

PHP et création d'images

1°) Créer une image vide:

```
$image=imagecreate(450,250);
```

La variable \$image désigne alors cette image créée avec ses dimensions 450 x 250

2°) Dessiner l'image

Définir une couleur RVB et remplir le fond:

```
$col_fond = ImageColorAllocate($image, 200,200,200);  
imagefill($image,0, 0,$col_fond);
```

Dessiner un rectangle rempli avec \$col1 qui a été définie:

```
imagefilledrectangle ($image,10,10,440,240,$col1);
```

Tracer une ligne du point de coordonnées (30 ; 220) au point (430 ; 220)

Comme en SVG, l'origine est au coin gauche supérieur.

```
imageline($image,30,220,430,220,$col4);
```

Une ligne en pointillé

```
imagedashedline($image,130,220,130,20,$col3);
```

Placer du texte de taille 2 avec le coin supérieur gauche à (120 ; 220) de la couleur \$col4

```
imagestring ($image,2,120,220,"m=1",$col4);
```

Dessiner un arc d'ellipse centré en (250 ; 250) de demi axes 50 et 100 entre 0 et 180° de couleur \$col3

```
imagearc ($image,250,250,50,100,0,180,$col3);
```

Colorier le pixel de coordonnées (200 ; 100) avec la couleur \$col2

```
imagesetpixel($image,200,100,$col2);
```

Dessiner un polygone rempli

```
$points=array(50,100,100,100,250,200,100,250);  
imagefilledpolygon($image,$points,4,$col3);
```

Les sommets sont stockés dans le tableau \$points, 4 est le nombre de sommets

Pour un polygone sans remplissage

```
$points=array(50,100,100,100,250,200,100,250);  
imagepolygon($image,$points,4,$col3);
```

3°) Afficher l'image dans la page HTML

Première méthode: créer sur le serveur un fichier image

Transformer l'image en un fichier GIF, PNG ou JPEG

Définir le nom du fichier

```
$filename="essai.png";      ou      $filename="essai.gif";
```

Créer le fichier image

```
imagePng($image,$filename);      ou      imagegif($image,$filename);
```

Placer la balise image dans le fichier HTML

```
echo "<img src='".$filename."' width='450' height='250'>";
```

Deuxième méthode: envoyer directement l'image au fichier HTML

Pour cela la balise image doit dans le paramètre src faire appel au fichier PHP qui crée l'image:
Vous devez donc mettre la balise:

```
echo "<BR><img src='vonkoch.php3?action=img&niveau=".$niveau."' width='450'  
height='250'>";
```

Le fichier PHP qui crée l'image peut être le même que celui qui crée la page HTML ou non.
Ne pas oublier d'envoyer les paramètres qui sont nécessaires à la construction de l'image.

Si c'est le même fichier, suivant le paramètre action (par exemple) il renvoie la page HTML ou l'image.

Pour envoyer l'image au navigateur, il suffit de ne pas mettre de nom de fichier:

```
imagePng($image);      ou      imagegif($image);
```

Suivant la distribution PHP, le type d'image supporté n'est pas le même, chez Free utilisez GIF, pour des distributions plus récentes utilisez PNG.

4°) Un exemple complet avec envoi direct de l'image

```
<?php
```

La fonction vonkoch qui soit dessine le segment, soit appelle récursivement la fonction de niveau supérieur.

```
function vonkoch($indice,$xp1,$yp1,$xp2,$yp2)  
{global $niveau,$image,$col2;  
if ($indice==$niveau) {imageline($image,$xp1,$yp1,$xp2,$yp2,$col2);}  
else
```

Appel récursif des 4 fonctions vonkoch avec de nouvelles coordonnées pour les extrémités du segment

```
{ $xp=round($xp1+($xp2-$xp1)/3); $yp=round($yp1+($yp2-$yp1)/3);  
vonkoch($indice+1,$xp1,$yp1,$xp,$yp);  
$xp=round($xp1+($xp2-$xp1)/3); $yp=round($yp1+($yp2-$yp1)/3);  
$xp3=round($xp+($xp2-$xp1)/6+1.732*($yp2-$yp1)/6); $yp3=round($yp-1.732*($xp2-$xp1)/6+($yp2-$yp1)/6);  
vonkoch($indice+1,$xp,$yp,$xp3,$yp3);  
$xp=round($xp1+2*($xp2-$xp1)/3); $yp=round($yp1+2*($yp2-$yp1)/3);  
vonkoch($indice+1,$xp3,$yp3,$xp,$yp);  
vonkoch($indice+1,$xp,$yp,$xp2,$yp2); }  
}
```

La fonction de construction de la page HTML

```
function faire_page()  
{global $niveau;  
echo "<html><body><FONT face='Verdana, Arial' size=2  
color='#000080'><center>\n";  
echo "Tracé de la courbe de Von Koch";  
echo "<BR>Choisissez le niveau de récursivité 0 - 6";  
echo "<form METHOD='GET' ACTION='vonkoch.php3?action=niv&'>\n";  
echo "<table border='1' width='350'>\n";  
echo "<tr><td width='30%'><b>Niveau de récursivité</b></td>\n";
```

```

echo "<td width='20%'><input type='text' name='niveau' size='10'
value='". $niveau. "'></td>\n";
echo "<td width='10%'><INPUT TYPE=SUBMIT VALUE='OK'>\n";
echo "</tr></table></form>\n";

```

Balise image avec dans src l'appel au fichier PHP avec ici deux paramètres action et niveau

```

echo "<BR><img src='vonkoch.php3?action=img&niveau=" . $niveau. "' width='450'
height='250'>";
echo "</body></html>";
}

```

La fonction qui crée l'image, initialise les couleurs, appelle la fonction vonkoch qui complète le dessin et enfin envoie l'image au navigateur

```

function faire_image($niveau)
{global $x_pt,$y_pt,$image,$col2;

```

Création de l'image

```

$image=imagecreate(450,250);

```

Définition des couleurs

```

$col_fond = ImageColorAllocate($image, 200,200,200);
$col1 = ImageColorAllocate($image,255,255,255);
$col2 = ImageColorAllocate($image,255,0,0);
$col3 = ImageColorAllocate($image,0,255,0);
$col4 = ImageColorAllocate($image,0,0,0);

```

Remplissage du cadre

```

imagefill($image,0, 0,$col_fond);
imagefilledrectangle ($image,10,10,440,240,$col1);

```

Appel de la fonction vonkoch qui dessine la courbe après initialisation des paramètres

```

$indice=0;$xd=25;$yd=230;$xa=425;$ya=230;
vonkoch($indice,$xd,$yd,$xa,$ya);

```

Envoi de l'image au navigateur

```

ImagePng($image);
}

```

Corps principal de la page:

Initialisation si aucun paramètre n'a été envoyé

```

if (isset($_GET[$niveau])) $niveau=$_GET[$niveau]; else $niveau=1;
if (isset($_GET[$action])) $action=$_GET[$action]; else $action="niv";

```

Appelle suivant le paramètre action la construction de la page HTML ou la construction de l'image.

```

if ($action=="img") {faire_image($niveau);} else {faire_page();}
?>

```