulisan Koran Atau Multiple Column Dengan CSS3. Duis au
26      cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat
27      proident, sunt in culpa qui officia deserunt mollit anim id est laborum.
28    </p>
29  </div>
30 </body>
31 </html>
```

style.css

```
2   background: #35A9DB;
3   font-family: roboto;
4   color: #fff;
5 }
6
7 h1,h2{
8   text-align: center;
9 }
10
11 .kotak{
12   width: 800px;
13   height: auto;
14   margin: 10px auto;
15 }
16
17
18 .koran{
19   /*jumlah column*/
20   -webkit-column-count: 3; /* support Chrome, Safari, Opera */
21   -moz-column-count: 3; /* support Firefox */
22   column-count : 3;
23
24   /*lebar column*/
25   -webkit-column-width: 100px; /* support Chrome, Safari, Opera */
26   -moz-column-width: 100px; /* support Firefox */
27   column-width :100px;
28 }
```



```
28
29 /*jarak column dengan column yang lain*/
30 -webkit-column-gap : 20px;
31 -moz-column-gap : 20px;
32 column-gap : 20px;
33
34 /*ketebalan garis*/
35 -webkit-column-rule-width : 1px;
36 -moz-column-rule-width : 1px;
37 column-rule-width : 1px;
38
39 /*pengaturan garis dengan format nilai jenis_garis warna*/
40 -webkit-column-rule : 1px solid blue;
41 -moz-column-rule : 1px solid blue;
42 column-rule : 1px solid blue;
43
44 /*mengatur span column*/
45 -webkit-column-span :all;
46 -moz-column-span :all;
47 column-span :all;
48 }
```

Membuat Tulisan Koran Atau Multiple Column Dengan CSS3 | www.malasngoding.com

Contoh Tulisan Koran

Membuat Tulisan Koran Atau Multiple Column Dengan CSS3. Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla

pariatur. Membuat Tulisan Koran Atau Multiple Column Dengan CSS3. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum. Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut

aliquip ex ea commodo consequat. Membuat Tulisan Koran Atau Multiple Column Dengan CSS3. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.



dan dapat teman-teman lihat pada gambar di atas. **tulisan dengan bentuk dan format koran** pun sudah jadi. perhatikan pada kegunaan property column di atas masing-masing.

column-count berfungsi untuk menentukan jumlah kolom yang ingin kita buat.

```
1 /*jumlah column*/
2 -webkit-column-count: 3; /* support Chrome, Safari, Opera */
3 -moz-column-count: 3; /* support Firefox */
4 column-count : 3;
```

column-width di gunakan untuk menentukan lebar dari masing-masing **multiple column**.

```
1 /*lebar column*/
2 -webkit-column-width: 100px; /* support Chrome, Safari, Opera */
3 -moz-column-width: 100px; /* support Firefox */
4 column-width :100px;
```

column-gap berfungsi untuk menentukan jarak masing-masing column.

```
1 /*jarak column dengan column yang lain*/
2 -webkit-column-gap : 20px;
3 -moz-column-gap : 20px;
4 column-gap : 20px;
```

column-rule-width untuk menentukan lebar atau ketebalan garis.

```
1 /*ketebalan garis*/
2 -webkit-column-rule-width : 1px;
3 -moz-column-rule-width : 1px;
4 column-rule-width : 1px;
```

column-rule untuk menentukan pengaturan garis. teman-teman boleh tidak menggunakannya jika tidak ingin menggunakan garis pada multiple column atau tulisan koran yang teman-teman ingin buat.

```
1 /*pengaturan garis dengan format nilai jenis_garis warna*/
2 -webkit-column-rule : 1px solid blue;
3 -moz-column-rule : 1px solid blue;
4 column-rule : 1px solid blue;
```

dan terakhir **column-span** untuk menentukan span column.

```
1 /*mengatur span column*/
2 -webkit-column-span : all;
3 -moz-column-span : all;
4 column-span : all;
```




adapun property lainnya yang bisa teman-teman coba adalah sebagai berikut untuk membuat multiple column atau tulisan koran dengan css3.

column-count	untuk menentukan jumlah column yang ingin di buat
column-width	untuk menentukan lebar column
column-gap	menentukan jarak column
column-rule	menentukan garis pada column
column-rule-style	menentukan jenis garis (solid,dotted dan lain-lain)
column-rule-color	menentukan warna garis
column-rule-width	menentukan lebar/ketebalan garis

Membuat Efek Kertas Dengan CSS3

Membuat Efek Kertas Dengan CSS3 – Kembali lagi di tutorial css3 di www.malasngoding.com. seperti yang udah teman-teman lihat di blog ini. malasngoding.com membahas tentang tutorial-tutorial pemrograman. dengan penjelasan yang terbaik. dan tutorial lengkap mulai dari html sampai dengan berbagai framework. hehe

Masuk ke materi kali ini saya ingin membahas tentang web design. saya ingin membagikan sebuah tutorial cara membuat efek kertas dengan css3. maksud nya adalah, efek kertas yang akan kita buat di tutorial ini adalah bentuk kertas yang saling tertimpa. dan saya rasa sangat bagus untuk tampilan website teman-teman. oke silahkan teman-teman simak penjelasan berikut untuk cara membuat effect kertas dengan css3. ***Membuat Efek Kertas Dengan CSS3.***

HTML

```
1 <div class="kertas">
2   <h2>Tutorial Web Design Dengan CSS3</h2>
3   <h3>Membuat Efek Kertas dengan CSS3</h3>
4   <br/>
5   <p>Tutorial pemrograman web, mobile dan web design di www.malasngoding.com. ini ada
6     quis rhoncus sapien facilisis. Mauris quis massa dui, rhoncus viverra quam. Nulla
7     eget laoreet. Integer sed nisl sed nibh pulvinar ornare quis nec quam. Aenean rho
8   <p>Malasngoding.com adalah website yang menyediakan tutorial-tutorial tentang pemro
9     erat in nibh auctor at aliquet velit vestibulum. Curabitur turpis diam, malesuado
10    metus, condimentum nec molestie et, egestas luctus libero.</p>
11  <p>Mauris porta odio augue. Nam turpis dui, volutpat id feugiat vitae, sagittis a j
12    a ut nulla.</p>
13  <p>Nullam tincidunt elit quis nibh blandit interdum. Duis ullamcorper pellentesque
14    enim nec ligula. Aliquam ipsum est, volutpat aliquam tincidunt id, sagittis et ve
15
16  <p>Aenean pulvinar, lacus ultrices euismod viverra, nulla diam interdum ligula, id
17    et elit consectetur rutrum. Integer lorem purus, ultricies ac laoreet quis, feugi
18 </div>
```




```
1 body {
2   background: #f2f5f7;
3   font: 14px sans-serif;
4   padding: 20px;
5 }
6
7 h2,
8 h3 {
9   text-align: center;
10 }
11
12 .kertas {
13   background: #fff;
14   box-shadow: 0 0 10px rgba(0, 0, 0, 0.3);
15   margin: 26px auto 0;
16   max-width: 550px;
17   min-height: 300px;
18   padding: 24px;
19   position: relative;
20   width: 80%;
21 }
22
23 .kertas:before,
24 .kertas:after {
25   content: "";
26   height: 98%;
27   position: absolute;
28   width: 100%;
29   z-index: -1;
30 }
31
```

```
31
32 .kertas:before {
33   background: #fafafa;
34   box-shadow: 0 0 8px rgba(0, 0, 0, 0.2);
35   left: -5px;
36   top: 4px;
37   transform: rotate(-2.5deg);
38 }
39
40 .kertas:after {
41   background: #f6f6f6;
42   box-shadow: 0 0 3px rgba(0, 0, 0, 0.2);
43   right: -3px;
44   top: 1px;
45   transform: rotate(1.4deg);
46 }
```

dan jalankan.



caranya sangat mudah. pertama kita hanya perlu membuat element class kertas menjadi bentuk posisi tertimpa. lalu kemudian tinggal kita putar saja agar miring dengan fungsi rotate css3. baca tutorial saya sebelumnya tentang [transform di css3](#).

```
1 .kertas:before {
2   background: #fafafa;
3   box-shadow: 0 0 8px rgba(0, 0, 0, 0.2);
4   left: -5px;
5   top: 4px;
6   transform: rotate(-2.5deg);
7 }
8
9 .kertas:after {
10  background: #f6f6f6;
11  box-shadow: 0 0 3px rgba(0, 0, 0, 0.2);
12  right: -3px;
13  top: 1px;
14  transform: rotate(1.4deg);
15 }
```



Cara Mengganti Warna Placeholder Dengan CSS

Cara Mengganti Warna Placeholder Dengan CSS – apa kabar teman-teman sekalian. hehe.. semoga sehat selalu ya. bertemu lagi lewat tulisan ini. kali ini kita akan membahas tentang ***cara mengganti warna placeholder dengan CSS***. seperti yang telah kita ketahui bersama, placeholder merupakan salah satu fitur baru di **HTML5**. placeholder sudah banyak di gunakan di hampir semua design website di tahun 2016. sudah dari beberapa tahun lalu sih sebenarnya. tetapi di tahun 2016 ini semakin banyak yang menggunakan placeholder untu memberikan penamaan pada form. ketika mendapatkan beberapa project membuat **aplikasi**, saya sendiri terkadang sering **menggunakan placeholder pada form**.

Pengertian Placeholder

Akan saya jelaskan juga **pengertian placeholder** dan **cara menggunakan placeholder pada form**. karena mungkin ada teman-teman yang baru mulai belajar dan baru mengetahui tentang placeholder ini.

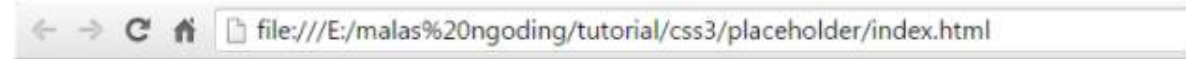
Placeholder adalah fitur dari HTML5. placeholder berguna untuk membuat semacam penamaan pada form. biasanya di gunakan untuk memberikan nama pada form.tetapi letakkan berada di dalam form. dan tulisan nya pun tidak akan berpengaruh atau mengganggu saat di submit. baca juga [tutorial penanganan form dengan php](#).

Cara Membuat Dan Menggunakan Placeholder

Untuk membuat placeholder pada form, kita tinggal menambahkan atribut "placeholder" dan isi kan value nya dengan text yang ingin di tampilkan pada form. contohnya :

index.html

```
2 <html>
3 <head>
4   <title>Cara Membuat Dan Mengganti Warna Placeholder | www.malasngoding.com</title>
5 </head>
6 <body>
7   <h1>Cara Membuat Dan Mengganti Warna Placeholder <br/> www.malasngoding.com</h1>
8
9   <input type="text" placeholder="Masukkan text di sini">
10
11 </body>
12 </html>
```



Cara Membuat Dan Mengganti Warna Placeholder www.malasngoding.com

Masukkan text di sini



cara membuat dan menggunakan placeholder

cara membuat dan menggunakan placeholder

Seperti yang teman-teman lihat pada gambar di atas. itulah **fungsi dari placeholder**. untuk membuat tulisan dengan tujuan memberikan penamaan pada form. nah sampai di sini kita sudah bisa **membuat placeholder**. lalu langkah selanjutnya bagaimana cara merubah warna tulisan placeholder? agar lebih menarik. langsung saja buat file [css](#) nya.



index.html

```
2 <html>
3 <head>
4   <title>Cara Membuat Dan Mengganti Warna Placeholder | www.malasngoding.com</title>
5   <link rel="stylesheet" type="text/css" href="style.css">
6 </head>
7 <body>
8   <h1>Cara Membuat Dan Mengganti Warna Placeholder <br/> www.malasngoding.com</h1>
9   <!-- contoh 1 -->
10  <input class="contoh1" type="text" placeholder="Masukkan text di sini">
11
12  <br/>
13  <!-- contoh 2 -->
14  <input class="contoh2" type="text" placeholder="Masukkan text di sini">
15
16  <br/>
17  <!-- contoh 3 -->
18  <input class="contoh3" type="text" placeholder="Masukkan text di sini">
19
20
21 </body>
22 </html>
```

style.css

```
2   background: #35A9DB;
3   font-family: roboto;
4   text-align: center;
5 }
6
7 h1{
8   color: #fff;
9 }
10
11 /*contoh1*/
12 /*support google chrome*/
13 .contoh1::-webkit-input-placeholder{
14   color: red;
15 }
16
17 /*support mozilla*/
18 .contoh1:-moz-input-placeholder{
19   color: red;
20 }
21
22 /*support internet explorer*/
23 .contoh1:-ms-input-placeholder{
24   color: red;
25 }
26
```



```
27
28 /*contoh2*/
29 /*support google chrome*/
30 .contoh2{
31     margin: 10px;
32     font-size: 12pt;
33     padding: 10px;
34 }
35
36 .contoh2::-webkit-input-placeholder{
37     color: blue;
38 }
39
40 /*support mozilla*/
41 .contoh2:-moz-input-placeholder{
42     color: blue;
43 }
44
45 /*support internet explorer*/
46 .contoh2:-ms-input-placeholder{
47     color: blue;
48 }
49
50
```

```
49
50
51 /*contoh3*/
52 /*support google chrome*/
53 .contoh3{
54     margin: 10px;
55     font-size: 13pt;
56     padding: 10px;
57     background: transparent;
58     color: #fff;
59     font-family: roboto;
60     border: none;
61 }
62
63 .contoh3:focus{
64     outline: none;
65 }
66
67 .contoh3::-webkit-input-placeholder{
68     color: white;
69 }
70
71 /*support mozilla*/
72 .contoh3:-moz-input-placeholder{
73     color: white;
74 }
75
76 /*support internet explorer*/
77 .contoh3:-ms-input-placeholder{
78     color: white;
79 }
```


hasilnya.



seperti yang teman-teman lihat pada contoh di atas. terdapat 3 buah contoh form yang saya buat dengan menggunakan placeholder juga. coba perhatikan pada css nya.

```
2 .contoh1::-webkit-input-placeholder{
3     color: red;
4 }
5
6 /*support mozilla*/
7 .contoh1:-moz-input-placeholder{
8     color: red;
9 }
10
11 /*support internet explorer*/
12 .contoh1:-ms-input-placeholder{
13     color: red;
14 }
```

agar di support di google chrome gunakan `::-webkit-input-placeholder`, untuk support di mozilla gunakan `:-moz-input-placeholder`. dan untuk support di internet explorer gunakan `-ms-input-placeholder`.

Untuk **design form** lebih lanjut nya perhatikan pada form contoh 3. hilangkan garis pinggir dan outline nya saat focus. berikan background nya transparent agar tembus pandang alias tidak memiliki warna. jadi warna yang di ambil adalah warna background nya.



Cara Membuat Animasi Preloader Dengan CSS3

Cara Membuat Animasi Preloader Dengan CSS3 – Terima kasih sebelumnya saya ucapkan kepada teman-teman yang sudah membaca tutorial-tutorial pemrograman di malasngoding.com. khususnya **tutorial membuat animasi preloader dengan css3** ini. sebenarnya saya agak bingung sih mau ngupdate tutorial apa lagi di malasngoding.com. teman-teman kalau ada yang mau request tutorial pemrograman bisa menghubungi saya melalui facebook. kebingungan saya hilang saat saya googling. dan menemukan tutorial yang saya rasa cocok untuk di angkat di www.malasngoding.com. yaitu **tutorial cara membuat efek animasi preloader dengan css3**.

Di tutorial ini akan saya jelaskan tentang cara membuat efek animasi yang bentuknya seperti efek loading. tutorial ini saya kutip dari website ternama. yaitu medialoot.com. jadi saya akan mencoba menjelaskan secara terperinci tentang komponen-komponen pembuatan **preloading dengan css3**. karena saya lihat di medialoot sendiri tidak menjelaskan secara terperinci. tapi yang penting terima kasih banyak juga untuk medialoot.com karena sudah bersedia membagikan ilmunya kepada kita secara gratis.

CSS

Animasi preloader 1 – Double Spinner

Langsung saja.

HTML

```
1 <div class="box">
2   <div id="preloader-1">
3     <span></span>
4     <span></span>
5   </div>
6 </div>
```

```
1 body{
2   background: #ffb83c;
3 }
4
5 #preloader-1 {
6   position: relative;
7   margin:0px auto;
8   width:100px;
9 }
10
11 #preloader-1 span {
12   position: absolute;
13   border: 8px solid #fff;
14   border-top: 8px solid transparent;
15   border-radius: 999px;
16 }
17
18 #preloader-1 span:nth-child(1) {
19   width: 80px;
20   height: 80px;
21   animation: spin-1 2s infinite linear;
22 }
23
24 #preloader-1 span:nth-child(2) {
25   top: 20px;
26   left: 20px;
27   width: 40px;
28   height: 40px;
29   animation: spin-2 1s infinite linear;
30 }
```

```
31
32 @keyframes spin-1 {
33   0% {
34     transform: rotate(360deg);
35     opacity: 1;
36   }
37   50% {
38     transform: rotate(180deg);
39     opacity: 0.5;
40   }
41   100% {
42     transform: rotate(0deg);
43     opacity: 1;
44   }
45 }
46
47 @keyframes spin-2 {
48   0% {
49     transform: rotate(0deg);
50     opacity: 0.5;
51   }
52   50% {
53     transform: rotate(180deg);
54     opacity: 1;
55   }
56   100% {
57     transform: rotate(360deg);
58     opacity: 0.5;
59   }
60 }
```

dan hasilnya

Membuat animasi preloader dengan css3
Double spinner
www.malasngoding.com



akan saa jelaskan sedikit. intinya ada di element span. ada dua buah span. dan keduanya di jadikan sebagai efek yang melingkar membentuk animasi loading seperti pada hasilnya.

