



Gambar 1. Sebuah Game Online yang ditulis dengan Action Script

Setelah kita mengenal VARIABLE dan bagaimana cara membuat variable dengan menggunakan Action Script, berikutnya kita akan belajar mengenai FUNCTION. Seperti halnya variable, function sebenarnya sudah sering kita buat meskipun belum mengetahui secara detail dengan maksud tulisan di dalam function tersebut. Sebagai contoh, ketika membuat tombol, maka kita akan menuliskan sebuah function yang akan dipanggil ketika tombol tersebut ditekan.

Nah, kali ini kita akan membahas secara lebih mendalam tentang function yang telah sering kita buat sebelumnya.

## APAKAH FUNCTION?

Function dan Variable adalah dua istilah yang sangat penting di dalam pemrograman. Keberadaan Function dan Variable di dalam Bahasa pemrograman apapun selalu dibutuhkan.

Jika kita mengulang kembali pelajaran matematika, maka kita bisa melihat kedua istilah tersebut selalu dipergunakan pada pelajaran matematika. Contoh  $x = 5$  ;  $y = 7$  ; di mana  $x$  dan  $y$  keduanya adalah **variable** dengan nilai yang telah ditentukan. Contoh lain  $y = 2x$  ; di mana  $y$  adalah sebuah **fungsi** dari  $x$ , sehingga ketika kita memasukkan sebuah nilai pada  $x$  maka akan dihasilkan  $y$ . Jika kita memasukkan nilai yang berbeda pada  $x$  maka akan dihasilkan  $y$  dengan nilai yang berbeda pula.

Di dalam pemrograman, sebuah **function** dapat berperan seperti halnya fungsi di dalam matematika. Akan tetapi, function di dalam pemrograman tidak terlalu tergantung pada input dan output. Sebuah function biasanya merupakan sebuah deretan perintah yang kemudian dikemas (*encapsulated*) dengan nama tertentu yang kemudian dapat dijalankan hanya dengan memanggil nama function tersebut.

Contohnya, misalnya kita membuat sebuah function untuk menggambar bintang yang mungkin akan terdiri dari 10 baris. Untuk itu, kita bisa memasukkan 10 baris tersebut ke dalam

sebuah function, sehingga untuk menjalankan perintah membuat gambar bintang kita bisa memanggil dengan nama function tersebut. Hal ini tentu akan jauh lebih mudah daripada kita menyalin 10 baris tersebut di setiap tempat.

Function adalah hal yang sangat mendasar di dalam bahasa pemrograman manapun. Dalam pemrograman berbasis objek, function dinamakan dengan method. Function dan method pada dasarnya adalah hal yang sama. Demikian halnya dengan Bahasa pemrograman lainnya, kita mengenal “subroutine”, “subprogram”, atau “procedure” yang pada prinsipnya adalah konsep yang sama.

## MEMBUAT FUNCTION

Sekarang, kita akan mencoba membuat sebuah function dengan menggunakan Action Script 3.0 seperti yang telah sering kita lakukan untuk membuat perintah pada button.

1. Buat dokumen baru dengan menggunakan Flash (Action Script 3.0).
2. Klik pada frame pertama (biasanya secara default sudah berada di frame 1).
3. Buka Action panel dengan **F9** atau menu: **Window > Action**
4. Klik pada ikon Pin untuk mengunci panel Action untuk memudahkan kembali lagi ke panel Action ini.
5. Ketik beberapa baris berikut:

```
function namafunction() {
    trace("Sedang mengerjakan sesuatu");
}
```

Sekarang, kita akan bahas maksud dari penulisan function di atas:

Keyword **function** adalah kata yang sudah ditandai dalam Action Script 3.0 seperti halnya kata **var** yang tujuannya adalah untuk deklarasi sebuah function.

Sesudah function kemudian diikuti dengan **namafunction()** dengan kurung buka dan kurung tutup. Di dalam kurung akan diisi dengan sebuah nilai yang akan dieksekusi ketika function dijalankan.

Sesudah namafunction() kemudian diikuti dengan kurung kurawal buka yang di dalamnya diisi dengan satu baris perintah dan ditutup dengan kurung kurawal tutup. Dengan demikian, kurung kurawal ini untuk mengawal baris-baris perintah tersebut sebagai bagian dari function. Jadi, setelah kurung kurawal tutup sudah tidak menjadi bagian dari function.

