



Gambar 1. Zuma buatan PopCap adalah game populer yang ditulis dengan Action Script 3.0

Setelah kita mengenal VARIABLE dan FUNCTION, maka sekarang kita akan belajar tentang PERCABANGAN. Istilah PERCABANGAN dikenal luas pada dunia pemrograman sebagai salah satu cara yang sangat berhubungan dengan LOGIKA PEMROGRAMAN. Dengan adanya PERCABANGAN, maka dimungkinkan bagi program untuk berjalan sesuai dengan pilihan-pilihan yang dilakukan oleh pengguna program.

Pernahkah Anda membaca buku ceritera bercabang? Buku ini adalah buku ceritera yang bersifat interaktif. Anda bisa membaca dan menentukan sendiri jalan ceriteranya dengan memilih salah satu dari beberapa pilihan yang ditawarkan. Dengan demikian, maka Anda seolah-olah turut menentukan jalan ceriteranya.

Mungkin Anda belum pernah membaca buku sejenis ini. Baiklah, secara singkat dapat dijelaskan seperti pada contoh berikut: Pada suatu ketika, Anda sedang dikejar oleh seekor harimau. Anda berhasil menyelinap pada sebuah lorong lalu berlari terus menyusuri lorong tersebut hingga ke ujung lorong. Sementara itu, harimau di belakang membuntuti Anda dan telah berhasil masuk ke lorong. Di ujung lorong terdapat dua pintu, satu di sebelah kiri lorong dan satu lagi di sebelah kanan lorong. Pintu mana yang akan Anda buka? Jika masuk ke pintu sebelah kiri, lanjutkan ke halaman 11. Jika masuk ke pintu sebelah kanan, lanjutkan ke halaman 25.

APAKAH PERCABANGAN?

Percabangan berhubungan langsung dengan LOGIKA PEMROGRAMAN. Berbeda dengan Variabel dan Function, Percabangan akan mengatur alur program yang ditentukan oleh pilihan-pilihan yang ada.

Contoh umum PERCABANGAN yang biasa kita temukan adalah pada saat kita berada pada halaman login. Kita mengisi form *username* dan *password* seperti yang diminta. Kemudian, setelah kita menekan tombol “*submit*”, maka program akan bekerja dengan melihat apakah *username* dan *password* yang kita masukkan sesuai dengan *username* dan *password* yang ada di database. Jika *username* dan *password* sesuai dengan salah satu data yang ada di database, maka Anda akan masuk ke halaman berikutnya. Jika tidak sesuai, maka Anda akan ditolak untuk melanjutkan ke halaman berikutnya.

Perintah yang dipergunakan pada percabangan dapat juga kita temukan pada bahasa pemrograman lainnya, yakni menggunakan perintah IF ... ELSE dan SWITCH ...CASE. Keduanya memiliki konsep yang hampir serupa, akan tetapi dengan detail yang berbeda.

PERCABANGAN DENGAN IF

Baiklah, sebetulnya percabangan dengan menggunakan keyword IF sudah kita pelajari sebelumnya. Contohnya pada penjelasan tentang FUNCTION terdapat Action Script yang menggunakan keyword IF.

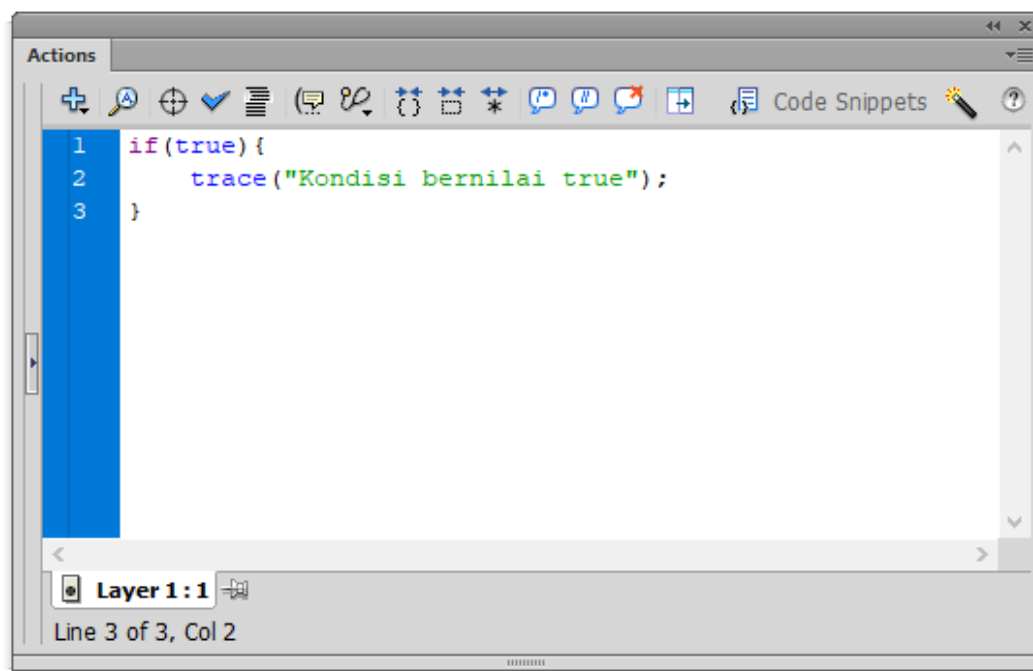
Keyword IF adalah salah satu keyword yang populer di kalangan programmer karena penggunaannya yang cukup mudah. Konstruksi IF biasa dipergunakan untuk mengatur alur program yang akan menyebabkan terdapat dua pilihan cabang. Adapun konstruksi IF tersebut dapat dijelaskan sebagai berikut:

```
if(true) {  
    Statement;  
}
```

Keyword IF diikuti dengan sebuah KONDISI yang ditempatkan di dalam tanda kurung. Jika KONDISI tersebut terpenuhi (bernilai TRUE), maka STATEMENT yang ditempatkan di dalam tanda kurung kurawal akan dieksekusi.

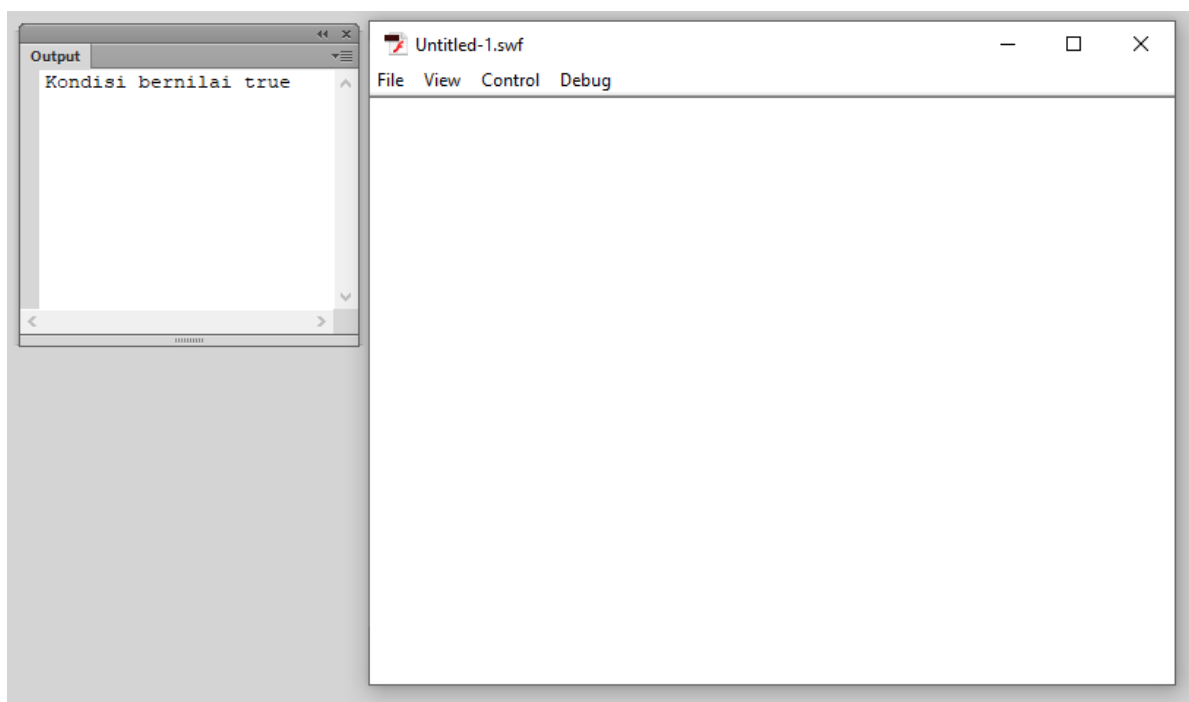
Adapun yang dimaksud dengan KONDISI dalam konstruksi IF adalah sebuah pernyataan LOGIKA yang akan bernilai TRUE jika kondisinya terpenuhi dan bernilai FALSE jika kondisinya tidak terpenuhi.