```
1 #preloader-1 span:nth-child(1) {
2   width: 80px;
3   height: 80px;
4   animation: spin-1 2s infinite linear;
5 }
6
7 #preloader-1 span:nth-child(2) {
8   top: 20px;
9   left: 20px;
10  width: 40px;
11  height: 40px;
12  animation: spin-2 1s infinite linear;
13 }
```

perhatikan pada syntax di atas.

```
1 #preloader-1 span:nth-child(1)
```

di gunakan untuk memilih span yang pertama pada css. dan

```
1 #preloader-1 span:nth-child(2)
```

untuk memilih span yang kedia. :nth-child() berfungsi untuk memilih **element HTML** berdasarkan turunan elament induk(child).

lalu kemudian kedua span tadi di berikan efek css3

```
2 0% {
3   transform: rotate(360deg);
4   opacity: 1;
5 }
6 50% {
7   transform: rotate(180deg);
8   opacity: 0.5;
9 }
10 100% {
11  transform: rotate(0deg);
12  opacity: 1;
13 }
14 }
15
16 @keyframes spin-2 {
17 0% {
18  transform: rotate(0deg);
19  opacity: 0.5;
20 }
21 50% {
22  transform: rotate(180deg);
23  opacity: 1;
24 }
25 100% {
26  transform: rotate(360deg);
27  opacity: 0.5;
28 }
29 }
```

`transform rotate()` sudah kita bahas pada tutorial `css3` di malasngoding.com. `rotate()` berfungsi untuk memutar element dengan `css3`.

contohnya.

```
1 0% {  
2   transform: rotate(360deg);  
3   opacity: 1;  
4 }  
5 50% {  
6   transform: rotate(180deg);  
7   opacity: 0.5;  
8 }  
9 100% {  
10  transform: rotate(0deg);  
11  opacity: 1;  
12 }
```

persen yang ada di atas untuk menentukan waktu dari animasi. berarti dari 0 opacity/ketransparannya di setting senilai 1. dengan putaran sebesar 360 derajat. dan pada pertengahan animasi atau 50% putarannya berubah menjadi 180 derajat. dan yang terakhir dari 50 sampai 100 persen atau selesai putaran di kembalikan ke nilai 0 derajat.

Animasi preloader 2 – Crossing Circle

Sesuai dengan nama preloader animation ini (crossing circle). animasi preloader ini berbentuk dua lingkaran kecil yang saling bersebrangan. oke kita lihat langsung source code nya ya.

HTML

```
1 <div class="box">
2   <center>
3     <h1>
4       Membuat animasi preloader dengan css3<br>
5       Crossing Circle<br/>
6       www.malasngoding.com
7     </h1>
8   </center>
9   <div id="preloader-2">
10    <span></span>
11    <span></span>
12  </div>
13 </div>
```

CSS

```
2   background-color: #4ad3b4;
3 }
4
5 h1 {
6   font-family: sans-serif;
7   color: #fcfcfc;
8   font-weight: 100;
9   font-size: 16pt;
10 }
11
12 #preloader-2 {
13   position: relative;
14   margin: 70px auto;
15   width: 100px;
16 }
17
18 #preloader-2 span {
19   position: absolute;
20   width: 30px;
21   height: 30px;
22   background: #fff;
23   border-radius: 999px;
24 }
25
26 #preloader-2 span:nth-child(1) {
27   animation: cross-1 1.5s infinite linear;
28 }
29
```



```
29
30 #preloader-2 span:nth-child(2) {
31   animation: cross-2 1.5s infinite linear;
32 }
33
34 @keyframes cross-1 {
35   0% {
36     transform: translateX(0px);
37     opacity: 1;
38   }
39   50% {
40     transform: translateX(80px);
41     opacity: 0.5;
42   }
43   100% {
44     transform: translateX(0px);
45     opacity: 1;
46   }
47 }
```

```
48
49 @keyframes cross-2 {
50   0% {
51     transform: translateX(80px);
52     opacity: 1;
53   }
54   50% {
55     transform: translateX(0px);
56     opacity: 0.5;
57   }
58   100% {
59     transform: translateX(80px);
60     opacity: 1;
61   }
62 }
```



Membuat animasi crossing circle ini juga menggunakan 2 buah element span. tapi sudah di buat dalam bentuk lingkaran(perhatikan syntax css di atas).

Animasi Preloader 3 – Circle Rotator

Circle Rotator adalah animasi preloader yang membentuk lingkaran yang berputar. sama seperti animasi preloader yang pertama kita bahas. bedanya circle rotator hanya memiliki satu garis.

HTML

```
1 <div class="box">
2   <center>
3     <h1>
4       Membuat animasi preloader dengan css3<br>
5       Circle Rotator<br/>
6       www.malasngoding.com
7     </h1>
8   </center>
9   <div id="preloader-3">
10    <span></span>
11  </div>
12 </div>
```

```
1 body {
2   background-color: #ab69d9;
3 }
4 h1 {
5   font-family: sans-serif;
6   color: #fcfcfc;
7   font-weight: 100;
8   font-size: 16pt;
9 }
10 #preloader-3 {
11   position: relative;
12   animation: rotate 1s infinite linear;
13   border: 4px solid rgba(255, 255, 255, .25);
14   width: 80px;
15   height: 80px;
16   border-radius: 999px;
17   margin: 70px auto;
18 }
19 #preloader-3 span {
20   position: absolute;
21   width: 80px;
22   height: 80px;
23   border: 4px solid transparent;
24   border-top: 4px solid #fff;
25   top: -4px;
26   left: -4px;
27   border-radius: 999px;
28 }
29 @keyframes rotate {
30   0% {transform: rotate(0deg);}
31   100% {transform: rotate(360deg);}
32 }
```

dan hasilnya.



```
1 @keyframes rotate {  
2   0% {transform: rotate(0deg);}  
3   100% {transform: rotate(360deg);}  
4 }
```

element di putar dengan 360 derajat secara berulang-ulang.

Thank You

ukrida.ac.id



UKRIDA
Universitas Kristen Krida Wacana



Modul Praktikum JavaScript

CYNTHIA HAYAT S.KOM., M.MSI

KRIDA WACANA CHRISTIAN UNIVERSITY
Faculty of Engineering and Computer Science
Departement of Information System



UKRIDA
Universitas Kristen Krida Wacana



Javascript merupakan bahasa pemrograman yang berguna untuk membuat interaksi pada website. sehingga website terlihat lebih hidup. seperti misalnya membuat efek-efek pada website. javascript sudah menjadi salah satu bahasa pemrograman yang wajib dan harus di kuasai untuk membangun website yang modern. javascript menjadi kebutuhan dan bahasa pemrograman yang harus di kuasai. apalagi bagi anda yang akan mengkhususkan diri di bagian web design. Adapun beberapa bahasa pemrograman yang wajib di kuasai oleh seorang web developer adalah:

- HTML
Dalam membangun sebuah website HTML berperan sebagai pembuat bentuk.
- CSS
Dalam membangun sebuah website CSS memiliki peran sebagai design bentuk, warna, tulisan dan sebagainya
- JavaScript
Dalam membangun sebuah website JavaScript berguna untuk membuat fitur-fitur serta efek yang membuat halaman website lebih interaktif.

Pada perkembangan dunia website sekarang ini javascript sudah menjadi bahasa **pemrograman** yang sangat di butuhkan. banyak website-website yang menggunakan javascript untuk membuat modal alert, pemberitahuan, form validation, efek, ajax, game dan sebagainya. ***Belajar JavaScript Pengertian dan Pengenalan JavaScript***



Apa saja yang dapat dilakukan dengan JavaScript

Untuk lebih menegaskan pemahaman tentang **pengertian javascript** dan **kegunaa javascript** berikut akan di jelaskan apa-apa saja yang dapat di lakukan oleh javascript:

- Javascript dapat mengganti atribut HTML
- Javascript dapat mengganti style css
- Javascript dapat membuat form validasi

Cara Penulisan Syntax Javascript

Kita mulai dengan cara penulisan syntax javascript yang baik dan benar. dan adapun pengetahuan dasar yang harus kita ketahui dari javascript adalah sebagai berikut.

- Di simpan dengan ekstensi .js.
- Syntax javascript yang di sisipkan pada halaman HTML di tulis di dalam tag `<script>`.
- Syntax javascript bisa di letakkan di bagian element tag `<head>` atau tag `<body>`.
- Javascript bersifat case sensitive.
- Di setiap akhir baris harus di tutup dengan titik koma/semicolon (;).

syntax javascript bisa di gunakan dengan 2 cara teman-teman. yaitu dengan cara di sisipkan pada halaman HTML langsung, dan 1 lagi dengan cara menyimpan file javascript dengan ekstensi .js. dan kemudian menginclude nya pada halaman HTML. agar lebih spesifik akan kita bahas satu persatu.

Menggunakan Javascript Secara Internal

Penulisan javascript secara internal adalah dengan cara menyisipkan langsung syntax javascript di dalam HTML. jika teman-teman menggunakan cara ini maka syntax javascript nya harus teman-teman letakkan di antara tag `<script>` pembuka dan tag `</script>` penutup. dan meletakkannya bisa di dalam tag `<head>`, bisa juga di dalam tag `<body>`. buat sebuah file HTML untuk membuat contoh penggunaan Javascript secara internal. bisa file html bisa juga php. karena javascript mampu berjalan di antara keduanya. di sini saya hanya membuat file html saja sebagai contoh. buat sebuah file html dengan nama belajar.html.

belajar.html

```
1 <!DOCTYPE html>
2 <html>
3 <head>
4   <title>Belajar Javascript Part 2 : Cara Penulisan Dan Penggunaan Javascript</title>
5
6 </head>
7 <body>
8   <h1>Belajar Javascript Part 2 : Cara Penulisan Dan Penggunaan Javascript</h1>
9   <h2>www.malasngoding.com</h2>
10
11   <div id="nama"></div>
12
13   <script>
14     document.getElementById("nama").innerHTML = "Nama Saya Malas Ngoding";
15   </script>
16 </body>
17
18 </html>
```

coba perhatikan pada contoh di atas. terdapat sebuah element div yang kita beri id="nama". dan kemudian kita memberikan sentuhan javascript di sana dengan perintah berikut. **Belajar Javascript Part 2 : Cara Penulisan Dan Penggunaan Javascript**

```
1 document.getElementById("nama").innerHTML = "Nama Saya Malas Ngoding";
```

document adalah syntax wajib di javascript, dan fungsi **getElementById()** berfungsi untuk memerintahkan mendapatkan element yang ber id sesuai dengan yang ada di dalam parameter fungsi **getElementById()** itu sendiri. penulisannya harus si perhatikan ya teman-teman besar kecil nya. karena javascript bersifat case sensitive. dan innerHTML berfungsi untuk menuliskan html. jadi secara lengkap intruksi yang kita serukan adalah kita menyerukan untuk menuliskan "Nama Saya Malas Ngoding" di element yang ber id "nama". jika di jalankan pada browser. maka hasilnya sebagai berikut.



Menggunakan Javascript Secara External

Setelah membahas cara penulisan javascript secara internal, atau dengan cara menyisipkan langsung syntax javascript pada html, maka di sini kita akan membahas cara lainnya untuk menggunakan javascript, yaitu dengan **cara menggunakan javascript secara external**. caranya buat lah dua buah file, yaitu :

- belajar.html
- belajar.js

di file belajar.html kita akan menuliskan syntax HTML nya saja. dan pada belajar.js kita akan menuliskan syntax **javascript** nya pada file belajar.js. dan cara agar kedua file ini terhubung adalah dengan cara menghubungkannya dengan syntax berikut. **Belajar Javascript Part 2 : Cara Penulisan Dan Penggunaan Javascript.**

```
1 <script type="text/javascript" src="belajar.js"></script>
```

belajar.html

```
1 <!DOCTYPE html>
2 <html>
3 <head>
4   <title>Belajar Javascript Part 2 : Cara Penulisan Dan Penggunaan Javascript</title>
5
6 </head>
7 <body>
8   <h1>Belajar Javascript Part 2 : Cara Penulisan Dan Penggunaan Javascript</h1>
9   <h2>www.malasngoding.com</h2>
10
11   <div id="nama"></div>
12
13 </body>
14   <script type="text/javascript" src="belajar.js"></script>
15 </html>
```

belajar.js

```
1 document.getElementById("nama").innerHTML = "Nama Saya Malas Ngoding";
```

dan jika di jalankan maka hasilnya akan sama.



Belajar Javascript Part 3 : Mengenal Variabel Pada Javascript

Cara Penulisan Variabel Di JavaScript

Yang harus di perhatikan di dalam membuat variabel di javascript adalah, semua penulisan variabel pada javascript tidak boleh mengandung spasi, harus memperhatikan penulisan huruf besar kecil. lebih lengkap nya perhatikan poin-poin cara penulisan variabel pada javascript yang benar.

Cara penulisan variabel di javascript:

- Penulisan nama variabel bisa menggunakan huruf, angka, tanda underscore dan tanda dollar.
- Penulisan nama variabel harus diawali dengan huruf.
- Penulisan nama variabel javascript harus memperhatikan huruf besar kecil (case sensitive).
- Nama variabel tidak boleh mengandung spasi.
- penulisan variabel javascript diawali dengan syntax "**var**".

Cara Membuat Variabel di JavaScript

baiklah teman-teman langsung saja kita buat contoh cara membuat variabel dengan javascript. silahkan teman-teman siapkan sebuah file html dengan nama belajar.html.

belajar.html

```
1 <!DOCTYPE html>
2 <html>
3   <head>
4     <title>Belajar Javascript Part 3 : Mengenal Variabel Pada Javascript</title>
5
6   </head>
7   <body>
8     <h1>Belajar Javascript Part 3 : Mengenal Variabel Pada Javascript</h1>
9     <h2>www.malasngoding.com</h2>
10
11     <div id="nama"></div>
12     <div id="alamat"></div>
13
14     <script>
15       var nama = "Malas Ngoding";
16       var alamat = "Jakarta";
17       document.getElementById("nama").innerHTML = nama;
18       document.getElementById("alamat").innerHTML = alamat;
19     </script>
20   </body>
21 </html>
```

seperti yang teman-teman perhatikan pada syntax di atas. saya membuat dua buah **variabel javascript** dengan nama "nama" dan "alamat". oh ya teman-teman jangan lupa untuk memberikan tanda titik koma atau semicolon(;) di akhir baris nya. karena ini ketentuan cara penulisan **javascript**.

```
1 var nama = "Malas Ngoding";
2 var alamat = "Jakarta";
```

dan kemudian saya menampilkan isi dari variabel ini ke dalam element nama dan alamat.

```
1 document.getElementById("nama").innerHTML = nama;
2 document.getElementById("alamat").innerHTML = alamat;
```

file:///E:/malas%20ngoding/tutorial/javascript/belajar.html

Belajar Javascript Part 3 : Mengenal Variabel Pada Javascript

www.malasngoding.com

Malas Ngoding
Jakarta

belajar javascript mengenal variabel pada javascript

note :

document.getElementById() berguna untuk menentukan element yang ber id sesuai dengan yang di dalam parameter fungsi ini. innerHTML berguna untuk menerapkan syntax html.



Belajar Javascript Part 4 : Operator Aritmatika Pada Javascript

Sebelum lebih jauh belajar **membuat operasi perhitungan aritmatika di javascript**, kita akan membahas terlebih dulu tentang operator-operator yang tersedia di **javascript**. yang bisa kita gunakan untuk **membuat operasi perhitungan matematika dengan javascript**.

beikut adalah **operator** yang dapat kita gunakan di javascript untuk membuat perhitungan aritmatika.

+	Penjumlahan
-	Pengurangan
*	Perkalian
/	Pembagian
%	Modulus
++	Increment
--	Decrement

Bagaimana Cara Membuat Perhitungan Aritmatika Di Javascript ?

Cara nya sangat mudah, perhatikan contoh berikut.

```
1 <!DOCTYPE html>
2 <html>
3   <head>
4     <title>Belajar Javascript Part 4 : Operator Aritmatika Pada Javascript</title>
5   </head>
6   <body>
7     <h1>Belajar Javascript Part 4 <br/> Operator Aritmatika Pada Javascript</h1>
8     <h2>www.malasngoding.com</h2>
9
10    <div id="penjumlahan"></div>
11
12    <script>
13      var bilangan1 = 6;
14      var bilangan2 = 2;
15
16      var penjumlahan = bilangan1 + bilangan2;
17
18      document.getElementById("penjumlahan").innerHTML = penjumlahan;
19    </script>
20  </body>
21 </html>
```

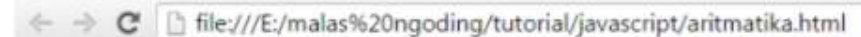
pada contoh di atas kita membuat contoh penjumlahan. dengan mengisi nilai pada variabel di atas terdapat dua buah variabel. yaitu variabel bilangan1 dan bilangan2. variabel bilangan1 berisi nilai 6. dan variabel bilangan2 berisi nilai 2. kemudian untuk menjumlahkannya kita tinggal menambahkan operator bilangannya seperti contoh di atas.

