

Programmation Web Avancée Côté Client

TP n° 7

Licence Informatique 2ème année
Université de Nice-Sophia Antipolis

Introduction

Dans cette séance, nous allons continuer le travail de la dernière séance (TP n° 6), en ajoutant de la persistance à l'aide de la sérialisation.

1 Sérialisation

1. Ajoutez un bouton « sérialiser » dans la page HTML.
2. Gérez un clique sur ce bouton en affichant la sérialisation de l'état courant de l'objet `labyrinthe` en bas de la page HTML. Pensez à écrire les filtres ou les méthodes `toJSON` éventuellement nécessaires pour que toute et seule l'information pertinente de chaque objet soit sérialisée.
3. Ajoutez à l'objet `Labyrinthe` une nouvelle méthode `fromJSON` qui accepte comme argument une chaîne de caractère en format JSON et reconstitue le labyrinthe à partir d'elle (pensez à ajouter une méthode `fromJSON` au prototype de `Tuile` aussi, qui transforme un objet désérialisé en une tuile)¹ ;
4. Une fois vérifié que la sérialisation et la désérialisation fonctionnent correctement, supprimez le bouton, mais gardez le code qui effectue la sérialisation pour le prochain exercice.

2 Persistance

Nous allons maintenant modifier notre script pour sauvegarder automatiquement l'état du labyrinthe (en fait, l'état du jeu : nous songerons à ce que toutes les informations pertinentes et nécessaires pour reconstituer une session du jeu soient, dans le futur, toujours incluses comme propriétés dans l'objet labyrinthe) à chaque fois qu'un changement se produit.

1. Ajoutez à l'objet `Labyrinthe` une méthode `sauvegarder` qui sérialise l'objet (utilisez à cet effet le code que vous avez écrit pour l'Exercice 1) et le stocke dans le *local storage*.

1. La nécessité de ce passage peut ne pas être immédiatement évidente : le problème est que JSON ne sérialise pas les méthodes ; donc, si on se limite à désérialiser un objet sérialisé, on aura perdu ses méthodes. La manière la plus directe de contourner ce problème consiste à créer un nouveau objet du type souhaité et à y copier toutes les propriétés à partir de l'objet désérialisé.

2. Ajoutez des appels à cette nouvelle méthode dans tous les endroits du script où l'état du jeu est modifié. Vous ne devrez jamais oublier de le faire à chaque fois que vous ajouterez des nouvelles fonctionnalités.
3. Modifiez le script de manière que, si l'état du jeu précédemment sauvegardé est disponible dans le *local storage*, il est utilisé pour reconstituer le labyrinthe. À partir de maintenant, le labyrinthe ne sera généré aléatoirement que si l'état précédent n'est pas disponible ou que l'utilisateur le demande explicitement en cliquant le bouton « Régénérer labyrinthe ».
4. Testez le fonctionnement de votre script en redémarrant votre navigateur : lorsque vous rouvrez la page HTML que vous avez développé, le labyrinthe devrait se trouver dans le même état qu'avant.