Sekarang, kita akan mencoba membuat sebuah aplikasi sederhana dengan Action Script 3.0. Kemudian kita ketikkan Action Script sederhana berikut ini:

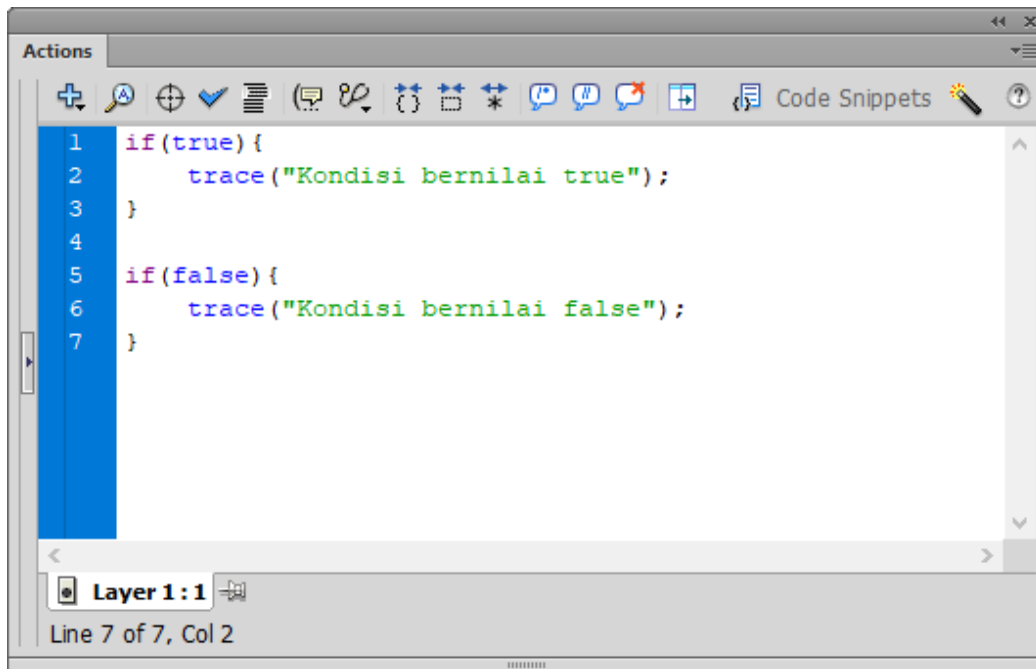


Jelas tertulis KONDISI pada pernyataan IF diisi dengan nilai TRUE, sehingga perintah TRACE akan dieksekusi dan akan muncul tulisan “Kondisi bernilai true” pada jendela Output.

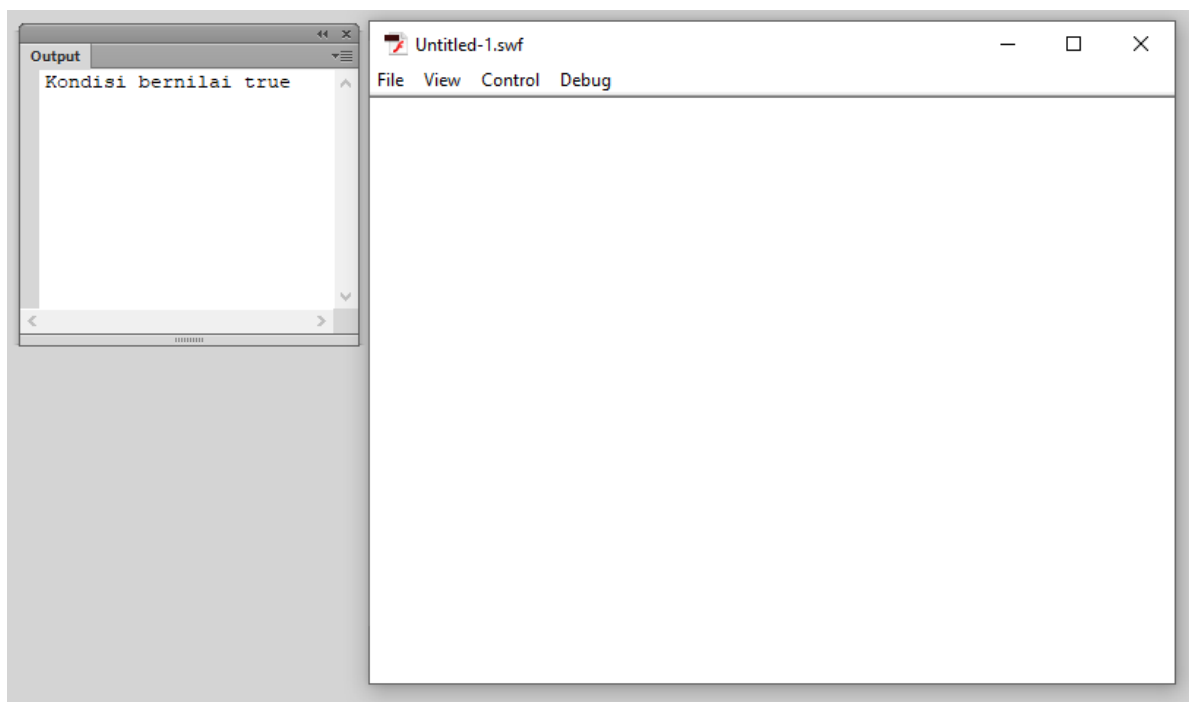
Jika movie Flash tersebut dijalankan dengan perintah **Test Movie** atau **Ctrl + ENTER**, maka akan ditampilkan hasilnya sebagai berikut:



Sekarang, kita akan mencoba memasukkan nilai FALSE pada kondisi di belakang IF dan melihat bagaimana hasilnya. Caranya adalah dengan menambahkan satu lagi keyword IF (bisa sebelum IF yang pertama, bisa juga setelah IF yang pertama), sebagai berikut:



Jika movie Flash tersebut dijalankan dengan perintah **Test Movie** atau **Ctrl + ENTER**, maka akan ditampilkan hasilnya sebagai berikut:

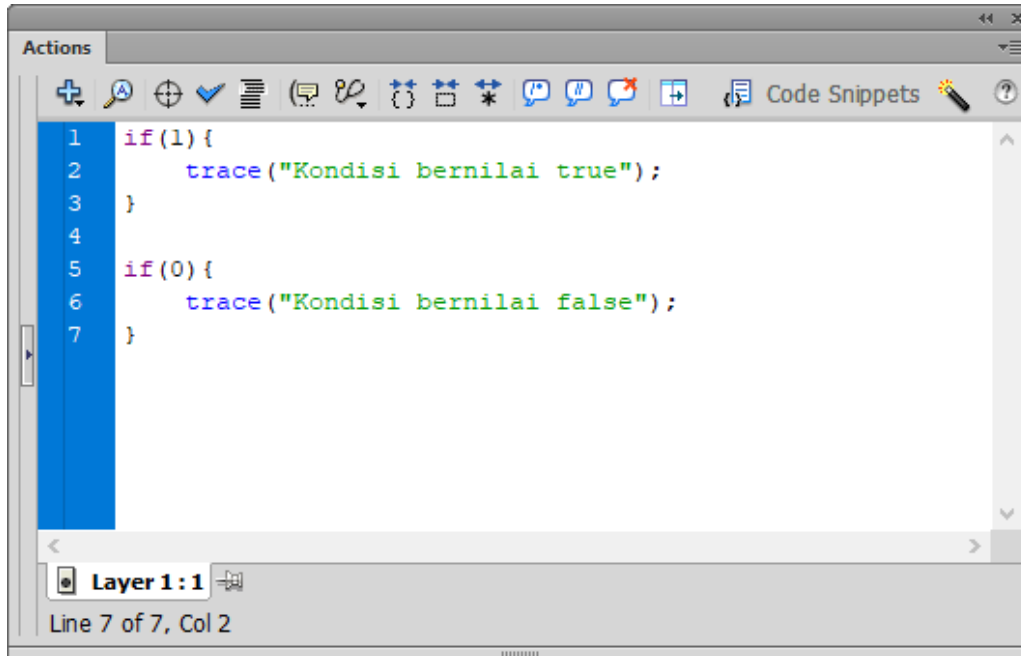


Tidak ada perbedaan hasil antara script yang pertama dengan script yang kedua. Dengan kata lain, Statement yang ada di belakang KONDISI yang bernilai FALSE tidak akan dieksekusi. Jadi, perintah TRACE dengan tulisan "Kondisi bernilai false" tidak akan pernah ditampilkan atau dijalankan.

OPERATOR LOGIKA

KONDISI pada pernyataan IF hanya berlaku jika bernilai TRUE atau FALSE. Kedua nilai tersebut disebut dengan BOOLEAN. Selain menggunakan nilai TRUE atau FALSE, tipe data Boolean juga dapat menggunakan angka 0 dan 1. Nilai 0 mewakili FALSE dan nilai 1 mewakili TRUE.

Dengan demikian, contoh di atas dapat ditulis sebagai berikut:



```

1  if(1){
2      trace("Kondisi bernilai true");
3  }
4
5  if(0){
6      trace("Kondisi bernilai false");
7  }
  
```

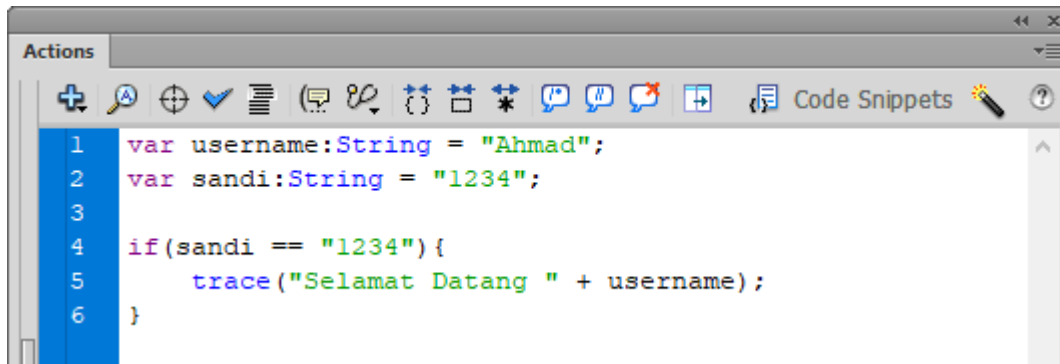
The screenshot shows a code editor window titled 'Actions' with a toolbar at the top. The code is written in a light blue font on a white background. The first if statement has a condition of 1 and a trace statement 'Kondisi bernilai true'. The second if statement has a condition of 0 and a trace statement 'Kondisi bernilai false'. The status bar at the bottom indicates 'Layer 1: 1' and 'Line 7 of 7, Col 2'.

Cara lain adalah dengan menggunakan Operator Logika, yakni tanda persamaan dan pertidaksamaan yang biasa digunakan pada matematika. Berikut ini adalah Operator Logika yang digunakan dalam pemrograman:

Operator	Nama	Bernilai TRUE jika	Contoh
==	Persamaan	Nilai kiri sama dengan kanan	25 == 25
!=	Pertidaksamaan	Nilai kiri tidak sama dengan kanan	5 != 47
<	Kurang dari	Nilai kiri lebih kecil dari kanan	10 < 15
<=	Kurang dari atau sama dengan	Nilai kiri lebih kecil atau sama dengan kanan	12 <= 17 17 <= 17
>	Lebih dari	Nilai kiri lebih besar dari kanan	30 > 21
>=	Lebih dari atau sama dengan	Nilai kiri lebih besar atau sama dengan kanan	50 >= 27 50 >= 50
&&	Logical AND	Nilai kiri bernilai TRUE dan nilai kanan bernilai TRUE	1 == 1 && 2 == 2
	Logical OR	Salah satu atau kedua sisi bernilai TRUE	1 == 1 3 == 4
!	Logical NOT	Kebalikan dari nilai kanan	!(5 == 7)

Persamaan

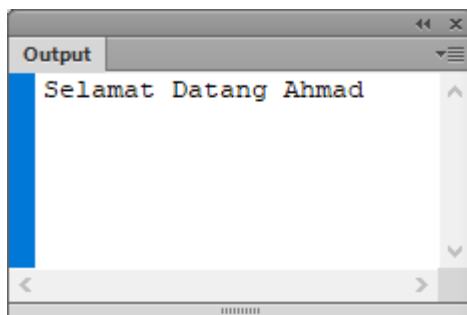
Jika nilai pada sisi kiri **SAMA DENGAN** nilai pada sisi kanan, maka KONDISI akan bernilai TRUE sehingga statement di dalam tanda kurung kurawal akan dieksekusi. Berikut ini adalah contoh Action Script yang menggunakan persamaan sebagai KONDISI pada statement IF:



```

1  var username:String = "Ahmad";
2  var sandi:String = "1234";
3
4  if(sandi == "1234"){
5      trace("Selamat Datang " + username);
6  }
  
```

Jika movie Flash dijalankan, maka akan ditampilkan hasilnya sebagai berikut:



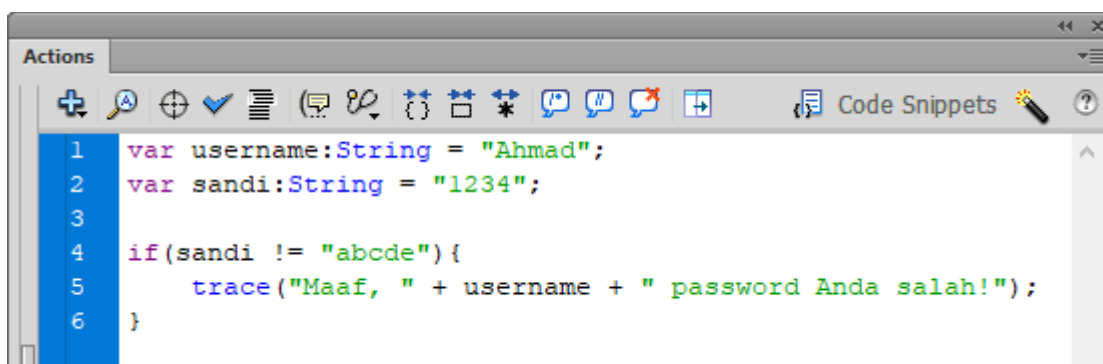
```

Output
Selamat Datang Ahmad
  
```

Cobalah untuk mengganti sisi kanan dari KONDISI pada statement IF di atas, maka ketika movie Flash dijalankan kita tidak akan melihat apapun pada panel Output.

Pertidaksamaan

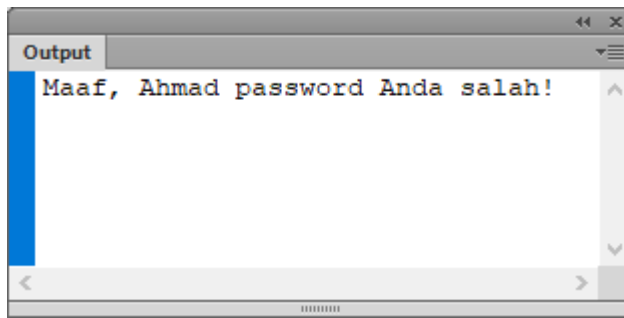
Jika nilai pada sisi kiri **TIDAK SAMA DENGAN** nilai pada sisi kanan, maka KONDISI akan bernilai TRUE sehingga statement di dalam tanda kurung kurawal akan dieksekusi. Berikut ini adalah contoh Action Script yang menggunakan persamaan sebagai KONDISI pada statement IF:



```

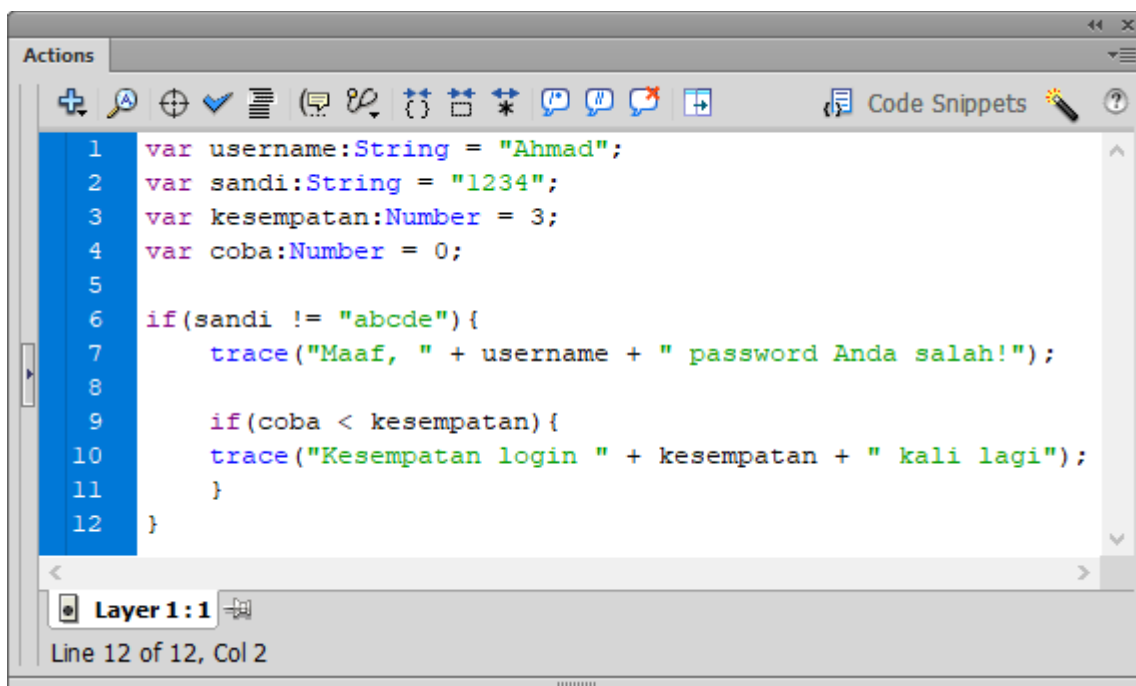
1  var username:String = "Ahmad";
2  var sandi:String = "1234";
3
4  if(sandi != "abcde"){
5      trace("Maaf, " + username + " password Anda salah!");
6  }
  
```

Jika movie Flash di atas dijalankan, maka akan ditampilkan hasilnya sebagai berikut:

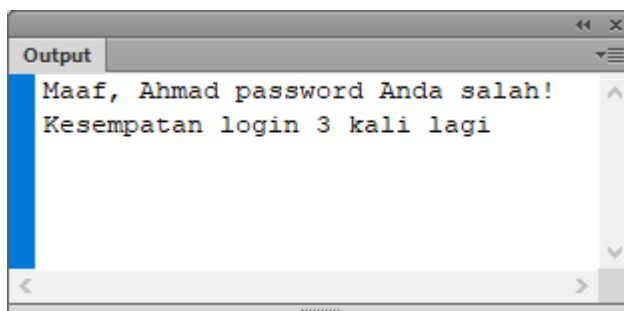


Kurang Dari

Jika nilai pada sisi kiri **LEBIH KECIL** dari nilai pada sisi kanan, maka KONDISI akan bernilai TRUE sehingga statement di dalam tanda kurung kurawal akan dieksekusi. Berikut ini adalah contoh Action Script yang menggunakan Kurang Dari sebagai KONDISI pada statement IF:

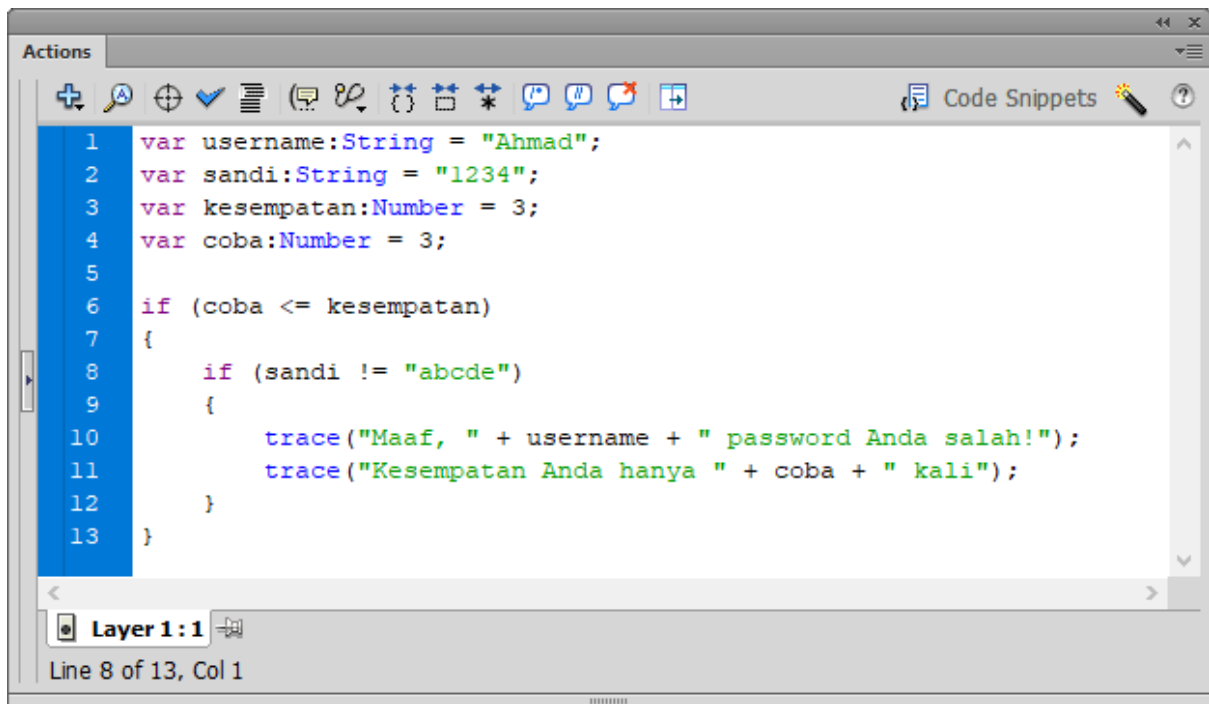


Jika movie Flash di atas dijalankan, maka akan ditampilkan hasilnya sebagai berikut:



Kurang Dari Atau Sama Dengan

Jika nilai pada sisi kiri **LEBIH KECIL** atau **SAMA DENGAN** nilai pada sisi kanan, maka KONDISI akan bernilai TRUE sehingga statement di dalam tanda kurung kurawal akan dieksekusi. Berikut ini adalah contoh Action Script yang menggunakan Kurang Dari sebagai KONDISI pada statement IF:



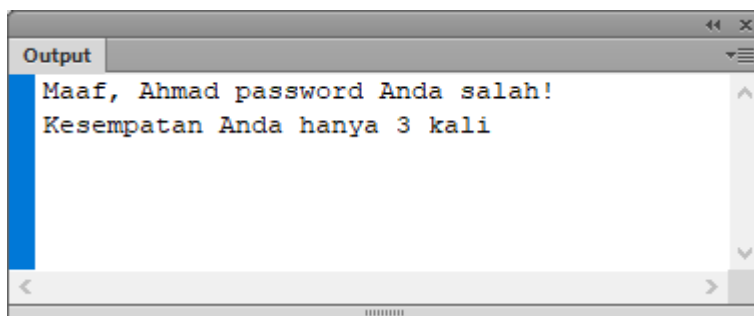
```

1  var username:String = "Ahmad";
2  var sandi:String = "1234";
3  var kesempatan:Number = 3;
4  var coba:Number = 3;
5
6  if (coba <= kesempatan)
7  {
8      if (sand != "abcde")
9      {
10         trace("Maaf, " + username + " password Anda salah!");
11         trace("Kesempatan Anda hanya " + coba + " kali");
12     }
13 }

```

Layer 1:1
Line 8 of 13, Col 1

Jika movie Flash di atas dijalankan, maka akan ditampilkan hasilnya sebagai berikut:



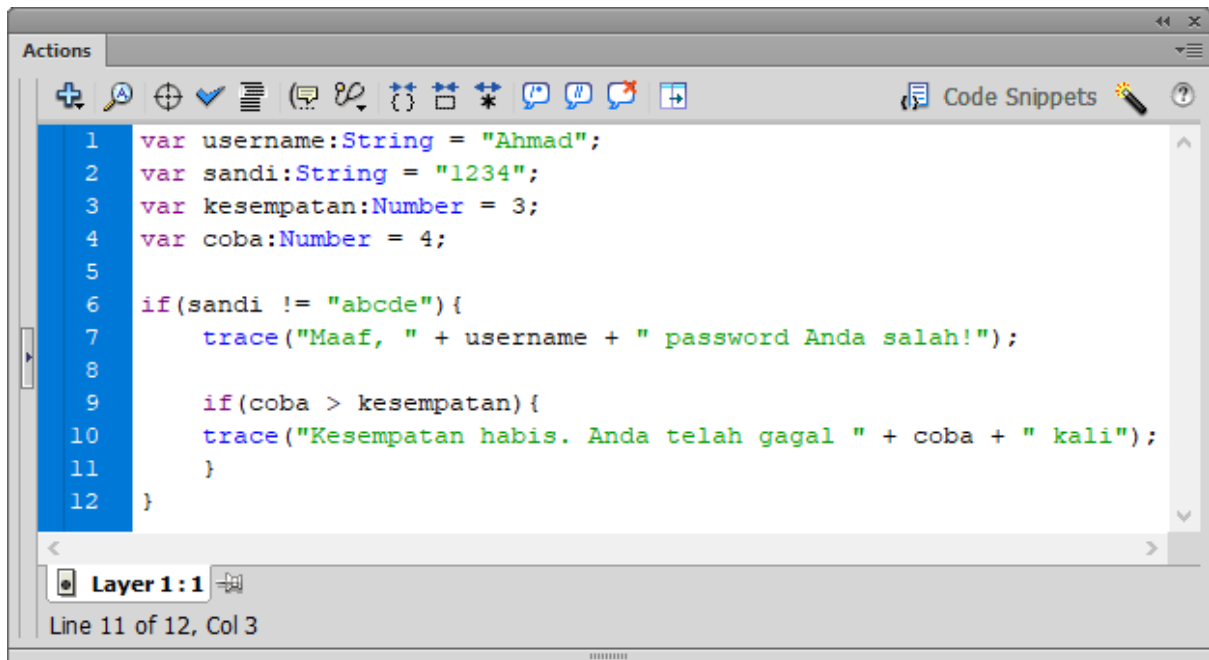
```

Output
Maaf, Ahmad password Anda salah!
Kesempatan Anda hanya 3 kali

```

Lebih Dari

Jika nilai pada sisi kiri **LEBIH BESAR** dari nilai pada sisi kanan, maka KONDISI akan bernilai TRUE sehingga statement di dalam tanda kurung kurawal akan dieksekusi. Berikut ini adalah contoh Action Script yang menggunakan Lebih Dari sebagai KONDISI pada statement IF:



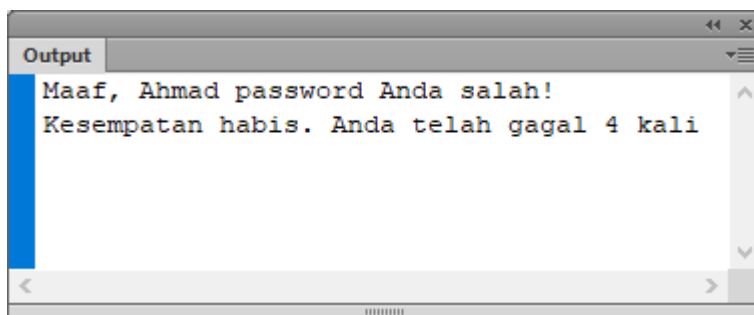
```

1  var username:String = "Ahmad";
2  var sandi:String = "1234";
3  var kesempatan:Number = 3;
4  var coba:Number = 4;
5
6  if(sandi != "abcde"){
7      trace("Maaf, " + username + " password Anda salah!");
8
9      if(coba > kesempatan){
10         trace("Kesempatan habis. Anda telah gagal " + coba + " kali");
11     }
12 }