```
1 var penjumlahan = bilangan1 + bilangan2;
```

sampai di sini variabel penjumlahan sudah berisi hasil dari bilangan1 di tambah bilangan2. 6+2.

kemudian hasilnya kita tampilkan pada element yang ber id penjumlahan.(tag div).

dan hasilnya.



Belajar Javascript Part 4 Operator Aritmatika Pada Javascript

www.malasngoding.com



```
1 <!DOCTYPE html>
2 <html>
3   <head>
4     <title>Belajar Javascript Part 4 : Operator Aritmatika Pada Javascript</title>
5   </head>
6   <body>
7     <h1>Belajar Javascript Part 4 <br/> Operator Aritmatika Pada Javascript</h1>
8     <h2>www.malasngoding.com</h2>
9
10    <div id="penjumlahan"></div>
11    <div id="pengurangan"></div>
12    <div id="perkalian"></div>
13    <div id="pembagian"></div>
14    <div id="modulus"></div>
15
16
17    <script>
18      var bilangan1 = 6;
19      var bilangan2 = 2;
20
21      var penjumlahan = bilangan1 + bilangan2;
22      var pengurangan = bilangan1 - bilangan2;
23      var perkalian = bilangan1 * bilangan2;
24      var pembagian = bilangan1 / bilangan2;
25      var modulus = bilangan1 % bilangan2;
26
27      document.getElementById("penjumlahan").innerHTML = penjumlahan;
28      document.getElementById("pengurangan").innerHTML = pengurangan;
29      document.getElementById("perkalian").innerHTML = perkalian;
30      document.getElementById("pembagian").innerHTML = pembagian;
31      document.getElementById("modulus").innerHTML = modulus;
32
33    </script>
34  </body>
35 </html>
```

dan hasilnya





Belajar Javascript Part 5 : Mengenal Tipe Data Pada Javascript

Mengenal Tipe Data Pada Javascript

Kita mulai dengan mengenal **tipe data** pada javascript. di **javascript** terdapat 5 buah tipe data yang masing-masing memiliki kegunaan dan fungsi nya masing-masing. di antaranya adalah :

- Number
- String
- Boolean
- Array
- Object

Dan adapun beberapa kegunaan dan penggunaan dari masing-masing tipe data **javascript** adalah sebagai berikut.

Tipe Data Number Di Javascript

Tipe data number atau integer adalah tipe data yang berbentuk **bilangan bulat**. contohnya adalah 0,1,2,3,4,5,6,7,8,9,10 dan seterusnya. cara menggunakan tipe data number adalah sebagai berikut.

```
2 <html>
3 <head>
4   <title>Belajar Javascript Part 5 : Mengenal Tipe Data Pada Javascript</title>
5 </head>
6 <body>
7   <h1>Tipe Data Number Pada Javascript</h1>
8   <h2>www.malasngoding.com</h2>
9
10  <!-- id hasil -->
11  <div id="hasil"></div>
12
13  <script>
14    // tipe data Number
15    var angka = 80;
16
17    // menampilkan tipe data number pada element id hasil
18    document.getElementById("hasil").innerHTML = angka;
19  </script>
20 </body>
21 </html>
```

seperti yang dapat teman-teman lihat pada contoh di atas. penulisan **tipe data number** tidak menggunakan tanda petik seperti *string*.

```
1 var angka = 80;
```

dan jika di jalankan pada browser maka hasilnya.

file:///E:/malas%20ngoding/tutorial/javascript/number.html

Tipe Data Number Pada Javascript

www.malasngoding.com

80



Tipe Data String Di Javascript

Di dalam pemrograman, tipe data string memiliki peran penting di dalamnya. tanpa tipe data string kita juga tidak bisa mendefinisikan suatu variabel dan lainnya. tipe data string biasanya berbentuk kata atau kalimat. **contoh tipe data string di javascript** sebagai berikut.

```
1 <!DOCTYPE html>
2 <html>
3 <head>
4   <title>Belajar Javascript Part 5 : Mengetahui Tipe Data Pada Javascript</title>
5 </head>
6 <body>
7   <h1>Tipe Data String Pada Javascript</h1>
8   <h2>www.malasngoding.com</h2>
9
10  <!-- id hasil -->
11  <div id="hasil"></div>
12
13  <script>
14    // tipe data string
15    var kalimat = "Belajar javascript dasar di malasngoding.com";
16
17    // menampilkan tipe data string pada element id hasil
18    document.getElementById("hasil").innerHTML = kalimat;
19  </script>
20 </body>
21 </html>
```

dapat di lihat pada contoh **tipe data string** di atas. terdapat sebuah variabel dengan tipe data string. dan kemudian kita tampilkan. Belajar Javascript Part 5 : Mengetahui Tipe Data Pada Javascript

```
1 var kalimat = "Belajar javascript dasar di malasngoding.com";
```

file:///E:/malas%20ngoding/tutorial/javascript/string.html

Tipe Data String Pada Javascript

www.malasngoding.com

Belajar javascript dasar di malasngoding.com

Tipe Data Array di Javascript

Tipe data array adalah tipe data yang mampu menyimpan banyak data di dalam satu buah variabel. dan data-data yang terdapat di dalam variabel tersebut di berikan nomor masing-masing sesuai urutannya, untuk memudahkan kita dalam menampilkan atau memanggil data yang di inginkan. perhatikan contoh **tipe data array di javascript** berikut.

```
1 <!DOCTYPE html>
2 <html>
3 <head>
4   <title>Belajar Javascript Part 5 : Mengetahui Tipe Data Pada Javascript</title>
5 </head>
6 <body>
7   <h1>Tipe Data Array Pada Javascript</h1>
8   <h2>www.malasngoding.com</h2>
9
10  <!-- id hasil -->
11  <div id="hasil"></div>
12
13  <script>
14    // tipe data Array
15    var hewan = ["Kambing", "Bebek", "Ayam"];
16
17    // menampilkan tipe data Array pada element id hasil
18    document.getElementById("hasil").innerHTML = hewan[1];
19  </script>
20 </body>
21 </html>
```

perhatikan pada contoh tipe data array di atas. terdapat tipe data array di dalam variabel hewan.

```
1 var hewan = ["Kambing", "Bebek", "Ayam"];
```

di dalam variabel array hewan ini terdapat data kambing, bebek dan ayam. berarti angka penyebut untuk kambing adalah 0. bebek 1 dan ayam 2. angka nya di mulai dari 0. dan untuk mengakses/menampilkan salah satu angka nya adalah sebagai berikut. **Belajar Javascript Part 5 : Mengetahui Tipe Data Pada Javascript**.

```
1 hewan[1]
```

ini berarti kita memilih data array di dalam variabel hewan dengan nomor urut 1. dan akan di tampilkan hewan bebek.

- kambing = 0
- bebek = 1
- ayam = 2

```
← → ↻ file:///E:/malas%20ngoding/tutorial/javascript/array.html
```

Tipe Data Array Pada Javascript

www.malasngoding.com

Bebek

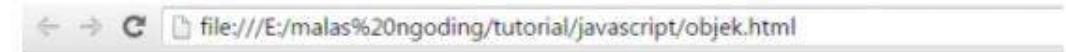
Tipe Data Object Pada Javascript

tipe data object berisi data yang banyak di dalam sebuah variabel. memiliki nama dan value nya masing-masing. contoh nye perhatikan

```
1 <!DOCTYPE html>
2 <html>
3 <head>
4   <title>Belajar Javascript Part 5 : Mengetahui Tipe Data Pada Javascript</title>
5 </head>
6 <body>
7   <h1>Tipe Data Object Pada Javascript</h1>
8   <h2>www.malasngoding.com</h2>
9
10  <!-- id hasil -->
11  <div id="hasil"></div>
12
13  <script>
14    // tipe data object
15    var manusia = {
16      nama : "malas ngoding",
17      alamat : "Jakarta",
18      umur : 30,
19      pekerjaan : "facebooker"
20    };
21
22    // menampilkan tipe data object pada element id hasil
23    document.getElementById("hasil").innerHTML = manusia.nama + " tinggal di " +
24  </script>
25 </body>
26 </html>
```

di sini terdapat tipe data object.

nama, alamat, umur, pekerjaan adalah nama. dan valuenya adalah malasngoding, jakarta, 30 dan facebooker. untuk mengakses data nya, pertama tuliskan nama variabel, dan kemudian di ikuti dengan nama data. misalnya jika kita ingin menampilkan data alamat. penulisannya adalah manusia.alamat, maka akan di tampilkan jakarta.



Tipe Data Object Pada Javascript

www.malasngoding.com

malas ngoding tinggal di Jakarta .dia sekarang berumur30 dan bekerja sebagai facebooker



Belajar Javascript Part 6 : Membuat Function di Javascript

Membuat Function Di Javascript

Untuk membuat function di javascript ada beberapa hal yang harus teman-teman pahami dulu. yaitu tentang penulisan function di javascript. berikut ini adalah contoh format penulisan function javascript.

```
1 <script>
2
3 /membuat function di javascript
4 function nama_function(){
5     // isi function nya di buat di sini
6 }
7
8 </script>
```

seperti yang teman-teman lihat pada contoh di atas. untuk membuat function, penulisannya harus diawali oleh syntax **function** dan kemudian dilanjutkan dengan nama function yang mau di buat. dan isi dari **function** di tuliskan di dalam tanda kurung kurawal pembuka "{" dan tanda kurung kurawal penutup "}".

langsung saja kita masuk ke tutorial membuat function dengan javascript. silahkan teman-teman perhatikan contoh berikut ini untuk membuat **function di javascript**.

```
1 <!DOCTYPE html>
2 <html>
3 <head>
4     <title>Belajar Javascript Part 6 : Membuat Function di Javascript</title>
5 </head>
6 <body>
7     <h1>Membuat Function di Javascript</h1>
8     <h2>www.malasngoding.com</h2>
9
10    <!-- id hasil -->
11    <div id="hasil"></div>
12
13    <script>
14        // membuat function tampilkan_nama
15        function tampilkan_nama(){
16            return "Malas Ngoding";
17        }
18
19        document.getElementById("hasil").innerHTML = tampilkan_nama();
20
21    </script>
22 </body>
23 </html>
```

di contoh di atas perhatikan

```
1 // membuat function tampilkan_nama
2 function tampilkan_nama(){
3     return "Malas Ngoding";
4 }
```

kita membuat function dengan nama tampilkan_nama(). yang berisi perintah untuk mengembalikan string "malas ngoding". jadi saat function tampilkan_nama() di panggil, hasilnya adalah "malas ngoding".



Belajar Javascript Part 7 : Mengenal Event Pada Javascript

Pengertian Event Pada Javascript

Event pada javascript adalah sesuatu yang terjadi pada element. pasti teman-teman pada bingung kan dengan pengertian event pada javascript. jadi gini, misalnya kita memiliki sebuah tombol di halaman website atau aplikasi yang kita bangun, dan kita ingin memberikan suatu aksi jika tombol tersebut di klik. jadi yang menjadi event di sini adalah "klik". misalnya jika tombol di klik maka akan di tampilkan pesan yang menampilkan sebuah kalimat. jadi event nya adalah klik. adapun beberapa event yang terdapat pada javascript adalah sebagai berikut.

Macam-macam event pada javascript

- onclick = adalah event jika sebuah element html di klik.
- onchange = adalah event jika sebuah element html berubah.
- onmouseover = adalah event jika sebuah element html di letakkan cursor mouse.
- onmouseout = adalah event jika saat cursor mouse meninggalkan element html.
- onkeydown = adalah event jika saat di terjadi pengetikan pada elemnet html.
- onload = adalah event ketika jika saat element atau halaman di buka.

Baiklah akan kita bahas beberapa **event javascript** yang sudah di jelaskan di atas.

Cara Membuat Event Javascript

Untuk membuat event dengan javascript, kita bisa menambahkan atribut dengan nama-nama event di atas, pada element html yang ingin kita berikan event. contohnya.

```
1 <button onclick="di sini berikan aksi yang ingin di lakukan">TOMBOL</button>
```


Contoh event pada Javascript

untuk contohnya saya akan membuat event klik pada sebuah tombol. jadi saya ingin menampilkan sebuah kalimat jika tombol tersebut di klik.

```
1 <!DOCTYPE html>
2 <html>
3 <head>
4   <title>Belajar Javascript Part 7 : Mengenal Event Pada Javascript</title>
5 </head>
6 <body>
7   <h1>Mengenal Event Pada Javascript</h1>
8   <h2>www.malasngoding.com</h2>
9
10  <!-- memberikan event pada element tombol -->
11  <button onclick="tampilkan_nama()">KLIK SAYA</button>
12
13  <!-- id hasil -->
14  <div id="hasil"></div>
15
16  <script>
17    // membuat function tampilkan_nama
18    function tampilkan_nama(){
19      document.getElementById("hasil").innerHTML = "<h3>Nama Saya Adalah Andi</h3>";
20    }
21
22  </script>
23 </body>
24 </html>
```

coba teman-teman perhatikan pada contoh di atas, terdapat sebuah tombol yang kita berikan **event** klik.

```
1 <button onclick="tampilkan_nama()">KLIK SAYA</button>
```

```
1 <button onclick="tampilkan_nama()">KLIK SAYA</button>
```

pada saat tombol ini di klik maka akan di jalankan function **tampilkan_nama()**. nah lalu kita buat function tampilkan_nama nya.

```
1 function tampilkan_nama(){
2   document.getElementById("hasil").innerHTML = "<h3>Nama Saya Adalah Andi</h3>";
3 }
```

isi dari function tampilkan_nama() ini adalah menampilkan kalimat "nama saya adalah andi" pada element id hasil.

perhatikan pada gambar berikut ini.



Mengenal Event Pada Javascript

www.malasngoding.com

KLIK SAYA

pada saat tombol di klik,



Mengenal Event Pada Javascript

www.malasngoding.com

KLIK SAYA

Nama Saya Adalah Andi

Belajar Javascript Part 8 : String Pada Javascript

Belajar String Pada Javascript

Seperti yang sudah kita ketahui **string** adalah tipe data yang tersusun dari huruf atau kata.

Untuk mengenal tipe data string pada javascript lebih jauh lagi, kita akan mulai dengan melihat bentuk penulisan tipe data string javascript dan cara membuatnya.

berikut adalah cara penulisan string ada javascript

```
1 var warna = "hitam";  
2 var warna = 'hitam';
```

seperti contoh penulisan string javascript di atas, terdapat 2 contoh variabel yang menyimpan data ber tipe data string.

untuk penulisan tipe data string, kita bisa menggunakan petik dua (") atau petik satu (').

Saya kira di sini teman-teman sudah paham jika sudah pernah membaca [tutorial tipe data di php](#) atau [tutorial tipe data javascript](#) yang sudah kita bahas dan pelajari sebelumnya.

Penggunaan tanda petik di dalam string

Lalu bagaimana jika kita ingin **menggunakan tanda petik di dalam tipe data string** ?

Caranya adalah dengan membalik tanda petik. jika string di bungkus oleh petik dua, maka petik yang di gunakan di dalam string adalah petik satu. begitu juga sebaliknya, jika string di bungkus oleh petik satu, maka petik yang harus di gunakan dalam string adalah petik dua. contoh nya :

```
1 var kata = "hari ini adalah hari jum'at";  
2 var kata = "nama saya 'diki'";  
3 var kata = 'nama saya "diki"';
```

cara penulisannya dan perbedaannya bisa teman-teman lihat pada contoh di atas.

yuk kita lihat hasil nya jika di jalankan pada browser.

```
1 <h2>www.malasngoding.com</h2>  
2  
3 <div id="tampil"></div>  
4  
5 <script>  
6   var kata = 'nama saya "diki"';  
7   document.getElementById("tampil").innerHTML=kata;  
8 </script>
```


Belajar Javascript Part 9 : Menampilkan Tanggal Dengan Javascript

Menampilkan Tanggal Dengan Javascript

Ada sebuah fungsi yang khusus di sediakan di javascript untuk menampilkan tanggal. yaitu kita bisa menggunakan fungsi Date(). dengan menggunakan fungsi **Date()** ini kita bisa menampilkan tanggal di php. dan untuk mengatur format tanggal yang di tampilkan, teman-teman bisa menyesuaikannya dengan memberikan/mengisi **format tanggal** nya sesuai dengan keinginan di dalam parameter **fungsi Date()**.

Perlu teman-teman perhatikan. seperti yang sudah kita bahas pada tutorial javasript dasar sebelumnya, bahwa javascript memiliki sifat case sensitive, yang artinya penulisan javascript sangat berpengaruh sekali tentang besar kecil huruf nya. jadi untuk membuat tanggal, teman-teman harus menuliskan fungsi date nya dengan benar. maksud nya huruf D nya besar.

contoh salah

date()

contoh benar

Date()

jadi jika teman-teman misalnya menuliskan fungsi Date() dengan menggunakan huruf kecil semua, maka hasilnya tidak akan muncul. jadi harus diawali dengan huruf kapital di awalnya, sesuai dengan ketentuan penulisan fungsi date.

Contoh Menampilkan Tanggal dengan Javascript

Pada sub judul ini kita akan membahas tentang cara menggunakan fungsi Date() pada javascript.

silahkan langsung teman-teman perhatikan lagi.

