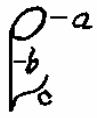
2. TEORI PENUNJANG

2.1. Musik

Dalam musik, pada umumnya digunakan dua sistem penulisan, yaitu sistem yang menggunakan not angka dan not balok. Sistem not angka menggunakan angka-angka untuk menggambarkan masing-masing not (1 = do, 2 = re, dst). Lain halnya dengan sistem not balok yang merupakan perpaduan beberapa unsur yang saling mendukung satu sama lain dengan sebutan abjad, yaitu : C D E F G A B C'.

2.1.1. Nada, Titinada dan Garis-Garis Para Nada

Nada mempunyai durasi (sifat panjang-pendek), *pitch* (tinggi-rendah), intensitas (keras-lemah). Dalam sistem not balok, nada dilambangkan dengan tanda gambar yang disebut titinada (not). Titinada memiliki bermacam-macam bentuk, ada yang bulat telur, terbuka, tertutup, dan ada yang memiliki bendera. Perbedaan bentuk tersebut mewakili arti dan nilai yang berbeda-beda pula.



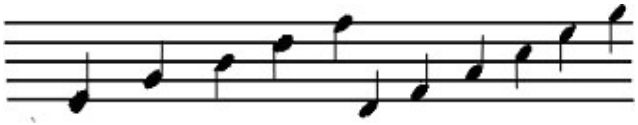
a. Kepala nada
b. Tangkai nada
c. Bendera.

Gambar 2.1. Titinada atau Not Balok

Dalam musik internasional dipergunakan 7 buah titinada pokok (nada pokok), yaitu : C, D, E, F, G, A, B. Titinada-titinada tersebut ditempatkan tepat pada garis-garis dan di antara garis-garis para nada. Garis-garis para nada terdiri dari 5 garis sejajar yang sama jaraknya.

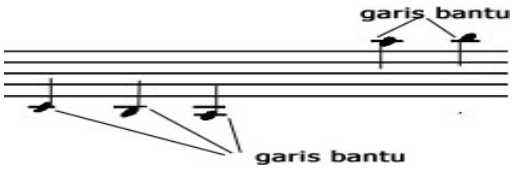
Garis ke 5 _____
4 _____
3 _____
2 _____
1 _____

Gambar 2.2. Garis Para Nada

Not yang ditempatkan pada garis para nada, disebut *Not Garis*, sedangkan nada-nada yang ditempatkan diantara garis para nada, disebut *Not Antara*.

Gambar 2.3. Not Garis dan Not Antara

Selain nada-nada tersebut, ada juga beberapa nada yang tinggi dan rendahnya tidak dapat ditulis dalam garis-garis para nada, melainkan ditulis di bawah dan di atas garis-garis para nada. Nada-nada tersebut ditulis dengan menggunakan garis penolong (garis bantu). Garis penolong (garis bantu) ialah garis-garis pendek yang sejajar dengan garis para nada dan terletak di atas atau di bawah garis para nada.

Cara membuat garis-garis pertolongan :

- Harus sejajar dan sama jaraknya.
- Panjangnya disesuaikan untuk melukiskan satu titinada saja.

Gambar 2.4. Garis Penolong atau Garis Bantu

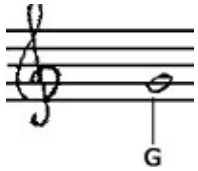
2.1.2. Kunci

Kunci ialah tanda yang ditempatkan pada permulaan garis para nada untuk menentukan nama nada dan menetapkan tinggi rendahnya nada dengan tepat. Sehingga, dengan memberikan tanda kunci di permulaan garus para nada, dapat ditentukan jenis (nama) titinada yang digunakan.

Ada 2 kunci yang sering digunakan, yaitu :

a. Kunci G / Kunci Biola

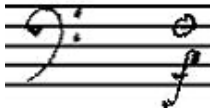
   Kunci ini letak dan bentuknya tidak berubah, yaitu pada garis para nada ke-2 dari bawah. Garis inilah yang menunjukkan letak nada G.



Gambar 2.5. Kunci G

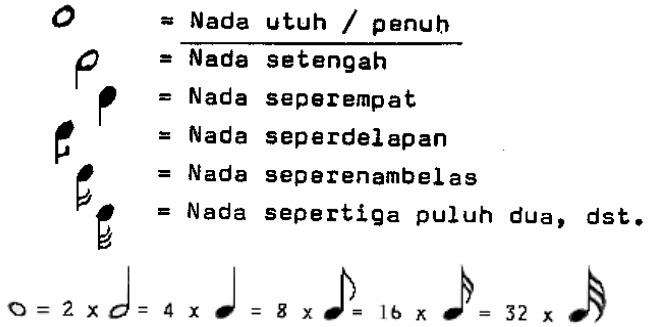
b. Kunci F / Kunci Bass

Kunci ini letak dan bentuknya juga harus tetap, yaitu pada garis para nada ke-4 dari bawah. Garis inilah yang menunjukkan letak nada F.

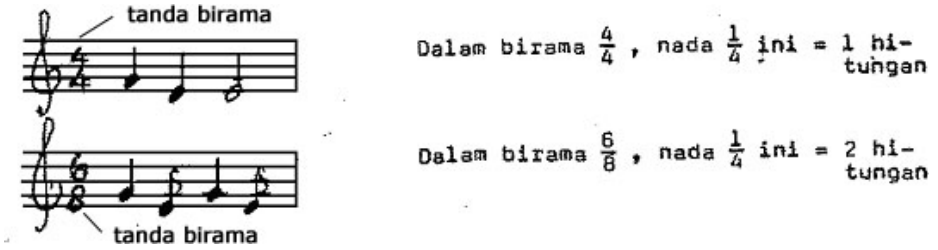


Gambar 2.6. Kunci F

2.1.3. Nilai Nada dan Tanda Istirahat

Nilai sebuah nada dapat dilihat oleh bentuknya. Jadi nilai nada ialah bentuk nada itu sendiri dimana bentuk tersebut menentukan panjang-pendeknya suatu nada, atau berapa lama nada tersebut harus ditahan (lama relatif).

Gambar 2.7. Nilai Nada

Masing-masing nada memiliki nama tersendiri dan nama tersebut tetap tidak berubah. Namun, lama relatif nada itu dimainkan, dipengaruhi oleh tanda birama. Tanda birama adalah angka-angka dalam bentuk pecahan, misalnya : 2/4, 3/4, 4/4, yang diletakkan pada permulaan garis para nada, dan penulisannya setelah kunci. Untuk selanjutnya dijelaskan pada 2.1.4.



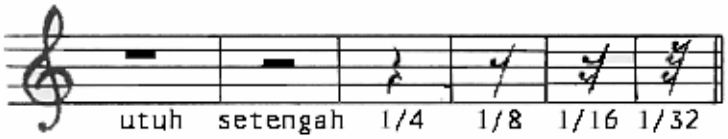
Gambar 2.8. Nilai Nada

Seperti yang telah dijelaskan, nilai nada menentukan lama relatif suatu nada. Lain halnya dengan tanda tempo. Tanda tempo menentukan lama absolut suatu nada. Maksudnya meskipun nilai nada, dan tanda biramanya sama, tetapi bila tanda temponya berbeda maka cara memainkan nada tersebut juga berbeda.

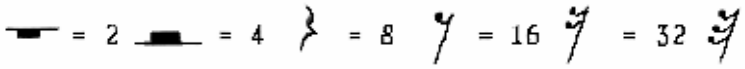
Gambar 2.9. Tanda Tempo

Nilai nada sering diukur dengan ketuk. Dalam hal ini tanda birama seperti yang telah dijelaskan sebelumnya, berpengaruh. Misalnya :

a. Lagu dengan tanda birama 4/4 : 𝅝 = 4 ketuk ; ♩ = 1 ketuk.
   𝅗𝅥 = 2 ketuk ; ♪ = ½ ketuk.
b. Lagu dengan tanda birama 4/2 : 𝅝 = 2 ketuk ; ♩ = ½ ketuk.
   𝅗𝅥 = 1 ketuk ; ♪ = ¼ ketuk.
c. Lagu dengan tanda birama 4/1 : 𝅝 = 1 ketuk ; ♩ = ¼ ketuk.
   𝅗𝅥 = ½ ketuk ; ♪ = 1/8 ketuk.

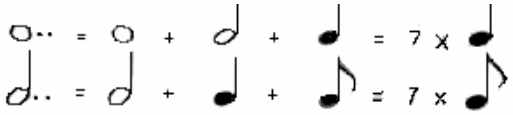
Seperti halnya not, tanda istirahat juga memiliki harga atau nilai, dan harganya sama dengan nilai nada. Tanda istirahat berguna untuk menjelaskan bahwa bila terdapat tanda tersebut, musik atau lagu tidak disuarakan (diam), tetapi ketukan tetap berjalan menurut hitungan tanda diam tersebut. Berikut ini adalah bentuk-bentuk tanda istirahat beserta dengan nilai-nilainya.

Gambar 2.10. Tanda Istirahat

Nilai tanda-tanda istirahat itu, juga nilai perbandingan seperti pada nilai nada.



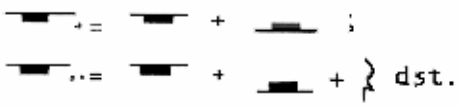
Gambar 2.11. Perbandingan Nilai Tanda Istirahat

Tanda titik di belakang nada berfungsi memperpanjang nilai nada dengan setengah kali nilai nada tersebut. Bila terdapat dua buah titik atau lebih, maka setiap titik nilainya = setengah dari titik di depannya.



Gambar 2.12. Nilai Titik di Belakang Not

Tanda titik di belakang tanda istirahat juga berfungsi sama, yakni memperpanjang waktu istirahat setengah kali dari nilai tanda istirahat itu.

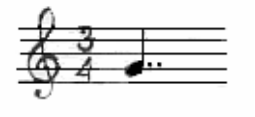


Gambar 2.13. Nilai Titik di Belakang Tanda Istirahat

Jadi ada perbedaan antara titik di belakang not balok dengan titik yang terdapat di belakang not angka. Nilai titik di belakang not balok memiliki nilai setengah kali dari not di depannya, sedangkan titik di belakang not angka bernilai satu not angka.

