

Manuel Utilisateur

Ligne de commande

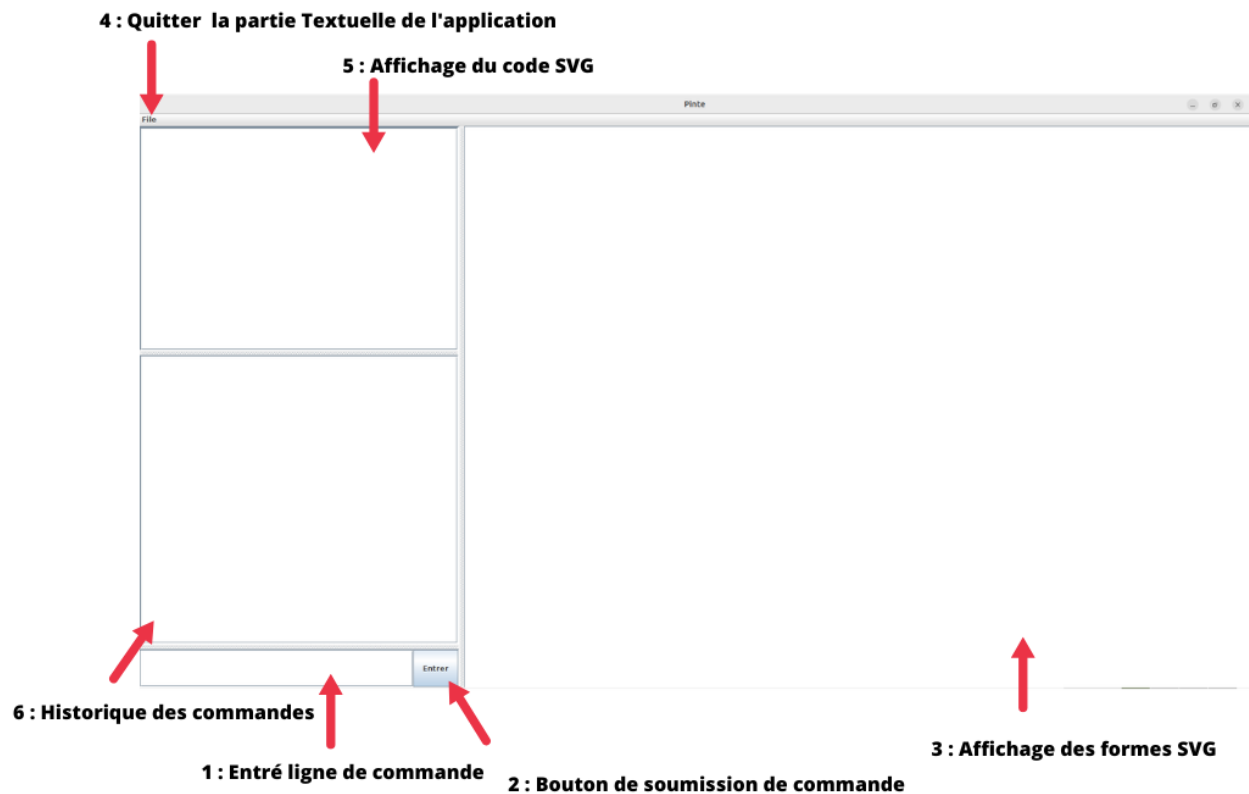
Cette partie du manuel utilisateur sert à décrire l'utilisation de l'application Pinte lorsque, dans le menu principal, vous choisissez l'option 'Créer une image SVG'.



Sommaire

Interface Utilisateur.....	3
Liste de commande.....	4
Création de forme.....	4
Modification de forme existante.....	7
Commandes générale.....	10

Interface Utilisateur



1 : Ce champ permet de rentrer les formes que l'on souhaite en respectant le format des commandes défini par l'application, lequel sera expliqué plus tard dans le manuel utilisateur.