```
1 <!DOCTYPE html>
2 <html>
3 <head>
4   <title>Belajar Javascript Part 9 : Menampilkan Tanggal Dengan Javascript</title>
5 </head>
6 <body>
7   <h1>Menampilkan Tanggal Dengan Javascript</h1>
8   <h2>www.malasngoding.com</h2>
9
10  <!-- id hasil -->
11  <div id="hasil"></div>
12
13  <script>
14    var tanggal = new Date();
15
16    document.getElementById("hasil").innerHTML = tanggal;
17  </script>
18 </body>
19 </html>
```

pada contoh ini, saya memasukkan fungsi date ke dalam variabel tanggal. dan kemudian menampilkannya.

Belajar Javascript Part 10 : Switch Case Di Javascript

Belajar Javascript Part 10 : Switch Case Di Javascript – Switch Case adalah fungsi yang berguna untuk membuat pengecekan sebuah nilai. dan nilai yang tersedia untuk pengecekan bisa banyak atau lebih dari satu. pada tutorial sebelumnya kita telah membahas tentang [cara menggunakan switch case pada php](#). dan di tutorial ini kita akan membahas tentang penggunaan switch case pada javascript. silahkan teman-teman perhatikan contoh penggunaan switch case pada javascript berikut.

Switch Case Pada Javascript

Pada contoh penggunaan switch case pada javascript di tutorial ini. kita akan membuat pengecekan sebuah variabel warna yang berisi merah. dan kemudian kita cek isi variabel warna ini. baca tutorial sebelumnya tentang [pengenalan variabel pada javascript](#).

switch.html

```
1 <!DOCTYPE html>
2 <html>
3 <head>
4   <title>Belajar Javascript Part 10 : Switch Case Di Javascript</title>
5 </head>
6 <body>
7   <h1>Switch Case Di Javascript</h1>
8   <h2>www.malasngoding.com</h2>
9
10  <!-- id hasil -->
11  <div id="hasil"></div>
12
13  <script>
14
15      var warna = "merah";
16
17      switch (warna){
18          case "hitam":
19              teks = "warna hitam";
20              break;
21          case "merah":
22              teks = "Warna merah";
23              break;
24          case "hijau":
25              teks = "Warna hijau";
26              break;
27          default:
28              teks = "Warna tidak terdeteksi";
29      }
30
31      document.getElementById("hasil").innerHTML = teks;
32  </script>
33 </body>
34 </html>
```

perhatikan pada contoh di atas. kita memiliki sebuah variabel bernama warna. dan berisi string "merah".

```
1 var warna = "merah";
```

kemudian kita mengecek isi variabel warna ini.

```
1 switch (warna){
2   case "hitam":
3     teks = "warna hitam";
4     break;
5   case "merah":
6     teks = "Warna merah";
7     break;
8   case "hijau":
9     teks = "Warna hijau";
10    break;
11  default:
12    teks = "Warna tidak terdeteksi";
13 }
```

jika warna menemukan data case nya yang sesuai dengan isi dari **variabel** warna maka proses pengecekan akan di hentikan. di sini data di temukan pada pengecekan kedua yaitu merah. kemudian kita menyimpan string ke dalam variabel teks. sampai di sini kita telah memiliki isi di dalam variabel teks. dan terakhir kita tampilkan.

```
1 document.getElementById("hasil").innerHTML = teks;
```

dan hasilnya.



Switch Case Di Javascript

www.malasngoding.com

Warna merah

belajar javascript switch case pada javascript

sedikit tambahan, pada bagian default adalah bagian dimana jika pengecekan tidak di temukan maka default yang akan di gunakan. misalnya jika pengecekan warna tidak ada yg sesuai maka default yang di jalankan.

Belajar Javascript Part 11 : Perulangan For Pada Javascript

Belajar Javascript Part 11 : Perulangan For Pada Javascript – Perulangan di dalam pemrograman sangat lah berguna. misalnya jika kita ingin menampilkan string secara berulang-ulang. jika kita ingin menampilkan sebuah kalimat sebanyak 100 kali. tidak mungkin kita menulis sebanyak 100 kali. jadi solusi di dalam pemrogramannya adalah menggunakan **for loop** atau perulangan for. tidak hanya string, kita juga bisa menjalankan sebuah method secara berulang-ulang.

fungsi **for** pada semua bahasa pemrograman berfungsi untuk membuat perulangan. dengan membuat perulangan kita bisa membuat semuanya secara berulang-ulang. baik itu menampilkan string secara berulang-ulang, angka dan fungsi method secara berulang-ulang. untuk lebih jelasnya silahkan teman-teman baca tutorial saya sebelumnya tentang [perulangan for pada php](#).

for.html

```
1 <!DOCTYPE html>
2 <html>
3 <head>
4   <title>Belajar Javascript Part 11 : Perulangan For Pada Javascript</title>
5 </head>
6 <body>
7   <h1>Perulangan For Pada Javascript</h1>
8   <h2>www.malasngoding.com</h2>
9
10
11  <button onclick="perulangan()">TOMBOL</button>
12
13  <!-- id hasil -->
14  <div id="hasil"></div>
15
16  <script>
17    function perulangan() {
18      var text = "";
19      var x;
20      for (x = 0; x <= 10; x++) {
21        text += "Angka " + x + "<br>";
22      }
23      document.getElementById("hasil").innerHTML = text;
24    }
25  </script>
26 </body>
27 </html>
```

di sini kita membuat sebuah tombol yang jika di klik maka akan menjalankan function perulangan(). baca tutorial sebelumnya di [malasngoding.com](#) tentang [pengenalan event pada javascript](#).



perulangan `for` pada javascript sama cara penulisannya dengan membuat perulangan `for` pada php. di sini kita membuat perulangan `for` dengan `x` sama dengan 0, `x` lebih kecil sama dengan 10. jadi yang di tampilkan adalah angka 0 sampai 10.

```
1 var text = "";  
2 var x;  
3 for (x = 0; x <= 10; x++) {  
4     text += "Angka " + x + "<br>";  
5 }  
6 document.getElementById("hasil").innerHTML = text;
```

file:///E:/malas%20ngoding/tutorial/javascript/for.html

Perulangan For Pada Javascript

www.malasngoding.com

TOMBOL
Angka 0
Angka 1
Angka 2
Angka 3
Angka 4
Angka 5
Angka 6
Angka 7
Angka 8
Angka 9
Angka 10

5 Macam Bentuk Perulangan pada Javascript

Apa yang akan kamu lakukan saat disuruh mencetak kalimat berulang-ulang?

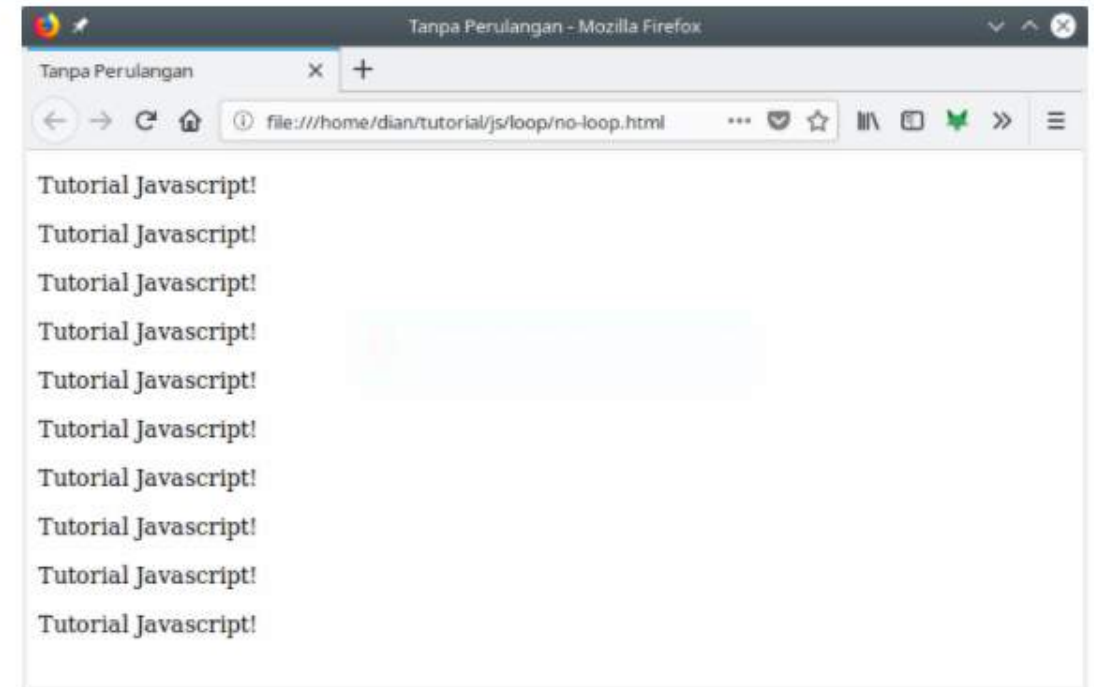
Misalnya:

“Tolong tampilkan kalimat "Tutorial Javascript!" di website saya sebanyak 10 kali”

Mungkin kamu bisa menuliskannya dengan fungsi `document.write()` sebanyak 10 kali seperti ini:

```
document.write("<p>Tutorial Javascript!</p>");  
document.write("<p>Tutorial Javascript!</p>");  
document.write("<p>Tutorial Javascript!</p>");  
document.write("<p>Tutorial Javascript!</p>");  
document.write("<p>Tutorial Javascript!</p>");  
document.write("<p>Tutorial Javascript!</p>");  
document.write("<p>Tutorial Javascript!</p>");  
document.write("<p>Tutorial Javascript!</p>");  
document.write("<p>Tutorial Javascript!</p>");  
document.write("<p>Tutorial Javascript!</p>");
```

Hasilnya:



Apakah boleh seperti ini?

Ya, boleh-boleh saja.

Tapi...

Bagaimana kalau nanti dia mau menampilkan sebanyak 1000 kali.



Perulangan akan membantu kita mengeksekusi kode yang berulang-ulang, berapapun yang kita mau.

Ada lima macam bentuk perulangan di Javascript. Secara umum, perulangan ini dibagi dua.

Yaitu: *counted loop* dan *uncounted loop*.

Perbedaanya:

- **Counted Loop** merupakan perulangan yang jelas dan sudah tentu banyak perulangannya.
- Sedangkan **Uncounted Loop**, merupakan perulangan yang tidak jelas berapa kali ia harus mengulan.

Perulangan yang termasuk dalam *Counted Loop*:

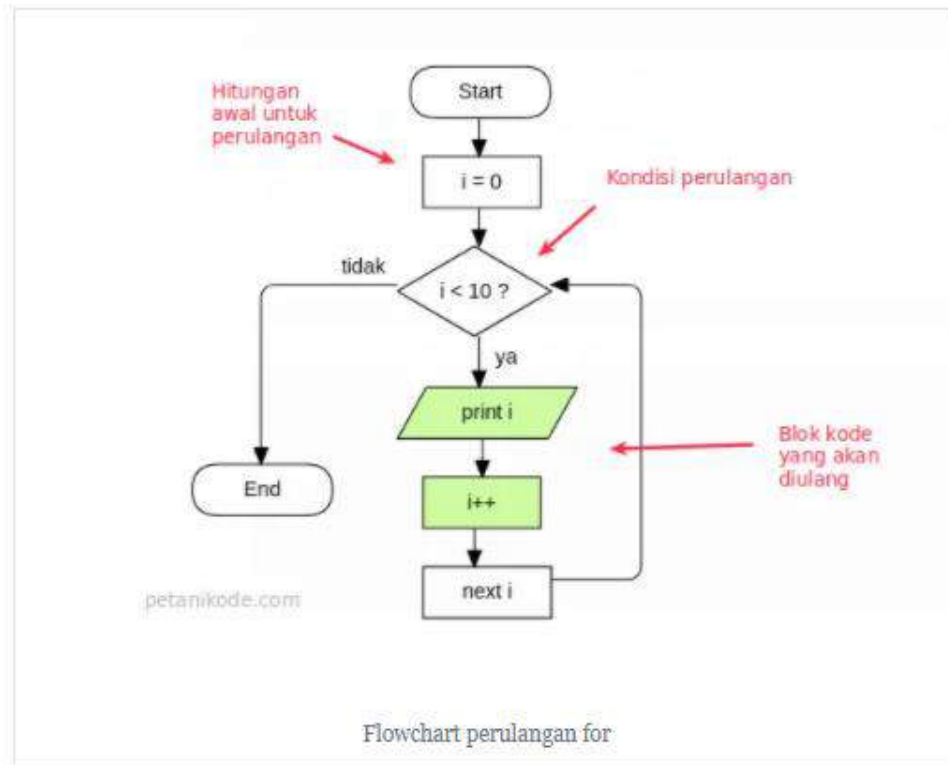
1. Perulangan For
2. Perulangan Foreach
3. Perulangan Repeat

Perulangan yang termasuk dalam *Uncounted Loop*:

1. Perulangan While
2. Perulangan Do/While

1. Perulangan For di Javascript

Perulangan `for` merupakan perulangan yang termasuk dalam *counted loop*, karena sudah jelas berapa kali ia akan mengulang.



Bentuknya kodenya seperti ini:

```
for(let i = 0; i < 10; i++){  
  document.write("<p>Perulangan ke-" + i + "</p>")  
}
```

Yang perlu diperhatikan adalah kondisi yang ada di dalam kurung setelah kata `for`.

Kondisi ini akan menentukan:

- Hitungan akan dimulai dari 0 (`i = 0`);
- Hitungannya sampai berapa? Sampai `i < 10`;
- Lalu di setiap perulangan `i` akan bertambah +1 (`i++`).

Variabel `i` pada perulangan `for` berfungsi untuk menyimpan nilai hitungan.

Jadi setiap perulangan dilakukan nilai `i` akan selalu bertambah satu. Karena kita menentukannya di bagian `i++`.

Apakah nama variabelnya harus selalu `i`?

Tidak.

Kita juga bisa menggunakan nama lain.

Misalnya:

```
for(counter = 0; counter < 50; counter+=2){  
    document.write("<p>Perulangan ke-"+counter+"</p>");  
}
```

Pada contoh tersebut, kita melakukan perulangan dimulai dari nol `0`. Lalu di setiap perulangan nilai variabel `counter` akan ditambah `2` (`counter+=2`).

Hasilnya:





Bagaimana kalau *counter* perulangannya dimulai dari akanga yang lebih besar sampai yang ke paling kecil?

Ini biasanya kita buat saat ingin menghitung mundur...

Caranya gampang.

Kita tinggal isi nilai *counter* dengan nilai terbesarnya.

Misalnya kita akan mulai hitungan dari 10 sampai ke 0.

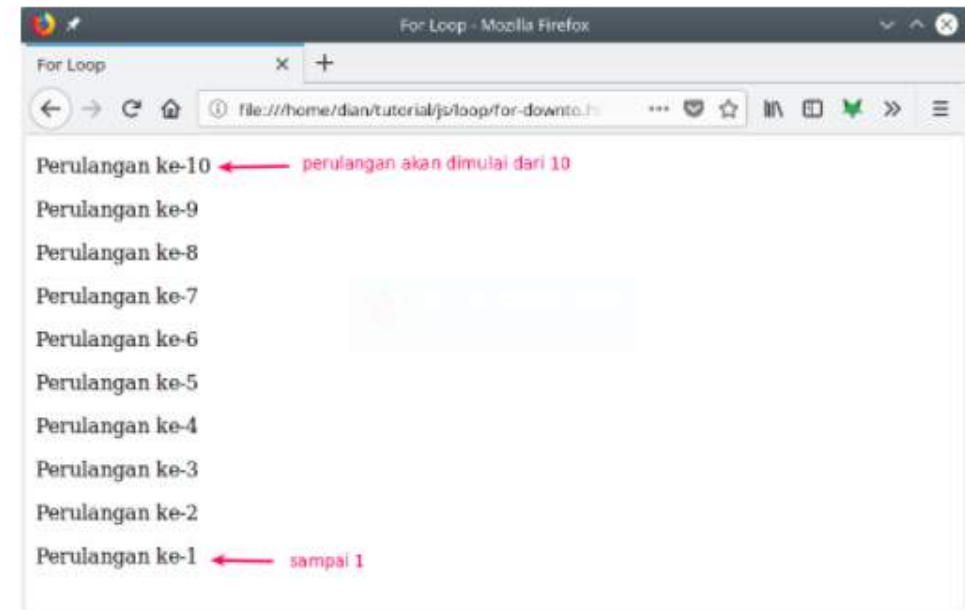
Maka nilai *counter*, kita isi awalnya dengan 10.

Lalu di kondisi perbandingannya, kita berikan *counter* > 0. Artinya perulangan akan dilakukan selama nilai *counter* lebih besar dari 0.

Lalu kita kurangi (-1) nilai *counter* di setiap perulangan (*counter--*).