Pada contoh ini diisi dengan satu baris perintah yakni **trace();** yang akan menghasilkan pesan di dalam panel output sesuai dengan nilai yang ada di dalam tanda kurung buka dan kurung tutup perintah trace tersebut.

```

function doSomething() {
    trace("Doing something.");
}

```

The diagram labels the components of the function syntax: **keyword** points to `function`, **name** points to `doSomething`, **parentheses** points to `()`, **opening brace** points to `{`, **body** points to the line `trace("Doing something.");`, and **closing brace** points to `}`.

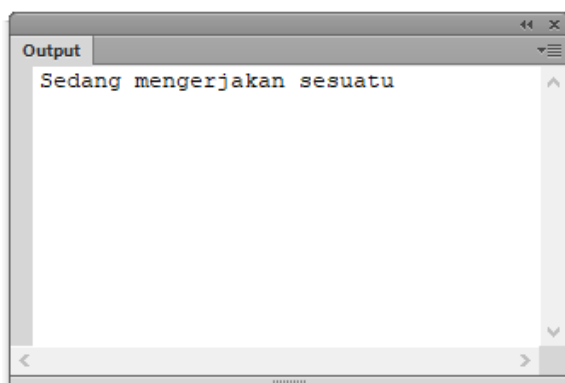
Perlu dicatat bahwa baris-baris perintah yang ada di dalam function tidak akan dieksekusi, kecuali jika function tersebut dipanggil dalam satu baris perintah. Jadi, pernyataan di atas hanyalah untuk mendeklarasikan sebuah function dan tidak akan dijalankan karena tidak ada baris perintah di luar function yang berfungsi untuk menjalankan function tersebut.

## MEMANGGIL FUNCTION

Untuk memanggil sebuah function, caranya adalah dengan menyebutkan nama function tersebut dalam satu baris perintah. Contoh:

```
namafunction();
```

Jika movie Flash tersebut ditest dengan Ctrl + ENTER, maka akan ditampilkan tulisan sebagai berikut di dalam panel output:

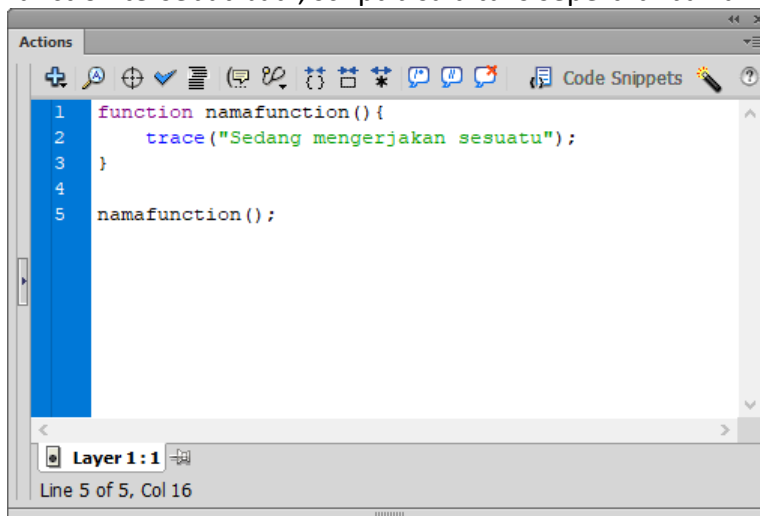


Seperti halnya nama variable, nama function juga bersifat case-sensitive. Artinya, “namafunction” dengan huruf kecil akan berbeda dengan “NAMAFUNCTION” dengan huruf besar atau “NamaFunction” dengan kombinasi huruf besar dan kecil.

Sebagai contoh, cobalah ganti nama function tersebut dengan huruf besar seluruhnya atau sebagian, kemudian test movie dan lihat hasilnya tidak akan berjalan seperti sebelumnya. Tidak ada pesan yang ditulis di dalam panel output.

Nama function harus selalu ditulis dengan menggunakan kurung buka dan kurung tutup di belakang nama function, meskipun tidak ada tulisan apapun di dalam tanda kurung tersebut. Dengan demikian penulisan function berbeda dengan variable yang tidak menggunakan kurung buka dan kurung tutup.