2 : Ce bouton permet de soumettre la commande entrée dans le champs 1, cependant l'appuie sur le bouton entrée de votre clavier permet aussi la soumission de la commande.

3 : Dans cet écran s'affichera les formes SVG créé via les commande entrée dans la ligne de commande.

4 : Dans "file" se trouve un champs "quitter" qui permet de retourner au menu principale de l'application.

5 : Cet écran permet d'afficher le code SVG des formes créé via le champs de saisie 1

6 : Dans cette zone d'affichage se trouve les commandes tapé dans le champs de saisie 1 et qui a ensuite été envoyé.

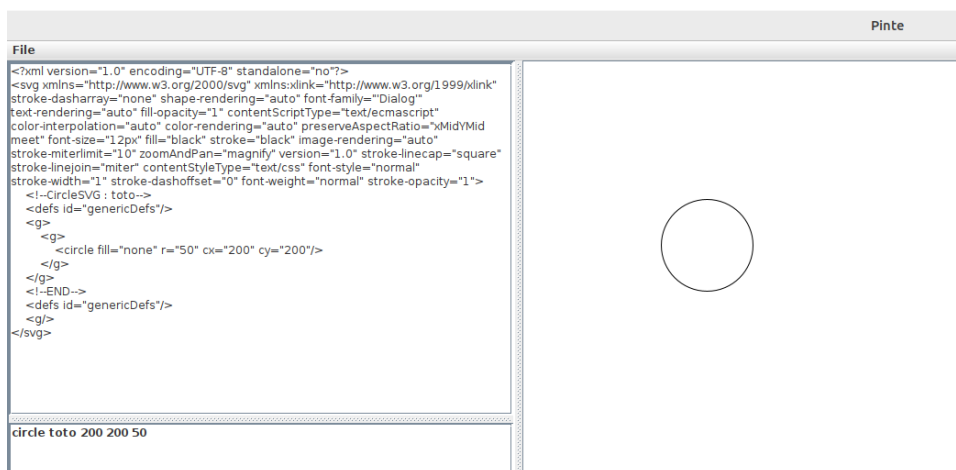
Liste de commande

Création de forme

On peut créer des formes différentes a partir de plusieurs commandes en ajoutant des options après le premier élément de la commande.

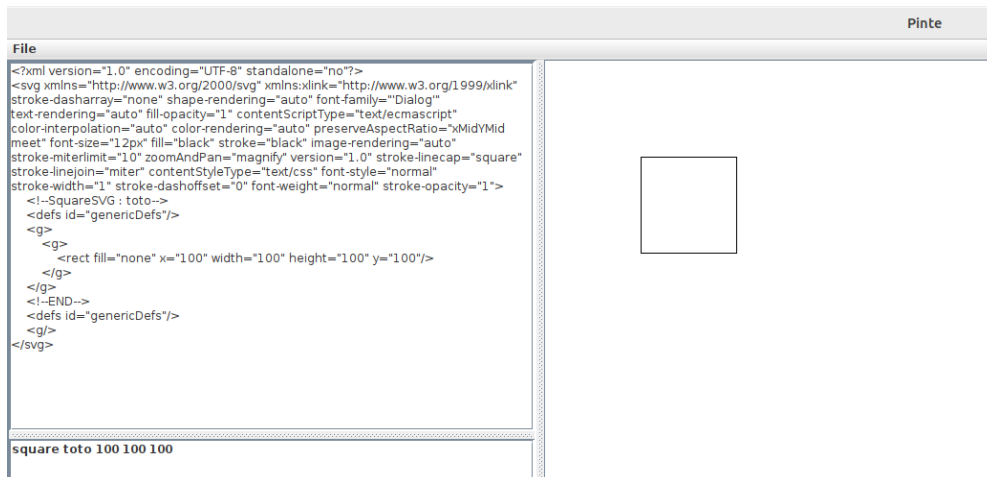
- *circle* <nom forme> <coordonnée x du centre> <coordonnée y du centre> <taille du rayon> <-s> <couleur bordure> <-f> <couleur remplissage>

Cette commande permet de créer un cercle, le nom de la forme est obligatoire ainsi que les coordonnées du centre et la taille du rayon, cependant “<-s> <couleur bordure> <-f> <couleur remplissage>” sont optionnels si l'ont veux changer les couleurs mais par défaut la couleur de bordure est noir et il n'y a pas de remplissage.