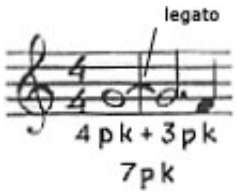
Lagu dengan tanda birama 3/4

5 . . → sol = 3 ketuk. (1 + 1 + 1)

Bila dalam not balok :

Nilai G = 1 ketuk + ½ ketuk + ¼ ketuk

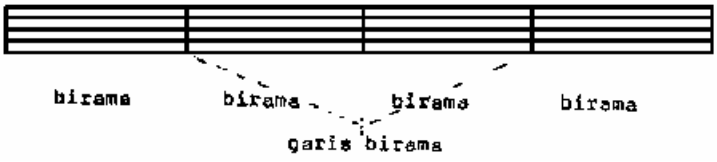
= 1 ¾ ketuk (sol = 1 ¾ ketuk)

Pada keadaan tertentu, ada not yang nilainya tidak dapat dilukiskan dengan titik, karena nilainya sangat lama, sehingga apabila menggunakan titik, harus menggunakan lebih dari dua titik. Penulisan titik lebih dari dua merupakan hal yang tidak umum. Untuk itu, dipakailah busur penghubung (*legato*).

Busur penghubung atau legato hanya boleh digunakan untuk menghubungkan dua nada yang sama tingginya. Apabila dua nada yang sama tingginya dihubungkan oleh legato maka, nada kedua tidak dimainkan tetapi ditahan dan lama ditahannya sama dengan nilai nada tersebut.

Gambar 2.14. Busur Penghubung atau Legato

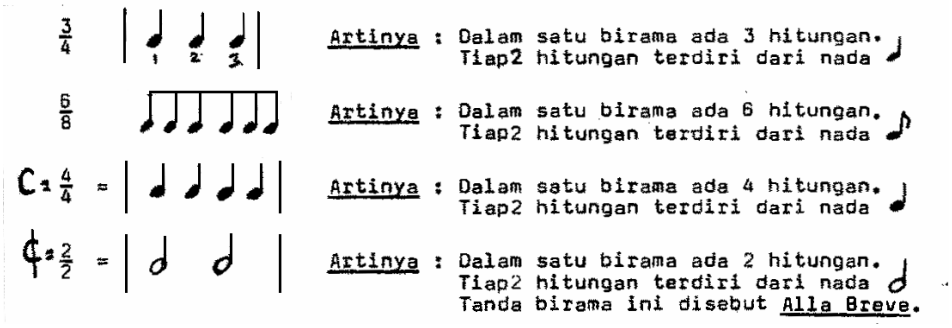
2.1.4. Birama

Sebuah lagu terdiri dari bagian-bagian yang sama yang disebut birama. Setiap birama dibatasi oleh garis tegak lurus yang memotong garis para nada, yang disebut garis birama. Apabila lagu tersebut selesai, maka pada akhir lagu diberi garis birama ganda.



Gambar 2.15. Birama dan Garis Birama

Tanda birama terdiri dari angka pecahan dimana angka yang di atas (pembilang) menunjukkan jumlah bagian (jumlah ketukan) dalam tiap-tiap birama. Sedangkan angka yang di bawah (penyebut) menunjukkan nilai nada dalam satu hitungan (ketukan).



Gambar 2.16. Tanda Birama

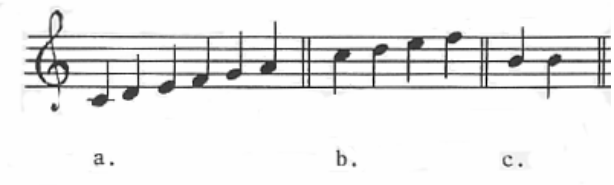
Dengan penjelasan pada Gambar 2.16, maka dalam mengelompokkan nilai nada dalam satu birama tidak bisa sembarangan. Contoh pengelompokan nada yang benar dan salah dapat dilihat pada Gambar 2.17.

Gambar 2.17. Pengelompokan Nilai Nada

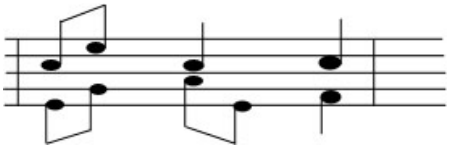
2.1.5. Cara Menulis Nada

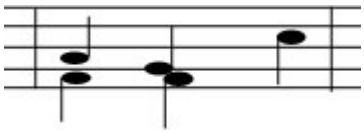
Cara menulis nada, tangkai nada dan bendera ada beberapa aturan, yaitu :

a. Tangkai ditulis sebelah kanan kepala nada, dengan arah ke atas, bila nada tersebut terdapat di bawah garis ke 3 dari garis para nada.
b. Tangkai ditulis di sebelah kiri kepala nada, dengan arah ke bawah, bila nada tersebut terdapat di atas garis ke 3 garis para nada.
c. Pada garis ke 3 garis para nada sendiri, tangkai boleh ditulis dengan arah ke atas maupun ke bawah melihat banyaknya nada lain sebelumnya yang bertangkai ke atas atau ke bawah. Yang terbanyak jumlahnya itu yang diikuti.

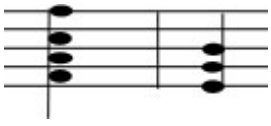


Gambar 2.18. Contoh Cara Menulis Not

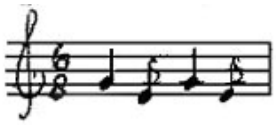
d. Pada notasi 2 suara pada kunci tangkai nada suara-suara atas ke atas, suara bawah ke bawah. Bila kedua suara sampai pada suatu nada yang sama dan nilai nadapun sama maka nada tersebut mendapat tangkai ke atas dan ke bawah.

Gambar 2.19. Contoh Cara Menulis Tangkai Nada

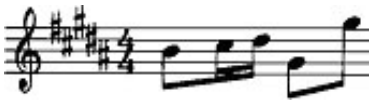
e. Dua nada utuh yang merupakan prime. Dua nada dengan tinggi yang sama dan nilai nada yang berbeda, serta membentuk jarak sekonde, ditulis sedekat mungkin.

Gambar 2.20. Contoh Cara Menulis Tangkai Nada

f. Nada akord dihubungkan dengan satu garis.



Gambar 2.21. Contoh Cara Menulis Akord

g. Bendera selalu ditulis di sebelah kanan dari kepala nada.

Gambar 2.22. Contoh Cara Menulis Bendera

h. Di samping bendera dapat juga dipergunakan garis-garis nilai nada. Garis-garis ini ditulis di atas nada bila kebanyakan dari nada-nada tersebut terdapat di bawah garis ke 3 dan sebaliknya ditulis ke bawah bila nada-nada tersebut di atas garis ke 3.

Gambar 2.23. Contoh Menulis Bendera Dengan Garis

2.1.6. Tanda-Tanda *Accidental* dan Tanda Asing

Tanda *accidental* (tanda kromatis) ialah tanda-tanda untuk menaikkan, menurunkan, dan mengembalikan nada-nada yang dinaikkan/diturunkan ke nada semula. Tanda kromatis ini ditulis di depan not yang akan diturunkan/dinaikkan. Berikut ini beberapa macam tanda kromatis, yaitu :

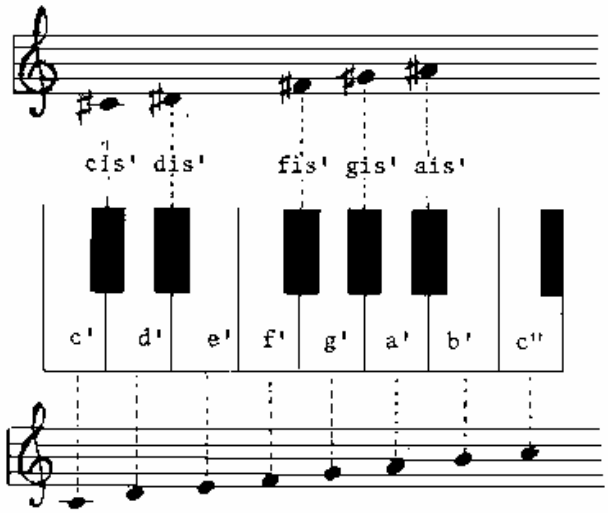
1. Tanda kruis, sharp, palang

   Tanda : #

   Tanda ini menyebabkan laras suatu nada naik setengah (½) laras. Nama nada yang dinaikkan ½ laras ini mirip nama nada aslinya ditambah akhiran “is”.

   - c $^{\#}$ → cis
   - d $^{\#}$ → dis
   - e $^{\#}$ → eis *)
   - f $^{\#}$ → fis
   - g $^{\#}$ → gis
   - a $^{\#}$ → ais
   - b $^{\#}$ → bis*)

   *) antara e - f dan antara b - c tidak ada “nada antaranya” (tuts hitam), maka nada eis persis sama dengan nada f, nada bis sama dengan c.



Gambar 2.24 Cara Menulis Nada Berkruis dan Tempatnya Pada Piano

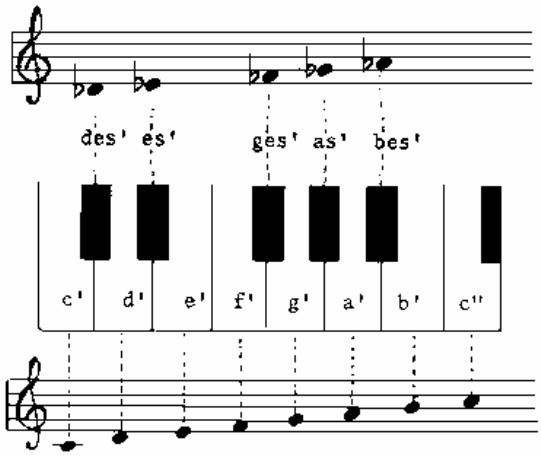
2. Tanda mol, flat

   Tanda : b

   Tanda ini menyebabkan laras suatu nada turun setengah (½) laras. Nama nada-nada yang diturunkan ½ laras ini mirip nama nada aslinya ditambah akhiran “es” atau “s”.

