

SUPPLEMENTS DIVERS

Code pour mouvements supplémentaires de Pinocchio

Dans la séquence, ajouter un calque au dessus du calque « textes » et nommer le « actions ». Taper le code suivant : (le rouge est du commentaire)

```

_root.onKeyDown=function(){           //création de la fonction appui sur une touche
    laTouche=Key.getCode();           //on obtient le code de la touche enfoncée
};
Key.addListener(_root);                // on place un espion, ou écoute ou
                                        //surveillance du clavier

_root.onEnterFrame= function() {      //exécution des instructions en fonction de la touche
    if (laTouche==16){                 // majuscule pour rotation horaire
pinoc._rotation+=10;
    }
    if (laTouche==17) {                // Ctrl pour rotation anti-horaire
pinoc._rotation-=10;
    }
    if (laTouche==37) {                //gauche pour déplacement à gauche
pinoc._x--;
    }
    if (laTouche==39) {                // droite pour déplacement à droite
pinoc._x++;
    }
    if (laTouche==38) {                //haut ...
pinoc._y--;
    }
    if (laTouche==40) {                //bas....
pinoc._y++;
    }
}};

```

Dans « textes », à côté de PINOCCHIO, ajouter : Agir également sur les touches de direction, Maj, et Ctrl. Espace pour stopper.

Remarque : pour un déplacement plus rapide à gauche, au lieu de pinoc._x--, choisir pinoc._x - = n avec n égal à 2 ; 3 ; 4 ; ou 5. Agir de façon analogue pour les autres déplacements.

Code total pour l'horloge. Ajout de la date.

```

//AFFICHAGE DE LA DATE DU JOUR ET DE L'HEURE
_root.onEnterFrame = fonction() {

    //déclaration de l'objet Date
    chron= new Date();

    //obtention de la « date »
    lejour= chron.getDay(); //le jour de la semaine est un rang de 0 à 6
    ladate=chron.getDate(); // nombre pour la date dans le mois
    lemois=chron.getMonth(); // rang du mois (0 à 11)
    lannee=chron.getFullYear(); //nombre pour l'année
    //tableau des jours(0 à 6)
    jours=["Dimanche","Lundi","Mardi","Mercredi","Jeudi","Vendredi","Samedi"];
    //tableau des mois(0 à 11)
    mois=["Janvier","Février","Mars","Avril","Mai","Juin","Juillet","Août","Septembre","Octobr
e","Novembre","Décembre"];
    //la variable
    aujourdhui=jours[lejour]+" "+ladate+" "+mois[lemois]+" "+lannee;
    //affichage de la variable dans un texte dynamique
    _root.afaujourdhui=aujourdhui;

    //obtention de l'heure( heures, minutes, secondes)
    heures=chron.getHours();
    minutes=chron.getMinutes();
    secondes=chron.getSeconds();
    //rotation des aiguilles
    aigheure._rotation=(heures*30+minutes*0.5);
    aigmin._rotation=(minutes*6+secondes*0.1);
    aigsec._rotation=secondes*6;
}

```

Voici le code total de l'horloge : ajouter l'encadré et ne pas oublier de noter la variable « afaujourdhui » dans les propriétés du texte dynamique.

Pour répondre à la question de Roger DECOURT qui veut afficher heures, minutes et secondes avec le nom entier prenant convenablement le pluriel ou le singulier, voilà la solution. Je donne l'exemple simplement pour les secondes. Essayer de bien comprendre : le ? est le symbole de test : le nombre de secondes est-il strictement supérieur à 1, si la réponse est oui on a le nombre de secondes suivi de **secondes**, si la réponse est non, (càd si c'est 0 ou 1), on a 0 ou 1 suivi de **seconde**.