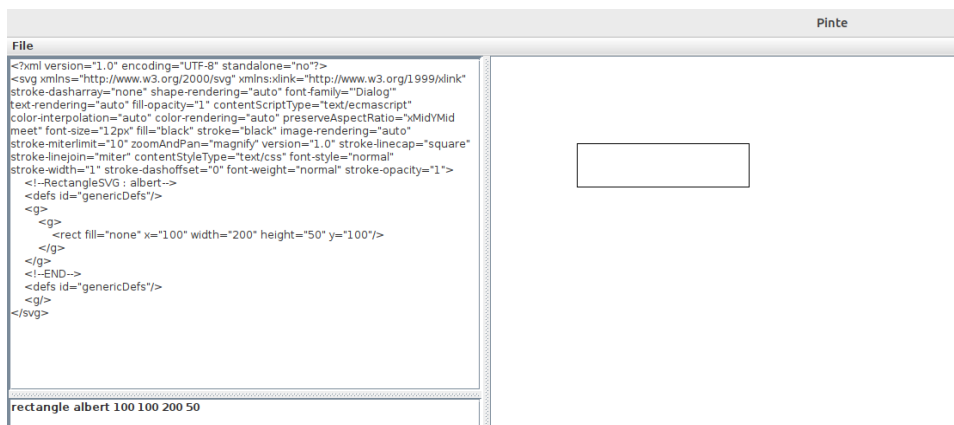
- *square* <nom forme> <coordonnée x du premier point> <coordonnée y du premier point> <longueur coté> <-s> <couleur bordure> <-f> <couleur remplissage>

Cette commande permet de créer un carré, le nom de la forme est obligatoire ainsi que les coordonnées du sommet du haut gauche et la taille d'un coté, cependant “<-s> <couleur bordure> <-f> <couleur remplissage>” sont optionnels si l'on veut changer les couleurs mais par défaut la couleur de bordure est noir et il n'y a pas de remplissage.



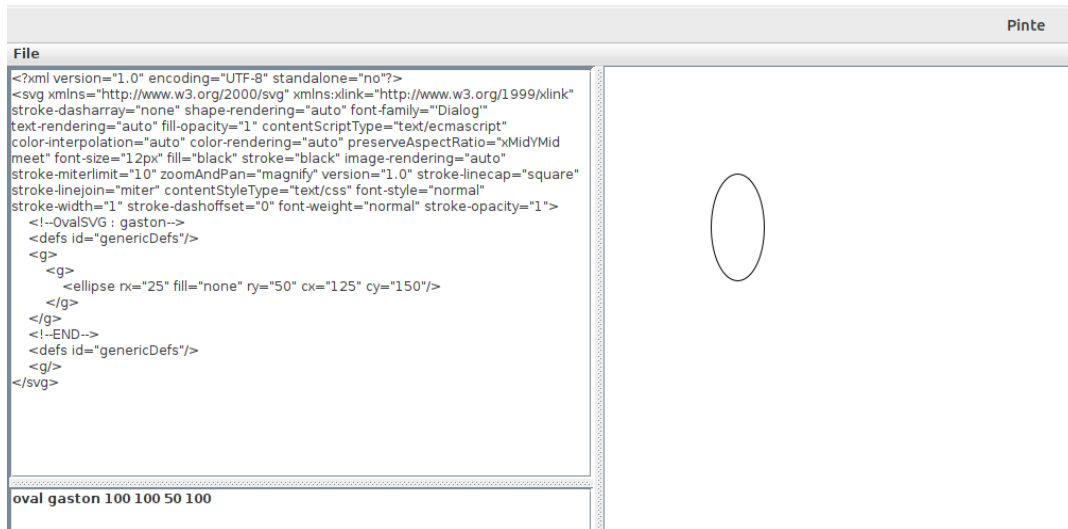
- *rectangle* <nom forme> <coordonnée x du premier point> <coordonnée y du premier point> <taille Longueur> <taille largeur> <-s> <couleur bordure> <-f> <couleur remplissage>