- c^{b} → ces *)
- d^{b} → des
- e^{b} → es
- f^{b} → fes *)
- g^{b} → ges
- a^{b} → as
- b^{b} → bes

*) antara e dan f dan antara b dan c tidak ada “nada antaranya” (tuts hitam), maka nada ces persis sama dengan nada b, nada fes sama dengan e.



Gambar 2.25. Cara Menulis Nada Bermol dan Tempatnya Pada Piano

3. Tanda pugar, natural

   Tanda : ♮

   Tanda ini akan mengembalikan nada-nada yang bertanda kruis, atau mol, kembali kepada nada asalnya.

4. Tanda kruis berganda atau palang berganda

   Tanda : X

   Tanda ini menyebabkan laras suatu nada naik satu laras. Nada-nada yang dinaikkan dengan tanda ini, mendapat akhiran “isis”.

- F ^{x} → Fisis.
- C ^{x} → Cisis, dll.

5. Tanda mol berganda

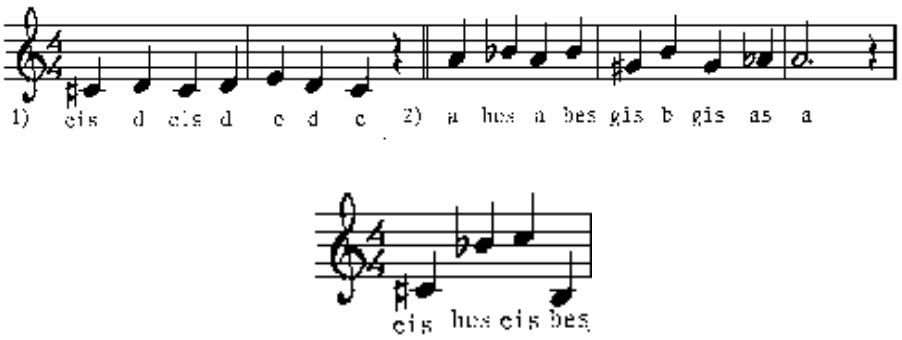
   Tanda : 𝄫

   Tanda ini menyebabkan laras suatu nada turun satu laras. Nama nada pokok yang diturunkan dengan tanda ini, mendapat akhiran “eses” atau “ses” untuk huruf hidup.

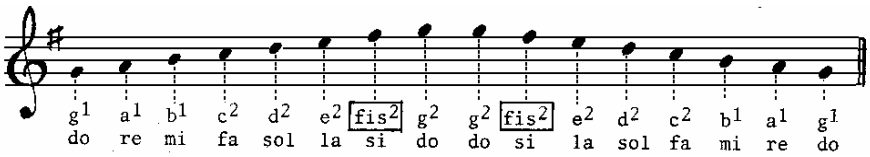
- B ^{bb} → Beses
- D ^{bb} → Deses
- A ^{bb} → Ases
- E ^{bb} → Eses, dll.

Nada pokok (asal) ialah nada yang tidak mempunyai tanda kromatis. Nada-nada asal tersebut adalah C, D, E, F, G, A, B. Sedangkan nada cabang adalah nada pokok yang mendapat tanda-tanda kromatis. Misalnya: Fis, Bes, dll.

Pengaruh tanda kromatis antara lain :

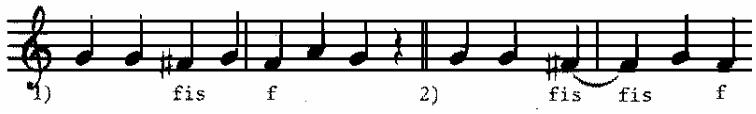
1. Tanda kromatis yang terletak di depan suatu nada, berlaku untuk semua nada yang sama tingginya dengan nada yang dikenakan tanda kromatis tersebut, selama masih terdapat di dalam satu birama. Perlakuan ini tidak berlaku untuk nada yang sama tingginya dalam birama berikutnya atau nada yang sama tetapi dalam oktaf yang berlainan. Tanda kromatis seperti ini disebut sebagai tanda asing (dapat dilihat pada Gambar 2.26).

Gambar 2.26. Tanda Kromatis sebagai Tanda Asing

2. Tanda kromatis (#,b) yang terletak pada permulaan garis para nada, berlaku untuk semua nada yang terletak pada tinggi yang sama dengan tanda kruis (#) atau mol (b) tersebut. Tanda kromatis semacam ini terletak dipermulaan garis para nada ini disebut tanda awal (mula). Tanda awal ini juga yang menunjukkan suatu lagu dimainkan dalam suatu tangga nada tertentu. (dapat dilihat pada Gambar 2.27).

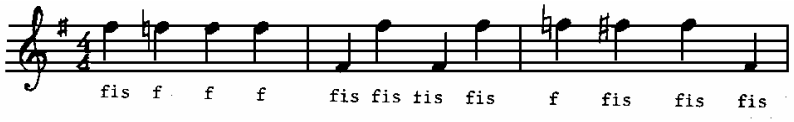


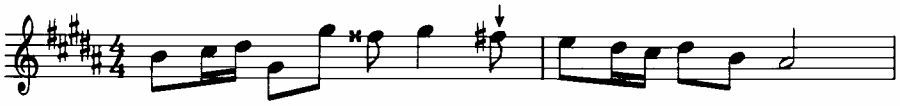
Gambar 2.27. Tanda Kromatis sebagai Tanda Awal

3. Tanda kromatis di depan suatu nada masih berlaku bagi nada-nada yang sama tingginya di dalam ruas berikutnya, asalkan kedua nada tersebut dihubungkan oleh garis legato.



Gambar 2.28. Pengaruh Tanda Kromatis dengan Adanya Legato

4. Tanda pugar juga hanya berlaku untuk satu birama saja.

Gambar 2.29. Pengaruh Tanda Pugar

5. Untuk menghilangkan tanda kruis berganda dan mol berganda, dapat langsung digunakan tanda kruis ataupun tanda mol secara langsung.

Gambar 2.30. Penghilangan Tanda Kruis Berganda dan Mol Berganda

2.1.7. Tangga Nada

Tangga nada adalah urutan naik atau turun dari deretan nada yang bertolak dari nada awal (tonika) hingga ulangan nada tersebut (finalis) dalam oktaf berikut, yang lebih tinggi atau lebih rendah. Tangga nada harus terdiri paling sedikit 5 nada. Tangga nada yang umum dibagi dalam dua kelompok besar :

1. Tangga nada utama :
   - Tangga nada mayor
   - Tangga nada minor relatif
   - Tangga nada kromatis

2. Tangga nada tambahan :
   - Tangga nada minor harmonis
   - Tangga nada minor melodis
   - Tangga nada tzigana
   - Tangga nada utuh

Dalam urutan nada-nada tersebut, ada jarak tertentu yang menyusunnya. Berikut ini adalah jarak untuk masing-masing tangga nada:

1. Tangga nada mayor

C	–	D	–	E	–	F	–	G	–	A	–	B	–	C’
	1		1		½		1		1		1		½	

2. Tangga nada minor harmonis

A	–	B	–	C	–	D	–	E	–	F	–	Gis	–	A’
	1		½		1		1		½		1½		½	

3. Tangga nada melodis

   Naik :

A	–	B	–	C	–	D	–	E	–	Fis	–	Gis	–	A’
	1		½		1		1		1		1		½	

   Turun :

A’	–	G	–	F	–	E	–	D	–	C	–	B	–	A
	1		1		½		1		1		½		1	

4. Tangga nada tzigana

A	–	B	–	C	–	Dis	–	E	–	F	–	Gis	–	A’
	1		½		1½		½		½		1½		½	

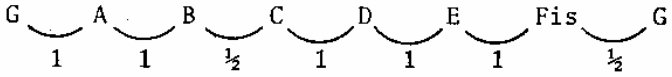
2.1.8. Tangga Nada Berkruis

Disebut tangga nada berkruis, karena di antara nada-nada yang tersusun dalam tangga nada tersebut mendapat tanda kruis (#) yaitu dinaikkan setengah jarak. Adanya kruis dalam tangga nada ini adalah untuk menyesuaikan jarak-jarak nadanya menurut ketentuan yang berlaku dalam susunan tangga nada diatonis mayor, yaitu :

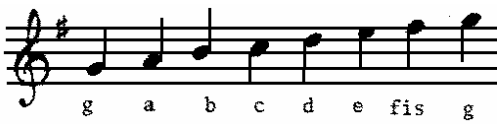
1 - 1 - ½ - 1 - 1 - 1 - ½

Tangga nada ber-kruis dan ber-mol ini merupakan susunan tangga nada baru, yang nada-nadanya berasal dari tangga nada diatonis C mayor. Tangga nada ini juga terdiri dari mayor dan minor.

Bagi tangga nada berkruis, nada dasar baru adalah nada kwint murni dari pada tangga nada lama (kwint murni adalah nada ke-5 dari nada dasar). Jadi kwint murni bagi tangga nada naturel (C) ialah nada G.



Gambar 2.31. G Mayor

Untuk memenuhi jarak yang sudah ditentukan seperti pada Gambar 2.31, maka nada F perlu dinaikkan ½ laras. Jadi nada ke-7 (F) harus berubah menjadi Fis. Oleh karena semua F harus berubah menjadi Fis, maka tanda kruis (#) ditempatkan tepat pada nada F pada permulaan suatu lagu, sehingga tanda # tersebut berfungsi merubah semua F menjadi Fis. Hal ini dijelaskan pada Gambar 2.32.