Memanggil function dapat dilakukan di manapun, sebelum atau setelah kita mendeklarasikan function tersebut. Jadi, script bisa ditulis seperti di bawah ini:

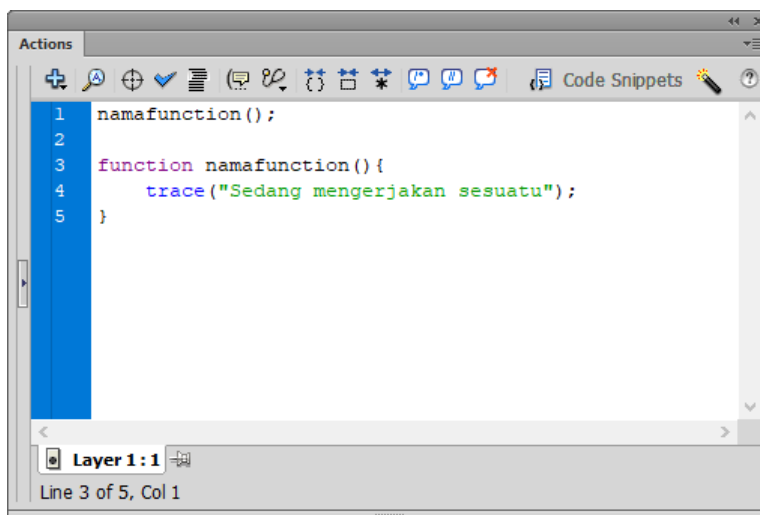


```

1 function namafunction(){
2     trace("Sedang mengerjakan sesuatu");
3 }
4
5 namafunction();
  
```

Layer 1: 1  
Line 5 of 5, Col 16

Atau seperti di bawah ini:



```

1 namafunction();
2
3 function namafunction(){
4     trace("Sedang mengerjakan sesuatu");
5 }
  
```

Layer 1: 1  
Line 3 of 5, Col 1

## PARAMETER PADA FUNCTION

Sebuah function bisa saja dijalankan tanpa menggunakan parameter seperti pada contoh di atas. Hal ini tidak akan menjadi masalah ketika hanya pekerjaan yang sama yang harus dilakukan oleh function tersebut. Contohnya misalnya ketika kita menginginkan sebuah function untuk menampilkan dan menjalankan video tertentu di layar ketika function tersebut dipanggil.

Akan berbeda keadaannya ketika kita menginginkan function untuk membuka file video, tetapi function tersebut kemudian meminta kita untuk menunjukkan file video mana yang akan dijalanannya. Dalam hal ini kita harus menggunakan parameter yang membuat function menjadi lebih fleksibel.

Parameter pada function dapat ditempatkan di dalam tanda kurung di belakang nama function. Contohnya sebagai berikut:

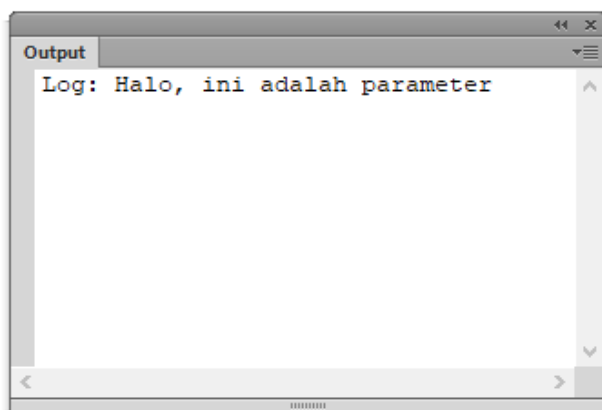
```
function log(pesan) {
    trace("Log: " + pesan);
}
```

Nah, sekarang kita bisa melihat bagaimana sebuah parameter disisipkan ke dalam sebuah function. Parameter di dalam function mirip dengan variable yang akan menampung sebuah nilai. Function akan mengambil nilai dari parameter tersebut dan kemudian menggunakannya di dalam lingkungannya sendiri (di antara kurung kurawal buka dan kurung kurawal tutup).

Dengan demikian, jika kita panggil function tersebut dengan menambahkan parameter seperti berikut ini:

```
log("Halo, ini adalah parameter");
```

Jika movie Flash tersebut dites dengan Ctrl + ENTER, maka akan ditampilkan tulisan sebagai berikut di dalam panel output:

