

TP 03 : Rapport Yahtzee

Pour ce Tp J'ai créé les classes **Yahtzee**, **Serveur**, **Pont** et comme extra j'ai créé **GenerateurAleatoire**.

La classe **Yahtzee** est une servlet qui contient la méthode doGet().

La classe **Serveur** est un programme qui contient la méthode main().

La classe **Pont** est un programme qui contient la méthode getNb() ainsi que deux constructeurs, l'un sans paramètres et l'autre avec (String ancienNb,String cbDe0,String cbDe1,String cbDe2,String cbDe3,String cbDe4) pour paramètres.

La classe **GenerateurAleatoire** contient la méthode getAl().

Yahtzee :

La méthode doGet() de la classe **Yahtzee** contient d'une part le suivi de session de l'utilisateur et le un appel a la classe **Pont**, d'autre part le code html généré par la servlet. Il s'agit donc de ce que l'utilisateur voit sur sa page web. Les sessions ne servent qu'a se souvenir des dés précédemment sortis, ils sont envoyé vers la classe pont uniquement si l'une des checkbox est cochée. Sinon la classe **Pont** est appelée sans rien en paramètre. On ne se soucie d'aucun autre traitement dans cette classe, on fait appel à la méthode getNb() de la classe pont qui fournit le String des dés et selon ce String on affiche différemment les images des dés que l'on trouve dans le répertoire image.

Serveur :

La méthode main() de la classe **Serveur** est l'unique méthode de ce serveur. En effet, il n'est pas très compliqué car il ne traite qu'une connexion à la foi. Il n'a pas besoin de plus car à chaque connexion il ne fait que générer 5 nombre aléatoires allant de 1 à 6, il les met dans un String, l'envoie au client connecté, et ferme la connexion. Il est ainsi à nouveau prêt pour une nouvelle connexion.

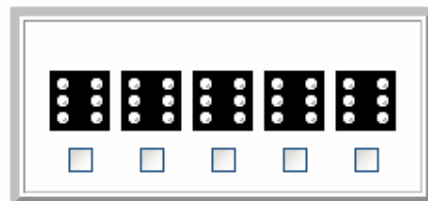
Pont :

La classe **Pont** comme son nom l'indique fait le pont entre les classes. C'est elle qui va être contactée par **Yahtzee** et c'est elle qui va aller interroger la classe **Serveur**. Quelle que soit son constructeur appelé elle va récupérer un nouveau set de dés aléatoires du serveur, ce qu'elle en fait change selon les constructeurs. Si on appelle son constructeur sans paramètre, elle stocke directement le String reçu du serveur dans la variable `finalDes`. Si on appelle son constructeur avec paramètres, le premier est considéré comme l'ancien numéro des dés, les cinq suivants représentent chacun un dé coché par l'utilisateur de la servlet, s'il est nul il n'est pas coché il faut donc mettre le nouveau dé à cette place, s'il est non nul il est coché, il faut donc remettre la valeur de l'ancien dé à cette place. Il fait également un pont avec la dernière classe : **GenerateurAleatoire**, qui est appelée au cas où le serveur ne serait pas en marche qui rend un nombre « totalement » aléatoire à la classe **Pont**.

GenerateurAleatoire :

La classe **GenerateurAleatoire** rend un String équivalent à 6 pour chacun des dés, par la méthode `getAI()`. C'est de l'humour de programmeur paraît-il...

Yahtzee



Jouer

Si l'interface est exactement pareille à celle de l'exemple présent sur le site du cours, toute ressemblance est fortuite.

Remarque :

Si la programmation ne m'a pris que peu de temps le codage en html m'a bloqué durant de nombreuses heures. Cependant, passer en paramètres des checkbox m'est désormais familier.