Gambar 2.32. G Mayor pada Nada Birama

Tangga nada G hanyalah salah satu dari tangga nada berkruis lainnya. Nada G adalah nada ke-5 (nada kwint) dalam deretan tangga nada C. Bila nada kwint dari suatu tangga nada dijadikan nada dasar baru, akan diperoleh tangga nada baru dengan tanda muka tambah satu kruis dari tangga nada yang lama. Kruis tambahan itu selalu jatuh pada nada ke-7 dari tangga nada baru itu. Jadi yang perlu diperhatikan untuk membentuk tangga nada baru (tangga nada berkruis) ialah :

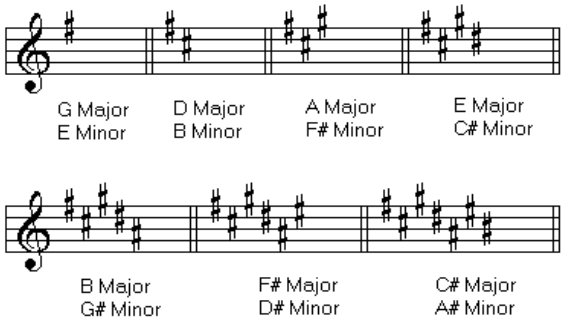
1. Jarak-jarak tetap : 1 – 1 – ½ – 1 – 1 – 1 – ½.
2. Nada dasar tangga nada baru ialah nada ke-5 (nada kwint) dari tangga nada lama.
3. Tanda kruis baru selalu jatuh pada nada ke-7 dari tangga nada baru.

Dalam praktek penggunaan tangga nada mayor (Gambar 2.33) tersebut biasanya hanya sampai dengan tangga nada Cis (7#), karena tangga nada Gis (8#) dan seterusnya lebih mudah ditulis dengan tangga nada bermol.

Tangga nada	Nada ke:	1	2	3	4	5	6	7	8
	Jarak-2	1	1	½	1	1	1	½	
C (nature1)		c	d	e	f	g	a	b	c
G (1 #)		g	a	b	c	d	e	fis	g
D (2 #)		d	e	fis	g	a	b	cis	d
A (3 #)		a	b	cis	d	e	fis	gis	a
E (4 #)		e	fis	gis	a	b	cis	dis	e
B (5 #)		b	cis	dis	e	fis	gis	ais	b
Fis (6 #)		fis	gis	ais	b	cis	dis	eis	fis
Cis (7 #)		cis	dis	eis	fis	gis	ais	bis	cis
Gis (8 #)		gis	ais	bis	cis	dis	cis	fisis	gis
Dis (9 #)		dis	eis	fisis	gis	ais	bis	cisis	dis
Ais (10#)		ais	bis	cisis	dis	eis	fisis	gisis	ais
Eis (11#)		eis	fisis	gisis	ais	bis	cisis	disis	eis
Bis (12#)		bis	cisis	disis	eis	fisis	gisis	aisis	bis

Gambar 2.33. Tangga Nada Mayor Kruis

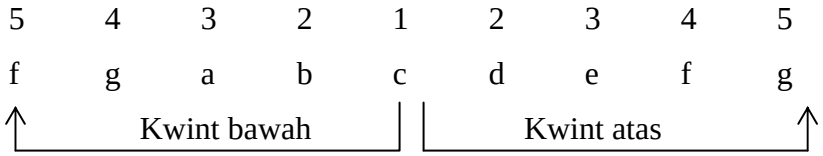
Tempat (letak) tanda kruis sebagai tanda muka (tanda awal) dari tangga nada G s/d tangga nada Cis dapat dilihat pada Gambar 2.34.



Gambar 2.34. Letak Kruis sebagai Tanda Awal

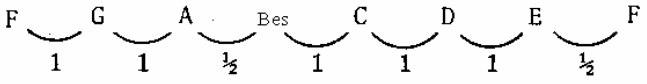
2.1.9. Tangga Nada Bermol

Disebut tangga nada bermol, karena di antara nada-nada yang tersusun dalam tangga nada tersebut mendapat tanda mol (b) yaitu diturunkan setengah jarak. Apabila dalam tangga nada berkruis, nada dasar dari tangga nada baru ialah nada kwint atas dari tangga nada lama, untuk memperoleh nada dasar dari tangga nada bermol agak berbeda, yakni nada kwint bawah dari tangga nada lama.



Gambar 2.35. Kwint Atas dan Kwint Bawah

Dari Gambar 2.35 dapat dilihat bahwa nada f adalah nada kwint bawah dari nada dasar C sekaligus nada kwart dari nada dasar C. Jadi tangga nada satu mol (1b) nada dasarnya ialah F. Dengan menggunakan jarak : 1 – 1 – ½ – 1 – 1 – 1 – ½, maka susunan tangga nada F ialah :



Gambar 2.36. F Mayor

Untuk memenuhi ketentuan jarak seperti pada Gambar 2.36, semua nada B harus berubah menjadi Bes, Agar tidak selalu menuliskan tanda mol (b) pada

nada B, maka tanda mol (b) ditulis sebagai tanda muka. Tangga nada minor dapat dilihat pada Gambar 2.37.

Beberapa hal yang perlu diperhatikan untuk menyusun tangga nada baru (tangga nada bermol), ialah:

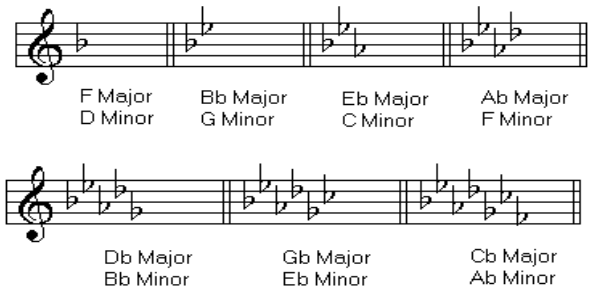
1. Jarak-jarak tetap : 1,1, ½, 1, 1, 1, ½, 1.
2. Nada dasar baru ialah kwint (nada ke-5) bawah atau kwart (nada ke-4) dari tangga nada lama.

Seperti pada tangga nada berkruis, pada praktek yang biasanya yang digunakan hanya sampai dengan 7 b.

Tangga nada	nada ke: / jarak-2	1 (1)	2 (1)	3 (½)	4 (1)	5 (1)	6 (1)	7 (1)	8 (½)
C (natural)		c	d	e	f	g	a	b	c
F (1 ♭)		f	g	a	bes	c	d	e	f
Bes (2 ♭)		bes	c	d	es	f	g	a	bes
Es (3 ♭)		es	f	g	as	bes	c	d	es
As (4 ♭)		as	bes	c	des	es	f	g	as
Des (5 ♭)		des	es	f	ges	as	bes	c	des
Ges (6 ♭)		ges	as	bes	ces	des	es	f	ges
Ces (7 ♭)		ces	des	es	fes	ges	as	bes	ces
Fes (8 ♭)		fes	ges	as	beses	ces	des	es	fes
Beses (9 ♭)		beses	ces	des	eses	fes	ges	as	beses
Eses (10 ♭)		eses	fes	ges	ases	beses	ces	des	eses
Ases (11 ♭)		ases	beses	ces	deses	eses	fes	ges	ases
Deses (12 ♭)		deses	eses	fes	geses	ases	beses	ces	deses

Gambar 2.37. Tangga Nada Mayor Mol

Tempat tanda mol sebagai tanda awal dapat dilihat pada Gambar 2.38.



Gambar 2.38. Letak Mol sebagai Tanda Awal

Bila tangga nada berkruis dan bermol kita bandingkan, maka akan diperoleh hasil, sbb. :

			1	2	3	4	5	6	7
a.	Letaknya tanda :	#	f	c	g	d	a	e	b
		b	b	e	a	d	g	c	f
b.	Nada dasarnya :	#	G	D	A	E	B	Fis	Cis
		b	F	Bes	Es	As	Des	Ges	Ces

Dari kedua hal tersebut dapat ditarik kesimpulan :

1. Letak tanda # berlawanan dengan letak tanda b.
2. Tanda # terakhir setiap tangga nada berkruis selalu merupakan nada si dari tangga nada itu. Jadi nada do (nada dasar) selalu terletak di atas # terakhir.
3. Tanda b terakhir dari setiap tangga nada bermol selalu merupakan nada fa dari tangga nada itu. Jadi nada do (nada dasarnya) merupakan kwint atas dari b terakhir itu atau kwart bawah dari b terakhir itu.

2.1.10. Tanda Tempo

Masing-masing lagu berbeda watak dan maksudnya. Ada lagu-lagu yang lambat, sedang dan ada juga yang cepat. Tanda tempo diberikan untuk membantu menetapkan kecepatan seluruh lagu, sesuai kehendak penggubah. Tanda tempo berupa suatu kata dari bahasa latin yang biasanya ditulis pada kiri atas sebuah gubahan. Berikut ini ialah beberapa contoh tanda tempo :

a. Largo = sangat perlahan.
b. Adagio = perlahan.
c. Andante = dalam keadaan bergerak.
d. Presto = sangat cepat.
e. Allegro = cepat.

Arti dari tempo utama ini dapat berubah artinya bila diberi akhiran issimo = lebih dari dan etto = lebih lambat dari atau kurang dari. Misalnya :

a. Prestissimo = lebih cepat dari presto
b. Larghisimo = lebih lambat dari largo
c. Allegretto = lebih lambat dari allegro

2.2. *Software* Aplikasi Penunjang

2.2.1. *Macromedia Flash MX*

Macromedia Flash adalah *software* yang dipakai luas oleh para profesional web karena kemampuannya yang mengagumkan dalam menampilkan multimedia, gabungan antara grafis, animasi, suara, serta interaktivitas bagi pengguna Internet. *Software* ini berbasis animasi vektor yang dapat digunakan untuk menghasilkan animasi web, presentasi, *game* atau bahkan film.

Oleh karena berbasis vektor, maka ketika dilakukan perubahan pada sebuah *vector graphic*, sebenarnya yang dirubah adalah properti dari garis dan kurva yang membentuknya. Sehingga *vector graphic* dapat dipindahkan, diubah ukuran, diubah bentuk, bahkan diubah warnanya tersebut tanpa harus mengurangi kualitas tampilannya.

