**ACTION SCRIPT**

**Diajukan Sebagai**

**Pemenuhan Tugas 8**

**Matakuliah Dasar Animasi**



**Oleh:**

1. **Astuti Wijayanti 702010053**
2. **Nila Isti Khoeriyah 702010059**
3. **Kartikaning Endah Setyorini 702100061**
4. **Octa Dilla Putri 702010062**
5. **Diah Oktie Utami 702010065**

**Fakultas Teknologi Informasi**

**Universitas Kristen Satya Wacana**

**Salatiga**

**Juni 2013**

**ANIMASI OBJEK 2 DIMENSI**

**ACTION SCRIPT**

Actionscript terdiri dari 2 kata yaitu : action (aksi) dan script (tulisan/naskah) yang berarti tulisan/naskah yang beraksi ,Actionscript adalah bahasa pemrograman yang digunakan di Flash. Hingga saat ini sudah mencapai 3 versi.

* *Actionscript 1.0 (tahun 2000 – tahun 2003) mulai dipergunakan pada Flash 5 dengan minimal - dimainkan di Flash Player 5*
* *Actionscript 2.0 (tahun 2003 - tahun 2006) mulai dipergunakan pada Flash MX 2004 dengan minimal dimainkan di Flash Player 7*
* *Actionscript 3.0 (tahun 2006 sampai sekarang) mulai dipergunakan pada Flash CS3 dengan minimal dimainkan di Flash Player 9*

Di Flash actionscript ditulis pada panel actions. Penulisan actionscript di panel actions dapat dilakukan pada 3 tempat yaitu pada movie clip, button, dan frame. Ketiga buah tempat tadi memiliki aturan penulisan yang berbeda.seperti :

1. *Untuk menjalankan event pada button, script pada button penulisannya diawali dengan on(){}*
2. *Untuk menjalankan event pada movie clip, script pada movie clip penulisannya diawali dengan on(){} atau onClipEvent(){}*
3. *Untuk menjalankan event pada frame, script pada frame penulisannya diawali dengan onLoad=function(){} atau onEnterFrame=function(){} atau onPress=function(){} atau onRelease=function(){} dsb.*

Ketiga cara penulisan script tersebut memiliki kekurangan dan kelebihan. Kami lebih merekomendasikan menulis script pada frame karena apabila terdapat kesalahan/error pada script yang anda tulis anda dapat cepat memperbaikinya. Coba bandingkan apabila anda menulis script pada movie clip atau button, jika sampai muncul kesalahan/error pada script yang anda tulis, tentunya anda harus mengeceknya satu per satu mana yang error.

Untuk mengecek apakah terdapat kesalahan pada script yang anda tulis anda dapat menggunakan tombol Check syntax pada panel actions. Ketika tombol Check syntax ini ditekan, apabila script yang anda tulis terdapat kesalahan maka akan muncul kotak dialog beserta panel output yang di dalamnya berisi informasi kesalahan pada script anda. Apabila script yang anda tulis sudah benar maka akan muncul kotak dialog bertuliskan This script contains no errors.

Seperti dalam kehidupan nyata tulisan pada selembar kertas harus ditulis dengan rapi agar enak membacanya. Begitu juga penulisan script, apabila kurang rapi penulisannya tentunya anda akan tidak enak membacanya atau malah sampai bingung bagaimana membacanya. Untuk itu di Flash sudah disediakan fasilitas untuk merapikan script yang anda tulis. Caranya cukup mudah, apabila anda sudah yakin tidak ada error pada script yang anda tulis, silahkan anda menekan tombol Auto format pada panel actions. Jika anda menekan tombol tersebut maka script yang tadinya penulisannya kurang rapi (ada yang terlalu kekanan, ada yang terlalu banyak spasinya dsb) akan menjadi teratur dan rapi secara otomatis.

Penjelasan pola penulisan action script pada button adalah sebagai berikut :

1. Action pada button harus dimulaidengan script on()
2. Di dalam tanda kurung () pada scripon berisi mouse event seperti release, press,dragover, rollout dan sebagainya. Mouseevent menyatakan sebuah kejadian yangakan di eksekusi oleh action tersebut saatmouse melakukan kegiatan tertentu.Sebagai contoh : mouse event release berarti perintah dalam blok {} akan dijalankan ketika mouse menekan tombol.
3. Tanda { merupakan awal suatu blok script. Yang dimaksud dengan blok scrip adalah suatu kumpulan perintah yang akandijalankan sepanjang movie event.
4. Tanda // merupakan tanda yang menyatakan komentar .
Baris yangmemiliki tanda tersebut tidak dianggap sebagai suatu perintah. Meskipun demikian tanda //sangatmembantu memberi informasi pada penulisan action scrip kita.
5. Baris trace(“tombol ditekan”);merupakan baris perintah. Perintah dalam buttonharus ditulis didalam sebuah blok mouseevent. Setiap akhir penulisan sebuah perintah selalu diakhiri dengan tanda ;
6. Tanda } merupakan penutup suatu blok script.Pola Penulisan Action Script padaMovieclip

