

## Cours 5 - Exercice (formatif)

L'interface Web présente douze tuiles illustrant chacune un *Lorem Ipsum*. Au clic d'une tuile, injectez au dos de celle-ci (`data-js-citation`) une citation aléatoire parmi son tableau de *Lorem Ipsum* (identifié en `dataset` de l'attribut `data-js-component`). Au clic suivant, réaffichez le côté face, mais avec une image aléatoire parmi ses 3 images associées.

Pour l'animation de la tuile, il suffit de faire basculer la classe CSS `ipsum--flip` sur l'élément parent (`.ipsum`).

Vous devez développer ces comportements à l'aide de classes ES6. Plus précisément, les classes `BobRoss`, `Cupcake`, `Fromage` et `SamuelLJackson` sont des classes enfants de `LoremIpsum`. Ces classes enfants doivent, lors de leur instantiation, définir leur tableau de citations et initialiser leurs comportements. Toutefois, pour éviter la redondance de code, les comportements génériques aux quatre *Lorem Ipsum* doivent être définis dans la classe parent `LoremIpsum` (héritage).

Notez que le *Lorem Ipsum* `BobRoss` a un comportement légèrement différent : le peintre est nommé en fin de citation. Il faudra redéfinir la méthode concernée à l'intérieur de cette classe enfant (polymorphisme).

Comme toujours, je vous invite à travailler par étape : faites d'abord un des quatre *Lorem Ipsum*, ensuite l'héritage puis le polymorphisme. Assurez-vous également d'utiliser le préfixe `#` pour déclarer comme *private* toutes propriétés et méthodes qui n'ont pas à être publiques ainsi que les accesseurs (*getter*) nécessaires.

Pour l'instanciation des classes, je vous lance le défi de factoriser leur appel. Tel que mentionné dans les notes de cours, il faudra boucler sur tous les éléments `data-js-component` et récupérer leur `dataset` respectif. L'idée est que chaque élément est un bloc comportemental requérant l'instanciation de la classe identifiée en `dataset`. Il y a toutefois une *twist*, l'opérateur `new` n'accepte pas les chaînes de caractères...