```

Layer 1:1
Line 11 of 12, Col 3

Jika movie Flash di atas dijalankan, maka akan ditampilkan hasilnya sebagai berikut:



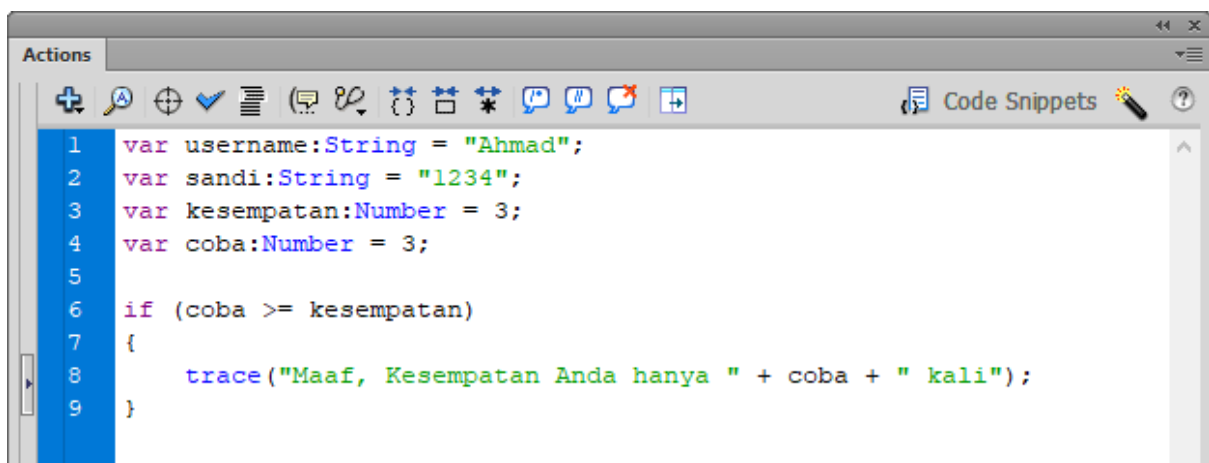
```

Output
Maaf, Ahmad password Anda salah!
Kesempatan habis. Anda telah gagal 4 kali

```

Lebih Dari Atau Sama Dengan

Jika nilai pada sisi kiri **LEBIH BESAR** atau **SAMA DENGAN** nilai pada sisi kanan, maka KONDISI akan bernilai TRUE sehingga statement di dalam tanda kurung kurawal akan dieksekusi. Berikut ini adalah contoh Action Script yang menggunakan Lebih Dari sebagai KONDISI pada statement IF:

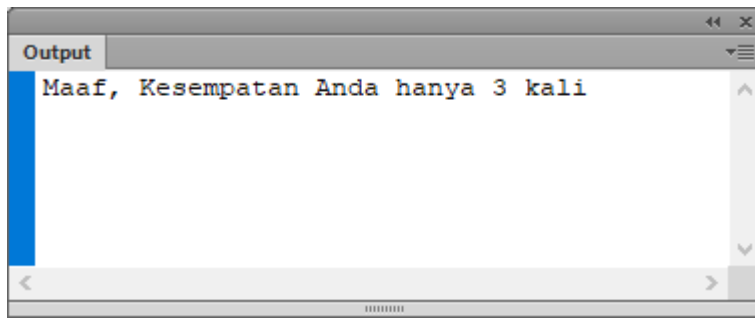


```

1  var username:String = "Ahmad";
2  var sandi:String = "1234";
3  var kesempatan:Number = 3;
4  var coba:Number = 3;
5
6  if (coba >= kesempatan)
7  {
8      trace("Maaf, Kesempatan Anda hanya " + coba + " kali");
9  }

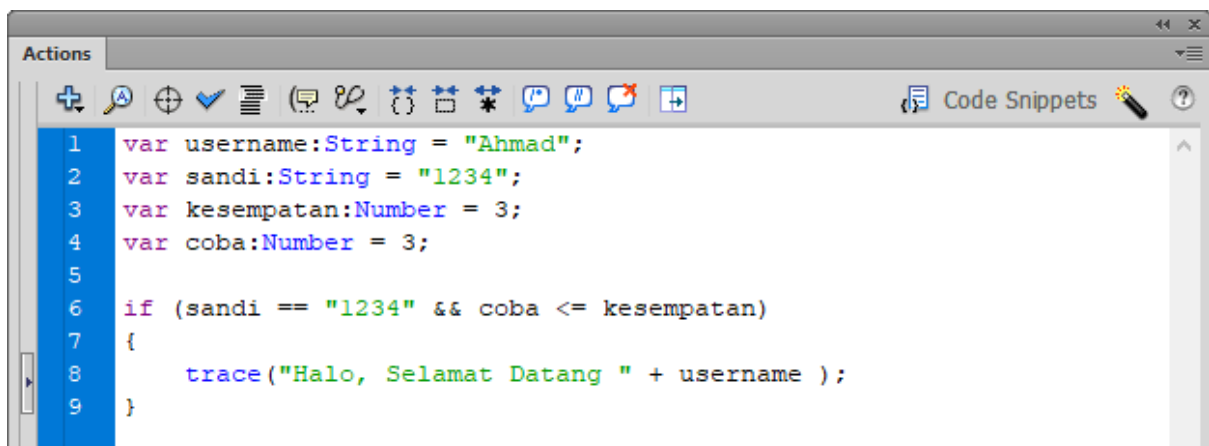
```

Jika movie Flash di atas dijalankan, maka akan ditampilkan hasilnya sebagai berikut:



Logical AND (&&)

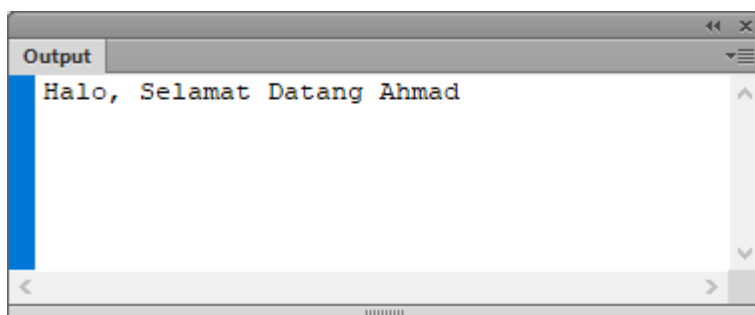
Jika nilai pertama bernilai TRUE **DAN** nilai kedua bernilai TRUE (dan nilai berikutnya bernilai TRUE), maka KONDISI akan bernilai TRUE sehingga statement di dalam tanda kurung kurawal akan dieksekusi. Berikut ini adalah contoh Action Script yang menggunakan Logical AND sebagai KONDISI pada statement IF:



Perhatikan bahwa KONDISI pada IF di atas diisi dengan dua KONDISI, yakni:

sandi == "1234" dan coba <= kesempatan

Statement IF di atas akan bernilai TRUE karena kedua kondisi tersebut bernilai TRUE. Dengan demikian, jika movie Flash di atas dijalankan, maka akan ditampilkan hasilnya sebagai berikut:



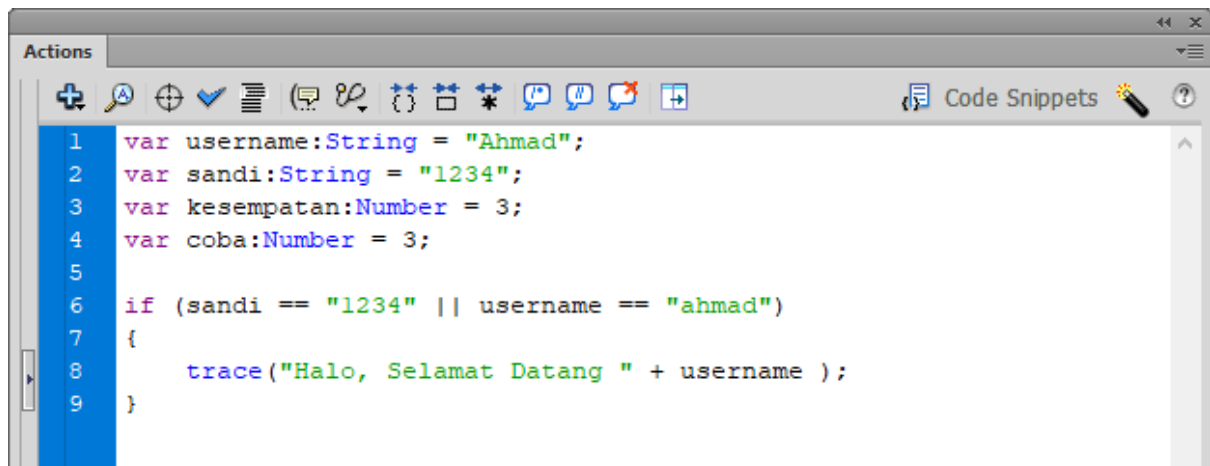
Logical AND atau **&&** tidak akan bernilai **TRUE** jika salah satu atau kedua kondisi tersebut bernilai **FALSE**. Contohnya jika kita ubah kondisi menjadi

sandi == "abcde" atau coba < kesempatan

maka perintah trace tidak akan dijalankan. Demikian halnya jika keduanya bernilai FALSE maka perintah trace tersebut juga tidak akan dijalankan.

Logical OR (||)

Jika nilai pertama bernilai TRUE **ATAU** nilai kedua bernilai TRUE (**ATAU** nilai berikutnya bernilai TRUE), maka KONDISI akan bernilai TRUE sehingga statement di dalam tanda kurung kurawal akan dieksekusi. Berikut ini adalah contoh Action Script yang menggunakan Logical OR sebagai KONDISI pada statement IF:



```

1  var username:String = "Ahmad";
2  var sandi:String = "1234";
3  var kesempatan:Number = 3;
4  var coba:Number = 3;
5
6  if (sandi == "1234" || username == "ahmad")
7  {
8      trace("Halo, Selamat Datang " + username );
9  }
  
```

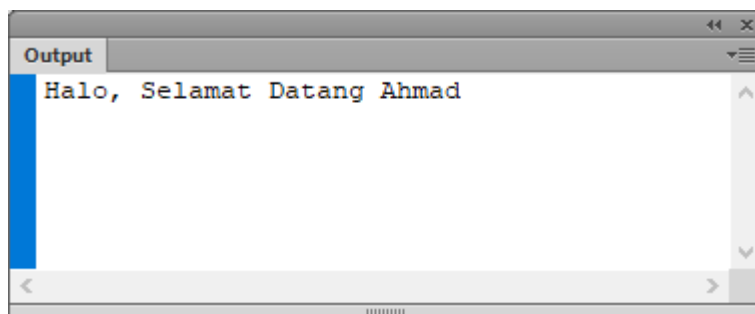
Pada contoh di atas terlihat bahwa KONDISI pertama:

sandi == "1234"

bernilai TRUE karena sisi kiri sama dengan sisi kanan, sementara KONDISI yang kedua:

username == "ahmad"

bernilai FALSE karena sisi kiri tidak sama dengan sisi kanan (huruf kecil tidak sama dengan huruf capital. Sehingga jika movie Flash di atas dijalankan, maka akan ditampilkan hasilnya sebagai berikut:

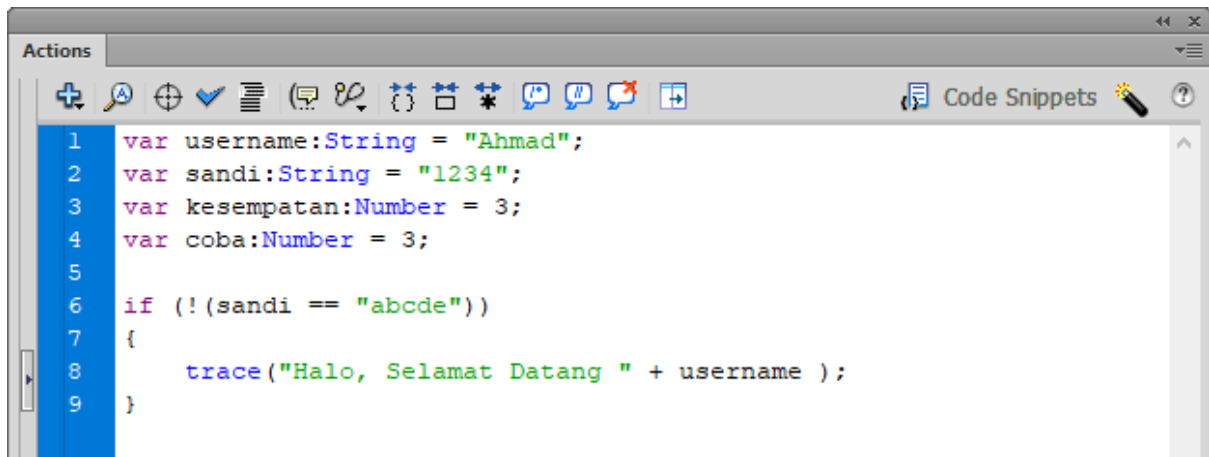


```

Output
Halo, Selamat Datang Ahmad
  
```

Logical NOT (!)

Logical NOT akan menghasilkan kebalikan dari nilai KONDISI. Jadi, jika KONDISI bernilai TRUE maka hasilnya akan menjadi FALSE. Sebaliknya, jika KONDISI bernilai FALSE, maka hasilnya akan bernilai TRUE. Berikut ini adalah contoh Action Script yang menggunakan Logical NOT sebagai KONDISI pada statement IF:



```

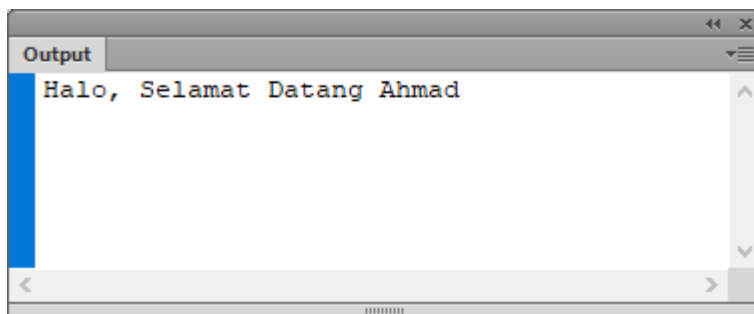
1  var username:String = "Ahmad";
2  var sandi:String = "1234";
3  var kesempatan:Number = 3;
4  var coba:Number = 3;
5
6  if (!(sandi == "abcde"))
7  {
8      trace("Halo, Selamat Datang " + username );
9  }