**Cara penulisan action pada movieclip adalah sebagai berikut:**

1. Buatlah sebuah obyek lingkaran denganmenggunakan oval tool. Perhatikan bahwa sebuah obyek dalam flash memiliki2 bagian yaitu stoke (garis tepi) dan fill(isi).Oval tool dan obyek dalam flash
2. Seleksi obyek lingkaran tersebutkemudian convert menjadi symbol denganmenekan tombol F8 (pilih menuinsert>convert to symbol). Pilih movieclip padaoption behaviour dan ketikan bola padaname.Convert to symbol
3. Klik movieclip bola, dan dalam kondisiterseleksi buka panel action ( tekan f9atau klik menu windows>action apabila panel action belum terbuka dan pastikanmode penulisan adalah mode expert jikamenggunakan flash MX atau Flash 8).
Kemudian ketikan script sebagai berikut :onClipEvent (enterFrame) {// menggerakan obyek kekanan \_x += 10;}action pada movieclip
4. Jalankan movie dengan menekan tombolCtrl+Enter. Maka bola akan bergerak kekanan sampai hilang dari layar. Tekantombol Ctrl+W untuk kembali ke stage

**Penjelasan pola penulisan action script pada movieclip adalah sebagai berikut :**

1. Action pada movieclip harus dimulaidengan script on() atau onClipEvent()
2. Di dalam tanda kurung () pada scrip onmaupun onClipEvent berisi movie eventseperti load, enterframe, mouse down,release dan sebagainya. Movie eventmenyatakan sebuah kejadian yang akan dieksekusi oleh action tersebut. Sebagaicontoh : movie event enterFrame berarti perintah dalam blok {} akan dijalankansepanjang frame tempat moviekliptersebut aktif dan movie event load berarti perintah dalam blok {} akan dijalankansatu kali saja yaitu ketika movieclip diload (ditampilkan) oleh flash player.
3. Tanda { merupakan awal suatu blok script. Yang dimaksud dengan blok scripadalah suatu kumpulan perintah yang akandijalankan sepanjang movie event.
4. Tanda // merupakan tanda yangmenyatakan komentar. Baris yangmemiliki tandatersebut tidak dianggap sebagai suatu perintah. Meskipun demikian tanda //sangatmembantu memberi informasi pada penulisan action scrip kita.
5. Baris \_x+=10; merupakan baris perintah.Perintah dalam movieclip harus ditulisdidalam sebuah blok movie event. Setiapakhir penulisan sebuah perintah selaludiakhiri dengan tanda
6. Tanda } merupakan penutup suatu blok script.Pola penulisan script pada movieclip

**Action Script 3.0**

ActionScript 3.0 mengalami banyak perubahan dari versi sebelumnya. Sangat berbeda hingga membutuhkan virtual machine yang baru untuk menjalankannya. Tapi bagaimana pun masih tetap ActionScript, yang tidak terlalu banyak perubahan dalam syntax. Seringkali ActionScript 3.0 untuk ActionScript 2.0 dengan fungsi yang sama akan terlihat sangat mirip. Tapi tidak selalu begitu. Cukup banyak perubahan yang membuat sulit konversi langsung ke AS 3.0.

***Ini yang baru di ActionScript 3.0 ....?***

* Runtime exceptions
* Runtime variable typing
* Sealed classes
* Method closures
* E4X
* Regular expressions
* Namespaces
* Int and uint data types
* New display list model
* New event model

***1.  Runtime Types***
Dalam ActionScript 2.0, tipe anotasi terutama bantuan untuk pengembang; saat runtime, semua nilai yang dynamically typed. Dalam ActionScript 3.0, tipe informasi yang disimpan saat runtime dan dimanfaatkan untuk beberapa tujuan. Flash Player runtime melakukan pemeriksaan pengetikan, meningkatkan sistem keamanan. Jenis informasi ini juga digunakan untuk mewakili variabel dalam mesin asli representasi, meningkatkan kinerja dan mengurangi penggunaan memori.

***2. Sealed Classes***
ActionScript 3.0 memperkenalkan konsep kelas yang tertutup rapat (Encapsulation). Sebuah kelas disegel hanya memiliki seperangkat tetap properti dan metode yang didefinisikan pada saat kompilasi; properti dan metode tambahan tidak dapat ditambahkan. Hal ini membuat waktu kompilasi yang lebih ketat, menghasilkan program yang lebih kuat. Ini juga meningkatkan penggunaan memori dengan tidak memerlukan tabel hash internal untuk setiap objek misalnya. Kelas dinamis juga mungkin menggunakan kata kunci dynamic.

