

## **BAB II**

### **LANDASAN TEORI**

#### **2.1 Pengertian Alat**

Alat adalah suatu benda yang dipakai untuk mengerjakan sesuatu; perkakas, perabot, yang dipakai untuk mencapai maksud (*Kamus Besar Bahasa Indonesia*, 2005, hal: 30).

#### **2.2 Pengertian Musik**

Musik adalah ilmu atau seni menyusun nada atau suara di urutan, kombinasi, dan hubungan temporal untuk menghasilkan komposisi (suara) yang mempunyai kesatuan dan kesinambungan. Nada atau suara yang disusun demikian rupa sehingga mengandung irama, lagu, dan keharmonisan (terutama yang menggunakan alat-alat yang dapat menghasilkan bunyi-bunyi itu), menurut *Kamus Besar Bahasa Indonesia*, 2005, hal: 134.

#### **2.3 Pengertian Alat Musik**

Alat Musik adalah suatu instrumen yang dibuat atau dimodifikasi untuk tujuan menghasilkan musik. Pada prinsipnya, segala sesuatu yang memproduksi suara, dan dengan cara tertentu biasa diatur oleh musisi, dapat disebut sebagai alat musik. Walaupun demikian, istilah ini umumnya diperuntukkan bagi alat yang khusus ditujukan untuk musik. Bidang ilmu yang mempelajari alat musik disebut organologi (*Seni Musik SMP*, 2006, hal: 122).

#### **2.4 Alat Musik Berdasarkan Cara Memainkan**

##### **2.4.1 Alat musik tiup**

Alat musik tiup menghasilkan suara sewaktu suatu kolom udara didalamnya digetarkan. Tinggi rendah nada ditentukan oleh frekuensi gelombang yang dihasilkan terkait dengan panjang kolom udara dan bentuk instrumen, sedangkan *timbre* dipengaruhi oleh bahan dasar, konstruksi instrumen dan cara

menghasilkannya (*Seni Musik SMP*, 2006, hal:124). Contoh alat musik ini adalah terompet dan *flute*.

#### **2.4.2 Alat musik pukul**

Alat musik pukul menghasilkan suara sewaktu dipukul atau ditabuh. Alat musik pukul dibagi menjadi dua yakni bernada dan tidak bernada. Bentuk dan bahan bagian-bagian instrumen serta bentuk rongga getar, jika ada, akan menentukan suara yang dihasilkan instrumen (*Seni Musik SMP*, 2006, hal: 125). Contoh alat musik ini adalah gendang dan kolintang.

#### **2.4.3 Alat musik petik**

Alat musik petik menghasilkan suara ketika senar digetarkan dengan cara dipetik. Tinggi rendah nada dihasilkan dari panjang pendeknya dawai (*Seni Musik SMP*, 2006, hal: 126). Contoh alat musik ini adalah gitar dan kecapi.

#### **2.4.3 Alat musik gesek**

Alat musik gesek menghasilkan suara ketika dawai digesek. Seperti alat musik petik, tinggi rendah nada tergantung panjang dan pendeknya dawai (*Seni Musik SMP*, 2006, hal: 127). Contoh alat musik ini adalah kepyak dan rebab.

### **2.5 Alat Musik Daerah**

#### **2.5.1 Jawa**

Kecapi adalah alat musik petik yang berasal dari daerah Jawa Barat. Bentuk organologi kecapi adalah sebuah kotak kayu yang di atasnya berjajar dawai/senar. Kotak kayu tersebut berguna sebagai resonatornya. Alat musik yang menyerupai Kecapi adalah siter dari daerah Jawa Tengah.

#### **2.5.2 Sumatera**

Talempong adalah alat musik tradisional Minangkabau yang biasanya terbuat dari kuningan dan bentuknya mirip dengan alat musik gamelan yang ada di Jawa. Talempong biasanya digunakan untuk mengiringi tari piring. Talempong ini telah disusun sesuai dengan tangga nada yang ditentukan oleh masing-masing lempengan batu tersebut, sehingga bisa dimainkan mengikuti irama lagu tradisional Minangkabau. Batu ini berjumlah 6 (enam) buah dan sudah ada sejak

dahulu yang terletak di 38 Km arah utara kota Payakumbuh dan 47 Km dari Sarilamak.

### **2.5.3 Sulawesi**

Kolintang adalah alat musik yang berasal dari Minahasa, Sulawesi Utara (Sulut) dan telah dikenal sejak puluhan tahun silam. Alat musik kolintang terbuat dari kayu lokal yang ringan namun kuat. Dulu alat musik ini biasa dimainkan untuk mengisi berbagai acara seperti pesta pernikahan, penyambutan, peresmian, pengucapan syukur, acara keagamaan dan acara pertandingan.