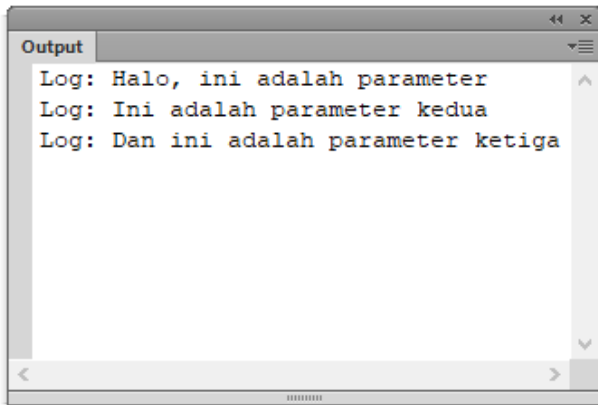


Jadi, parameter akan dibaca oleh function dan kemudian disertakan di dalam perintah trace karena ada parameter "pesan" di dalam perintah trace. Untuk lebih memperjelas, cobalah membuat beberapa baris perintah untuk memanggil function, namun dengan parameter yang berbeda. Contoh:

```
function log(pesan) {
    trace("Log: " + pesan);
}

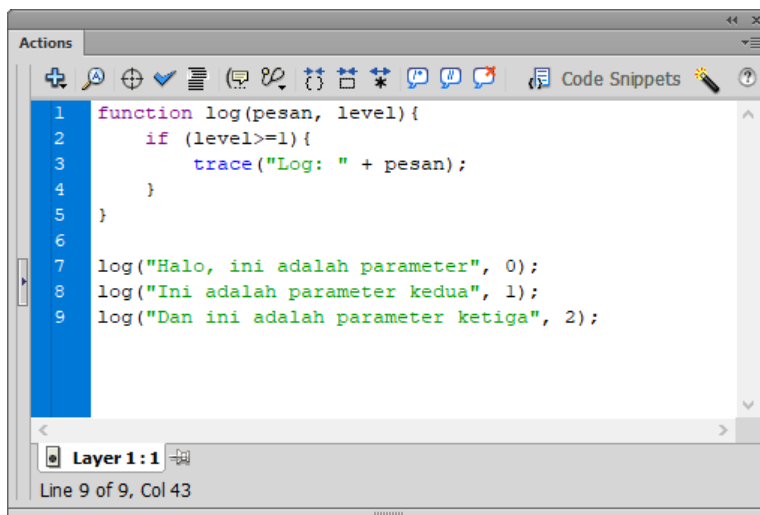
log("Halo, ini adalah parameter");
log("Ini adalah parameter kedua");
log("Dan ini adalah parameter ketiga");
```

Jika movie Flash tersebut dites dengan Ctrl + ENTER, maka akan ditampilkan tulisan sebagai berikut di dalam panel output:

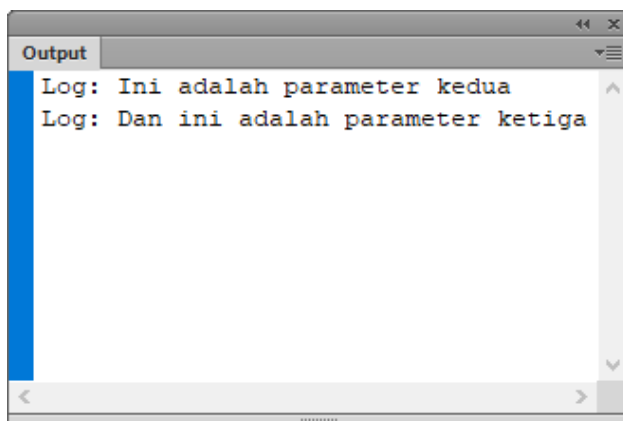


## MENAMBAHKAN PARAMETER LAIN

Jika dibutuhkan beberapa parameter pada sebuah function, maka kita dapat menderetkan parameter-parameter tersebut dengan pemisah tanda koma. Contoh:



Jika movie Flash tersebut di test dengan Ctrl + ENTER, maka akan ditampilkan tulisan sebagai berikut di dalam panel output:



## TIPE DATA PADA PARAMETER

Seperti halnya variable, parameter pada function juga dapat menggunakan tipe data. Itulah sebabnya seringkali kita menuliskan sebuah function untuk tombol yang di-klik dengan penulisan sebagai berikut:

```
function klikhome(e:Event) {
    baris perintah;
}
```

Dalam contoh ini, Event adalah tipe data dari parameter “e”.