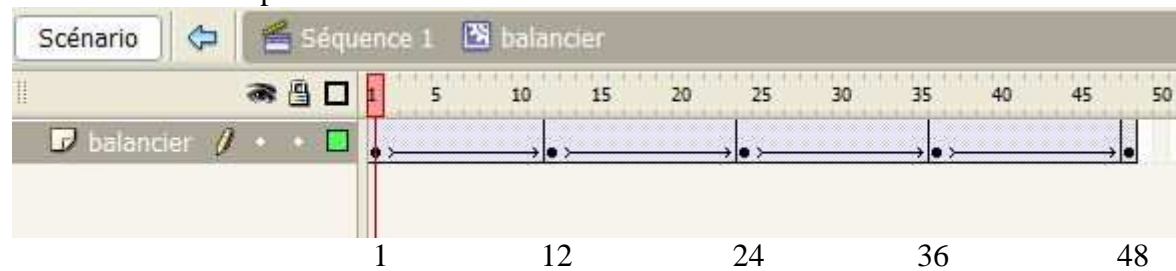
```






secondes = secondes >1? secondes+" secondes":secondes+" seconde";
_root.afsec="et "+secondes;

```

Ajout d'un balancier.

La période du balancier doit être de 2 secondes. Rappel : la période est le temps qui s'écoule entre 2 passages consécutifs du balancier au même point. La cadence du document étant fixé à 24 im/s, on va effectuer un aller-retour (2 balancements) du balancier sur 48 images. Un balancement correspondra à 1 seconde.

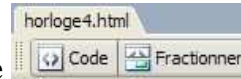


<p>Positions : faire le balancier vertical en 1. Insérer une image clé aux positions suivantes. Passer aux positions de 1, 24, et 48 par pivotement.</p>					
<p>Pivoter</p>	15 °	0°	-15°	0°	15°
<p>Accélération interpolation</p>	-100	100	-100	100	rien

Affichage des flashes en mode transparent

- Insérer le fichier (.swf) dans un tableau sur la page web.

- Faire apparaître le code en cliquant sur l'icône



- Ajouter ce qui est en rouge dans le code relatif au flash (exemple ci-dessous pour dujhorl4.swf).

```
<td><object classid="clsid:D27CDB6E-AE6D-11cf-96B8-444553540000"
codebase="http://download.macromedia.com/pub/shockwave/cabs/flash/swflash.cab#version
=7,0,19,0" width="450" height="500" align="left">
  <param name="WMODE" value="Transparent" />
  <param name="movie" value="../dosflash/dujhorl4.swf" />
  <param name="quality" value="high" />

  <embed src="../dosflash/dujhorl4.swf" width="450" height="500" align="left"
wmode="transparent" quality="high"
pluginspage="http://www.macromedia.com/go/getflashplayer" type="application/x-
shockwave-flash"></embed>
</object></td>
```

`<param name="WMODE" value="Transparent" />` est ajouté en tête des commandes **param name**

`wmode="transparent"` est ajouté dans la commande **embed**

Quelques informations sur la couleur.

En vidéo, le rouge, le vert, le bleu sont les couleurs « primaires ».

Le dosage de la couleur se fait à partir d'une échelle de 0 à 255. (donc sur un octet (2^8))

Le dessin ci-contre explique l'obtention des couleurs cyan, magenta et jaune (C, M, J), à partir de 2 des couleurs R, V, B.

Le blanc est obtenu à partir des 3 couleurs R, V, B.

Les dosages sont au maximum 255 pour les couleurs présentes.