Cette commande permet de créer un rectangle, le nom de la forme est obligatoire ainsi que les coordonnées du sommet du haut gauche et la taille de la longueur et de la largeur, cependant “<-s> <couleur bordure> <-f> <couleur remplissage>” sont optionnels si l’on veut changer les couleurs mais par défaut la couleur de bordure est noir et il n’y a pas de remplissage.



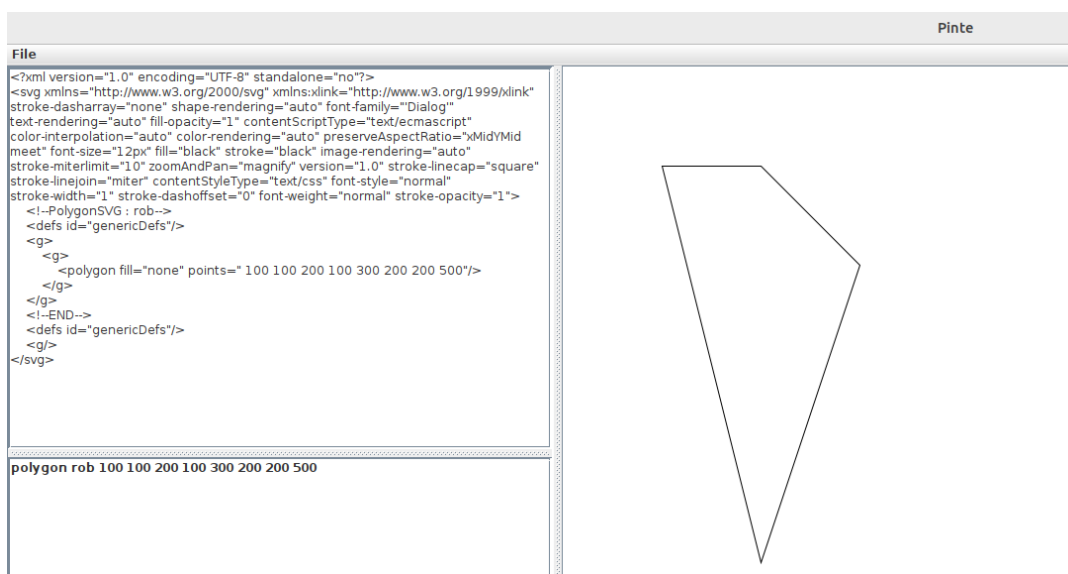
- *oval* <nom forme> <coordonnée x du centre> <coordonnée y du centre> <taille largeur> <taille hauteur> <-s> <couleur bordure> <-f> <couleur remplissage>

Cette commande permet de créer un ovale, le nom de la forme est obligatoire ainsi que les coordonnées du centre et la taille de la largeur et de la hauteur, cependant “<-s> <couleur bordure> <-f> <couleur remplissage>” sont optionnels si l’on veut changer les couleurs mais par défaut la couleur de bordure est noir et il n’y a pas de remplissage.



- *polygone* *<nom forme>* *<coordonnée x du point 1>* *<coordonnée y du point 1>* *<-s>* *<couleur bordure>* *<-f>* *<couleur remplissage>*

Cette commande permet de créer un polygone, le nom de la forme est obligatoire ainsi que les coordonnées x et y de chaque point qui compose le polygone, pour un quadrilatère il faudra 4 points donc 4 coordonnées x et 4 coordonnées y, cependant “*<-s>* *<couleur bordure>* *<-f>* *<couleur remplissage>*” sont optionnels si l’on veut changer les couleurs mais par défaut la couleur de bordure est noir et il n’y a pas de remplissage.