### **2.5.4 Kalimantan**

Sampek (sampe/sapek) adalah alat musik yang bentuknya menyerupai gitar atau alat musik petik dengan dawai berjumlah tiga atau empat, yang berasal dari daerah Kalimantan. Alat musik ini terbuat dari bahan kayu yang dipenuhi dengan ornamen/ukiran yang indah, khas suku Dayak.

### **2.5.5 NTB dan NTT**

Serunai merupakan alat musik yang berasal dari Nusa Tenggara Barat. Alat musik ini digunakan dengan cara ditiup dan mempunyai suara yang cukup merdu. Alat ini biasa dipergunakan untuk acara adat.

Sasando adalah sebuah alat instrumen petik musik. Instrumen musik berasal Rote, Nusa Tenggara Timur. Bagian utama sasando berbentuk tabung panjang yang biasa terbuat dari bambu. Lalu pada bagian tengah, melingkar dari atas ke bawah diberi ganjalan-ganjalan dimana senar-senar (dawai-dawai) yang direntangkan ditabung, bertumpu dari atas kebawah. Alat musik ini ditempatkan pada tempat yang mempunyai model serupa dengan tempat minum tuak yang disebut '*haik*'.

### **2.5.6 Maluku dan Papua**

Tifa adalah alat musik yang berasal dari maluku. Alat musik ini mempunyai fungsi untuk acara-acara adat istiadat di pulau Maluku. Alat ini merupakan jenis alat yang ditabuh, bahan yang digunakan adalah sebuah bambu yang berukuran sedang.

## 2.6 Sekilas *Macromedia Flash 8*

*Macromedia flash 8* merupakan sebuah program aplikasi yang distandarkan untuk menggambar grafis dan animasi yang dipasang pada *website*. Dalam perkembangannya, *Macromedia flash 8* mulai beradaptasi dengan penggunaannya (*user*) selain sebagai pembuatan *website flash* kini digunakan juga sebagai pembuatan logo, *movie*, *game*, presentasi, *screen saver* serta CD interaktif. Hal yang paling baru adalah *flash* kini dapat berkomunikasi dengan telepon dan pembuatan aplikasi-aplikasi lainnya (*Belajar Sendiri Macromedia Flash 8*, 2003, hal: 55).

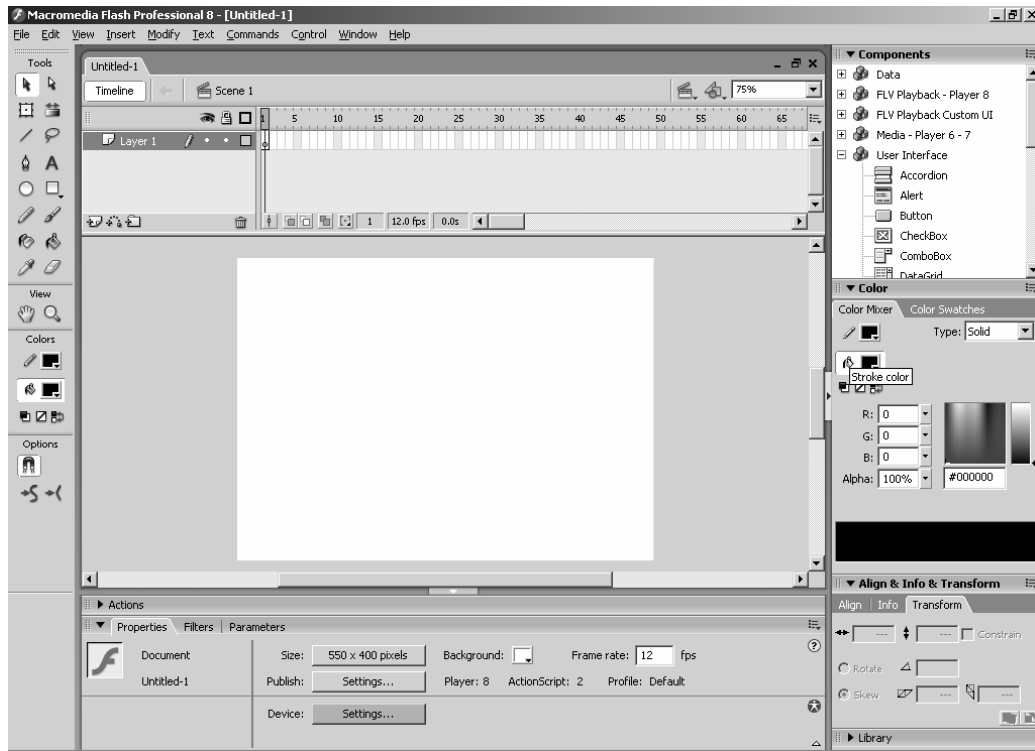
Sedangkan *movie-movie* yang terdapat dalam *flash* mempunyai ukuran yang kecil sehingga mudah untuk di-*upload* atau di-*download*. *Flash 8* juga mempunyai keteraturan dalam pembuatan objek, contohnya *layer* (lapisan) karena dengan *layer* ini pengguna bisa memisah-misahkan objek sesuai dengan keinginan pemakai. Program aplikasi *Flash 8* juga kini sudah dilengkapi dengan bahasa pemrograman *ActionScript versi 2.0* yang dapat berinteraksi dengan bahasa pemrograman yang lain seperti *Java Script*, *Visual Basic 6.0*, *CSS*, *SQL*, *Microsoft Access* dan program-program lainnya.