(R, V, B) = (255, 0, 0) = Rouge

(R, V, B) = (0, 255, 0) = Vert

(R, V, B) = (0, 0, 255) = Bleu

(R, V, B) = (0, 255, 255) = Cyan

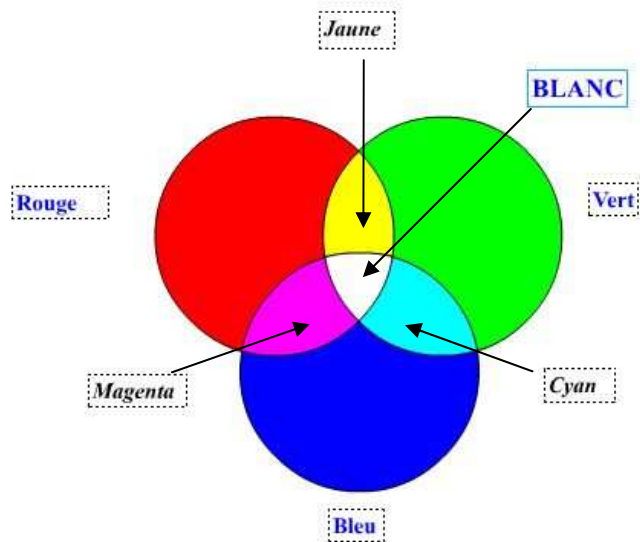
(R, V, B) = (255, 0, 255) = Magenta

(R, V, B) = (255, 255, 0) = Jaune

(R, V, B) = (255, 255, 255) = Blanc

(R, V, B) = (0, 0, 0) = Noir

En combinant avec les différents dosages, on peut donc obtenir théoriquement 256^3 couleurs soit 16 777 216.



Chaque couleur est donc définie par un code (R, V, B) = (n, p, q) où n, p, q sont des nombres de 0 à 255.

Le code de la couleur peut aussi s'exprimer en utilisant **la base 16 ou hexadécimale**. Dans cette base, on utilise pour symboles de numération les chiffres de 0 à 9 suivis des lettres A, B, C, D, E, F.

Symboles et correspondance.

Base 10	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Base 16	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F

Conversion.

Au 1^{er} rang à droite, valeur du symbole, au 2^e rang une unité vaut 16, au 3^e rang 16^2 , au 4^e rang 16^3 , au 5^e rang 16^4 , etc.

EXEMPLE :

$$\begin{array}{cccccc}
 16^4 & 16^3 & 16^2 & 16^1 & 16^0 & \\
 A & D & 5 & F & C & \longrightarrow
 \end{array}
 \begin{array}{l}
 10 \times 16^4 + 13 \times 16^3 + 5 \times 16^2 + 15 \times 16^1 + 12 \times 16^0 \\
 10 \times 65536 + 13 \times 4096 + 5 \times 256 + 15 \times 16 + 12 \times 1 \\
 655\,360 + 53\,248 + 1280 + 240 + 12 \\
 \text{AD5FC} \longrightarrow 710\,140
 \end{array}$$

En comptant en commençant par la droite : $12 \times 16^0 + 15 \times 16^1 + 5 \times 16^2 + 13 \times 16^3 + 10 \times 16^4$

Remarque : Si dans le nombre, on numérote les colonnes de 0 à n à partir de la droite, la valeur de l'unité va de 16^0 à 16^n .

Le code d'une couleur en hexadécimal est annoncé par le symbole dièse #. Ce code comprend en réalité 3 tranches de 2 symboles. En commençant par la gauche, la 1^{ère} tranche correspond au rouge, la 2^{ème} correspond au vert et la 3^{ème} au bleu. Chacune de

ces tranches exprime un nombre en décimal de 0 à 255. (Code de 00 à FF)

1^{er} exemple

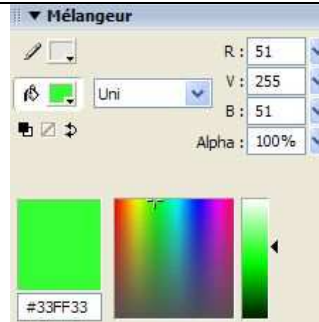
(R, V, B) = (51, 255, 51)

33 FF 33

R V B

33(hexa) = 3 x 16 + 3 = 51(décimal)

FF(hexa) = 15x16 + 15 = 255



2^{ème} exemple

(R, V, B) = (255, 153, 204)

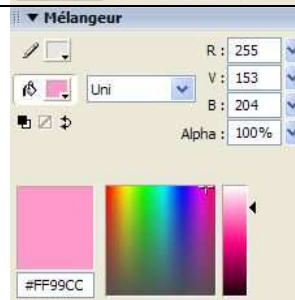
FF 99 CC

R V B

FF(hexa) = 15 x 16 + 15 = 255 (décimal)

99(hexa) = 9 x 16 + 9 = 153

CC(hexa) = 12 x 16 + 12 = 204



3^{ème} exemple

(R, V, B) = (38, 168, 217)