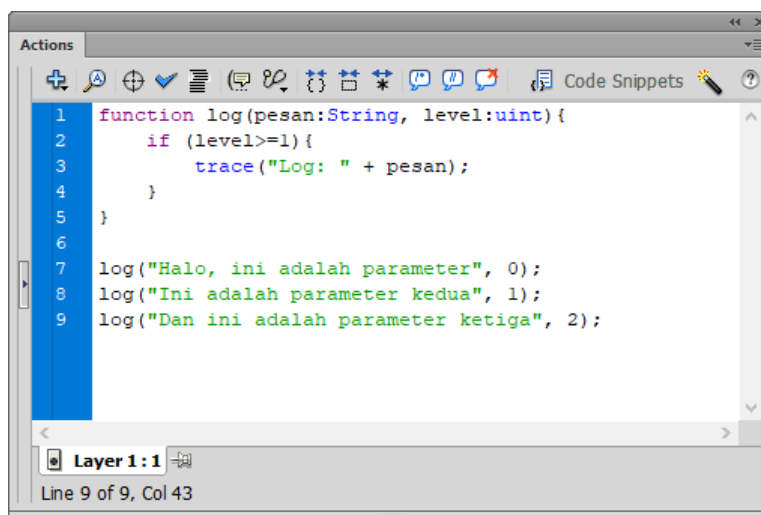
Dengan demikian, jika perintah di dalam function sebelumnya akan kita lengkapi dengan tipe data, maka function tersebut dapat ditulis sebagai berikut:

```
function log(pesan:String) {
    trace("Log: " + pesan);
}

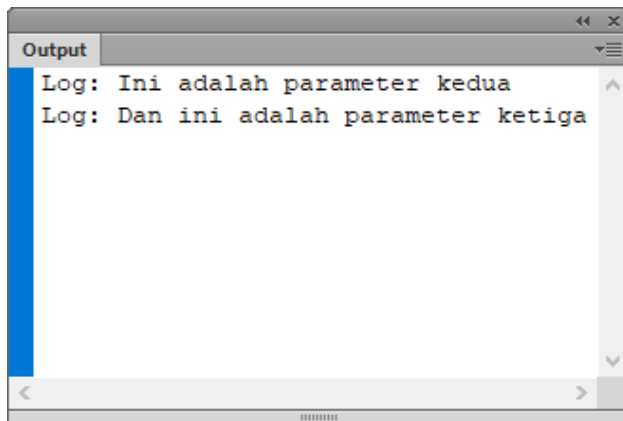
log("Halo, ini adalah parameter");
```

Jadi, **String** adalah tipe data dari parameter “pesan”. Demikian halnya **Event** adalah tipe data dari parameter “e”. Lalu, apa saja tipe data yang ada di dalam Action Script 3.0? Berikut ini adalah beberapa tipe data dalam Action Script 3.0: Array, Boolean, int, MovieClip, Null, Number, Shape, String, uint, void, Object, dan lain-lain.

Contoh lain:



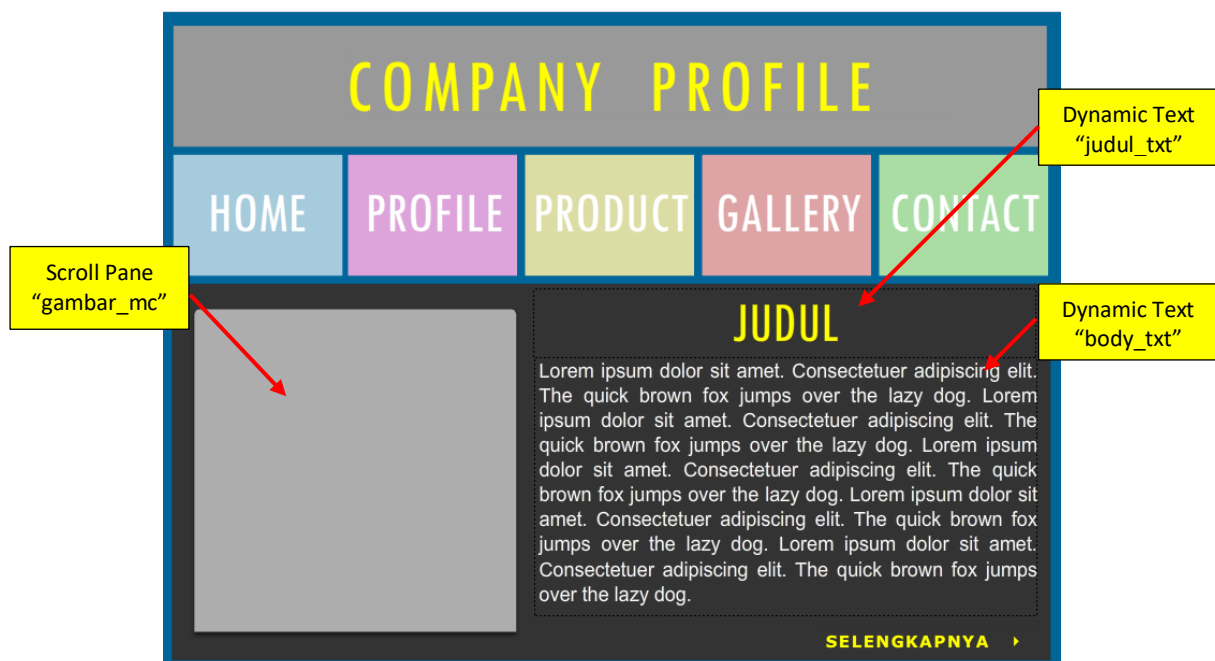
Jika movie Flash tersebut di-test dengan Ctrl + ENTER, maka akan ditampilkan tulisan sebagai berikut di dalam panel output:



## MEMBUAT HALAMAN INTERAKTIF

Berikut ini adalah contoh penerapan dari function dan parameter yang ditempatkan ke dalam bentuk halaman interaktif. Contoh ini dimaksudkan untuk menjelaskan bagaimana function dan parameter dimanfaatkan untuk menampilkan tulisan tertentu sesuai dengan tombol yang di-klik.