## 2.7 Pengenalan *Interface* Pada *Macromedia Flash 8*

Saat pertama kali menjalankan program *flash* secara *default* (standar), *flash* akan menampilkan sebuah area kerja yang terdiri dari *stage*, *Panel toolbox*, *timeline*, *color mixer*, *properties* dan *action*. Tujuannya adalah untuk mempermudah pengguna (*user*) untuk menggunakan *Macromedia flash*. Berikut ini akan diuraikan beberapa panel yang terdapat di *Macromedia flash 8*.

### 2.7.1 Layar *Flash* dan *Stage*

*Stage* adalah sebuah area kerja untuk membuat suatu objek baik dalam bentuk bergerak (animasi) maupun dalam bentuk *vector* (diam). *Stage* juga dapat digunakan untuk menampilkan berbagai *movie*.



**Gambar 2.1** Area Kerja pada *Macromedia Flash 8*

### 2.7.2 Panel *Toolbox*

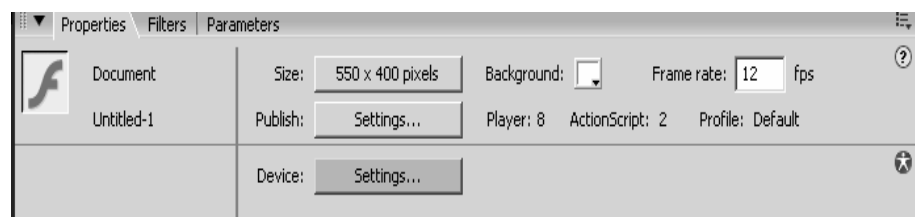
*Toolbox* merupakan kumpulan *tool* yang sering digunakan dalam pembuatan animasi baik sebagai seleksi, pemberian warna pada objek, maupun memodifikasi objek dan masih banyak lagi kegunaan *tool* dalam *flash 8*.



**Gambar 2.2 Panel Toolbox**

### 2.7.3 Panel *Properties Inspector*

Panel *Properties Inspector* adalah sebuah jendela panel yang berisikan berbagai atribut objek. Contohnya adalah pengaturan *stage* tampilan panel *properties inspector* secara otomatis dapat berganti-ganti tergantung objek yang kita pilih.

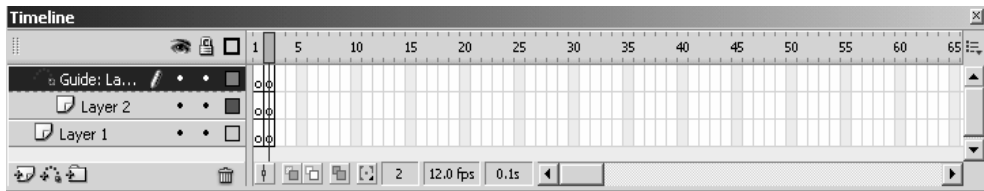


**Gambar 2.3 Panel *Properties Inspector***

### 2.7.4 Panel *Timeline*

Panel *timeline* adalah sebuah jendela yang berfungsi sebagai pengatur pergerakan suatu animasi dan juga sebagai pengatur isi sebuah *movie*. Di dalam *timeline* suatu animasi dapat diatur baik kecepatan maupun kelambatan animasi.

Di dalam *timeline* pengguna juga dapat menggunakan sebuah *layer* (lapisan) yang berfungsi sebagai pemisah antara satu objek dengan objek yang lain.