```
for(counter = 10; counter > 0; counter--){
  document.write("<p>Perulangan ke "+counter+"</p>");
}
```

Hasilnya:



Mengapa tidak sampai nol (0)?

Karena kondisi yang kita berikan *counter* > 0. Apabila *counter* bernilai 0, maka kondisi ini akan menjadi *false*.

Kecuali kita menggunakan operator lebih besar sama dengan (>=), maka jika *counter* bernilai 0, kondisi akan menjadi *true*.

2. Perulangan While di Javascript

Perulangan `while` merupakan perulangan yang termasuk dalam perulangan *uncounted loop*.

Perulangan `while` juga dapat menjadi perulangan yang *counted loop* dengan memberikan `counter` di dalamnya.

Untuk memahami perulangan ini...

...mari kita coba lihat contohnya:

```
var ulangi = confirm("Apakah anda mau mengulang?");
var counter = 0;

while(ulangi){
  var jawab = confirm("Apakah anda mau mengulang?")
  counter++;
  if(jawab == false){
    ulangi = false;
  }
}

document.write("Perulangan sudah dilakuakn sebanyak "+ counter +" kali");
```

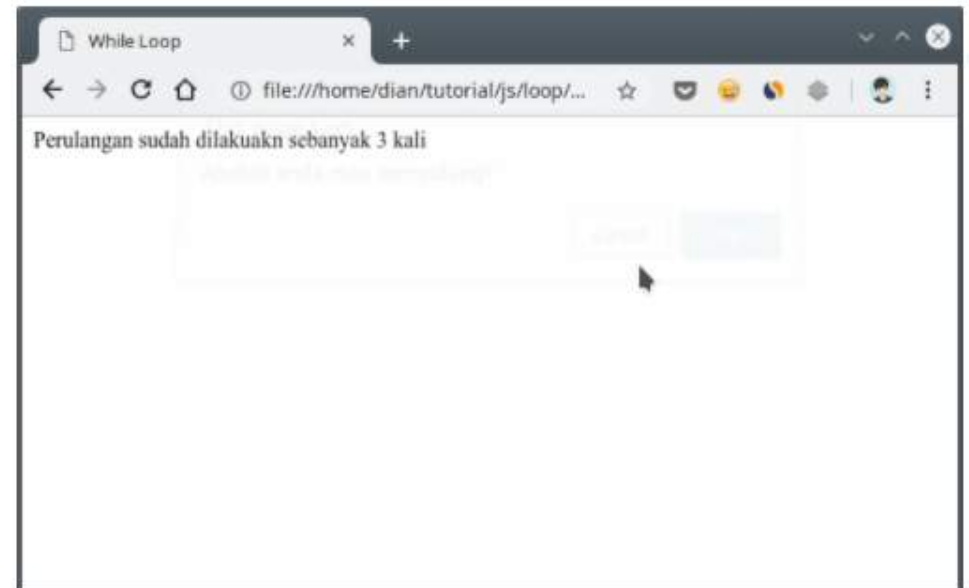
Bisa disederhanakan menjadi:

```
var ulangi = confirm("Apakah anda mau mengulang?");
var counter = 0;

while(ulangi){
  counter++;
  ulangi = confirm("Apakah anda mau mengulang?");
}

document.write("Perulangan sudah dilakuakn sebanyak "+ counter +" kali");
```

Hasilnya:





Coba perhatikan blok kode `while`:

```
while(ulang){
  counter++;
  ulang = confirm("Apakah anda mau mengulang?");
}
```

Di sana... Perulangan akan terjadi selama variabel `ulang` bernilai `true`.

Lalu kita menggunakan fungsi `confirm()` untuk menampilkan dialog konfirmasi.

Selama kita memilih **Ok** pada dialog konfirmasi, maka variabel `ulang` akan terus bernilai `true`.

Tapi kalau kita pilih **Cancel**, maka variabel `ulang` akan bernilai `false`.

Saat variabel `ulang` bernilai `false`, maka perulangan akan dihentikan.

3. Perulangan Do/While di Javascript

Perulangan `do/while` sama seperti perulangan `while`.

Perbedaanya:

Perulangan `do/while` akan melakukan perulangan sebanyak 1 kali terlebih dahulu, lalu mengecek kondisi yang ada di dalam kurung `while`.

Bentuknya seperti ini:

```
do {  
  // blok kode yang akan diulang  
} while (<kondisi>);
```

Jadi perbedaanya:

Perulangan `do/while` akan mengecek kondisi di belakang (sesudah mengulang), sedangkan `while` akan mencek kondisi di depan atau awal (sbelum mengulang).

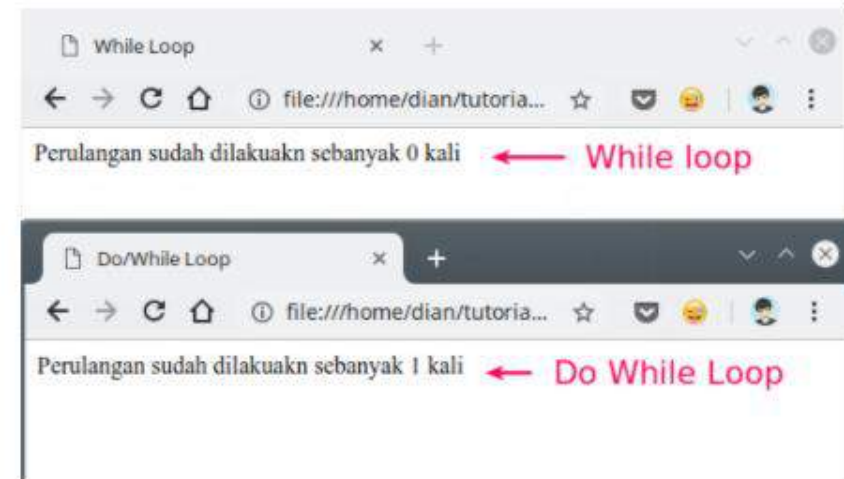
Mari kita coba lihat contohnya:

```
var ulangi = confirm("Apakah anda mau mengulang?");  
var counter = 0;  
  
do {  
  counter++;  
  ulangi = confirm("Apakah anda mau mengulang?");  
} while(ulangi)  
  
document.write("Perulangan sudah dilakuakn sebanyak "+ counter +" kali");
```

Contoh tersebut sama seperti contoh pada perulangan `while`.

Saat perulangan pertama, cobalah untuk membatalkan perulangannya dengan memilih **Cancel**.

Maka hasilnya:



4. Perulangan Foreach di Javascript

Perulangan `foreach` biasanya digunakan untuk mencetak item di dalam array.

Perulangan ini termasuk dalam perulangan *counted loop*, karena jumlah perulangannya akan ditentukan oleh panjang dari array.

Ada dua cara menggunakan perulangan `foreach` di Javascript:

1. Menggunakan `for` dengan operator `in`;
2. Menggunakan method `forEach()`.

Contoh:

Berikut ini bentuk perulangan *“foreach”* tanpa menggunakan operator `in`:

```
var languages = ["Javascript", "HTML", "CSS", "Typescript"];

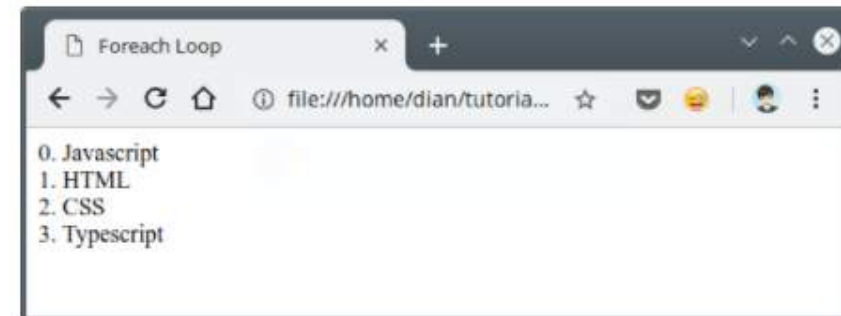
for(i = 0; i < languages.length; i++){
    document.write(i+. " + languages[i] + "<br/>");
}
```

Perulangan ini dapat dibuat lebih sederhana lagi dengan menggunakan operator `in` seperti ini:

```
var languages = ["Javascript", "HTML", "CSS", "Typescript"];

for(i in languages){
    document.write(i+. " + languages[i] + "<br/>");
}
```

Hasilnya:



Cara kedua membuat perulangan `foreach` ialah dengan menggunakan method `forEach()` dari array.

Contoh:

```
// kita punya array seperti berikut
var days = ["Senin", "Selasa", "Rabu", "Kamis", "Jum'at", "Sabtu", "Minggu"];

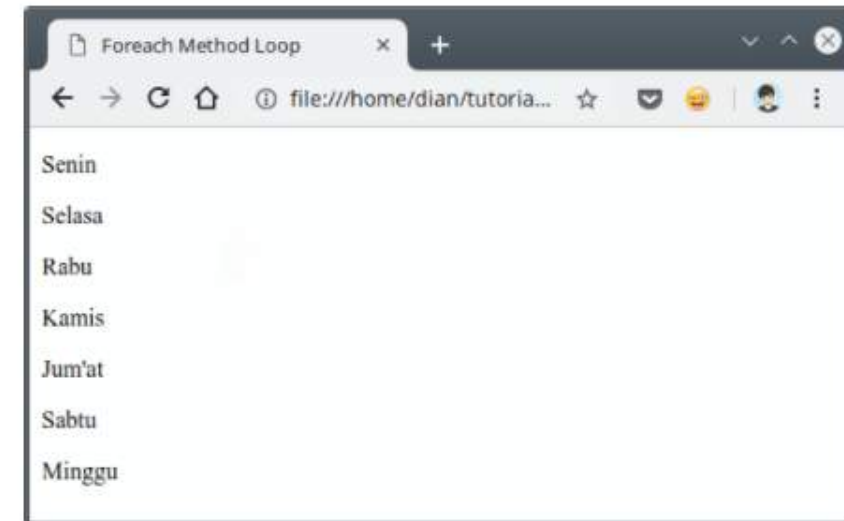
// Kemudian kita tampilkan semua hari
// dengan menggunakan method foreach
days.forEach(function(day){
    document.write("<p>" + day + "</p>");
});
```

Method `forEach()` memiliki parameter berupa fungsi *callback*. Sebenarnya kita juga bisa menggunakan *arrow function* seperti ini:

```
// kita punya array seperti berikut
var days = ["Senin", "Selasa", "Rabu", "Kamis", "Jum'at", "Sabtu", "Minggu"];

// Kemudian kita tampilkan semua hari
// dengan menggunakan method foreach
days.forEach((day) => {
    document.write("<p>" + day + "</p>");
});
```

Hasilnya:



5. Perulangan dengan Method repeat()

Perulangan dengan method atau fungsi `repeat()` termasuk dalam perulangan *counted loop*.

Fungsi ini khusus digunakan untuk mengulang sebuah teks (string).

Bisa dibilang:

Ini merupakan singkat dari perulangan `for`.

Contoh:

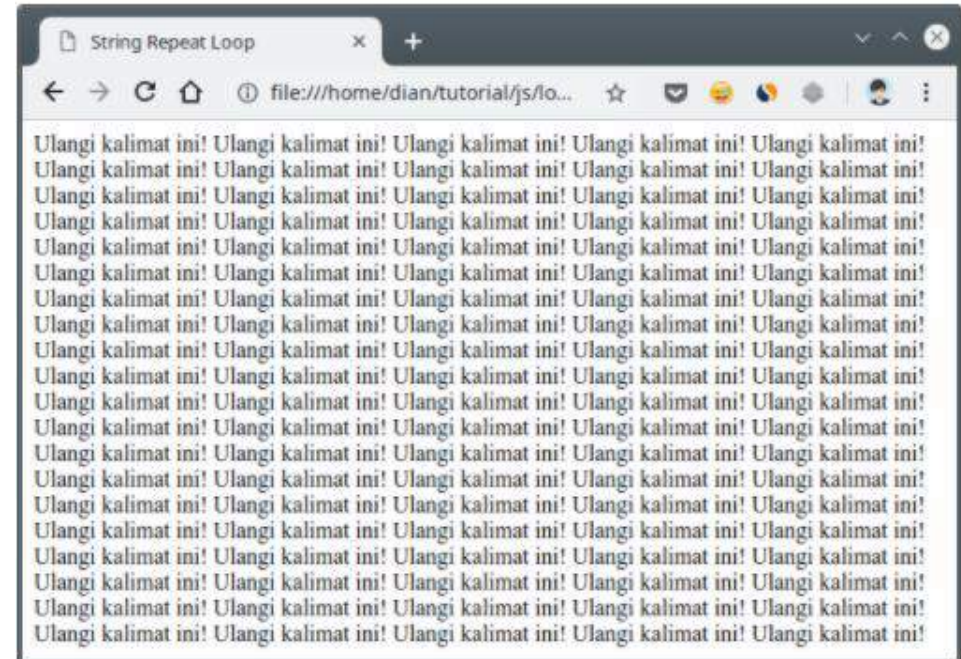
Apabila kita menggunakan perulangan `for`:

```
for( let i = 0; i < 100; i++){  
  document.write("Ulangi kalimat ini!");  
}
```

Apabila kita menggunakan fungsi `repeat()`:

```
document.write("Ulangi kalimat ini! ".repeat(100));
```

Hasilnya:



6. Bonus: Perulangan Bersarang (*Nested*)

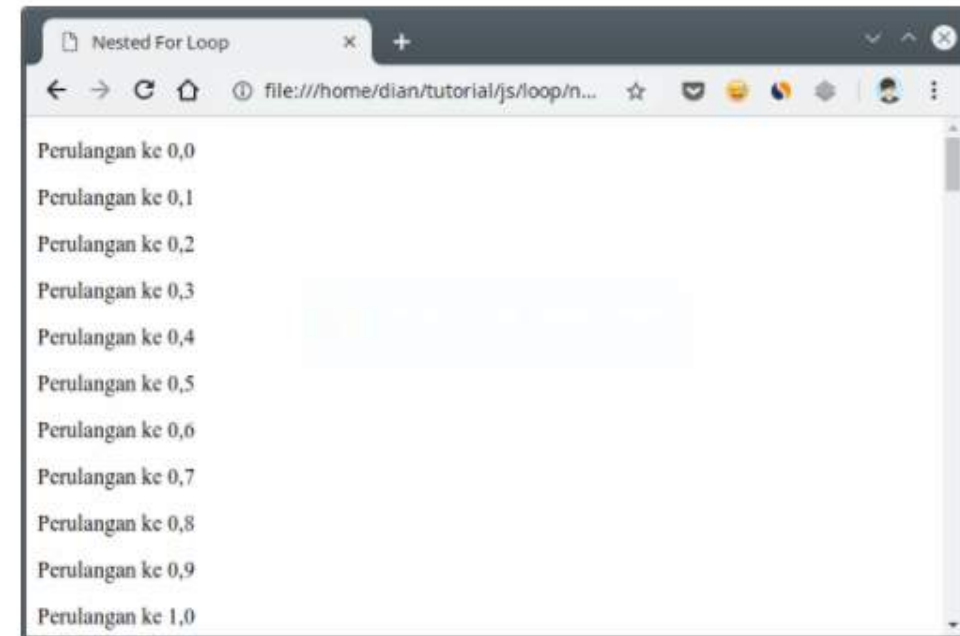
Di dalam blok perulangan, kita juga dapat membuat perulangan.

Ini disebut dengan *nested loop* atau perulangan bersarang atau perulangan di dalam perulangan.

Mari kita coba lihat contohnya:

```
for(let i = 0; i < 10; i++){  
  for(let j = 0; j < 10; j++){  
    document.write("<p>Perulangan ke " + i + "," + j + "</p>");  
  }  
}
```

Hasilnya:



Pada perulangan tersebut, kita menggunakan dua perulangan *for*.

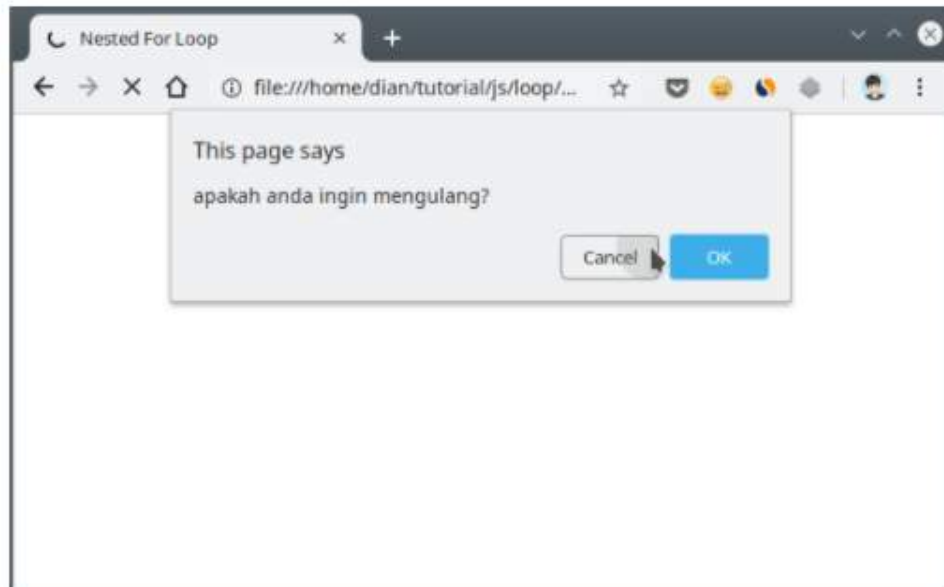
Perulangan pertama menggunakan variabel *i* sebagai *counter*, sedangkan perulangan kedua menggunakan variabel *j* sebagai *counter*.

Contoh lain:

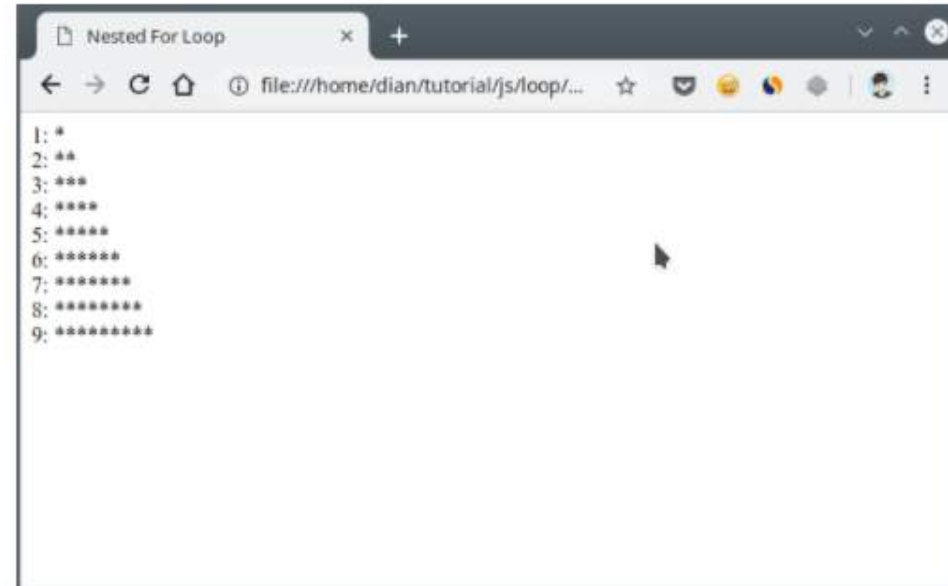
```
var ulangi = confirm("apakah anda ingin mengulang?");
var counter = 0;

while (ulangi) {
  counter++;
  var bintang = "*".repeat(counter) + "<br>";
  document.write(counter + ": " + bintang);
  ulangi = confirm("apakah anda ingin mengulang?");
}
```

Hasilnya:



Hasilnya:





Belajar Javascript Part 12 : Manipulasi String Pada Javascript

Berikut ini beberapa **function javascript** untuk **memanipulasi string** dengan penjelasan dan kegunaannya masing-masing.

Nama Fungsi	Kegunaan
length	Menghitung jumlah karakter pada string
indexOf()	Menemukan letak string dalam sebuah string (di urutan ke berapa)
LastindexOf()	Menemukan letak string dalam sebuah string (di urutan ke berapa), di hitung dari akhir
search()	Mencari string di dalam string
slice()	Menampilkan string dari karakter ke berapa sampai yang ke berapa
substring()	Menampilkan string dari karakter ke berapa sampai yang ke berapa (sama seperti slice())
substr()	Menampilkan string dari karakter ke berapa sampai yang ke berapa (sama seperti slice())
replace()	Mengganti string
toUpperCase()	Mengubah string menjadi huruf besar semua
toLowerCase()	Mengubah string menjadi huruf kecil semua
concat()	Menggabungkan string
split()	Mengubah string menjadi array

Menghitung jumlah karakter pada string dengan length

Berikut adalah contoh cara menghitung jumlah karakter pada sebuah string dengan menggunakan javascript.