***3. Method Closures***
Event handling disederhanakan dalam ActionScript 3.0 berkat method closures, yang menyediakan built-in delegasi. Dalam ActionScript 2.0, closure tidak akan ingat objek itu berasal, menyebabkan perilaku tak terduga ketika closure itu digunakan. Kelas mx.utils.Delegate solusi yang populer, untuk menggunakannya, perlu menulis kode sebagai berikut :
*myButton.addEventListener("click", Delegate.create(this, someMethod));
Delegate.create(this, someMethod);*
Kelas ini tidak lagi diperlukan karena dalam ActionScript 3.0, sebuah metode penutupan akan dihasilkan ketika someMethod adalah ditunjuk. Metode Closures akan secara otomatis mengingat objek aslinya misalnya. Sekarang, bisa saja ditulis:
*myButton.addEventListener("click", someMethod);*

***4. ECMAScript Untuk XML (E4X)***
ActionScript 3.0 mengimplementasi penuh ECMAScript untuk XML (E4X), baru-baru ini sebagai standar ECMA-357. E4X secara natural menawarkan fasih bahasa set konstruksi untuk memanipulasi XML. Tidak seperti tradisional parsing XML API, E4X membuat XML merasa seperti tipe data asli dari bahasa. E4X mengeffektifkan pengembangan aplikasi yang memanipulasi XML dengan secara drastis mengurangi jumlah kode yang dibutuhkan.

***5. Regular expressions***
ActionScript 3.0 meliputi dukungan asli untuk ekspresi reguler sehingga Anda dapat dengan cepat mencari dan memanipulasi string. ActionScript 3.0 mengimplementasikan ekspresi reguler didefinisikan dalam Bahasa Spesifikasi ECMAScript (ECMA-262).

***6. Namespaces***
Namespaces adalah mekanisme baru yang inovatif untuk mengendalikan visibilitas deklarasi. Serupa dengan specifiers akses tradisional yang digunakan untuk mengontrol visibilitas deklarasi (public, private, protected), ruang nama kustom dasarnya specifiers akses, yang dapat memiliki nama yang Anda pilih. Flex framework, misalnya, menggunakan namespace untuk mx\_internal data internal. Namespaces dilengkapi dengan Universal Resource Identifier (URI) untuk menghindari bentrokan, dan juga digunakan untuk mewakili XML namespaces ketika bekerja dengan E4X.

***7. New Primitive Types***
ActionScript 2.0 memiliki tipe numerik tunggal, Number ganda presisi floating point nomor. Suatu penambahan ActionScript 3.0 yang baru int tipe 32-bit integer yang memungkinkan penandaan kode ActionScript mengambil keuntungan dari kemampuan matematika integer cepat dari CPU. Type data int bagus untuk loop counter dan hampir di mana saja titik desimal tidak diperlukan. Jenis baru lainnya uint sebuah 32-bit unsigned integer mirip dengan tipe int

***8. Flash Player API features***
Flash Player API adalah kumpulan kelas dan fungsi yang mengekspos kemampuan Flash Player dengan bahasa ActionScript. Fungsi ini adalah jembatan antara inti/core bahasa ActionScript dan sisanya dari platform. Ini adalah sumber dari banyak keunggulan yang tersedia untuk aplikasi Flash dan sangat penting untuk melengkapi bahasa inti. Meskipun tidak ada ruang di sini untuk menutupi API secara rinci.

***9. DOM3 Event Model***
Model event menyediakan cara standar untuk menghasilkan dan penanganan pesan event sehingga objek di dalam aplikasi dapat berinteraksi dan berkomunikasi, menjaga status dan merespons perubahan. Berpola setelah W3C Acara DOM3 spesifikasi, model ini memberikan gambaran yang lebih jelas dan lebih efisien daripada acara mekanisme sistem yang tersedia dalam versi sebelumnya ActionScript. Framework aplikasi Flex menggunakan model peristiwa yang sama seperti Flash Player API, sehingga sistem acara dipersatukan di platform dari atas ke bawah.

***Tujuan Pengembangan ActionScript 3.0 :***

Tujuan pengembangan ActionScript 3.0 adalah untuk menciptakan bahasa dengan cepat, yang cocok untuk membangun aplikasi Internet yang kaya, yang telah menjadi bagian penting dari pengalaman web. Developer ingin ActionScript 3.0 untuk memberikan perbaikan , model  pemrograman yang konsisten; sesuai dengan standar industri, dan kinerja suatu urutan besarnya lebih besar dari pada yang bisa dilakukan versi sebelumnya. Meskipun ActionScript 3.0 mewakili sebuah model pemrograman baru untuk runtime, ini adalah salah satu yang akan dikenal oleh pengembang dengan pengetahuan dasar tentang pemrograman berorientasi objek.

***Tujuan perancangan ActionScript 3.0 adalah:***

* Keamanan: bahasa mendukung keamanan pengetikan sehingga para pengembang dapat menulis jelas (tidak ambigu) mudah maintainable code.
* Kesederhanaan: Bahasa ini cukup intuitif bagi pengembang untuk bisa membaca dan menulis program tanpa terus-menerus berkonsultasi dengan referensi manual.
* Kinerja: bahasa memungkinkan pengembang untuk menulis program yang kompleks dengan efisien dan responsif.
* Kompatibilitas: Bahasa yang pendek menyediakan kompatibilitas ke belakang dan ke depan jalan dan tumpang tindih yang signifikan dengan standar industri.

ActionScript 3.0 berdasarkan spesifikasi ECMAScript yang telah di formalkan  dari fitur ActionScript 2.0, menambah kemampuan ECMAScript untuk XML (E4X), dan menyatukan bahasa menjadi sebuah kesatuan yang utuh.