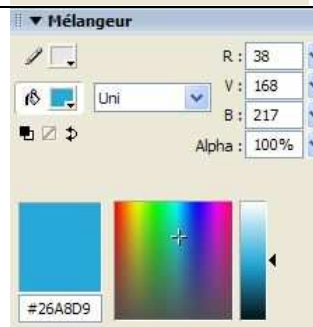
26 A8 D9

R V B

26(hexa) = 2 x 16 + 6 = 38 (décimal)

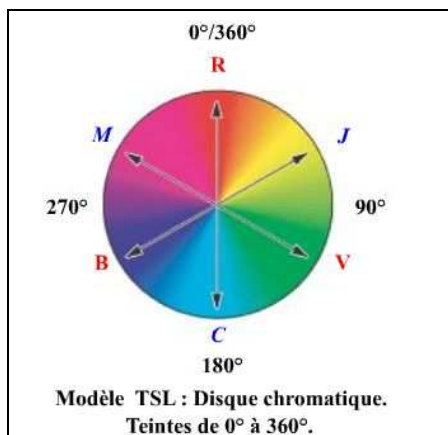
A8(hexa) = 10 x 16 + 8 = 168

D9 (hexa) = 13 x 16 + 9 = 217



Après les valeurs R, V, B de chaque couleur, on trouve la valeur **Alpha** en pourcentage. Elle définit le degré de transparence de la couleur.

Le codage TSL (HSL en anglais).



Il consiste à décomposer la couleur suivant 3 facteurs :

- **Teinte** : elle correspond à la perception de la couleur. Elle est exprimée en degrés (de 0° à 360°).
- **Saturation** : elle correspond au caractère vif ou terne de la couleur. Elle est exprimée en pourcentage (de 0 à 100).
- **Luminance** : elle indique la quantité de lumière, donc l'aspect clair ou sombre. Elle est exprimée en pourcentage (de 0 à 100).

On peut voir l'utilisation du code TSL dans Flash et Photoshop, dans les panneaux **Mélangeur de couleurs** et **Couleurs**. Voyons des exemples dans Flash.

On obtient le menu d'options en cliquant.
On passe du code RVB au code TSL.

Dans ce mélangeur de couleurs, la teinte en abscisse varie de **0° à 360°**.
La saturation en ordonnée varie de **0 % à 100 %**.
La luminance au sélecteur de droite varie de **0 % à 100%**.