**Gambar 2.4 Panel *Timeline***

Berikut penjelasan mengenai atribut-atribut yang terdapat pada *timeline*:

**a. *Frame***

*Frame* merupakan segmen-segmen dari *movie* yang akan dijalankan secara bergantian. *Frame* juga sering digunakan sebagai pengontrol untuk jalannya animasi. Biasanya *frame* secara *default* akan berada pada *line* pertama dan dalam kondisi yang *blank* (kosong).

**b. *Layer***

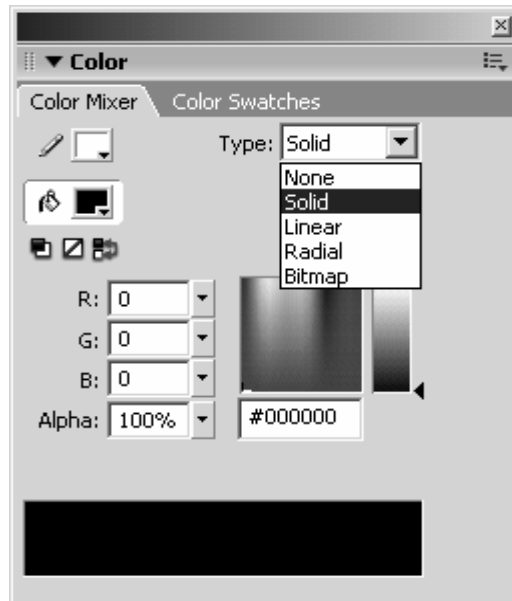
*Layer* merupakan lapisan-lapisan yang berfungsi sebagai pemisah antara satu objek dengan objek yang lain. Urutan posisi *layer* menentukan tampilan masing-masing *layer* tersebut yang akan dijalankan secara bersamaan.

**c. *Framerate***

*Framerate* adalah ukuran kecepatan *frame* yang akan dijalankan di dalam *stage*. Dengan adanya *framerate*, animasi dapat diatur kecepatan dan kelambatannya. Ukuran *frame* menggunakan satuan fps (*frame per second*). *Frame rate* mempunyai ukuran antara 0.01 sampai dengan 120 fps. *Framerate* 30 fps artinya setiap detik *frame* yang ditampilkan sebanyak 30 secara *default frame* dalam *flash mx* adalah 12 fps.

### 2.7.5 Panel *Color Mixer*

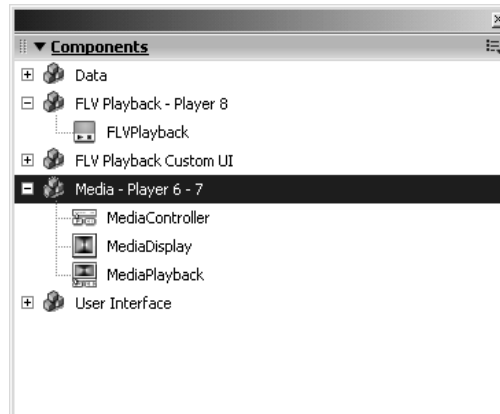
Panel *color mixer* adalah sebuah jendela panel yang digunakan untuk mengedit suatu warna. Dalam *color mixer* terdapat tiga jenis warna yaitu *solid*, *linear* dan *radial*. Ketiga jenis warna ini dapat diubah-ubah sesuai dengan keinginan.



**Gambar 2.5** Panel *Color Mixer*

### 2.7.6 Panel *Components*

Panel *components* adalah sebuah jendela Panel yang berisikan klip-klip *movie* yang kompleks dan memiliki parameter-parameter yang telah didefinisikan dan serangkaian metode-metode *ActionScript* yang dapat diset ulang dan diberi opsi-opsi tambahan sesuai dengan kebutuhan. Perlu diketahui bahwa panel *components* menggantikan fungsi *smart clips* pada versi *macromedia flash* yang terdahulu.



**Gambar 2.6 Panel Components**

### 2.7.7 Panel Library

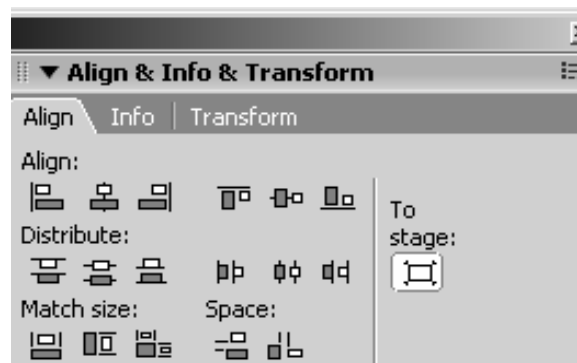
Panel *Library* di dalam *macromedia flash 8* digunakan sebagai tempat menyimpan objek yang telah dibuat di dalam *stage*. Objek yang ada di *library* dapat digunakan secara berulang-ulang dalam *library* banyak terdapat objek seperti gambar, tombol, *sound* dan *movie clip*. Di bawah ini merupakan *library button* yang disediakan dari standar *Macromedia Flash 8*.