Berbeda dengan *bitmap graphic*, mengubah sebuah *bitmap graphic* dapat mengubah kualitas tampilannya karena ketika mengubah *bitmap graphic*, sebenarnya sama dengan sedang memodifikasi titik-titik penyusunnya, bukan garis dan kurvanya.

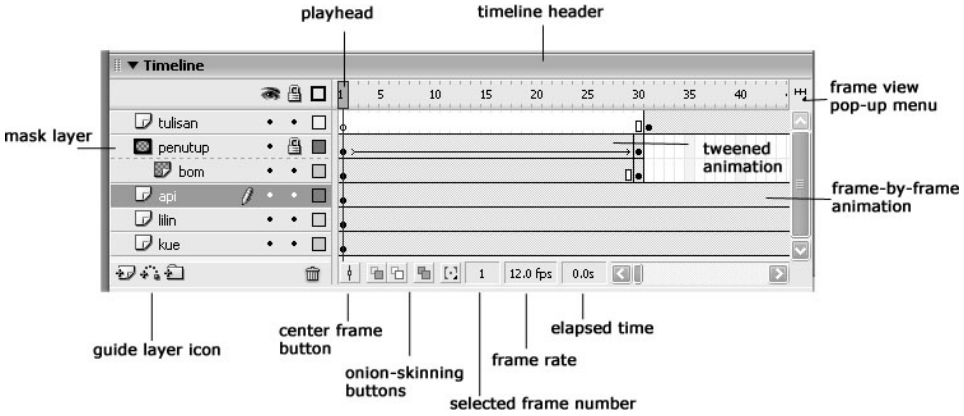
Macromedia Flash MX adalah versi terbaru dari *Macromedia Flash* sebelumnya (5.0). *Flash MX* merupakan perubahan besar-besaran dari segi UI (*User Interface*) sampai *ActionScript*.

2.2.2. Fitur-fitur dalam *Flash MX*

Pada sub bab berikut ini dijelaskan beberapa perubahan ataupun fitur-fitur baru yang ada dalam *Flash MX* sekaligus perbandingannya dengan *Flash 5.0.*

2.2.2.1.*Stage* dan *Timeline*

Layaknya sebuah film, *flash movies* membagi durasi menjadi *frame-frame*. Timeline adalah koordinat waktu kapan sebuah animasi dimainkan. Stage adalah tempat yang disediakan untuk mengisi setiap *frame* tersebut. *Layer* berperan seperti lembaran-lembaran yang ditumpuk.



Gambar 2.39. *Timeline* dan S*tage*

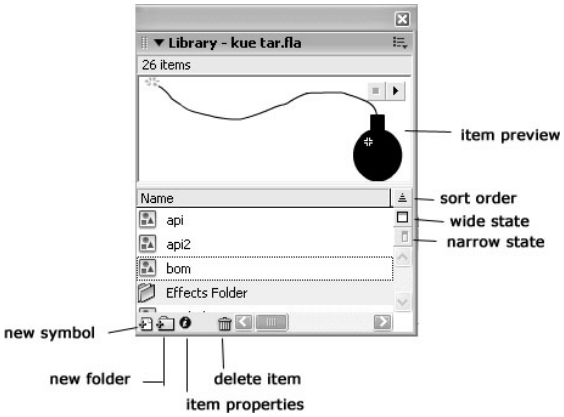
2.2.2.2.*Symbol* dan *Instances*

Symbol dapat berupa grafik, tombol, *movie clip, sound files*, atau huruf adalah elemen yang dapat digunakan bersama dengan *movie*. Setelah sebuah *symbol* dibuat, *symbol* tersebut disimpan di *library*. Pada saat *symbol* digunakan di *stage*, maka dapat dibuat *instance symbol* tersebut.

Symbol mengurangi ukuran *file*. Tidak berpengaruh seberapa banyak *instance* yang dibuat dari *symbol* tersebut, *Flash* hanya menyimpan *symbol* untuk satu kali saja.

2.2.2.3.*The Library Window*

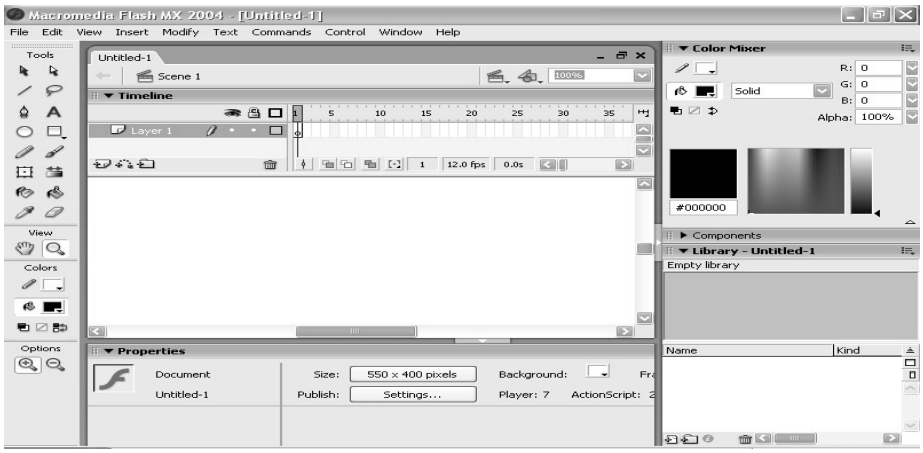
Library window adalah tempat dimana tersimpan semua *symbol* yang dibuat, termasuk juga *imported file, sound files, bitmap graphic.*



Gambar 2.40. *Library Window*

2.2.2.4.*User Interface*

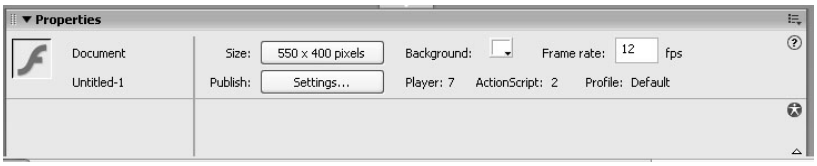
Setiap *user interface* dalam keluarga *MX* baik itu *Flash MX*, *Dreamweaver MX*, ataupun *Fireworks MX* memiliki kesamaan.



Gambar 2.41. *User Iinterface Flash MX*

Perubahan yang terjadi antara lain adanya *Floating Panel* yang diletakkan semuanya di sebelah kanan. Panel di sebelah kanan tersebut dapat dibuka dan ditutup. Hal ini memudahkan untuk mencari sesuatu. Sedangkan *Action Panel* dan *Properties Panel* diletakkan di bagian bawah.

Properties Panel ini sering juga disebut sebagai *Property Inspector*. *Properties Panel* ini merupakan pengganti *Frame Panel*, *Info Panel*, *Sound Panel*, dan lain-lain.



Gambar 2.42. *Properties Panel*

2.2.2.5.Komponen

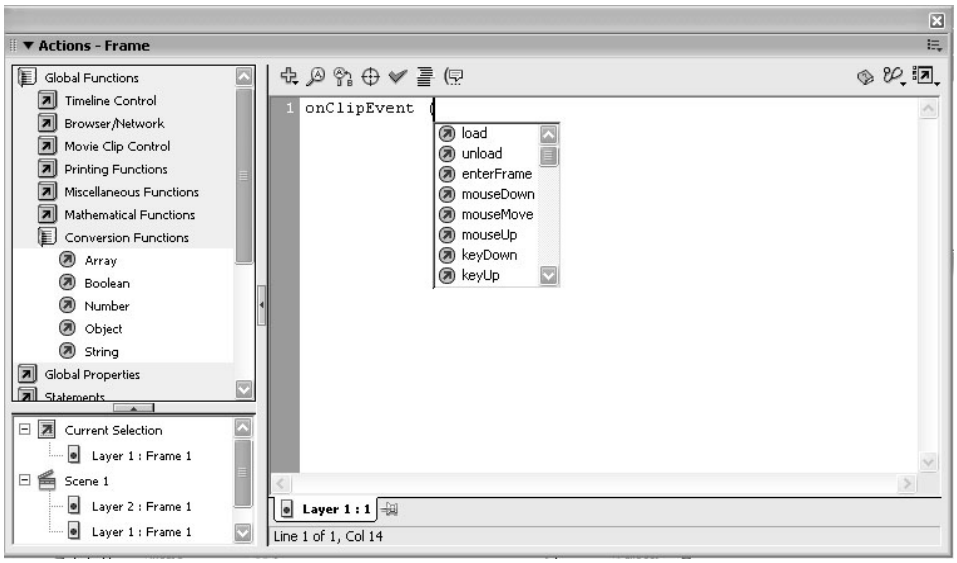
Ada beberapa fitur yang ditambahkan antara lain *UI Component,* yang juga menggantikan *Smart Clips.* Dengan *UI Component* ini, *user* dapat dengan mudah membuat berbagai componen seperti *scroll bar*, tidak perlu menggunakan *ActionScript.*

Gambar 2.43. *UI Component*

2.2.2.6.*ActionScript*

Action Panel memiliki banyak peningkatan dibandingkan dengan sebelumnya. Misalnya dengan penambahan warna, indexing *Action* berdasar abjad, penambahan angka baris dan *Code Hint.*

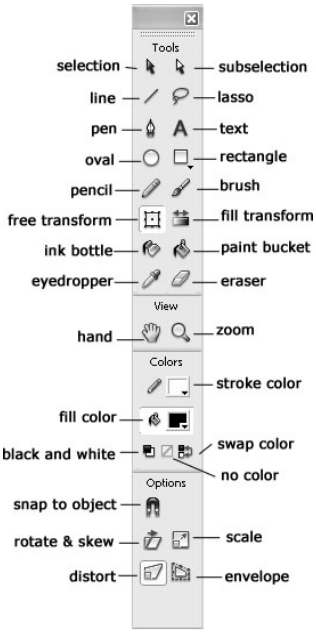
Pembagian kategori *ActionScript* di *Action Panel* memiliki banyak perubahan. *ActionScript* yang sekarang lebih bersifat *Object Oriented*, misalnya dengan adanya penambahan *object*, *event* dan sekarang hampir setiap elemen yang ada di *stage* bisa menjadi *instance object*. Misalnya tombol bisa memiliki nama *instance*.