```

Pada contoh di atas, terlihat bahwa KONDISI:

sandi == "abcde"

diberi Logical NOT di bagian depan sehingga yang tadinya bernilai FALSE menjadi bernilai TRUE. Sehingga jika movie Flash di atas dijalankan, maka akan ditampilkan hasilnya sebagai berikut:



```

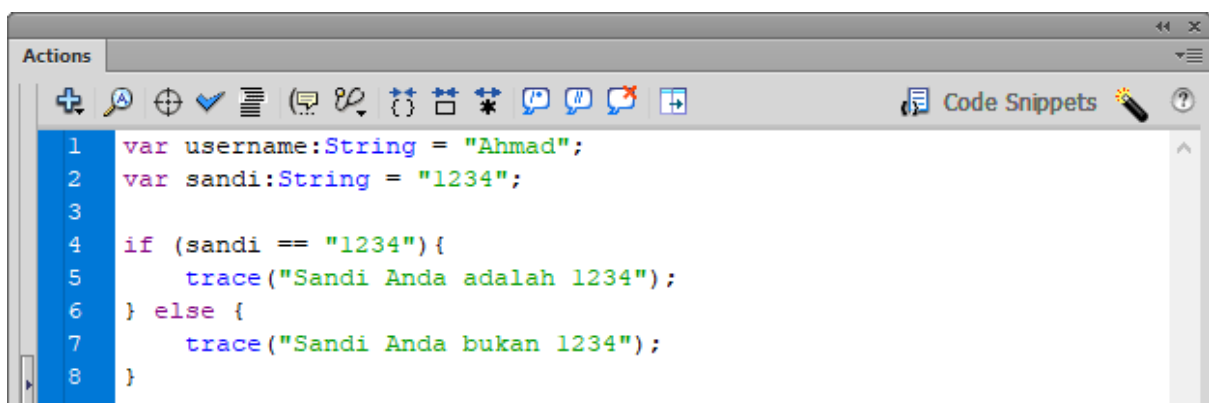
Output
Halo, Selamat Datang Ahmad

```

IF DAN ELSE

Selain IF, masih ada turunan dari keyword tersebut, yakni keyword ELSE. Berfungsi sebagai cabang lain dari IF ketika KONDISI di dalam IF tidak bisa dipenuhi. Dengan kata lain, ketika KONDISI pada IF bernilai TRUE, maka ELSE akan diabaikan. Sebaliknya, ketika KONDISI di dalam IF bernilai FALSE, maka ELSE akan dijalankan. Seperti halnya IF, satu set perintah di antara kurung kurawal buka dan kurung kurawal tutup di belakang ELSE akan dieksekusi.

IF dan ELSE dapat dijelaskan dengan script berikut:



```

1  var username:String = "Ahmad";
2  var sandi:String = "1234";
3
4  if (sandi == "1234"){
5      trace("Sandi Anda adalah 1234");
6  } else {
7      trace("Sandi Anda bukan 1234");
8  }

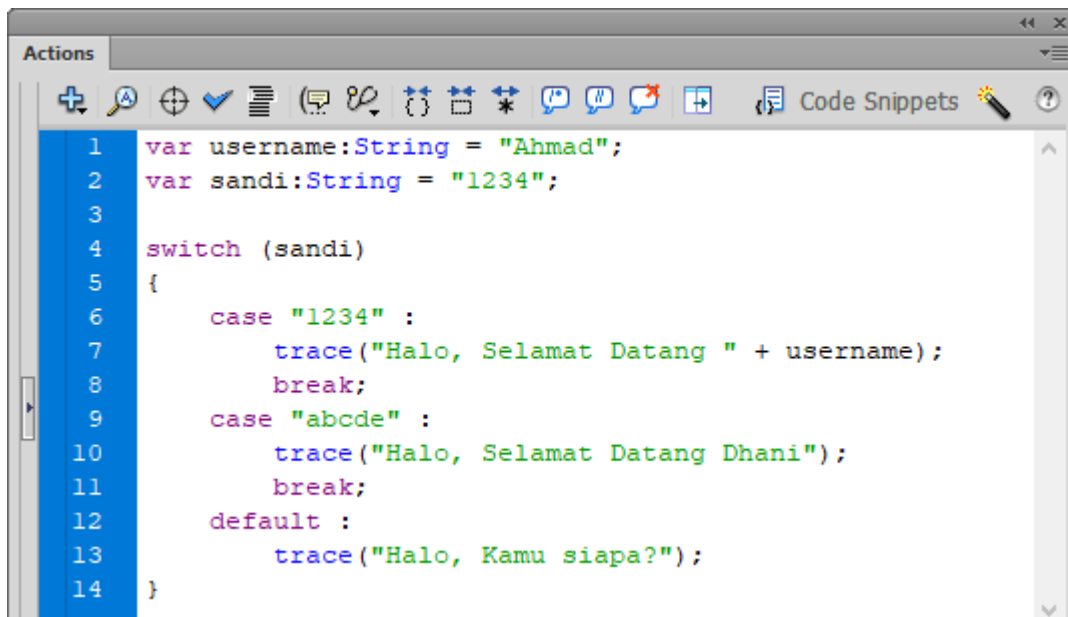
```

Pada script di atas, karena variable sandi bernilai “1234”, maka KONDISI pada IF akan bernilai TRUE. Akan tetapi jika misalnya sandi bernilai selain dari “1234”, misalnya “abcde”, maka KONDISI pada IF akan bernilai FALSE sehingga aliran program akan mengeksekusi ELSE. Satu set perintah di belakang ELSE akan dieksekusi.

SWITCH CASE

Selain menggunakan keyword IF dan ELSE, serta ELSE IF, percabangan juga dapat menggunakan keyword SWITCH CASE.

Perhatikan contoh blok program berikut ini:

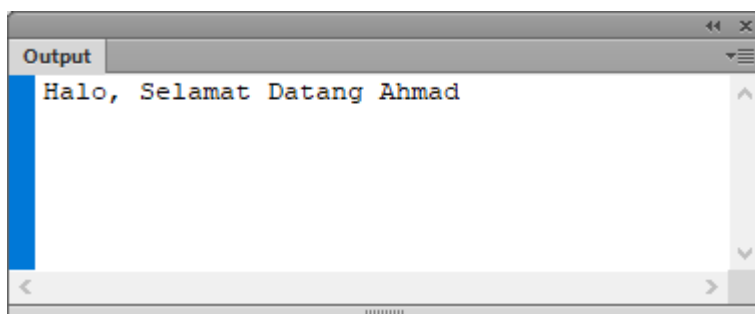


```

1  var username:String = "Ahmad";
2  var sandi:String = "1234";
3
4  switch (sandi)
5  {
6      case "1234" :
7          trace("Halo, Selamat Datang " + username);
8          break;
9      case "abcde" :
10         trace("Halo, Selamat Datang Dhani");
11         break;
12     default :
13         trace("Halo, Kamu siapa?");
14 }

```

Pada contoh di atas, variable **sandi** bernilai “1234” sehingga jika movie Flash ini dijalankan maka hasilnya adalah sebagai berikut:



```

Output
Halo, Selamat Datang Ahmad

```

Jika variable **sandi** bernilai “abcde” maka aliran program akan masuk ke case yang kedua, sehingga hasilnya akan ditampilkan tulisan “Halo, Selamat Datang Dhani”.

Terakhir, jika sandi tidak memiliki nilai “1234” atau “abcde”, maka aliran program akan masuk ke default sehingga hasilnya akan ditampilkan tulisan “Halo, Kamu siapa?”.

Demikian penjelasan mengenai percabangan dengan IF, ELSE, dan SWITCH CASE. Semoga bermanfaat.