Modification de forme existante

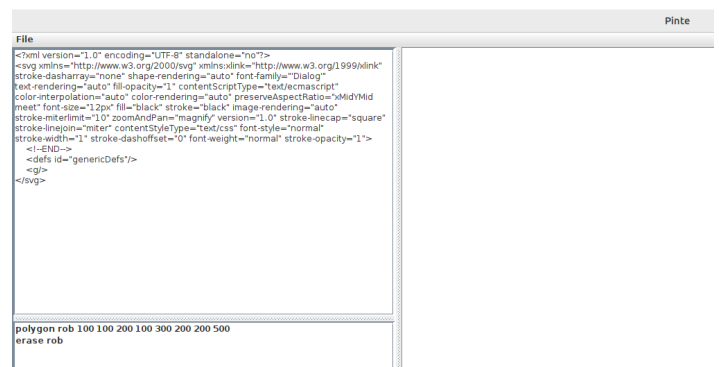
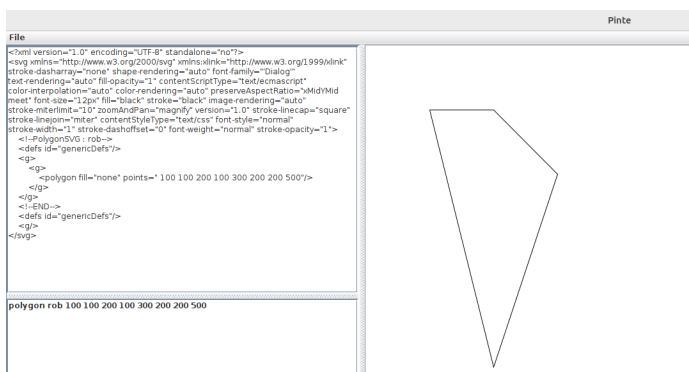
Une forme créée peut être modifiée avec différentes commandes en spécifiant le nom de la forme créée.

- *<commande forme> <nom forme existante> <parametre> <parametre> <parametre> -s <couleur bordure> -f <couleur remplissage>*

Si vous utilisez le nom d'une forme déjà existante vous pouvez utiliser les commandes de création de forme pour redéfinir totalement la forme.



- *erase <nom forme existante>*

La commande "erase" permet d'effacer une forme existante et ainsi la faire disparaître de l'écran qui l'affiche ainsi que du code svg, cependant vous gardez une trace de la forme dans l'historique.





- **recolor** <nom forme existante> -s <couleur bordure> -f <couleur remplissage>

La commande “recolor” permet de changer la couleur de bordure et ou de remplissage de la forme spécifiée.

<pre>File <?xml version="1.0" encoding="UTF-8"?><svg xmlns="http://www.w3.org/2000/svg" xmlns:xlink="http://www.w3.org/1999/xlink" stroke-dasharray="none" shape-rendering="auto" font-family="Dialog" text-rendering="auto" fill-opacity="1" contentScriptType="text/ecmascript" color-interpolation="auto" color-rendering="auto" preserveAspectRatio="xMidYMid meet" font-size="12" fill="black" stroke="black" image-rendering="auto" stroke-miterlimit="10" zoomAndPan="magnify" version="1.0" stroke-linecap="square" stroke-linejoin="miter" contentStyleType="text/css" font-style="normal" stroke-width="1" stroke-dashoffset="0" font-weight="normal" stroke-opacity="1"> <!--CircleSVG: toto--> <defs id="genericDefs"/> <g> <g fill="blue" stroke="blue"> <circle fill="none" r="50" cx="100" cy="100"/> </g> </g> <!--END--> <defs id="genericDefs"/> </g> </svg></pre>		<pre>File <?xml version="1.0" encoding="UTF-8"?><svg xmlns="http://www.w3.org/2000/svg" xmlns:xlink="http://www.w3.org/1999/xlink" stroke-dasharray="none" shape-rendering="auto" font-family="Dialog" text-rendering="auto" fill-opacity="1" contentScriptType="text/ecmascript" color-interpolation="auto" color-rendering="auto" preserveAspectRatio="xMidYMid meet" font-size="12" fill="black" stroke="black" image-rendering="auto" stroke-miterlimit="10" zoomAndPan="magnify" version="1.0" stroke-linecap="square" stroke-linejoin="miter" contentStyleType="text/css" font-style="normal" stroke-width="1" stroke-dashoffset="0" font-weight="normal" stroke-opacity="1"> <!--CircleSVG: toto--> <defs id="genericDefs"/> <g> <g fill="lime" stroke="lime"> <circle fill="none" r="50" cx="100" cy="100"/> </g> </g> <!--END--> <defs id="genericDefs"/> </g> </svg></pre>	
<pre>circle toto 100 100 50 -s blue [+] toto</pre>		<pre>circle toto 100 100 50 -s blue [+] toto recolor toto -s green -f red >> Recolor executed</pre>	