Gambar 2.44. *Action Panel*

2.2.2.7.*Toolbar*

Toolbar tidak banyak mengalami perubahan. Salah satunya yang cukup penting adalah pada *Free Transform Tool* yang sekarang menjadi *icon* tersendiri. *Tool* ini memiliki beberapa pilihan lain seperti *Distort* dan *Envelope*.



Gambar 2.45. *Toolbar*

Distort Tool berfungsi seperti *Perspective Tool* di *Photoshop*. Sedangkan jika *Envelope Tool* diaktifkan, obyek seakan-akan memiliki "sangkar" yang terdiri atas titik-titik kontrol. Titik-titik ini memiliki *handel bezier* yang bisa digeser-geser.

Fitur menarik lainnya adalah bisa dilakukan *mask* pada *movie clip* dan sebaliknya, *movie clip* dapat menjadi *mask.* Selain itu, fitur lain adalah *Distribute to Layers*. Dengan menggunakan fitur ini, sebuah obyek dapat dipecah menjadi berbagai bagian yang masing-masing bagian dapat ditempatkan dalam satu *layer*. Hal ini menambah efisiensi dalam bekerja.

Fasilitas lain adalah *Snap to Pixel*. Dengan mengaktifkan fasilitas ini, menggambar dapat dilakukan dengan lebih presisi atau tepat. *Timeline* juga memiliki banyak penambahan. Seperti *Layer Fólder* untuk memudahkan mengorganisasikan *layer.*

2.2.3. Pemrograman Menggunakan *Flash MX*

ActionScript adalah bahasa pemrograman di *Flash.* Ada banyak kesamaan antara *ActionScript* dengan *Javascript*.

2.2.3.1. Fungsi *ActionScript*

ActionScript memiliki beberapa fungsi atau peran yang dapat membantu seseorang merancang situs atau *movie flash*.

1. Membuat sistem navigasi situs

   Hal ini berguna bagi seorang perancang web, yakni bagaimana *ActionScript* dapat mengantar *user* menjelajahi situsnya. Dengan demikian situs tersebut tidak hanya menjadi "film iklan" belaka. Kecuali memang menginginkan demikian (dalam kasus film kartun menggunakan *flash*).
2. Menambahkan interaktivitas dengan *user*

   Dengan *ActionScript* seseorang dapat berinteraksi dengan setiap elemen di dalam *movie flash*, baik itu tombol, *movie clip*, maupun teks.
3. Membuat situs yang dinamis

   ActionScript dapat digabungkan dengan berbagai bahasa lain, seperti *Javascript, ASP, PHP, CGI, MySQL, XML (Extensible Markup Language)* untuk membuat berbagai aplikasi yang sulit atau tidak dapat dibuat hanya dengan *ActionScript* saja. Misalnya untuk membuat sistem *database* sehingga situs *flash* dapat menjadi lebih dinamis.
4. Mempertahankan *user* untuk kembali mengunjungi situs

   ActionScript dapat membuat situs *flash* yang menarik, interaktif, yang menambah kualitas *user experience*.

Pada prinsipnya, *ActionScript* hanya merupakan alat bantu yang mempermudah seorang perancang atau pembuat *movie*.

2.2.3.2. Obyek dan *Class*

Obyek adalah tipe data seperti suara, gambar, teks yang digunakan untuk mengontrol *movie.* Semua obyek merupakan bagian dari suatu *class*. *Flash* memiliki variasi bentuk obyek yang sangat beragam dari gambar yang kelihatan,

sampai dengan yang abstrak seperti tanggal, data atau deteksi *input* dari *mouse*. Sebelum menggunakan suatu obyek, obyek tersebut terlebih dahulu harus diberi nama.

Di samping obyek yang dibuat sendiri, *flash* juga memiliki *predefined class* yang berisi obyek-obyek yang bisa dipakai dalam *movie*. Obyek-obyek ini berada dalam kategori *objects* di dalam *ActionScript*, misalnya *MovieClip, Color, Sound,* dan lain-lain. Obyek inipun harus diberi nama. Proses pemberian nama ini disebut dengan *intantiating.*

2.2.3.3. Metode dan Properti

Setiap obyek selain berbeda dalam nama, juga berbeda dalam karakteristik. Di dalam *ActionScript* perbedaan-perbedaan ini dinamakan *properties*. Misalnya kelas *movie clip* memiliki *property* seperti *_height, _width* yang mengukur dimensi danorientasi dari objeck *movie clip* tersebut.

Obyek juga melakukan suatu tugas. Pekerjaan ini dalam *ActionScript* dinamakan *method*. Misalnya kelas *sound* memiliki *method setvolume* yang dapat membuat suara lebih keras atau lebih lembut.

2.2.3.4. Struktur Bahasa *ActionScript*

Seperti halnya dalam berbahasa, diperlukan kata benda, kata sifat, dan kata kerja dalam susunan yang dimengerti orang lain. Di sini yang disusun adalah obyek, properti, dan metode sampai membentuk *statement* yang dimengerti *flash*.

Flash menggunakan struktur *Dot Syntax*, yang menggantikan *Slash Syntax* pada versi terdahulu. *Dot* atau titik (.) digunakan untuk menunjukkan properti atau metode yang terkait dengan obyek. Contoh :

```
Balerina.berat = 50
Flinstone.berat = 65
```

Statement tersebut menetapkan nilai 50 untuk properti berat badan balerina dan nilai 65 untuk berat badan flinstone.

```
Balerina.pakaian.warna = putih
```

Statement tersebut menjelaskan putih sebagai properti warna pada obyek pakaian yang terkait dengan obyek balerina.

```
Balerina.menari ()
```

Statement tersebut menjelaskan metode menari yang diterapkan pada obyek balerina. Tanda kurung buka dan kurung tutup memberi tanda kalau menari adalah metode.

```
Balerina.menari (cepat)
Balerina.menari (lambat)
```

Metode seringkali memiliki *argument* atau parameter, yang menerangkan bagaimana metode dijalankan.

2.2.3.5. Tanda Baca dalam *ActionScript*

Seperti halnya dalam bahasa tulisan yang memiliki tanda baca untuk mengakhiri kalimat, atau mengelompokkan berbagai kalimat menjadi satu paragraf, *flash* juga memiliki tanda baca seperti itu.

2.2.3.5.1. *Curly braces* dan *semicolon*

Curly braces atau kurung kurawal ({ }) digunakan untuk melakukan pengelompokkan, sedangkan *semicolon* atau titik koma (;) digunakan untuk mengakhiri suatu *statement*.

```
On (release) {
      stopAllSounds () ;
      play ();
}
```

Dalam *script* di atas, metode *stopAllSounds* dan *play* akan dieksekusi saat tombol mouse dilepaskan karena berada dalam satu kelompok (seperti paragraf).

Semicolon berfungsi seperti tanda akhir kalimat yang setelah kalimat pertama berakhir kemudian beralih ke kalimat berikutnya. Dalam *ActionScript*, setiap *statement* dieksekusi dari atas ke bawah secara berurutan. Jadi seperti contoh di atas, setelah metode *stopAllSounds* selesai baru kemudian *play*.

2.2.3.5.2. *Comment*

Comment adalah keterangan yang ditambahkan di dalam *script* untuk menjelaskan apa arti kode tersebut. *Comment* dapat membantu memahami *script* apabila hendak me-*review* kembali kode tersebut untuk perbaikan. *Comment* ditandai dengan *double slash* (//).

```
on  (release) {
      // matikan semua suara
      stopAllSounds ();
      // mainkan
      play ();
}
```

2.2.3.6. Variabel dan Tipe Data

Variabel adalah suatu "kontainer" yang mengandung suatu informasi tertentu. Kontainer atau wadahnya selalu sama tetapi informasi yang dapat terkandung di dalam suatu variabel dapat berubah-ubah bergantung pada tipe informasi atau data. Ada empat macam tipe data, yaitu :

Tabel 2.1. Tipe Data

Tipe Data	Keterangan	Contoh
Number	Angka tertentu	*myScore* = 100
String	Kumpulan berbagai angka, huruf, atau simbol. *String* selalu berada di dalam tanda petik	*myName* = "abcd"
Boolean	Nilai yang berisi *true* atau *false*. Dapat juga dipakai angka 1 untuk *true* atau 0 untuk *false*.	*radioButton = true* *radio Button = false*
Object	Nama obyek yang dibuat berdasar proses *initiate*.	*myColor = new Color* (movieklip)

Cara menuliskan nama variabel, yang umum digunakan misalnya seperti *myColor, playerscore*. Sebaiknya, setiap nama variabel spesifik terhadap informasinya dan tidak terdapat *space* atau jarak di antara kata dan membuat huruf besar setiap kata kecuali kata pertama.

Selain itu terdapat beberapa kata yang tidak dapat digunakan karena kata-kata tersebut digunakan untuk berbagai fungsi atau *keyword* di *ActionScript*, seperti *break, for, new, var, continue, function, return, void, delete, if , this, while, else, in, typeof, with, true,* dan *false*.

2.2.3.7. *String* dan Ekspresi

Ketika menggunakan *action set variable* dalam *Normal Mode* untuk memasukkan informasi ke dalam variabel, terdapat kotak *Expression checkbox*

yang bisa diisi atau dikosongkan. Membiarkan kotak *expression* terisi mengakibatkan tanda petik hilang dan mengakibatkan tipe variabel menjadi *expression* daripada *string*.

Expression adalah suatu *statement* yang harus dievaluasi sebelum *flash* dapat menentukan nilainya. Sedangkan *string* adalah *statement* yang digunakan *flash* apa adanya dan dipandang hanya sebagai kumpulan karakter.

2.2.3.8. Operator

Suatu variabel dapat dimodifikasi dengan menggunakan operator. Variabel dapat diganti nilainya dengan informasi terbaru dari kondisi *movie clip*, mengganti suatu nilai dengan *true* atau *false*, atau dengan menambah, mengurangi, mengalikan. Tugas utk memodifikasi informasi ini merupakan kerja dari operator.