```
1 <h2>www.malasngoding.com</h2>
2
3 <p id="contoh"></p>
4
5 <script>
6 var kalimat = "saya belajar javascript di www.malasngoding.com";
7 document.getElementById("contoh").innerHTML = kalimat.length;
8 </script>
```

di sini kita buat sebuah variabel 'kalimat' yang menyimpan sebuah string "saya belajar javascript di www.malasngoding.com". lalu kita hitung jumlah karakter dari string tersebut dengan length.

```
1 document.getElementById("contoh").innerHTML = kalimat.length;
```

dan hasilnya :



The screenshot shows a live code editor interface. On the left, the HTML editor contains the code from the previous block. On the right, the result editor shows the output of the code: the string "www.malasngoding.com" and the number "47". The interface includes a "Resources" tab at the bottom left, a "Run" button at the bottom right, and zoom controls (1x, 0.5x, 0.25x) in the bottom center.

Menemukan letak string yang di cari dengan indexOf() dan LastindexOf()

Untuk menemukan letak dari string yang di cari, kita bisa menggunakan fungsi yang sudah di sediakan oleh javascript. yaitu indexOf() atau LastindexOf() jika ingin menghitung urutannya dari akhir.

contoh :

```
1 <h2>www.malasngoding.com</h2>
2
3 <p id="contoh"></p>
4
5 <script>
6
7     var kalimat = "Selamat datang di tutorial string javascript";
8     var temukan = kalimat.indexOf("tutorial");
9     document.getElementById("contoh").innerHTML = temukan;
10
11 </script>
```

silahkan teman-teman tes juga dengan mengganti **indexOf** menjadi **LastindexOf**. karena cara penulisannya sama saja.



The screenshot shows a CodePen live editor interface. The 'HTML' pane contains the following code:

```
<h2>www.malasngoding.com</h2>
<p id="contoh"></p>
<script>
    var kalimat = "Selamat datang di tutorial string javascript";
    var temukan = kalimat.indexOf("tutorial");
    document.getElementById("contoh").innerHTML = temukan;
</script>
```

The 'Result' pane shows the output of the code, which is the string `www.malasngoding.com` followed by the number `18`. The interface also includes a 'LIVE' indicator, a 'Resources' section, and zoom controls (1x, 0.5x, 0.25x) and a 'Run' button.

Mencari string dengan search()

Manipulasi String Pada Javascript. Cara mencari string di dalam string kita bisa menggunakan fungsi search(). contohnya:

```
2
3 <p id="contoh"></p>
4
5 <script>
6   var kalimat = "belajar javascript di www.malasngoding.com";
7   var cari = kalimat.search("javascript");
8   document.getElementById("contoh").innerHTML = cari;
9 </script>
```



Filter string dengan slice()

ada 2 parameter yang harus di isi pada fungsi slice(). yang pertama isikan dari urutan ke berapa filter di mulai. dan parameter kedua sampai urutan ke berapa batas filternya.

```
1 slice(mulai, batas akhir)
```

contonya

```
2
3 <p id="contoh"></p>
4
5 <script>
6   var kalimat = "belajar javascript di www.malasngoding.com";
7   var filter = kalimat.slice(22, 42);
8   document.getElementById("contoh").innerHTML = filter;
9 </script>
```

dan hasilnya adalah "www.malasngoding.com". karena kita memerintahkan untuk menampilkan karakter ke 22 sampai 42. seperti yang dapat kita lihat pada contoh di atas.

```
1 kalimat.slice(22, 42)
```



penggunaan **substring()** dan **substr()** juga sama. bisa teman-teman coba sendiri. **Manipulasi String Pada Javascript**

Mengganti string dengan replace()

cara penggunaan fungsi replace() pada javascript untuk mengganti string adalah sebagai berikut.


```
1 replace('yang ingin di ganti','ganti nya')
```

jadi pada parameter pertama, kita masukkan misalnya kata yang ingin kita ganti. dan pada parameter kedua nya kita masuk kan kata yang menjadi penggantinya.

```
1 <h2>www.malasngoding.com</h2>
2
3 <p id="contoh"></p>
4
5 <script>
6   var kalimat = "belajar javascript di www.malasngoding.com";
7   var ganti = kalimat.replace('javascript', 'pemrograman');
8   document.getElementById("contoh").innerHTML = ganti;
9 </script>
```

Maka yang sebelumnya isi variabel kalimat adalah "belajar javascript di www.malasngoding.com", maka berubah menjadi "belajar pemrograman di www.malasngoding.com".

karena di sini kita mengubah kata 'javascript' menjadi 'pemrograman' dengan menggunakan fungsi replace di javascript.



```
HTML Result EDIT ON CODEPEN
<h2>www.malasngoding.com</h2>
<p id="contoh"></p>
<script>
  var kalimat = "belajar javascript di www.malasngoding.com";
  var ganti = kalimat.replace('javascript', 'pemrograman');
  document.getElementById("contoh").innerHTML = ganti;
</script>
www.malasngoding.com
belajar pemrograman di www.malasngoding.com
Resources 1x 0.5x 0.25x Run
```

Huruf besar kecil dengan toUpperCase() dan toLowerCase()

Jika kita ingin mengubah isi string dengan huruf besar semua atau huruf kecil semua, kita bisa mengandalkan kedua fungsi ini, yaitu **toUpperCase()** untuk mengubah string menjadi huruf besar atau huruf kapital, dan **toLowerCase()** berguna untuk mengubah string menjadi huruf kecil semua. Manipulasi String Pada Javascript

```
1 <h2>www.malasngoding.com</h2>
2
3 <p id="besar"></p>
4 <p id="kecil"></p>
5
6 <script>
7   var kalimat = "Belajar JAVASCRIPT di www.malasngoding.com";
8
9   // mengubah ke huruf besar
10  var a = kalimat.toUpperCase();
11
12  // mengubah ke huruf kecil
13  var b = kalimat.toLowerCase();
14
15  document.getElementById("besar").innerHTML = a;
16  document.getElementById("kecil").innerHTML = b;
17 </script>
```



EDIT ON CODEPEN

HTML Result

LIVE

```
<h2>www.malasngoding.com</h2>
<p id="besar"></p>
<p id="kecil"></p>
<script>
  var kalimat = "Belajar JAVASCRIPT di www.malasngoding.com";
  // mengubah ke huruf besar
  var a = kalimat.toUpperCase();
  // mengubah ke huruf kecil
  var b = kalimat.toLowerCase();
  document.getElementById("besar").innerHTML = a;
  document.getElementById("kecil").innerHTML = b;
</script>
```

www.malasngoding.com

BELAJAR JAVASCRIPT DI
WWW.MALASNGODING.COM

belajar javascript di www.malasngoding.com

Rerun

1x 0.5x 0.25x

Menggabungkan string dengan concat()

Untuk contoh kita akan mencoba menggabungkan 2 string. sebagai pengganti "+" sebagai tanda penggabung string.

```
1 <h2>www.malasngoding.com</h2>
2
3 <p id="contoh"></p>
4
5 <script>
6     var a = "diki alfarabi";
7     var b = "hadi";
8
9     // menggabungkan string
10    var gabung = a.concat(" ",b);
11    document.getElementById("contoh").innerHTML = gabung;
12 </script>
```

pada contoh penggunaan concat() javascript ini, kita membuat sebuah variabel a yang berisi "diki alfarabi". dan variabel b berisi "hadi". sebenarnya saat di tampilkan bisa saja dengan cara a+" "+b. tapi dengan javascript bisa lebih mudah, yaitu menggunakan fungsi concat. yang hasilnya akan menjadi "diki alfarabi hadi".



```
HTML | Result | EDIT ON CODEPEN
```

```
<h2>www.malasngoding.com</h2>
<p id="contoh"></p>
<script>
  var a = "diki alfarabi";
  var b = "hadi";
  // menggabungkan string
</script>
```

www.malasngoding.com
diki alfarabi hadi

Resources | 1x | 0.5x | 0.25x | Run

Convert string ke array dengan split()

di rekomendasikan kepada teman-teman yang belum mempelajari tentang array untuk membaca tutorial www.malasngoding.com sebelumnya tentang tutorial array javascript. agar lebih mudah memahami fungsi split ini. karena di sini kita akan sedikit bermain dengan array pada javascript.

```
2
3 <p id="contoh"></p>
4
5 <script>
6   var huruf = "a,b,c";
7
8   // string jadi array
9   var convert = huruf.split(",");
10  document.getElementById("contoh").innerHTML = convert[0];
11 </script>
```

Pada contoh ini, kita membuat sebuah variabel bernama huruf. dan berisi 'a,b,c'. kemudian kita pecahkan string ini menjadi array dengan split(). dan menetapkan tanda ',' sebagai pemisah.

```
1 var convert = huruf.split(",");
```

nah, maka jadilah array yang tersimpan dalam variabel convert. dan menampilkan array urutan pertama

```
1 document.getElementById("contoh").innerHTML = convert[0];
```

urutan pertama pada array (0) adalah huruf a. maka hasilnya :



```
HTML Result EDIT ON CODEPEN
<h2>www.malasngoding.com</h2>
<p id="contoh"></p>
<script>
  var huruf = "a,b,c";
  // string jadi array
  var convert = huruf.split(",");
Resources 1x 0.5x 0.25x Run
```



Membuat Format Rupiah Dengan Javascript

Membuat Format Rupiah Dengan Javascript, Seperti yang kita sudah ketahui, javascript memiliki kegunaan yang besar dalam pengembangan sebuah website atau aplikasi. salah satunya adalah kita bisa membuat **format rupiah javascript**.

Jika mungkin teman-teman sedang mengerjakan sebuah aplikasi keuangan yang mengharuskan teman-teman untuk membuat format rupiah secara otomatis pada saat mengetik angka. maka angka akan secara otomatis menampilkan pemisah satuannya.

Misalnya pada saat angka yang di ketik berjumlah ribuan, maka akan muncul titik pada 3 angka terakhir, begitu juga dengan format angka jutaan, milyaran dan seterusnya.

Pada tutorial **Membuat Format Rupiah Dengan Javascript** ini. kita akan belajar tentang cara membuat **format rupiah javascript**.

Kita akan membuat sebuah form input, dimana form input tersebut akan kita buat sebuah fitur yang akan mengubah angka yang di input secara otomatis menjadi format angka.

Sekedar pengingat kembali, silahkan teman-teman baca juga tutorial javascript dasar di www.malasngoding.com. pada tutorial javascript sebelumnya sudah di jelaskan tentang dasar-dasar dari javascript. jadi akan sangat memudahkan teman-teman dalam mengikuti dan mempelajari materi tutorial **Membuat Format Rupiah Dengan Javascript** ini.

- [Tutorial dasar javascript lengkap bahasa indonesia](#)

Langsung saja, kita buat dulu sebuah form sebagai contoh, project ini saya buat dalam folder dengan nama "rupiah_javascript". atau teman-teman bisa langsung menerapkan pada project teman-teman.

index.html

```
1 <!DOCTYPE html>
2 <html>
3 <head>
4   <title>Membuat Format Rupiah Dengan Javascript - www.malasngoding.com</title>
5 </head>
6 <body>
7
8   <style type="text/css">
9     body {
10       font-family: sans-serif;
11     }
12     .kotak {
13       width: 350px;
14       margin: auto;
15       margin-top: 15px;
16       padding: 10px;
17     }
18
19     p{
20       margin-bottom: 20px;
21       color: #0004ff;
22     }
23
24     input {
25       text-align: right;
26       width: 100%;
27       margin-bottom: 20px;
28       margin-top: 10px;
29       padding: 7px 10px;
30       font-size: 18px;
31     }
32   </style>
33
```

```
34 <center>
35   <h1>Membuat Format Rupiah Dengan Javascript <br/> www.malasngoding.com</h1>
36 </center>
37
38 <div class="kotak">
39   <p>Ketik jumlah nominal pada form di bawah ini.</p>
40   <span>Nominal Rupiah. :</span>
41   <input type="text" id="rupiah"/>
42 </div>
43
44
45 <script type="text/javascript">
46
47   var rupiah = document.getElementById('rupiah');
48   rupiah.addEventListener('keyup', function(e){
49     // tambahkan 'Rp.' pada saat form di ketik
50     // gunakan fungsi formatRupiah() untuk mengubah angka yang di ketik menjadi
51     rupiah.value = formatRupiah(this.value, 'Rp. ');
52   });
53
54   /* Fungsi formatRupiah */
55   function formatRupiah(angka, prefix){
56     var number_string = angka.replace(/[,|\d]/g, '').toString(),
57     split          = number_string.split(','),
58     sisa           = split[0].length % 3,
59     rupiah         = split[0].substr(0, sisa),
60     ribuan         = split[0].substr(sisa).match(/\d{3}/gi);
61
62     // tambahkan titik jika yang di input sudah menjadi angka ribuan
63     if(ribuan){
64       separator = sisa ? '.' : '';
65       rupiah += separator + ribuan.join('.');
66     }
67
68     rupiah = split[1] != undefined ? rupiah + ',' + split[1] : rupiah;
69     return prefix == undefined ? rupiah : (rupiah ? 'Rp. ' + rupiah : '');
70   }
71 </script>
72 </body>
73 </html>
```


akan saya jelaskan terlebih dulu, baru kemudian kita akan melihat hasilnya.

Pertama kita buat sebuah form penginputan untuk angka.

```
1 <div class="kotak">
2   <p>Ketik jumlah nominal pada form di bawah ini.</p>
3   <span>Nominal Rupiah. :</span>
4   <input type="text" id="rupiah"/>
5 </div>
```

ktop/rupiah_javascript/index.html

Membuat Format Rupiah Dengan Javascript www.malasngoding.com

Ketik jumlah nominal pada form di bawah ini.

Nominal Rupiah. :

dan kemudian kita berikan sentuhan CSS nya, supaya lebih terlihat ganteng.

```
1 <style type="text/css">
2 body {
3   font-family: sans-serif;
4 }
5 .kotak {
6   width: 350px;
7   margin: auto;
8   margin-top: 15px;
9   padding: 10px;
10 }
11
12 p{
13   margin-bottom: 20px;
14   color: #0004ff;
15 }
16
17 input {
18   text-align: right;
19   width: 100%;
20   margin-bottom: 20px;
21   margin-top: 10px;
22   padding: 7px 10px;
23   font-size: 18px;
24 }
25 </style>
```

dan terakhir pada bagian javascript nya. kita deklarasikan id rupiah ke dalam variabel.

kemudian kita beri perintah untuk mengirim data yang di input ke function formatRupiah().

dan terakhir pada bagian javascript nya. kita deklarasikan id rupiah ke dalam variabel.
kemudian kita beri perintah untuk mengirim data yang di input ke function formatRupiah().

```
1 rupiah.addEventListener('keyup', function(e){
2 // tambahkan 'Rp.' pada saat form di ketik
3 // gunakan fungsi formatRupiah() untuk mengubah angka yang di ketik menjadi format ang
4 rupiah.value = formatRupiah(this.value, 'Rp. ');
```

dan function formatRupiah berisi syntax untuk membuat format angka. dengan menggunakan titik.