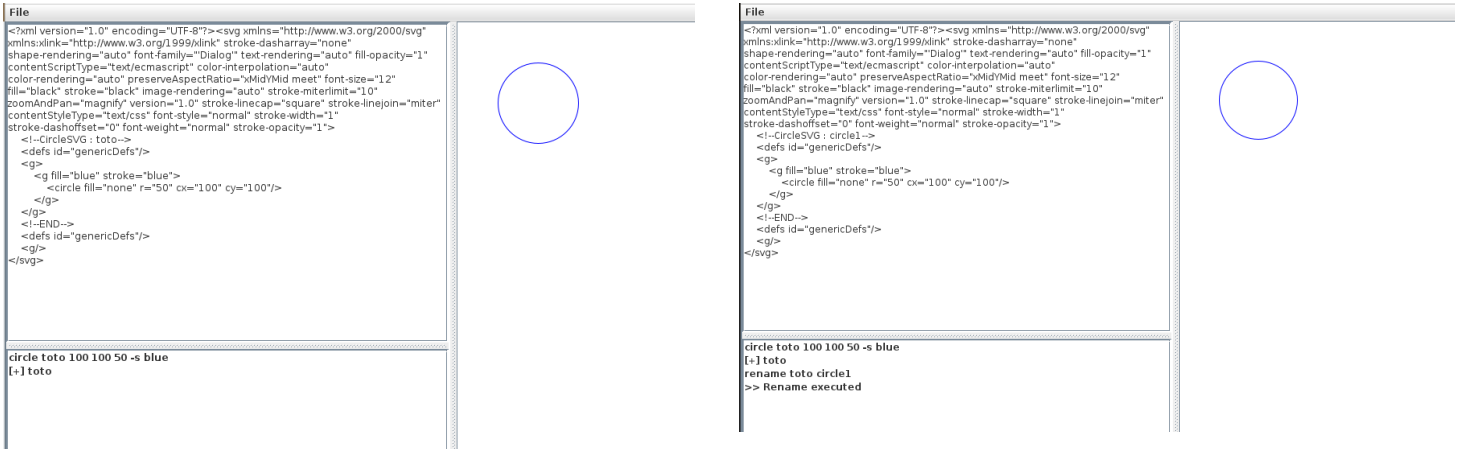
- **resize** <nom forme existante> -h <hauteur de la forme> -w <largeur de la forme>

La commande “resize” permet de redimensionner la forme en changeant la hauteur et la largeur de la forme donnée via le nom.

