SVG - Paramètres HTTP - Formulaires

Exercice 1 :

SVG est un langage XML consacré au dessin vectoriel (les figures sont décrites par leur caractéristiques géométriques). HTML5 permet d'incorporer des dessins SVG directement au sein du source HTML, dans un élément `<svg>`.

En SVG, les coordonnées s'expriment par un simple nombre (sans unité). L'axe des y est orienté vers le bas. Nous n'utiliserons que 3 types de figures :

élément	attributs obligatoires
`circle`	`cx`, `cy` (coordonnées du centre), `r` (rayon)
`rect`	`x`, `y` (coordonnées du coin en haut à gauche), `width`, `height`
`polygon`	`points` (liste des coordonnées des points : p_1x p_1y p_2x p_2y p_3x p_3y...)

Exemples :

```
<circle cx="100" cy="150" r="70" />
<rect x="0" y="50" width="100" height="50" />
<polygon points="200 200 250 250 200 250" />
```

NB : ces 3 éléments sont sans contenu, donc leurs balises sont auto-fermantes (se terminent par `/>`). Les attributs figurent nécessairement entre guillemets.
L'élément `<g>` ...`</g>` permet de regrouper plusieurs éléments SVG .
L'attribut `transform` est autorisé pour tous les éléments. Nous utiliserons une seule transformation : la rotation. Par exemple `transform="rotate(30,0,50)"` indique qu'il faut appliquer à l'élément une rotation de 30 degrés autour du point de coordonnées 0 50.
Le fichier `exempleSVG.html` est un fichier HTML5 qui comporte un dessin en SVG. Il vous est fourni à titre d'exemple. Ouvrez le dans un éditeur ; **examinez attentivement** le contenu de l'élément `<svg id="dessin">`

Q 0 .

```
├── exempleSVG.html
├── figures.php
├── lib
│   └── fonctionsSVG.php
└── views
    ├── pageErreur.html
    └── pageFigures.php
```

Le code est organisé comme suit :

- à la racine du projet : le fichier `figures.php` sera le script directement invoqué par les utilisateurs
- dossier `lib` : bibliothèques et classes
- dossier `views` : ensemble des contenus que `figures.php` est susceptible d'afficher : ils sont destinés à être inclus par `figures.php` mais PAS à être appelés appelés directement par l'utilisateur

L'accès aux dossiers `lib` et `views` devrait être interdit d'accès direct depuis le serveur (voir fichier `.htaccess` de la feuille précédente).

`pageFigures.php` est dérivé de `exempleSVG.html` : le code statique des éléments SVG dessinant des figures a été remplacé par des appels à des fonctions PHP.
Consultez également le contenu de `figures.php` pour constater qu'on se contente, pour l'instant, de définir 3 variables et d'inclure `pageFigures.php`

Q 1 . *Compléter la **mini** bibliothèque PHP dédiée à la génération de code SVG.*
La bibliothèque `lib/fonctionsSVG.php` vous est fournie avec la fonction `cercle()` déjà écrite, il s'agira de la compléter et de faire des tests.

1. voici les caractéristiques du *« carré inscrit dans le cercle* (cx, cy, r) *»* et possédant un côté horizontal :
   - longueur des côtés : $l = r\sqrt{2}$
   - coordonnées du coin inférieur : $(cx - \frac{l}{2}, cy - \frac{l}{2})$

   Écrivez une fonction `carreInscrit($cx,$cy,$r)` qui renvoie une chaîne contenant le texte de l'élément `<rect>` dessinant ce carré.

   Complétez le fichier `pageFigures.php` pour obtenir le dessin d'un cercle et de son carré inscrit. Vous devez obtenir un résultat analogue à la figure ci-dessous (sans le triangle). Testez avec différentes valeurs

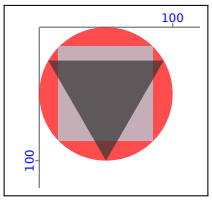

FIGURE 1 – Exemple d'un cercle de centre $(50, 50)$ et de rayon 50 avec le carré (à un côté horizontal) et le triangle (à base horizontale) inscrits

2. ajoutez à la fonction `carreInscrit()`, un paramètre `angle` optionnel (valeur par défaut : 0) qui a pour effet de pivoter le carré autour de son centre. Testez.
3. le triangle équilatéral à base horizontale **inscrit** dans le cercle (cx, cy, r) possède les 3 sommets $(cx, cy + r)$, $(cx - \frac{r\sqrt{3}}{2}, cy - \frac{r}{2})$, $(cx + \frac{r\sqrt{3}}{2}, cy - \frac{r}{2})$.
   Créez la fonction `triangleInscrit($cx,$cy,$r,$angle=0)` qui renvoie le code de l'élément `<polygon>` représentant ce triangle. Le paramètre *angle* a le même rôle que pour le carré.
4. ajoutez dans `pageFigures.php` le dessin du triangle.

Q 2 . *Paramétrage du script*

Dans cette question, vous allez faire en sorte que la page `figures.php` prenne en compte des paramètres (en mode GET) indiquant les caractéristiques de figures à afficher.

1. Commençons par 3 paramètres HTTP : `centreX` , `centreY`, `rayon` vérifiant les contraintes :
   — ils sont obligatoires
   — ils représentent chacun un nombre (entier ou flottant)
   — `rayon` est positif ou nul
   Pour vérifier la présence et la validité des 3 arguments. Les fonctions PHP `isset()` et `is_numeric()` vous seront utiles.
   Si l'un des arguments est absent ou incorrect, le script `figures.php` affichera la page d'erreur `views/pageErreur.html` qui vous est fournie. **Ne recopiez pas ce fichier**, utilisez `require()` pour l'inclure, évidemment.
   Si les 3 arguments sont corrects, le script définira les 3 variables `$centreX $centreY` et `$rayon` recevant respectivement les valeurs des 3 paramètres puis affichera la page contenant les figures.
   Testez le script en lui passant différents paramètres (corrects, incorrects, absents) que vous coderez directement dans l'url.
2. ajoutez la prise en compte d'un paramètre HTTP **optionnel** nommé `angle`, nombre qui désigne l'angle à appliquer au carré et au triangle. En cas d'absence du paramètre, on considèrera que l'angle vaut 0. On fera de même si la valeur fournie est la chaîne vide. Par contre toute autre valeur incorrecte entraînera l'affichage de la page d'erreur.

Q 3 . *Création d'un formulaire*
Créez une page `formulaireFigures.html` qui contient un formulaire permettant à l'utilisateur de choisir les caractéristiques des figures à dessiner. Après validation, la page `figures.php` sera invoquée.
Q 4 . *Ajout d'un nouveau paramètre*
On ajoute un paramètre optionnel nommé `fig` dont la valeur est nécessairement `cercle, carre` , `triangle`. ou `all` Dans les 3 premiers cas, seule la figure choisie sera dessinée. Dans le cas de `all`. (qui est la valeur par défaut) toutes les figures le seront. produira la page d'erreur. Toute autre valeur fournie entraînera l'affichage de la page d'erreur. Modifiez les scripts nécessaires puis complétez le formulaire. Vous ferez 2 versions du formulaire : l'une utilisant un `select` et l'autre utilisant des boutons `radio`.