```
1 /* Fungsi formatRupiah */
2 function formatRupiah(angka, prefix){
3     var number_string = angka.replace(/[,|\d]/g, '').toString(),
4     split          = number_string.split(','),
5     sisa           = split[0].length % 3,
6     rupiah         = split[0].substr(0, sisa),
7     ribuan         = split[0].substr(sisa).match(/\d{3}/gi);
8
9     // tambahkan titik jika yang di input sudah menjadi angka ribuan
10    if(ribuan){
11        separator = sisa ? '.' : '';
12        rupiah += separator + ribuan.join('.');
13    }
14
15    rupiah = split[1] != undefined ? rupiah + ',' + split[1] : rupiah;
16    return prefix == undefined ? rupiah : (rupiah ? 'Rp. ' + rupiah : '');
17 }
```

hasil akhirnya kita return atau kembalikan penulisan yang dalam bentuk Rp.

Di sini kita juga menyeleksi penginputan selain angka akan di hapus jadi fungsi ini akan menghapus inputan selain angka.

Coba kita lihat hasilnya.

/Desktop/rupiah_javascript/Index.html

Membuat Format Rupiah Dengan Javascript www.malasngoding.com

Ketik jumlah nominal pada form di bawah ini.

Nominal Rupiah. :

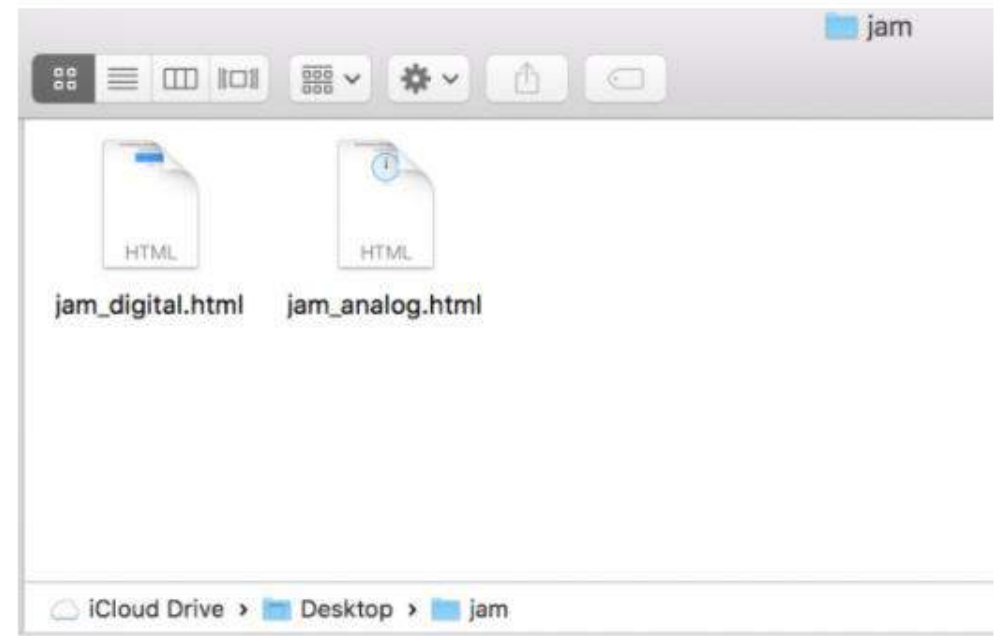
membuat format rupiah dengan javascript

Nah selesai, **format rupiah javascript** pun selesai.

Membuat Jam Analog dan Digital Dengan Javascript

Membuat Jam Analog dan Digital Dengan Javascript

Saya asumsikan kepada teman-teman untuk membaca terlebih dulu tutorial-tutorial javascript dasar di www.malasngoding.com. agar lebih mudah memahami syntax atau coding jam dengan javascript yang sebentar lagi akan kita buat.



membuat jam analog dan digital dengan javascript

Di tutorial ini kita akan belajar membuat jam digital dengan javascript. dan belajar membuat jam analog sederhana dengan javascript. dimana nantinya jam yang kita buat ini bisa teman-teman terapkan codingnya ke blog atau website teman-teman sebagai **widget**

Membuat Jam Digital Dengan Javascript

Untuk cara membuat jam digital dengan javascript silahkan teman-teman ketik coding berikut. dan menyimpannya di dalam file HTML atau PHP. di sini saya menyimpannya dalam file dengan nama jam_digital.html

jam_digital.html

```
1 <!DOCTYPE html>
2 <html>
3 <head>
4   <title>Membuat Jam Digital Dengan Javascript - www.malasngoding.com</title>
5 </head>
6 <body>
7
8   <style>
9     h1,h2,p,a{
10       font-family: sans-serif;
11       font-weight: normal;
12     }
13
14     .jam-digital-malasngoding {
15       overflow: hidden;
16       width: 330px;
17       margin: 20px auto;
18       border: 5px solid #efefef;
19     }
20     .kotak{
21       float: left;
22       width: 110px;
23       height: 100px;
24       background-color: #189fff;
25     }
```

```
25   }
26   .jam-digital-malasngoding p {
27     color: #fff;
28     font-size: 36px;
29     text-align: center;
30     margin-top: 30px;
31   }
32
33
34 </style>
35
36 <center>
37   <h1>Tutorial Membuat Jam Digital Dengan Javascript</h1>
38   <h2>www.malasngoding.com</h2>
39 </center>
40
41 <div class="jam-digital-malasngoding">
42   <div class="kotak">
43     <p id="jam"></p>
44   </div>
45   <div class="kotak">
46     <p id="menit"></p>
47   </div>
48   <div class="kotak">
49     <p id="detik"></p>
50   </div>
51 </div>
52
53 <center>
54   <a href="https://www.malasngoding.com/membuat-jam-analog-dan-digital-dengan-javas
55 </center>
56
```

```
57 <script>
58   window.setTimeout("waktu()", 1000);
59
60   function waktu() {
61     var waktu = new Date();
62     setTimeout("waktu()", 1000);
63     document.getElementById("jam").innerHTML = waktu.getHours();
64     document.getElementById("menit").innerHTML = waktu.getMinutes();
65     document.getElementById("detik").innerHTML = waktu.getSeconds();
66   }
67 </script>
68 </body>
69 </html>
```

Dan jika di jalankan maka hasilnya akan seperti berikut.

Tutorial Membuat Jam Digital Dengan Javascript

www.malasngoding.com



TUTORIAL

membuat jam digital dengan javascript

Seperti yang terlihat pada gambar jam digital di atas. jam, menit, detiknya pun muncul *lengan real time*

Penjelasan Jam Digital Javascript

Perhatikan pada syntax di atas tadi. kita membuat beberapa element HTML yang kita beri penamaan `class` dan `id` masing-masing.

```
1 <div class="jam-digital-malasngoding">
2   <div class="kotak">
3     <p id="jam"></p>
4   </div>
5   <div class="kotak">
6     <p id="menit"></p>
7   </div>
8   <div class="kotak">
9     <p id="detik"></p>
10  </div>
11 </div>
```

ada 3 element penting di sana, yang kita beri id jam, id menit, dan id detik. pada masing-masing id tersebut akan kita tampilkan sesuai dengan nama id nya masing-masing dengan `javascript`. misalnya pada id jam kita tampilkan jam sekarang, pada id menit kita tampilkan menit sekarang, dan pada id detik kita tampilkan detik yang berjalan sekarang secara *real-time*.

perhatikan pada bagian javascript.

```
1 <script>
2   window.setTimeout("waktu()", 1000);
3
4   function waktu() {
5     var waktu = new Date();
6     setTimeout("waktu()", 1000);
7     document.getElementById("jam").innerHTML = waktu.getHours();
8     document.getElementById("menit").innerHTML = waktu.getMinutes();
9     document.getElementById("detik").innerHTML = waktu.getSeconds();
10  }
11 </script>
```

di sini kita memerintahkan menjalankan `function waktu()` saat 1 detik pertama halaman ini di jalankan.

dan perhatikan pada `function waktu()`. di sana kita membuat sebuah variabel dengan nama "waktu". dan menyimpan data waktu atau tanggal sekarang.

```
1 var waktu = new Date();
```

kemudian menjalankan lagi function waktu() setelah 1 detik. ini untuk membuat perintah agar fungsi ini di jalankan setiap detik. sesuai dengan jam digital yang menampilkan perubahan detiknya setiap 1 detik sekali.

dan terakhir, data waktu atau tanggal yang telah tersimpan dalam variabel waktu tadi kita pecah-pecah isinya dengan cara menampilkan jam (`waktu.getHours()`) pada id jam. menampilkan menit (`waktu.getMinutes()`) pada id menit. serta menampilkan detik (`waktu.getSeconds()`) pada id detik.

```
1 document.getElementById("jam").innerHTML = waktu.getHours();
2 document.getElementById("menit").innerHTML = waktu.getMinutes();
3 document.getElementById("detik").innerHTML = waktu.getSeconds();
```

Membuat Jam Analog Dengan Javascript

Kita lanjutkan dengan belajar membuat jam analog dengan javascript.

Perhatikan syntax berikut.

```
1 <!DOCTYPE html>
2 <html>
3 <head>
4   <title>Membuat Jam Digital Dengan Javascript - www.malasngoding.com</title>
5   <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.3.1/jquery.min.js"></script>
6 </head>
7 <body>
8   <style type="text/css">
9
10  h1,h2,p,a{
11    font-family: sans-serif;
12    font-weight: normal;
13  }
14
15  .jam_analog_malasngoding {
16    background: #e7f2f7;
17    position: relative;
18    width: 240px;
19    height: 240px;
20    border: 16px solid #52b6f0;
21    border-radius: 50%;
22    padding: 20px;
23    margin: 20px auto;
24  }
25
26  .xxx {
27    height: 100%;
28    width: 100%;
29    position: relative;
30  }
```

```
31
32  .jarum {
33    position: absolute;
34    width: 50%;
35    background: #232323;
36    top: 50%;
37    transform: rotate(90deg);
38    transform-origin: 100%;
39    transition: all 0.05s cubic-bezier(0.1, 2.7, 0.58, 1);
40  }
41
42  .lingkaran_tengah {
43    width: 24px;
44    height: 24px;
45    background: #232323;
46    border: 4px solid #52b6f0;
47    position: absolute;
48    top: 50%;
49    left: 50%;
50    margin-left: -14px;
51    margin-top: -14px;
52    border-radius: 50%;
53  }
54
55  .jarum_detik {
56    height: 2px;
57    border-radius: 1px;
58    background: #F0C952;
59  }
60
61  .jarum_menit {
62    height: 4px;
63    border-radius: 4px;
64  }
65
```




```
65
66   .jarum_jam {
67     height: 8px;
68     border-radius: 4px;
69     width: 35%;
70     left: 15%;
71   }
72 </style>
73
74
75 <center>
76   <h1>Tutorial Membuat Jam Digital Dengan Javascript</h1>
77   <h2>www.malasngoding.com</h2>
78 </center>
79
80 <div class="jam_analog_malasngoding">
81   <div class="xxx">
82     <div class="jarum jarum_detik"></div>
83     <div class="jarum jarum_menit"></div>
84     <div class="jarum jarum_jam"></div>
85     <div class="lingkaran_tengah"></div>
86   </div>
87 </div>
88
89 <center>
90   <a href="https://www.malasngoding.com/membuat-jam-analog-dan-digital-dengan-java
91 </center>
92
93
94
95
```

```
95
96 <script type="text/javascript">
97   const secondHand = document.querySelector('.jarum_detik');
98   const minuteHand = document.querySelector('.jarum_menit');
99   const jarum_jam = document.querySelector('.jarum_jam');
100
101   function setDate(){
102     const now = new Date();
103
104     const seconds = now.getSeconds();
105     const secondsDegrees = ((seconds / 60) * 360) + 90;
106     secondHand.style.transform = `rotate(${secondsDegrees}deg)`;
107     if (secondsDegrees === 90) {
108       secondHand.style.transition = 'none';
109     } else if (secondsDegrees >= 91) {
110       secondHand.style.transition = 'all 0.05s cubic-bezier(0.1, 2.7, 0.58, 1)';
111     }
112
113     const minutes = now.getMinutes();
114     const minutesDegrees = ((minutes / 60) * 360) + 90;
115     minuteHand.style.transform = `rotate(${minutesDegrees}deg)`;
116
117     const hours = now.getHours();
118     const hoursDegrees = ((hours / 12) * 360) + 90;
119     jarum_jam.style.transform = `rotate(${hoursDegrees}deg)`;
120   }
121
122   setInterval(setDate, 1000)
123 </script>
124 </body>
125 </html>
```

Coba jalankan pada browser dan hasilnya seperti berikut.

Tutorial Membuat Jam Digital Dengan Javascript

www.malasngoding.com



membuat jam analog dengan javascript

Dan selesai. caranya tidak jauh berbeda dengan penjelasan sebelumnya.

Teman-teman bisa menerapkan syntax jam analog atau jam digital dengan javascript ini langsung pada website atau blog teman-teman. dan bisa mengubah css nya agar tampilan jam nya bisa sesuai dengan keinginan teman-teman.

Membuat Inputan Hanya Angka Dengan Javascript

Terkadang di dalam membuat sebuah aplikasi atau website, kita diharuskan untuk membuat form input yang hanya memperbolehkan untuk menginput angka. hal ini bertujuan untuk meminimalisir kesalahan di dalam sebuah aplikasi yang kita buat. untuk **cara membuat validasi angka dengan javascript** silahkan teman-teman perhatikan pada penjelasan berikut.

Buat sebuah file html atau php terserah teman-teman. di sini saya menyediakan sebuah file dengan nama angka.html. di mana di file ini akan kita buat validasi angka atau membuat validasi hanya angka dengan javascript. Pertama kita akan membuat sebuah form dulu teman-teman.

angka.html

```
1 <!DOCTYPE html>
2 <html>
3 <head>
4   <title>Membuat Inputan Hanya Angka Dengan Javascript | www.malasngoding.com</title>
5 </head>
6 <body>
7   <h1>Membuat Inputan Hanya Angka Dengan Javascript</h1>
8   <h2>www.malasngoding.com</h2>
9
10  <input type="text" onkeypress="return hanyaAngka(event)"/>
11
12  <script>
13    function hanyaAngka(evt) {
14      var charCode = (evt.which) ? evt.which : event.keyCode
15      if (charCode > 31 && (charCode < 48 || charCode > 57))
16
17        return false;
18      return true;
19    }
20  </script>
21 </body>
22 </html>
```

coba teman-teman perhatikan pada contoh syntax membuat form input angka di atas.

pertama kita buat dulu sebuah form biasa.

coba teman-teman perhatikan pada contoh syntax membuat form input angka di atas.
pertama kita buat dulu sebuah form biasa.

```
1 <input type="text" onkeypress="return hanyaAngka(event)">
```

tapi di form ini kita memberikan event **onkeypress=""**. untuk membuat aksi pada saat form ini di ketikkan atau di input. coba teman-teman baca juga tutorial saya sebelumnya tentang [Pengenalan event pada javascript](#). untuk menambah pemahaman lebih pada **event javascript**. jadi pada saat terjadi pengetikan pada form ini, maka akan di jalankan function **hanyaAngka()**.

```
1 onkeypress="return hanyaAngka(event)"
```

sekalian mereturn nilai dari fungsi ini.

selanjutnya teman-teman perhatikan pada function hanyaAngka().

```
1 function hanyaAngka(evt) {  
2   var charCode = (evt.which) ? evt.which : event.keyCode  
3   if (charCode > 31 && (charCode < 48 || charCode > 57))  
4  
5     return false;  
6     return true;  
7 }
```

di sini kita melakukan pengecekan jika yang di input adalah angka, maka akan mengembalikan nilai true pada form.

di sini kita melakukan pengecekan jika yang di input adalah angka, maka akan mengembalikan nilai true pada form.

```
1 if (charCode > 31 && (charCode < 48 || charCode > 57))
```

sekarang coba jalankan pada browser.



Membuat Inputan Hanya Angka Dengan Javascript

www.malasngoding.com



Hanya angka yang bisa di ketik

www.malasngoding.com

membuat inputan hanya angka dengan javascript



Bagaimana Membuat Batas maksimal digit angka yang di masukkan ?

Untuk membuat maksima angka yang di masukkan teman-teman tinggal menambahkan atribut **maxlength** pada element form tersebut. contohnya.

```
1 <input type="text" maxlength="2" onkeypress="return hanyaAngka(event)"/>
```

nah, pada contoh di atas berarti kita hanya memperbolehkan 2 digit angka yang di input. selanjutnya tidak akan bisa terketik pada form.



Membuat Form Validasi dengan Javascript

Membuat Form Validasi dengan Javascript

Untuk teman-teman yang belum membaca tutorial form validasi lainnya di www.malasngoding.com silahkan baca tutorial berikut ini.

- [Form validasi dengan PHP](#)
- [Form validasi dengan JQuery](#)

Oke yuk langsung simak apa saja yang kita butuhkan untuk membuat form validasi dengan javascript. silahkan teman-teman ikuti panduan dan penjelasan berikut ini untuk **membuat validasi pada form dengan mudah**.

validasi yang kita buat di tutorial ini adalah form validasi pendaftaran sederhana. dimana nantinya teman-teman bisa mengembangkannya lagi sesuai dengan kebutuhan aplikasi yang akan teman-teman ingin buat.

Membuat Form Validasi dengan Javascript

Pertama kali yang paling penting. kita akan buat dulu sebuah form sederhana. yaitu form pendaftaran. dimana pada form ini terdapat **form input nama, email dan alamat**.

buat sebuah file html atau php, dengan nama apa saja bebas terserah teman-teman. tapi pada tutorial ini saya akan membuat file php pada htdocs dengan nama file daftar.php(silahkan sesuaikan dengan keinginan).

daftar.php

```
1 <!DOCTYPE html>
2 <html>
3 <head>
4   <title>Membuat Form Validasi dengan Javascript - WWW.MALASNGODING.COM</title>
5   <link rel="stylesheet" type="text/css" href="style.css">
6 </head>
7 <body>
8   <center><h2>WWW.MALASNGODING.COM</h2></center>
9   <div class="login">
10    <form action="#" method="POST" onSubmit="validasi()">
11      <div>
12        <label>Nama Lengkap:</label>
13        <input type="text" name="nama" id="nama" />
14      </div>
15      <div>
16        <label>Email:</label>
17        <input type="email" name="email" id="email" />
18      </div>
19      <div>
20        <label>Alamat:</label>
21        <textarea cols="40" rows="5" name="alamat" id="alamat"></textarea>
22      </div>
23      <div>
24        <input type="submit" value="Daftar" class="tombol">
25      </div>
26    </form>
27  </div>
28 </body>
```

```
29 <script type="text/javascript">
30   function validasi() {
31     var nama = document.getElementById("nama").value;
32     var email = document.getElementById("email").value;
33     var alamat = document.getElementById("alamat").value;
34     if (nama != "" && email!="" && alamat != "") {
35       return true;
36     }else{
37       alert('Anda harus mengisi data dengan lengkap !');
38     }
39   }
40 </script>
41 </html>
```

kemudian buat file dengan nama style.css. karena kita akan **mendesain tampilan form pendaftaran** ini. agar lebih terlihat menarik.