<pre>File <?xml version="1.0" encoding="UTF-8"?><svg xmlns="http://www.w3.org/2000/svg" xmlns:xlink="http://www.w3.org/1999/xlink" stroke-dasharray="none" shape-rendering="auto" font-family="Dialog" text-rendering="auto" fill-opacity="1" contentScriptType="text/ecmascript" color-interpolation="auto" color-rendering="auto" preserveAspectRatio="xMidYMid meet" font-size="12" fill="black" stroke="black" image-rendering="auto" stroke-miterlimit="10" zoomAndPan="magnify" version="1.0" stroke-linecap="square" stroke-linejoin="miter" contentStyleType="text/css" font-style="normal" stroke-width="1" stroke-dashoffset="0" font-weight="normal" stroke-opacity="1"> <!--RectangleSVG: toto--> <defs id="genericDefs"/> <g> <g fill="blue" stroke="blue"> <rect fill="none" x="100" width="50" height="50" y="100"/> </g> </g> <!--END--> <defs id="genericDefs"/> </g> </svg></pre>		<pre>File <?xml version="1.0" encoding="UTF-8"?><svg xmlns="http://www.w3.org/2000/svg" xmlns:xlink="http://www.w3.org/1999/xlink" stroke-dasharray="none" shape-rendering="auto" font-family="Dialog" text-rendering="auto" fill-opacity="1" contentScriptType="text/ecmascript" color-interpolation="auto" color-rendering="auto" preserveAspectRatio="xMidYMid meet" font-size="12" fill="black" stroke="black" image-rendering="auto" stroke-miterlimit="10" zoomAndPan="magnify" version="1.0" stroke-linecap="square" stroke-linejoin="miter" contentStyleType="text/css" font-style="normal" stroke-width="1" stroke-dashoffset="0" font-weight="normal" stroke-opacity="1"> <!--RectangleSVG: toto--> <defs id="genericDefs"/> <g> <g fill="blue" stroke="blue"> <rect fill="none" x="100" width="200" height="100" y="100"/> </g> </g> <!--END--> <defs id="genericDefs"/> </g> </svg></pre>	
<pre>rectangle toto 100 100 50 50 -s blue [+] toto</pre>		<pre>rectangle toto 100 100 50 50 -s blue [+] toto resize toto -w 200 -h 100 >> Resize executed</pre>	

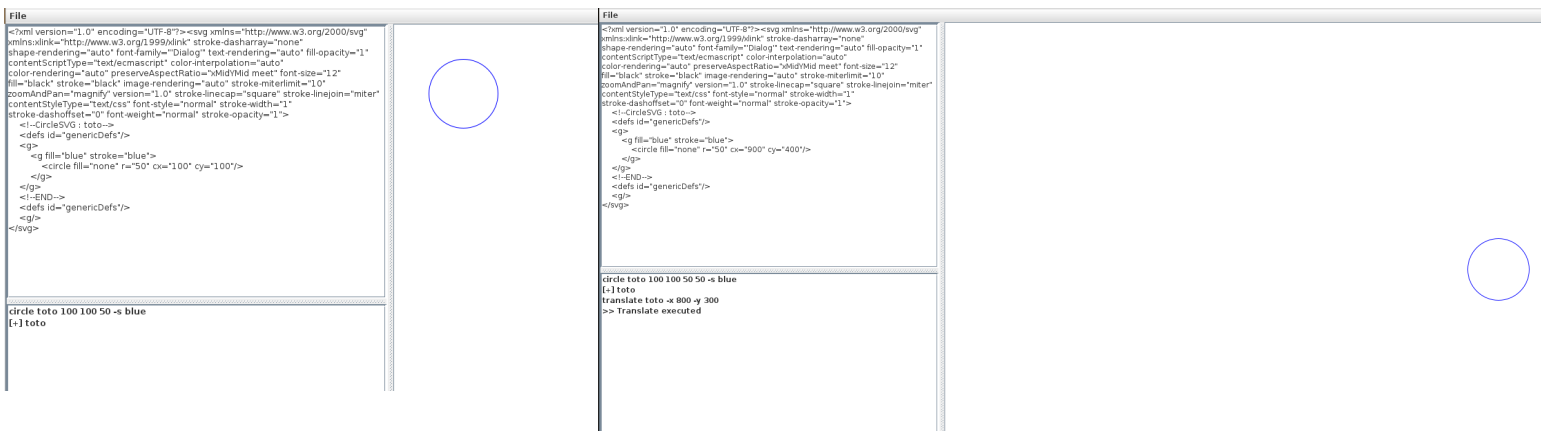
- *rename <nom forme existante> <nouveau nom forme>*

La commande “rename” permet de changer le nom d’une forme existante.