**Gambar 2.7 Panel Library**

### 2.7.8 Panel Align

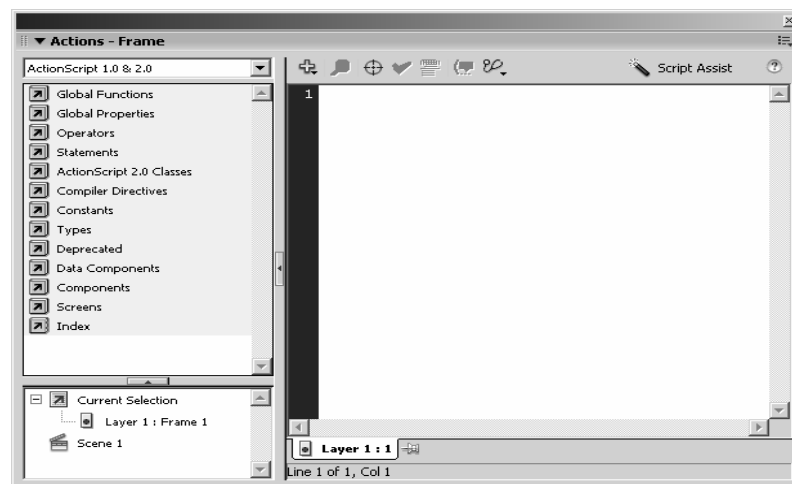
Panel *align* merupakan suatu jendela perintah yang berfungsi untuk menentukan posisi suatu objek di dalam *stage*. Panel *align* terdiri dari empat atribut yaitu *align*, *distributor*, *match size* dan *space*.



**Gambar 2.8 Panel Align**

### 2.7.9 Panel Actions

Panel *actions* merupakan sebuah panel yang berfungsi untuk membuat atau menyediakan kebutuhan untuk berinteraksi dengan bahasa pemrograman *ActionScript* dan pada Panel *action* juga dapat digunakan interaksi antara bahasa pemrograman yang lain.



**Gambar 2.9 Panel ActionScript**

## 2.8 Terminologi-Terminologi Macromedia Flash 8

Pemahaman dari beberapa istilah dan terminologi-terminologi yang ada dalam aplikasi *Macromedia Flash 8* dapat mempermudah pembahasan dalam menggunakan aplikasi tersebut dan mempermudah melakukan pekerjaan pembuatan proyek aplikasi latihan. Berikut beberapa bagian dari terminologi-terminologi yang terdapat di dalam *Macromedia Flash 8*:

### 2.8.1 *Artwork*

Ada bermacam cara untuk membuat dan memodifikasi *artwork* dalam fasilitas pada *Macromedia Flash MX 8.0*. Di dalam fasilitas *Artwork* juga kita bisa mengimpor dari aplikasi lain. *Artwork* dapat berupa objek *vector*, *image bitmap*, objek teks, video suara dan lain-lain.

### 2.8.2 *Symbol*

*Symbol* (simbol) dalam *Macromedia Flash 8* adalah sebuah elemen yang terdiri dari objek grafik, objek tombol, objek *movieclip* dan juga *file* suara atau *font* yang digunakan berulang kali dalam sebuah *library*. Simbol dalam *flash* secara *default* diletakkan di dalam panel *library*.

### 2.8.3 *Instance*

*Instance* adalah sebuah tempat untuk memberi nama variabel ke sebuah simbol-simbol yang bisa berupa *button*, *movieclip* dan simbol *graphic*. Dalam *instance* ini juga kita bisa memodifikasi sebuah simbol tanpa harus merubah nama utama dari simbol tersebut.

### 2.8.4 *Animasi*

Animasi adalah suatu objek yang terdiri lebih dari satu yang mampu bergerak baik secara *random* (acak) maupun berubah bentuk, berubah ukuran, berubah warna, berotasi (berputar), berubah keburaman (*opacity*) dan perubahan-perubahan yang lainnya.