```

1 body {
2   background: #3498db;
3   font-family: sans-serif;
4 }
5
6 h2 {
7   color: #fff;
8 }
9
10 .login {
11   padding: 1em;
12   margin: 2em auto;
13   width: 17em;
14   background: #fff;
15   border-radius: 3px;
16 }
17
18 label {
19   font-size: 10pt;
20   color: #555;
21 }
22
23 input[type="text"],
24 input[type="email"],
25 textarea {
26   padding: 8px;
27   width: 95%;
28   background: #efefef;
29   border: 0;
30   font-size: 10pt;
31   margin: 6px 0px;
32 }
33
34 .tombol {
35   background: #3498db;
36   color: #fff;
37   border: 0;
38   padding: 5px 8px;
39 }

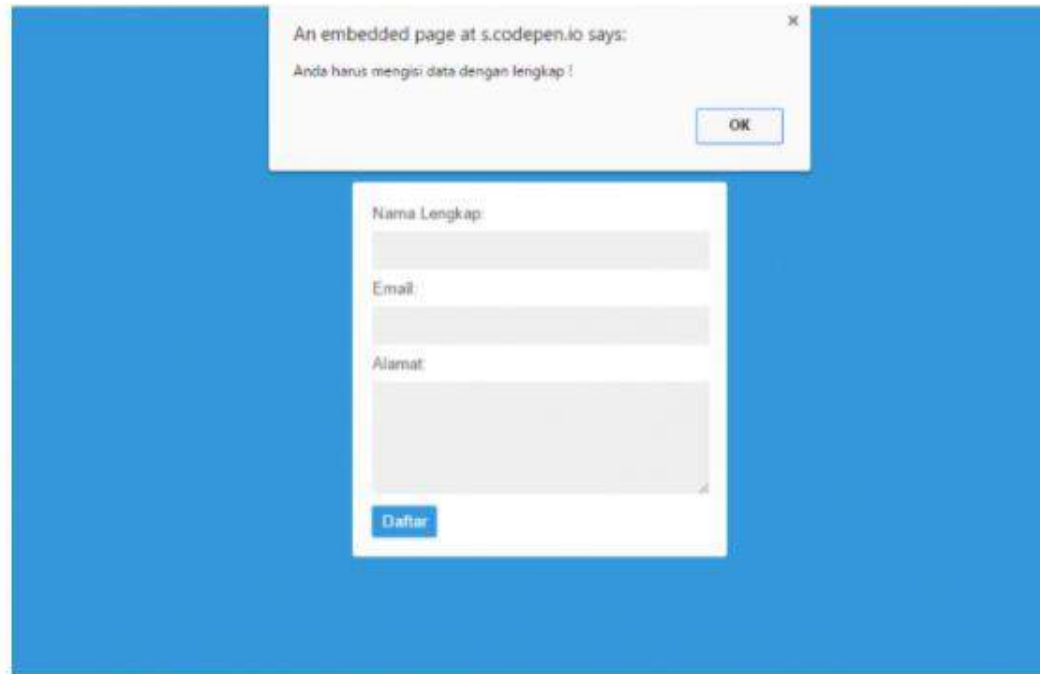
```



sekarang coba akses *daftar.php* yang sudah kita buat untuk melihat hasil **form validasi pendaftaran dengan javascript**.

membuat form validasi dengan javascript

dan coba klik tombol daftar tanpa mengisi form. maka akan muncul pemberitahuan alert "*Anda harus mengisi data dengan lengkap !*".



membuat validasi form pendaftaran dengan javascript

Dan pemberitahuan untuk mengisi form dengan lengkap pun berhasil di tampilkan..

Penjelasan

nah. coba teman-teman perhatikan pada *form pendaftaran* yang sudah kita buat.

Di sini kita memerintahkan untuk menjalankan function `validasi()` saat form di submit. `onSubmit="validasi()"`.

```
1 <form action="#" method="POST" onSubmit="validasi()">
2   <div>
3     <label>Nama Lengkap:</label>
4     <input type="text" name="nama" id="nama" />
5   </div>
6   <div>
7     <label>Email:</label>
8     <input type="email" name="email" id="email" />
9   </div>
10  <div>
11    <label>Alamat:</label>
12    <textarea cols="40" rows="5" name="alamat" id="alamat"></textarea>
13  </div>
14  <div>
15    <input type="submit" value="Daftar" class="tombol">
16  </div>
17 </form>
```

yang mana function validasi sudah kita buat untuk mengecek inputan.

```
1 function validasi() {
2   var nama = document.getElementById("nama").value;
3   var email = document.getElementById("email").value;
4   var alamat = document.getElementById("alamat").value;
5   if (nama != "" && email != "" && alamat != "") {
6     return true;
7   }else{
8     alert('Anda harus mengisi data dengan lengkap !');
9   }
10 }
```

kita menangkap nilai value dari form input. dan memasukkannya ke dalam [variabel](#).

```
1 var nama = document.getElementById("nama").value;
2 var email = document.getElementById("email").value;
3 var alamat = document.getElementById("alamat").value;
```

dan kemudian mengecek masing-masing variabel.

```
1 if (nama != "" && email != "" && alamat != "") {
2   return true;
3 }else{
4   alert('Anda harus mengisi data dengan lengkap !');
5 }
```

jika nama, email dan alamat di isi. atau tidak kosong. maka akan di kembalikan nilai TRUE pada form agar dapat di teruskan. jika tidak maka tampilkan pesan alert yang menampilkan "Anda harus mengisi data dengan lengkap !".

Thank You

ukrida.ac.id



UKRIDA
Universitas Kristen Krida Wacana



Modul Praktikum JavaScript Manipulasi String Pada Javascript

CYNTHIA HAYAT S.KOM., M.MSI

KRIDA WACANA CHRISTIAN UNIVERSITY
Faculty of Engineering and Computer Science
Departement of Information System



UKRIDA
Universitas Kristen Krida Wacana



Belajar Javascript Part 12 : Manipulasi String Pada Javascript

Berikut ini beberapa **function javascript** untuk **memanipulasi string** dengan penjelasan dan kegunaannya masing-masing.

Nama Fungsi	Kegunaan
length	Menghitung jumlah karakter pada string
indexOf()	Menemukan letak string dalam sebuah string (di urutan ke berapa)
LastindexOf()	Menemukan letak string dalam sebuah string (di urutan ke berapa), di hitung dari akhir
search()	Mencari string di dalam string
slice()	Menampilkan string dari karakter ke berapa sampai yang ke berapa
substring()	Menampilkan string dari karakter ke berapa sampai yang ke berapa (sama seperti slice())
substr()	Menampilkan string dari karakter ke berapa sampai yang ke berapa (sama seperti slice())
replace()	Mengganti string
toUpperCase()	Mengubah string menjadi huruf besar semua
toLowerCase()	Mengubah string menjadi huruf kecil semua
concat()	Menggabungkan string
split()	Mengubah string menjadi array

Menghitung jumlah karakter pada string dengan length

Berikut adalah contoh cara menghitung jumlah karakter pada sebuah string dengan menggunakan javascript.

```
1 <h2>www.malasngoding.com</h2>
2
3 <p id="contoh"></p>
4
5 <script>
6 var kalimat = "saya belajar javascript di www.malasngoding.com";
7 document.getElementById("contoh").innerHTML = kalimat.length;
8 </script>
```

di sini kita buat sebuah variabel 'kalimat' yang menyimpan sebuah string "saya belajar javascript di www.malasngoding.com". lalu kita hitung jumlah karakter dari string tersebut dengan length.

```
1 document.getElementById("contoh").innerHTML = kalimat.length;
```

dan hasilnya :



The screenshot shows a live code editor interface with two main panels: 'HTML' and 'Result'. The 'HTML' panel contains the code from the previous block. The 'Result' panel shows the output of the code, which is 'www.malasngoding.com' and '47'. The editor also has a 'Resources' panel at the bottom left and a 'Run' button at the bottom right.

Menemukan letak string yang di cari dengan indexOf() dan LastindexOf()

Untuk menemukan letak dari string yang di cari, kita bisa menggunakan fungsi yang sudah di sediakan oleh javascript. yaitu indexOf() atau LastindexOf() jika ingin menghitung urutannya dari akhir.

contoh :

```
1 <h2>www.malasngoding.com</h2>
2
3 <p id="contoh"></p>
4
5 <script>
6
7     var kalimat = "Selamat datang di tutorial string javascript";
8     var temukan = kalimat.indexOf("tutorial");
9     document.getElementById("contoh").innerHTML = temukan;
10
11 </script>
```

silahkan teman-teman tes juga dengan mengganti **indexOf** menjadi **LastindexOf**. karena cara penulisannya sama saja.



The screenshot shows a CodePen live editor interface. On the left, the HTML editor contains the code from the previous block. On the right, the result editor shows the output of the JavaScript code: 'www.malasngoding.com' and the value '18'. The interface includes a 'LIVE' indicator, zoom controls (1x, 0.5x, 0.25x), and a 'Run' button.

Mencari string dengan search()

Manipulasi String Pada Javascript. Cara mencari string di dalam string kita bisa menggunakan fungsi search(). contohnya:

```
2
3 <p id="contoh"></p>
4
5 <script>
6   var kalimat = "belajar javascript di www.malasngoding.com";
7   var cari = kalimat.search("javascript");
8   document.getElementById("contoh").innerHTML = cari;
9 </script>
```



Filter string dengan slice()

ada 2 parameter yang harus di isi pada fungsi slice(). yang pertama isikan dari urutan ke berapa filter di mulai. dan parameter kedua sampai urutan ke berapa batas filternya.

```
1 slice(mulai, batas akhir)
```

contonya

```
2
3 <p id="contoh"></p>
4
5 <script>
6   var kalimat = "belajar javascript di www.malasngoding.com";
7   var filter = kalimat.slice(22, 42);
8   document.getElementById("contoh").innerHTML = filter;
9 </script>
```

dan hasilnya adalah "www.malasngoding.com". karena kita memerintahkan untuk menampilkan karakter ke 22 sampai 42. seperti yang dapat kita lihat pada contoh di atas.

```
1 kalimat.slice(22, 42)
```



penggunaan **substring()** dan **substr()** juga sama. bisa teman-teman coba sendiri. **Manipulasi String Pada Javascript**

Mengganti string dengan replace()

cara penggunaan fungsi replace() pada javascript untuk mengganti string adalah sebagai berikut.

```
1 replace('yang ingin di ganti','ganti nya')
```

jadi pada parameter pertama, kita masukkan misalnya kata yang ingin kita ganti. dan pada parameter kedua nya kita masuk kan kata yang menjadi penggantinya.

```
1 <h2>www.malasngoding.com</h2>
2
3 <p id="contoh"></p>
4
5 <script>
6   var kalimat = "belajar javascript di www.malasngoding.com";
7   var ganti = kalimat.replace('javascript', 'pemrograman');
8   document.getElementById("contoh").innerHTML = ganti;
9 </script>
```

Maka yang sebelumnya isi variabel kalimat adalah "belajar javascript di www.malasngoding.com", maka berubah menjadi "belajar pemrograman di www.malasngoding.com".

karena di sini kita mengubah kata 'javascript' menjadi 'pemrograman' dengan menggunakan fungsi replace di javascript.



The screenshot shows a live code editor interface. On the left, the HTML and JavaScript code is displayed. The HTML code includes an h2 tag with the text 'www.malasngoding.com' and a p tag with id 'contoh'. The JavaScript code defines a variable 'kalimat' with the value 'belajar javascript di www.malasngoding.com', then uses the 'replace' method to replace 'javascript' with 'pemrograman', and finally updates the innerHTML of the 'contoh' element with the result. On the right, the 'Result' tab shows the rendered output: 'www.malasngoding.com' in a large font, followed by 'belajar pemrograman di www.malasngoding.com' in a smaller font. The editor also features a 'Resources' tab at the bottom left and zoom controls (1x, 0.5x, 0.25x) and a 'Run' button at the bottom right.

Huruf besar kecil dengan toUpperCase() dan toLowerCase()

Jika kita ingin mengubah isi string dengan huruf besar semua atau huruf kecil semua, kita bisa mengandalkan kedua fungsi ini, yaitu **toUpperCase()** untuk mengubah string menjadi huruf besar atau huruf kapital, dan **toLowerCase()** berguna untuk mengubah string menjadi huruf kecil semua. Manipulasi String Pada Javascript

```
1 <h2>www.malasngoding.com</h2>
2
3 <p id="besar"></p>
4 <p id="kecil"></p>
5
6 <script>
7   var kalimat = "Belajar JAVASCRIPT di www.malasngoding.com";
8
9   // mengubah ke huruf besar
10  var a = kalimat.toUpperCase();
11
12  // mengubah ke huruf kecil
13  var b = kalimat.toLowerCase();
14
15  document.getElementById("besar").innerHTML = a;
16  document.getElementById("kecil").innerHTML = b;
17 </script>
```



EDIT ON CODEPEN

HTML Result

LIVE

```
<h2>www.malasngoding.com</h2>
<p id="besar"></p>
<p id="kecil"></p>
<script>
  var kalimat = "Belajar JAVASCRIPT di www.malasngoding.com";
  // mengubah ke huruf besar
  var a = kalimat.toUpperCase();
  // mengubah ke huruf kecil
  var b = kalimat.toLowerCase();
  document.getElementById("besar").innerHTML = a;
  document.getElementById("kecil").innerHTML = b;
</script>
```

www.malasngoding.com

BELAJAR JAVASCRIPT DI
WWW.MALASNGODING.COM

belajar javascript di www.malasngoding.com

Resources 1x 0.5x 0.25x Rerun

Menggabungkan string dengan concat()

Untuk contoh kita akan mencoba menggabungkan 2 string. sebagai pengganti "+" sebagai tanda penggabung string.

```
1 <h2>www.malasngoding.com</h2>
2
3 <p id="contoh"></p>
4
5 <script>
6     var a = "diki alfarabi";
7     var b = "hadi";
8
9     // menggabungkan string
10    var gabung = a.concat(" ",b);
11    document.getElementById("contoh").innerHTML = gabung;
12 </script>
```

pada contoh penggunaan concat() javascript ini, kita membuat sebuah variabel a yang berisi "diki alfarabi". dan variabel b berisi "hadi". sebenarnya saat di tampilkan bisa saja dengan cara a+" "+b. tapi dengan javascript bisa lebih mudah, yaitu menggunakan fungsi concat. yang hasilnya akan menjadi "diki alfarabi hadi".



The screenshot shows a live code editor interface. On the left, the HTML and JavaScript code is displayed in a dark theme. On the right, the rendered output is shown in a light theme. The output displays the text "www.malasngoding.com" in a large font, followed by "diki alfarabi hadi" in a smaller font. The editor includes a "LIVE" indicator and a "Run" button at the bottom right.

```
HTML
```

```
<h2>www.malasngoding.com</h2>
<p id="contoh"></p>
<script>
  var a = "diki alfarabi";
  var b = "hadi";
  // menggabungkan string
  var gabung = a.concat(" ",b);
  document.getElementById("contoh").innerHTML = gabung;
</script>
```

```
Result
```

```
www.malasngoding.com
diki alfarabi hadi
```

EDIT ON CODEPEN

Resources 1x 0.5x 0.25x Run

Convert string ke array dengan split()

di rekomendasikan kepada teman-teman yang belum mempelajari tentang array untuk membaca tutorial www.malasngoding.com sebelumnya tentang tutorial array javascript. agar lebih mudah memahami fungsi split ini. karena di sini kita akan sedikit bermain dengan array pada javascript.

```
2
3 <p id="contoh"></p>
4
5 <script>
6   var huruf = "a,b,c";
7
8   // string jadi array
9   var convert = huruf.split(",");
10  document.getElementById("contoh").innerHTML = convert[0];
11 </script>
```

Pada contoh ini, kita membuat sebuah variabel bernama huruf. dan berisi 'a,b,c'. kemudian kita pecahkan string ini menjadi array dengan split(). dan menetapkan tanda ',' sebagai pemisah.

```
1 var convert = huruf.split(",");
```

nah, maka jadilah array yang tersimpan dalam variabel convert. dan menampilkan array urutan pertama

```
1 document.getElementById("contoh").innerHTML = convert[0];
```

urutan pertama pada array (0) adalah huruf a. maka hasilnya :



```
HTML Result EDIT ON CODEPEN
LIVE
<h2>www.malasngoding.com</h2>
<p id="contoh"></p>
<script>
  var huruf = "a,b,c";

  // string jadi array
  var convert = huruf.split(",");
Resources 1x 0.5x 0.25x Rerun
```