Cliquer

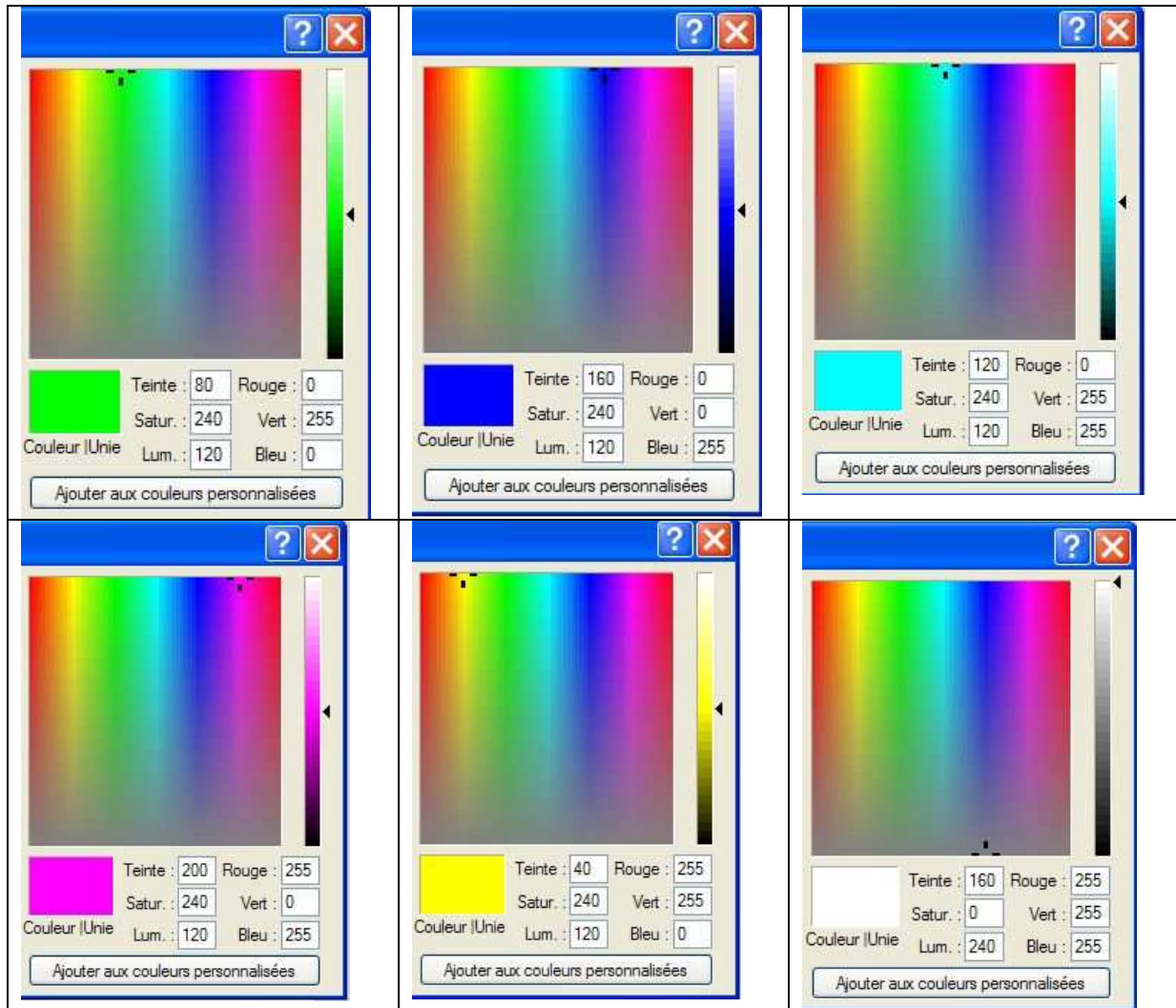
PARTIE DROITE DU PANNEAU COULEURS

Les échelles ne sont plus les mêmes.

Luminance (0 à 240)
Teinte (0 à 240)

Saturation (0 à 240)

Ici le rouge (R, V, B) = (255, 0, 0) équivaut à (T, S, L) = (0, 240, 120)

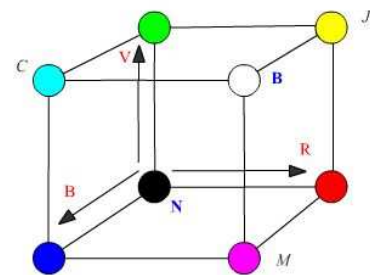


Remarques.

- En dehors des modèles **RVB** et **TSL**, il existe d'autres modèles colorimétriques. Les modèles **CMJN** (cyan, magenta, jaune, noir) et **L*a*b** apparaissent dans PHOTOSHOP.
- **Extrait de liste de couleurs et autre représentation du modèle RVB.**

Nom de la couleur	Codage RVB	Aperçu
aliceblue	#F0F8FF	
antiquewhite	#FAEBD7	
aqua	#00FFFF	
aquamarine	#7FFFD4	
azure	#F0FFFF	
beige	#F5F5DC	
bisque	#FFE4C4	
black	#000000	
blanchedalmond	#FFEBCD	
blue	#0000FF	
blueviolet	#8A2BE2	
brown	#A52A2A	
burlywood	#DEB887	
cadetblue	#5F9EA0	
chartreuse	#7FFF00	
chocolate	#D2691E	
coral	#FF7F50	
cornflowerblue	#6495ED	
cornsilk	#FFF8DC	
crimson	#DC143C	
cyan	#00FFFF	

... la suite sur Internet.



Représentation cubique du modèle RVB.