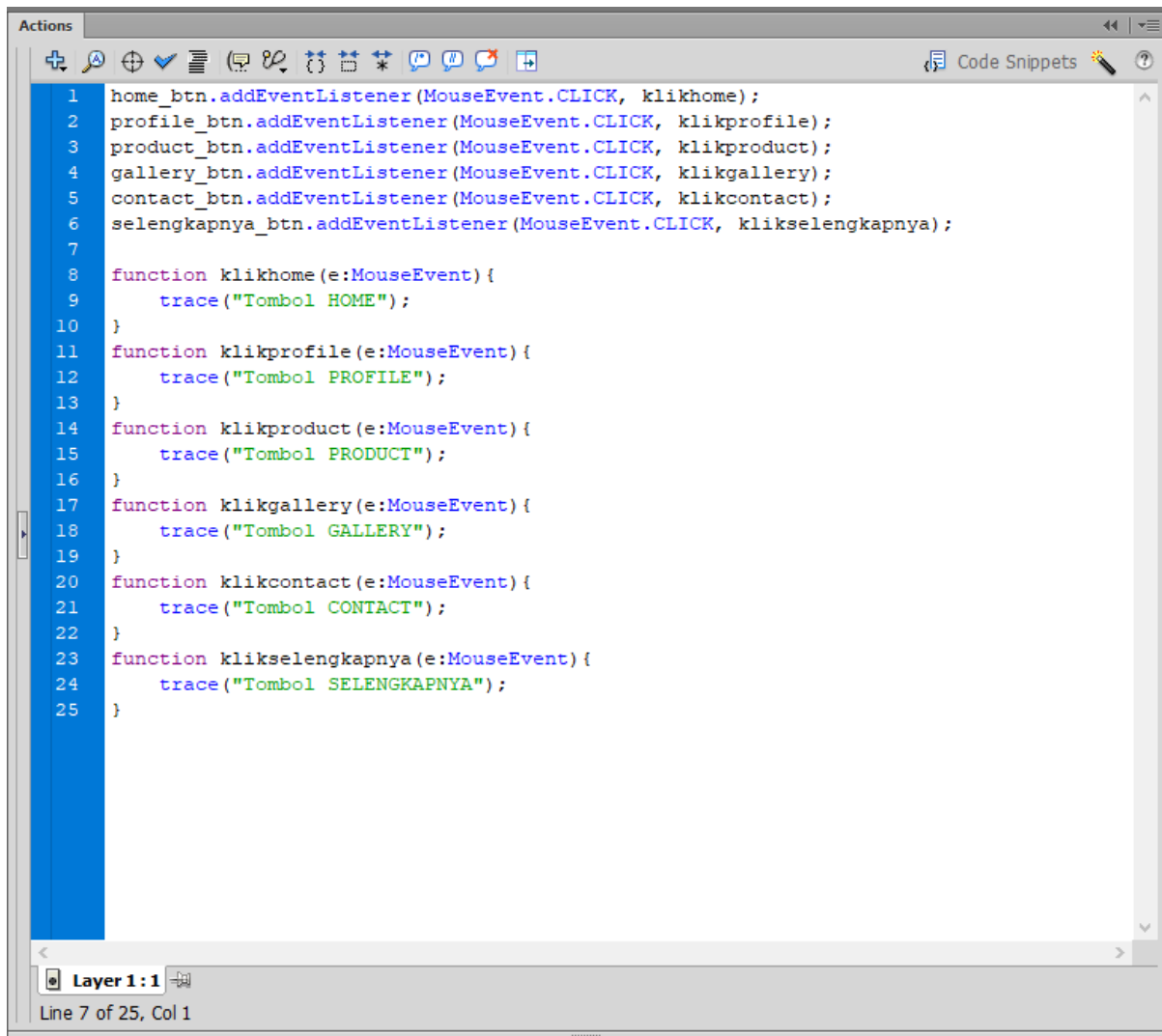
Antarmuka berikut ini dibuat dengan menggunakan Adobe Flash dan Action Script 3.0. Untuk memperlihatkan bagaimana function bekerja, kita dapat menggunakan Dynamic Text pada bagian JUDUL dan KONTEN, sedangkan untuk gambar menggunakan Scroll Pane. Masing-masing diberi instance name: judul\_txt, konten\_txt, dan gambar\_mc:



Dalam contoh ini terdapat 6 tombol dengan instance name sebagai berikut: home\_btn, profile\_btn, product\_btn, gallery\_btn, contact\_btn, dan selengkapnya\_btn.