### 2.8.5 *Movie*

*Movie* adalah serangkaian animasi yang dibuat berdasarkan satu alur cerita. Di dalam *movie* tersebut terdiri dari beberapa *scene* yang memiliki sebuah *timeline* dan juga animasi. Sebuah *scene* terdiri atas beberapa *frame*. Sebagian besar *movie* terdiri atas grafik *vector*, *bitmap*, teks, animasi, video, suara dan aplikasi untuk kebutuhan suatu situs.

## 2.9 *Penggunaan ActionScript*

*ActionScript* adalah sebuah bahasa pemrograman yang digunakan oleh *Macromedia Flash 8* untuk mempermudah pembuatan animasi. Dasar

pemrograman *actionscript* berasal dari bahasa C. Dalam bahasa pemrograman *ActionScript* mempunyai sifat *Case Sensitive* (Huruf besar dan kecil dibedakan).

### 2.9.1 Fungsi *ActionScript*

*ActionScript* memiliki beberapa fungsi yang sangat membantu para *user* untuk merancang sebuah aplikasi, sebagai berikut:

#### a. Membuat sistem aplikasi

Pembuatan aplikasi menggunakan *ActionScript* merupakan suatu solusi atau terobosan untuk membuat aplikasi lebih interaktif dan menarik. Dengan demikian aplikasi yang telah dibuat dapat menarik bagi *user* (pemakai).

#### b. Membuat aplikasi lebih mudah

Dengan *ActionScript*, *user* (pemakai) dapat lebih mengembangkan aplikasinya. Selain itu dengan *ActionScript* kita diberi kemudahan dalam menjalankan program *ActionScript*.

#### a. Membuat situs atau program yang dinamis

*ActionScript* merupakan sebuah program yang mampu berinteraksi dengan bahasa pemrograman lain seperti *JavaScript*, *PHP*, *MYSQL* karena pemrograman *ActionScript* mempunyai sifat *Case Sensitive*. Untuk membuat aplikasi dengan menggunakan bahasa pemrograman *ActionScript*, kita terlebih dahulu harus mengerti tentang struktur bahasa pemrograman *ActionScript*.

### 2.9.2 Struktur Bahasa Pemrograman *ActionScript*

Seperti menggunakan bahasa sehari-hari, kita perlu menempatkan kata benda, kata sifat, dan kata kerja dengan susunan yang dimengerti orang lain. Di dalam pemrograman *ActionScript* penyusunan sebuah objek dan properti harus membentuk *statement* yang dimengerti oleh *flash*.

*Flash 8* menggunakan bahasa *dot syntax*, yang berfungsi menggantikan *slash syntax* pada versi yang terdahulu. Dot atau titik (.) digunakan untuk menunjukkan properti atau metode yang terkait dengan objek.

### 2.9.3 Tanda Baca Dalam *ActionScript*

Tanda baca dalam pemrograman *ActionScript* terdiri dua macam, yaitu:

#### a. *Curly braces dan semicolon*

*Curly braces* atau kurung kurawal ({} ) digunakan untuk melakukan pengelompokkan, atau sering disebut juga sebagai blok program. Sedangkan *semicolon* atau titik koma (;) digunakan untuk mengakhiri sebuah program atau *statement*.

#### b. *Comments*

*Comments* dalam pemrograman *ActionScript* adalah sebuah komentar untuk menjelaskan apa arti kode tersebut. *Comments* dapat membantu bagi *user* untuk memahami *script* apabila hendak me-*review* kode tersebut. *Comments* ditandai dengan tanda *slash (/)*. *Comments* juga dapat ditandai dengan tanda bintang setelah tanda *slash (/\*)* dan diakhiri dengan tanda bintang dan *slash (/\*)*.

### 2.9.4 Kategori Dalam *ActionScript*

Didalam pemrograman *ActionScripts* terdapat dua kategori keadaan yaitu:

#### a. **Kategori dalam keadaan *Expert Mode***

*Expert Mode* adalah sebuah keadaan dimana seorang *user* (pemakai) atau *programmer* tidak lagi memerlukan bantuan (*guide*) dari *ActionScript*. Biasanya *Expert Mode* digunakan oleh para *user* (pemakai) atau *programmer* yang sudah *expert* (ahli).

#### b. **Kategori dalam keadaan *Normal Mode***

*Normal Mode* adalah keadaan dimana seorang *user* (pemakai) atau *programmer* menggunakan bantuan (*guide*) dari *ActionScript*. *Normal Mode* digunakan untuk para pemula. Di dalam *Normal Mode* terbagi ke dalam beberapa kategori berikut ini:

##### 1. *Actions*

*Actions* adalah sebuah pernyataan yang menginstruksikan sebuah *movie* untuk melakukan suatu perintah pada saat *flash exe* dijalankan.