Di bawah ini adalah beberapa macam operator yang biasa digunakan:

1. *Arithmetic* Operator

   Operator ini terdiri dari simbol-simbol yang biasa digunakan dalam matematika

Tabel 2.2. *Arithmetic* Operator

Simbol	Nama	Keterangan
-	*Minus*	Pengurangan
+	*Addition*	Penambahan
%	*Modulo*	Mencari nilai sisa dari pembagian antara variabel 1 dan variabel 2. Misalnya 12%5 hasilnya adalah 2
*	*Multiplication*	Perkalian
/	*Division*	Pembagian

2. *Assignment* Operator

   Operator ini adalah operator yang menetapkan suatu nilai untuk suatu variabel.

Tabel 2.3. *Assignment* Operator

Simbol	Nama	Keterangan
-=	*Minus Assignment*	Mengurangi suatu nilai pada variabel x -= 5 sama nilainya dengan x=x-5
%=	*Modulo Assigment*	Menambahkan nilai modulo pada variabel

		x%=Y sama nilainya dengan x=x%Y
*=	*Addition Assignment*	Mengalikan suatu nilai pada variabel x*=5 sama nilainya dengan x=x*5
/=	*Division Assignment*	Membagi suatu nilai pada variabel x/=5 sama nilainya dengan x=x/5
+=	*Addition Assignment*	Menambah suatu nilai pada variabel x+=5 sama nilainya dengan x=x+5
=	*Assignment*	Memberikan suatu nilai pada variabel

3. *Bitwise* Operator

   Digunakan untuk menghitung dua *expression* atau variabel menggunakan format *binary* (0 dan 1).

4. *Comparison* Operator

   Operator ini membandingkan kedua *expression* di antara simbol operator dan mengeluarkan suatu nilai *true* atau *false*.

Tabel 2.4. *Comparison* Operator

Simbol	Nama	Keterangan
!=	*Inequality*	Mengecek apakah nilai suatu variabel tidak sama dengan variabel yang lain
!==	*Strict Inequality*	Mengecek apakah nilai dan tipe data suatu variabel tidak sama dengan variabel yang lain
<	*Less Than*	Mengecek apakah suatu variabel kurang dari variabel lain
<=	*Less Than or Equal To*	Mengecek apakah suatu variabel kurang atau sama dengan variabel lain
==	*Equality*	Mengecek dua variabel apakah memiliki kesamaan nilai
===	*Strict Equality*	Mengecek dua variabel apakah memiliki kesamaan nilai dan tipe data
>	*Greater Than*	Mengecek apakah suatu variabel lebih dari variabel yang lain
>=	*Greater Than or Equal To*	Mengecek apakah suatu variabel lebih atau sama dengan variabel yang lain

5. *Logical* Operator

   Operator ini digunakan sebagai operator *boolean*, misalnya *NOR, AND* dan *OR*.

Tabel 2.5. *Logical* Operator

Simbol	Nama	Keterangan
!	*NOT*	Membalik nilai dari suatu variabel Misalnya (x) nilainya *true*, ¡(x) menjadi *false*
&&	*AND*	Nilai expression dua atau lebih variabel adalah *true* jika semua variabel *true*
||	*OR*	Nilai *expression* dua atau lebih variabel adalah *true* jika salah satu variabel *true*

6. *Miscelanous* Operator

   Operator ini digunakan untuk mengurangi, menambah atau memeriksa suatu kondisi.

Tabel 2.6. *Miscelanous* Operator

Simbol	Nama	Keterangan
--	*Decrement*	Mengurangi nilai dengan 1 (*decrement*) x—sama nilainya dengan x=x-1
?:	*Conditional*	Apabila var 1 nilainya *true*, maka nilai yang dihasilkan sama dengan variabel 2, jika tidak nilainya sama dengan variabel 3 Formatnya adalah var 1 ?var 2 : var 3 Misalnya : x = 3 y = 7 z= (x<4) ? y :6, maka nilai z=7
++	*Increment*	Menambah nilai dengan 1 (*increment*) x++ sama nilainya dengan x=x+1

2.2.3.9. Kategori *ActionScript*

Di dalam *Action panel*, *action* dibagi dalam berbagai kategori. Pembagian tersebut dapat dilihat pada Tabel 2.7.

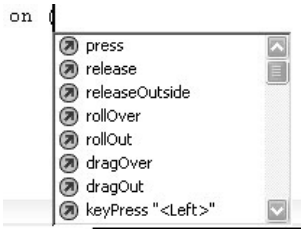
Tabel 2.7. Pembagian Kategori *Action*

No	Kategori	Keterangan
1	*Actions*	Kategori ini menampung berbagai *action* yang sering digunakan untuk *flash movie*
2	*Operators*	Kategori ini berisi simbol yang digunakan misalnya untuk

		operasi logika, matematika, seperti tambah, kurang, kali dan bagi
3	*Functions*	Kategori ini meliputi berbagai *predefined function* yang sudah tersedia di *flash. Function* berisi *actions* yang dapat menerima data tertentu yang kemudian menghasilkan informasi yang dapat kita gunakan
4	*Constant*	Kategori ini memuat berbagai nilai konstan suatu variabel, misalnya *true* dan *false*
5	*Properties*	Kategori ini berisi properti obyek yang dapat dimodifikasi. Sebagian besar properti ini digunakan untuk obyek *movie clip*
6	*Objects*	*Flash* memiliki berbagai kelas obyek yang sudah didefinisikan (*predefined class*). Kelas-kelas ini berada dalam kategori *object* di *ActionScript.* *Flash MX* memiliki banyak obyek baru seperti *button, stage*
7	*Deprecated*	Kategori ini meliputi berbagai *action* yang sudah jarang dipakai dan dianjurkan oleh *macromedia* untuk tidak dipakai lagi. *Action deprecated* ini sudah digantikan oleh *action* lain
8	*Flash UI Component*	Kategori ini memuat berbagai *action* yang berkaitan dengan *UI Component*
9	*Index*	Kategori ini merupakan daftar *action* yang disusun berdasar abjad.

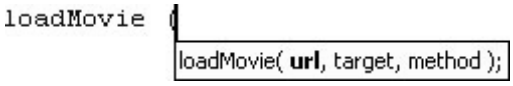
2.2.3.10. Menggunakan *Code Hint*

Code hint adalah alat bantu untuk menuliskan *ActionScript* di *Action Panel*. Biasanya *Code hint* digunakan pada *Expert Mode*. Secara *default Code hint* ini aktif. Penggunaan *Code* hint sangat membantu karena kelupaan akan *action, event* sering terjadi pada saat menulis *script*.



Gambar 2.46. *Code Hint*

Code dapat tampil dalam bentuk lain, yaitu membantu mengingatkan tentang sintaks dan pemakain *action.*



Gambar 2.47. Bentuk Lain dari *Code Hint*

Code hint juga dapat membantu menuliskan *dot* sintaks untuk *instance* tertentu, tapi syaratnya *instance* tersebut harus memiliki akhiran khusus sesuai jenis *instance*-nya. Misalnya *instace movie clip* harus memiliki nama dengan akhiran _mc.

Tidak hanya untuk *movie clip*, hampir semua *instance* obyek bisa ditampilkan *code hint action*-nya, tapi nama *instance* obyek tersebut harus memiliki akhiran tertentu pula. Beberapa kelas obyek dan akhiran yang diperlukan dapat dilihat pada Tabel 2.8.

Tabel 2.8. Nama *Instance* dan akhirannya

Akhiran	Obyek
_mc	*Movie Clip*
_array	*Array*
_str	*String*
_btn	*Button*
_txt	*Text Field*
_fmt	*Text Format*
_date	*Date*
_sound	*Sound*
_xml	*XML*
_color	*Color*
_video	*Video*

2.2.3.11. *Debugging*

Tanpa disadari seringkali dalam menuliskan sintaks *ActionScript* terjadi banyak kesalahan. Untuk itu perlu dilakukan proses *debugging*. *Debugging* adalah proses untuk menganalisa, menemukan kesalahan, dan kemudian memperbaiki kesalahan tersebut. *Flash MX* menyediakan *tool* untuk melakukan proses ini. Kelebihan *Debugger* adalah dapat memeriksa kesalahan *step-by-step* dengan menggunakan *breakpoint.*

2.3. XML (*Extensible Markup Language*)

XML banyak digunakan sebagai format penyimpanan data. XML sendiri bukan merupakan bahasa pemrograman. XML merupakan *markup language*, sama seperti HTML (*Hyper Text Markup Language*), namun XML didesain untuk tujuan yang berbeda. XML didesain untuk mendeskripsikan data dan berfokus pada data yang dibawa, sedangkan HTML didesain untuk memunculkan data dan berfokus pada bagaimana tampilan dari data. Jadi, HTML lebih berorientasi pada bagaimana menampilkan informasi, sedangkan XML berorientasi pada bagaimana mendeskripsikan informasi.

XML memiliki beberapa keunggulan, seperti :

- XML bersifat bebas. XML memperbolehkan *developer* membuat *tag*-nya sendiri dan struktur dokumennya.
- XML bisa digunakan dalam berbagai sistem operasi.
- XML mudah dimengerti. Dengan menggunakan *tag* yang disesuaikan dengan bahasa *developer*, maka bisa digunakan bahasa yang mudah dimengerti.
- XML bisa dibuat dengan *text editor* biasa seperti *notepad*.