- *translate <nom forme existante> -x <taille translation en x> -y <taille translation y>*

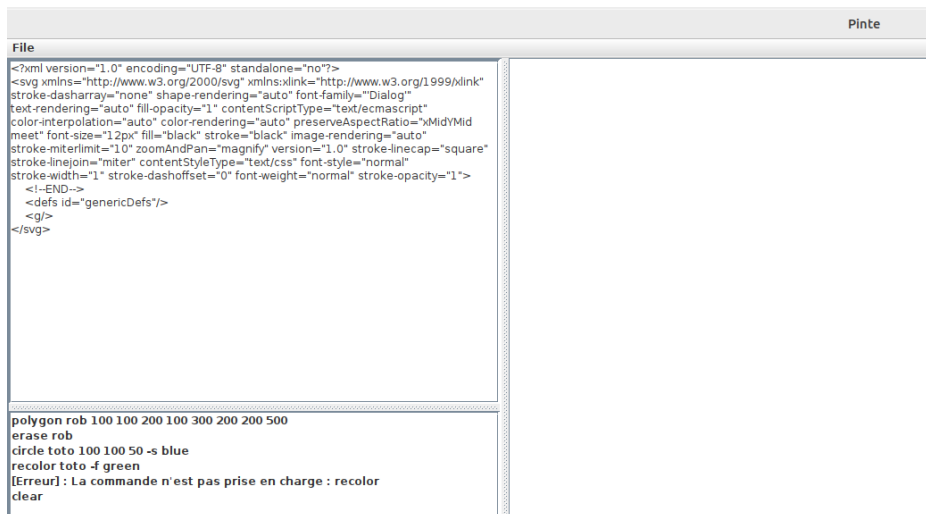
La commande “translate” permet de traduire une forme en précisant de combien en x et ou de combien en y.



Commandes générale

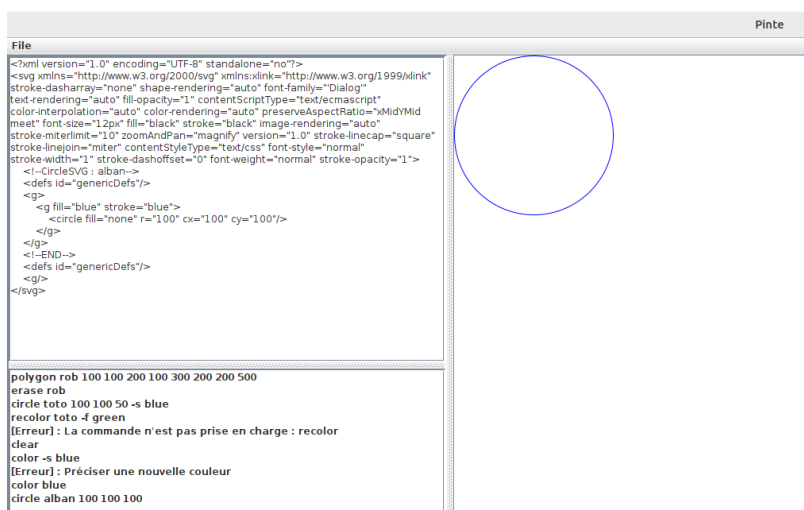
- **clear**

La commande “clear” permet d’effacer toute les formes dans l’affichage, cependant il reste une trace dans l’historique.



- **color <couleur bordure>**

La commande “color” permet de changer la couleur par default de bordure des prochaines formes qui seront créées sans le specifier en argument lors de la création de la forme.



- **save**

La commande “save” permet de sauvegarder le code svg dans un fichier sous le nom de “canva.svg”