## 2. **Boolean**

*Boolean* adalah sebuah nilai yang mempunyai nilai “Benar” atau “Salah” atau dapat juga bernilai 0 dan 1.

## 3. **Classes**

*Classes* adalah sebuah jenis-jenis yang dapat ditentukan sebagai jenis objek baru dengan membuat sebuah fungsi *constructors*.

## 4. **Constants**

*Constants* adalah elemen-elemen tetap yang berguna untuk membandingkan sebuah nilai.

## 5. **Constructors**

*Constructors* adalah sebuah fungsi yang digunakan untuk menentukan *properties* dan *methods* dari sebuah *class*.

## 6. **Data Type**

*Data Type* adalah sebuah kumpulan nilai-nilai dan operasi berupa *string*, *number*, *Boolean*, *Object*, *Movieclip*, *Function* dan *Null*.

## 7. **Events**

*Events* adalah *actions* yang terjadi ketika sebuah *movie* dijalankan atau di-*load* ketika *playhead* mencapai sebuah *frame*, ketika seorang meng-klik sebuah tombol atau klip *movie* atau mengetikkan *keyboard*.

## 8. **Events Heandlears**

*Event Heandlears* adalah *action-action* khusus yang mengatur sebuah *Event script*. *Event Heandlears* biasanya diletakkan di dalam *frame*.

## 9. **Expressions**

*Expressions* adalah berbagai gabungan simbol-simbol *ActionScript* yang mewakili sebuah nilai. *Expressions* terdiri dari *Operator* dan

*Operand*. Contoh, dalam *expressions*  $x+6$ , huruf “x” dan angka “6” merupakan *Operand*, sedangkan “+” adalah sebuah *Operator*.

#### **10. Functions**

*Functions* adalah sebuah *statement* yang dapat menghasilkan sebuah nilai dan dapat dipakai berulang-ulang.

#### **11. Identifiers**

*Identifiers* adalah nama-nama untuk mengenali sebuah variabel, properti, objek. Huruf pertama dalam penulisan *identifiers* adalah garis bawah “\_”, atau tanda dollar “\$”.

#### **12. Instances**

*Instances* adalah objek yang terjadi pada beberapa *class*. Masing-masing *class* berisi semua *properties* dan *method class*. Semua klip *movie* adalah sebuah *instance* dengan *properties* dan *method class movieclip*.

#### **13. Instance Name**

*Instance Name* adalah sebuah nama untuk digunakan sebagai target *instance movieclip* dan tombol dalam *script*.

#### **14. Keyword**

*Keyword* adalah kata-kata tertentu yang mempunyai arti khusus. Sebuah *keyword* tidak dapat digunakan sebagai *identifiers* atau variabel.

#### **15. Methods**

*Methods* adalah sebuah fungsi-fungsi yang digunakan untuk memberi tanda pada sebuah objek fungsi yang telah digunakan dan dapat dikatakan sebagai sebuah *method object*.

#### **16. Object**

*Object* adalah beberapa kumpulan *properties* dan *method*. Masing-masing objek memiliki nama dan sebuah *instance* sebagai bagian dari *class*.

#### **17. Operator**

*Operator* adalah persyaratan menghitung sebuah nilai baru dari satu atau beberapa nilai. Nilai-nilai yang digunakan bersama *operator* disebut *operand*.

### **18. Parameter (Argument)**

*Parameter* adalah tempat untuk nilai- nilai yang melewati *function*.

### **19. Properties**

*Properties* adalah sebuah atribut-atribut yang digunakan untuk menentukan sebuah objek.

### **20. Target Path**

*Target Path* adalah pengamatan nama-nama *instance* klip movie, variabel dan objek dalam sebuah *movie* hierarki sebuah target dapat digunakan untuk menunjuk sebuah *action* pada sebuah klip *movie*.

### **21. Variabel**

Variabel adalah *identifier* yang memegang nilai-nilai berbagai *data type*. Di dalam variabel kita dapat mengubah-ubah nilai yang terdapat dalam variabel tersebut. Variabel dapat digunakan untuk menampung sebuah nilai.

## **2.10 Suara**

Suara merupakan bagian dari *movie flash* yang sangat penting. Terkadang dengan aspek suara dapat membuat sebuah *movie flash* menjadi sangat menarik dan lebih kreatif. Di dalam aplikasi *Macromedia Flash 8* dapat menggunakan format suara seperti format WAV, MP3, dan AIF. Untuk dapat menggunakan suara tersebut caranya ialah dengan mengklik **File > Import**. Setelah diimpor, *file* suara akan berada di *library*. Untuk menggunakan suara di dalam *movie flash*, pilih simbol suara dari *library* dan geser ke *stage*. Di dalam *Movie flash* kita juga dapat meng-*edit* sebuah *sound* yang berada di *Sound Panel*.