2.3.1. Sintaks XML

Aturan sintaks XML sangat sederhana dan sangat ketat. Semua elemen XML harus mempunyai *tag* penutup. Menggunakan XML tanpa *tag* penutup bersifat ilegal. *Tag* XML bersifat *case sensitive*, tidak seperti HTML. Dalam XML, *tag* <from> dengan <From> berbeda artinya. *Tag* pembuka dan penutup harus dituliskan dengan tulisan yang sama. Lihat contoh berikut ini :

```
<to>Tove</to>
<from>Jani</from>
<heading>Reminder</heading>
<body>Don’t forget me this weekend</body>
```

Semua dokumen XML harus mempunyai sebuah elemen *root* dan harus punya satu pasang *tag* yang mendefinisikan elemen *root.* Semua elemen harus terdapat dalam elemen *root* berikutnya. Semua elemen dapat mempunyai sub elemen (*child* elemen). Sub elemen harus mempunyai sarang yang benar dalam elemen atasnya. Lihat contoh di bawah ini :

```
<root>
      <child>
      <subchild> ..... </subchild>
      </child>
</root>
```

2.4. Psikologi Pendidikan

Berdasarkan buku “Materi Dasar Pendidikan Program Akta Mengajar V” yang dibuat oleh Departemen Pendidikan dan Kebudayaan tahun 1981, dikatakan bahwa dalam hidup, manusia perlu selalu belajar. Dengan senantiasa belajar, manusia dapat mengembangkan dirinya. Belajar dapat dilakukan dimana saja, di rumah, kantor, jalan, bahkan di kamar mandi, dan tentu saja di lembaga pendidikan formal.

Oleh karena belajar merupakan suatu hal yang penting, maka banyak ahli-ahli psikologi yang memikirkan untuk menjawab persoalan-persoalan mengenai belajar. Konsep, prinsip dan informasi lain yang telah tersusun sangat berguna untuk memahami, merangsang dan memberi arah dalam kegiatan belajar.

2.4.1. Persoalan-Persoalan Pokok dalam Belajar

Seseorang dapat dikatakan telah belajar apabila terjadi perubahan pada orang tersebut. Misalnya, dari tidak bisa berbahasa Inggris menjadi mahir berbahasa Inggris. Namun, ada juga perubahan yang terjadi pada diri seseorang bukan karena proses belajar, melainkan karena pendewasaan diri. Misalnya, bayi yang tidak dapat berjalan, menjadi bisa berjalan dan berlari. Selain dua hal tersebut, masih ada lagi satu perubahan yang tidak disebabkan karena proses belajar. Perubahan ini adalah perubahan yang terdapat pada seseorang dan sifatnya sangat singkat, kemudian hilang lagi. Misalnya, secara kebetulan, seseorang dapat memperbaiki radio atau memecahkan sebuah persoalan, tapi ketika diminta untuk mengerjakan sekali lagi, tidak dapat. Satu hal perlu diingat, perubahan sebagai hasil belajar pada seseorang adalah hasil usaha orang yang bersangkutan.

Dari uraian di atas, dapat disimpulkan bahwa kegiatan belajar memiliki ciri-ciri sebagai berikut :

- Belajar adalah aktivitas yang menghasilkan perubahan pada diri seseorang.
- Perubahan tersebut ditandai dengan didapatnya kemampuan baru dan sifatnya relatif lama.
- Perubahan dapat terjadi karena ada usaha.

Berhasil atau tidaknya proses belajar seseorang dapat dilihat dari hasil atau *output* yang diperoleh, yaitu apa yang dapat dilakukan orang tersebut. Tentang sumber sebab atau *input*, ada sebagian yang dapat dilihat atau ditelaah, tapi ada juga yang tidak dapat dilihat.

Persoalan dalam belajar dapat diidentifikasikan sebagai berikut :

- Persoalan *input*, yaitu faktor-faktor yang mempengaruhi belajar.
- Persoalan proses, yaitu mengenai bagaimana belajar itu berlangsung dan prinsip apa saja yang mempengaruhi proses belajar itu.
- Persoalan *output*, yaitu hasil yang dicapai apakah sudah sesuai dengan tujuan belajar.

2.4.2. Faktor-Faktor yang Mempengaruhi Proses dan Hasil Belajar

Secara garis besar, ada empat faktor yang mempengaruhi proses dan hasil belajar, yakni :

- Bahan atau hal yang harus dipelajari.

  Belajar keterampilan dan belajar untuk memecahkan suatu soal tidak sama. Perbedaan ini melahirkan konsep yang berbeda, misalnya apa inti belajar dan bagaimana peranan ulangan atau latihan dalam belajar.
- Faktor lingkungan.

  Lingkungan yang dimaksud meliputi lingkungan secara alami dan sosial. Lingkungan alami seperti keadaan suhu, kelembaban udara mempengaruhi proses belajar. Di Indonesia misalnya, saat yang paling baik untuk belajar adalah di pagi hari, ketika keadaan udara masih segar. Lingkungan sosial mencakup orang lain dan representasinya. Seseorang yang sedang belajar akan terganggu bila di dekatnya ada orang lain yang mondar-mandir. Rekaman suara, suara gaduh dari

mesin pabrik, hiruk pikuk lalu lintas juga merupakan lingkungan sosial yang merugikan.

- Faktor instrumental.

  Faktor ini mencakup perangkat keras seperti gedung, perlengkapan belajar, alat praktikum, maupun beberapa perangkat lunak seperti kurikulum, pedoman belajar, dan lain-lain.

- Faktor individu pelajar.

  Kondisi individu ini terdiri dari kondisi fisiologis dan panca indera seseorang. Kondisi fisiologis sangat berpengaruh terhadap belajar seseorang (sub bab 2.3.3). Keadaan jasmani yang segar membantu seseorang pada saat belajar. Gizi yang cukup membentuk kemampuan belajar seseorang, karena orang yang kurang gizi akan mudah lelah, mengantuk dan sulit menerima pelajaran. Kondisi panca indera terutama penglihatan dan pendengaran yang termasuk kondisi fisiologis adalah hal yang tak kalah pentingnya, karena sebagian besar orang belajar mempergunakan penglihatan dan pendengaran.

2.4.3. Faktor-Faktor Fisiologis yang Berpengaruh

Berikut ini adalah faktor-faktor fisiologis yang berpengaruh terhadap proses belajar seseorang.

A. Pengamatan

Cara manusia mengenali suatu obyek adalah dengan melihat, mendengar, meraba, membau dan mengecapnya. Cara ini disebut dengan mengamati. Modal pengamatan itu dibedakan menurut panca indera yang digunakan untuk mengamati. Dari kelima modalitas pengamatan itu yang telah diteliti secara psikologis adalah penglihatan.

- **Penglihatan**

  Berdasar obyeknya, masalah penglihatan menyangkut pada melihat bentuk dan warna. Dari sudut pandang psikologi ada 2 hal yang penting mengenai warna, yaitu nilai efektif warna dan nilai lambang warna.

Nilai efektif warna adalah masing-masing nilai yang dimiliki yang ikut membentuk medan perilaku dan memberi motif kepada perbuatan dan perilaku seseorang. Oleh karena itu, pada umumnya orang akan mencat bagian dalam rumahnya dengan warna-warna seperti putih, hijau muda atau warna-warna lain yang sejuk dan tenang.

Sedangkan nilai lambang warna adalah warna yang digunakan orang untuk melambangkan sesuatu, misalnya warna hitam untuk melambangkan kegelapan, kesedihan. Warna putih melambangkan kesucian, dsb.

- **Pendengaran**

  Modalitas pengamatan yang kedua adalah pendengaran. Pendengaran dan suara memelihara komunikasi vokal antara makhluk yang satu dengan lainnya. Bunyi sebenarnya adalah pernyataan yang dapat dimengerti oleh lawan dengan suatu arti tertentu.

B. Motif

Motif adalah keadaan dalam diri seseorang yang mendorong orang tersebut untuk melakukan aktifitas-aktifitas tertentu guna mencapai suatu tujuan. Ada dua motivasi, yakni dari luar (ekstrinsik) dan dari dalam (instrinsik). Aktifitas yang didorong oleh motif dari dalam diri sendiri (intrinsik) lebih berhasil dibanding dengan yang didorong oleh motif dari luar (ekstrinsik). Karena itu, untuk mempelajari sesuatu, sangat penting untuk menumbuhkan rasa minat terlebih dahulu terhadap hal yang akan dipelajari.

2.4.4. Penggunaan Teknologi Informasi dalam Proses Pembelajaran

Pemanfaatan teknologi dalam proses pembelajaran juga dapat memberikan pengaruh yang positif pada motivasi seseorang yang ingin belajar sesuatu. Fungsi teknologi dapat dikatakan lebih efektif dan tepat dalam memfasilitasi proses pembelajaran jika dalam menggunakannya juga diperhatikan faktor yang terkait dengan proses pembelajaran itu sendiri.

Salah satu faktor atau komponen yang utama dalam pembelajaran dengan teknologi adalah orang yang ingin belajar (siswa). Perlu diperhatikan apakah siswa tersebut sudah terbiasa dengan teknologi, dalam hal ini mengakses komputer. Oleh sebab itu, penulis melakukan survei untuk mengetahui seberapa besar penggunaan teknologi komputer di antara anak muda saat ini. Hal ini dikarenakan program belajar piano secara mandiri yang dibuat sangat bergantung pada teknologi komputer.

Teknologi memiliki potensi merubah peran pengajar, karena dengan teknologi peran siswa yang biasanya pasif (menerima) dapat menjadi aktif. Dengan teknologi, siswa dapat diminta untuk bertanggung jawab terhadap perkembangannya sendiri, misalnya memonitor kemajuan, membuat prioritas dan mengendalikan metode belajar yang sesuai dengan dirinya sendiri.

2.3.4.1. Perangkat Teknologi

Tidak semua perangkat teknologi dapat digunakan untuk proses pembelajaran. Tergantung juga pada sasaran dari pembelajaran itu sendiri, sehingga perlu dipilih perangkat teknologi yang mendukung sasaran tersebut.

Multimedia dipilih penulis untuk mendukung program belajar piano secara mandiri karena tujuan penulis adalah membuat program yang bersifat mandiri. Multimedia dapat mewakili tenaga pengajar. Selain itu, multimedia juga dapat membangun interaksi antara program dengan pengguna. Sehingga dengan aplikasi teknologi semacam ini (multimedia), siswa dapat belajar sambil praktek, yang merupakan prinsip penting yang sudah umum dalam pembelajaran dengan teknologi.