Selanjutnya, kita akan buat Action Script untuk menguji masing-masing tombol sebagai berikut:

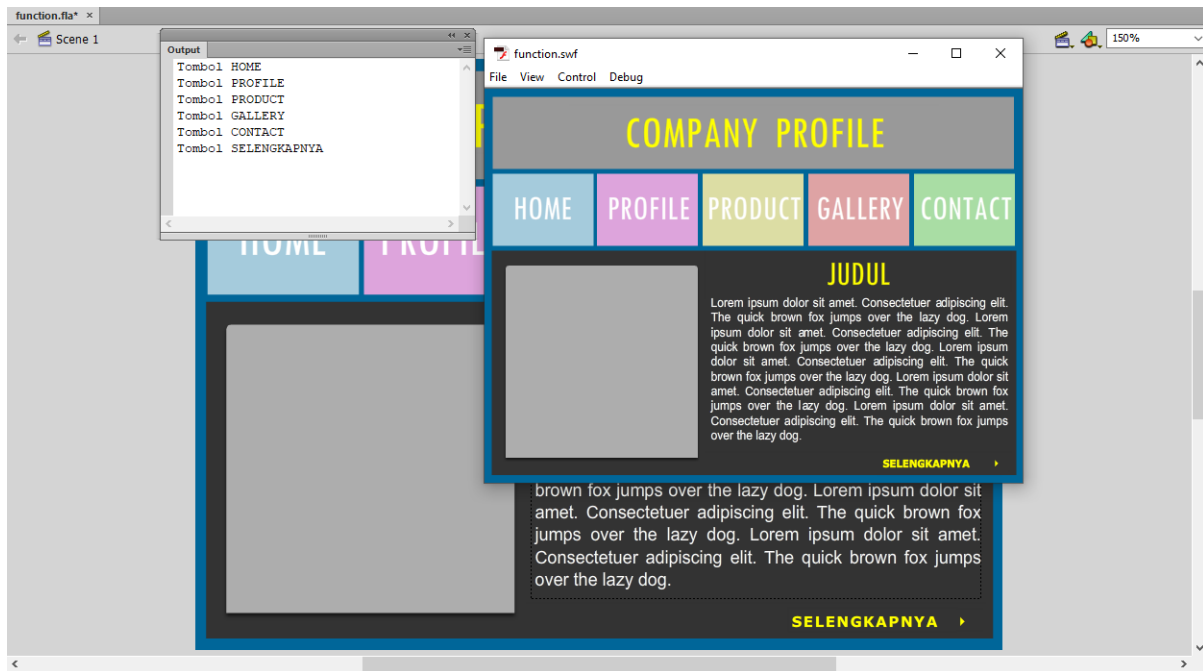




```
1 home_btn.addEventListener(MouseEvent.CLICK, klikhome);
2 profile_btn.addEventListener(MouseEvent.CLICK, klikprofile);
3 product_btn.addEventListener(MouseEvent.CLICK, klikproduct);
4 gallery_btn.addEventListener(MouseEvent.CLICK, klikgallery);
5 contact_btn.addEventListener(MouseEvent.CLICK, klikcontact);
6 selengkapnya_btn.addEventListener(MouseEvent.CLICK, klikselengkapnya);
7
8 function klikhome(e:MouseEvent){
9     trace("Tombol HOME");
10 }
11 function klikprofile(e:MouseEvent){
12     trace("Tombol PROFILE");
13 }
14 function klikproduct(e:MouseEvent){
15     trace("Tombol PRODUCT");
16 }
17 function klikgallery(e:MouseEvent){
18     trace("Tombol GALLERY");
19 }
20 function klikcontact(e:MouseEvent){
21     trace("Tombol CONTACT");
22 }
23 function klikselengkapnya(e:MouseEvent){
24     trace("Tombol SELENGKAPNYA");
25 }
```

Layer 1:1  
Line 7 of 25, Col 1

Test Movie, kemudian klik masing-masing tombol sehingga kita bisa melihat log pesan yang ditampilkan pada panel Output sebagai berikut:



Setelah semua tombol lengkap dan dapat bekerja dengan baik, sekarang kita akan membuat sebuah function yang akan kita jadikan sebagai function yang akan diakses oleh semua tombol. Scriptnya adalah sebagai berikut:

```
function setContent(judul:String, konten:String, imageUrl:String, link:String) {
    trace("judul: " + judul);
    trace("konten: " + konten);
    trace("image: " + imageUrl);
    trace("link: " + link);
}
```

Dengan adanya function setContent() tersebut, maka kita dapat membuat perintah pada function di setiap tombol yang masing-masing diisi dengan parameter untuk setiap tombol. Scriptnya adalah sebagai berikut:

```
var linkUrl:String;

function klikhome(e:MouseEvent) {
    trace("Tombol HOME");
    setContent("Judul Satu", "Ini tulisan untuk konten satu", "gambar1.jpg", "http://www.yai.ac.id/home");
}

function klikprofile(e:MouseEvent) {
    trace("Tombol PROFILE");
    setContent("Judul Dua", "Ini tulisan untuk konten dua", "gambar2.jpg", "http://www.yai.ac.id/profile");
}

function klikproduct(e:MouseEvent) {
    trace("Tombol PRODUCT");
    setContent("Judul Tiga", "Ini tulisan untuk konten tiga", "gambar3.jpg", "http://www.yai.ac.id/product");
}
```

```

}

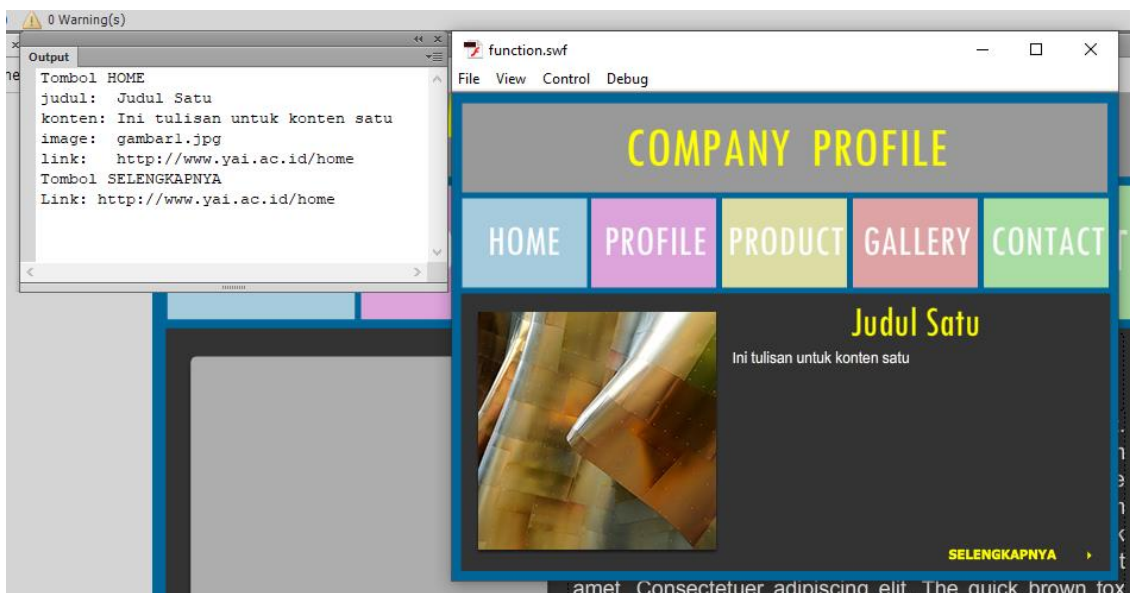
function klikgallery(e:MouseEvent) {
    trace("Tombol GALLERY");
    setContent("Judul Empat", "Ini tulisan untuk konten empat",
"gambar4.jpg", "http://www.yai.ac.id/gallery");
}

function klikcontact(e:MouseEvent) {
    trace("Tombol CONTACT");
    setContent("Judul Lima", "Ini tulisan untuk konten lima",
"gambar5.jpg", "http://www.yai.ac.id/contact");
}

function klikselengkapnya(e:MouseEvent) {
    trace("Tombol SELENGKAPNYA");
    trace("Link: " + linkUrl);
}

```

Test Movie Flash dengan Ctrl + ENTER untuk melihat hasil dari Action Script yang telah lengkap tersebut. Cobalah untuk meng-klik salah satu tombol, misalnya tombol HOME, kemudian tombol SELENGKAPNYA:



Nah, dengan contoh ini mudah-mudahan kita bisa memahami bagaimana membuat function dan bagaimana function tersebut bekerja.