### **2.10.1 Streaming sound**

Ketika seorang *user* (pemakai) men-*download* sebuah *movie* dari internet, *streaming sound* dapat dimainkan walau hanya sedikit bagian *sound* yang sudah di-*download*.

### 2.10.2 *Event Sound*

*Event Sound* adalah sebuah kejadian yang menjalankan *sound* terus-menerus secara beraturan. *Event Sound* terdiri dari:

- a. ***Loop***: Merupakan sebuah *looping* (pengulangan) berapa kali suara akan dimainkan.
- b. ***Edit***: Efek suara dapat di-*edit* secara manual dengan meng-klik tombol *edit*.
- c. ***Effect***: Efek ini digunakan untuk menambah suara (*sound*) agar lebih bervariasi, contohnya *fade in* dan *fade out*.

### 2.10.3 Menggunakan Suara dari *File SWF*

Setiap kali menggunakan *Lingage*, suara akan ditambahkan ke dalam *file SWF* sehingga ukuran *file* tersebut menjadi sangat besar. Untuk mengatasi hal ini suara dapat dipisahkan dalam *file SWF* yang lain. Untuk menggunakan suara tersebut cukup dengan membuat *file SWF* tersebut dengan *action loadMovieNum* dan untuk menghentikannya cukup dengan *action unLoadMovieNum*.

## 2.11 Menggunakan *Fscommand*

Apabila *movie* yang akan kita publikasikan sebagai *projector* atau *stand alone player* dapat kita gunakan *action fscommand*. Untuk mengatur tampilan *projector* tersebut. Seleksi *frame* pertama, kemudian pada *Action Panel* klik +> *Basic Action* > *fscommand Action* ini muncul dalam format:

*Fscommand (command, argument)*.

*Fscommand* memiliki beberapa pilihan *command* dan *argument*, diantaranya:

#### 1. ***Fullscreen (true/false)***

Pada saat *movie* dimainkan dengan kondisi *true* maka *projector* akan menjadi *fullscreen* (layar penuh).

### 2. *Allowscale (true/false)*

Pada kondisi *true*, apabila *window* proyektor diperkecil, *movie* didalamnya juga ikut mengecil.

### 3. *Showmenu (true/false)*

Pada *argument false*, menu *pull-down* yang muncul karena klik kanan *mouse* dimatikan.

### 4. *Quit*

*Command quit* dapat digunakan pada tombol untuk keluar dari aplikasi.

## 2.12 Publikasi *File Flash*

Di dalam *flash* dapat dipublikasi dalam berbagai format, diantaranya *Flash* (swf), HTML, *Quicktime*(mov), *RealPlayer* (smil), dan lain-lain. *Movie flash* juga dapat di-*publish* ke dalam *flash projector* (*exe*). Dengan *flash projector*, seorang *user* (pemakai) tidak perlu memiliki *Flash Player*.

Untuk membuat format *publish* pilih menu **File > Publish Setting**. Pada kotak dialog *publish setting* ini juga dapat mengatur kualitas suatu gambar dan suara yang diekspor ke dalam *movie flash*. Tata letak, posisi, kualitas *movie*, dan dimensi *movie flash* di dalam format HTML juga dapat diatur di sini.

## 2.13 Mengoptimalkan *Movie Flash*

Supaya *flash movie* yang kita buat berjalan dengan baik atau mempunyai kualitas yang bagus maka *flash movie* dapat dioptimalkan dengan cara sebagai berikut:

1. Mempertahankan ukuran *file flash* sekecil mungkin. Semakin kecil berarti semakin cepat waktu untuk men-*download*.
2. Membuat jalannya animasi menjadi halus. Hal ini biasanya terkait dengan spesifikasi komputer yang *user* (pemakai) gunakan.
3. Menghindari pembuatan objek yang dianimasikan bersamaan.

4. Membuat revisi *file Flash* menjadi lebih sederhana. Salah satu kritik yang banyak dilontarkan pada animasi *flash* adalah biasanya *movie Flash* hanya dibuat sekali dan jarang di-*update*.
5. Gunakan nama *instance* secara sederhana, mudah dimengerti dan konsisten.
6. Gunakan *comment* untuk menjelaskan *script* sehingga mudah untuk dimengerti.
7. Hindari penggunaan lebih dari satu *scene*.
8. Jangan terlalu sering menggunakan warna *gradient* dan transparansi.