

## Examen Commun de janvier 2023 – Durée 1 h 30

Documents seuls autorisés : transparents de cours.

Barème indicatif.

## 1 Problème : gestion de livraison de colis

On veut étudier la mise en place d'une application dans un navigateur pour gérer le livraison de colis d'une entreprise de transport de marchandises.

Dans tout le sujet, vous n'utiliserez ni `innerHTML`, ni `outerHTML`.

Pour simplifier ici tout le code sera écrit dans le fichier `livraison.js`.

### 1.1 Colis (5 points)

Les colis sont de formes prédéfinies identifiées par un format (un numéro compris entre 1 et 5).

Un colis est défini par son identifiant (une chaîne de caractères), son format et son poids (réel) en kg.

Le prix d'un colis est déterminé par son format et son poids et vaut  $2 \times \text{format} + 5 \times \text{poids}$ .

**Écrivez** la « classe » `Colis` représentant un colis :

- un colis stocke son identifiant, son format et son poids ;
- le constructeur prend en paramètre l'identifiant, le format et le poids ;
- on veut pouvoir connaître l'identifiant, le format, le poids et le prix du colis (méthodes `identifiant()`, `format()`, `poids()` et `prix()`) ;
- pour faciliter l'échange de données avec l'extérieur, la méthode `toObjet()` doit renvoyer un objet ayant les propriétés `identifiant`, `format` et `poids` valant respectivement l'identifiant, le format et le poids du colis, et la méthode de classe `depuisObjet(obj)` doit renvoyer un colis correspondant à l'objet `obj` contenant dans ses propriétés `identifiant`, `format` et `poids` l'identifiant, le format et le poids du colis.

**Comment** indique-t-on qu'une propriété d'un objet est « privée » ?

**Écrivez** la ligne de code qui crée la variable `c` valant un colis d'identifiant `TOT03654`, de format 4 et de 1,4 kg.

**Ré-écrivez** cette ligne de code en utilisant cette fois-ci `depuisObjet`.

### 1.2 Affichage de la liste des colis d'une livraison (7 points)

Une livraison de plusieurs colis est représentée par un objet ayant pour propriétés `idlivraison` contenant l'identifiant de la livraison et `colis` tableau contenant les colis de la livraison.

On veut afficher une livraison dans la page HTML :

```
<div id="livraison">
  <p>Identifiant livraison</p>
  <ul>
    <li><!-- un item pour chaque colis -->
      Colis <code>identifiant colis</code> :
      catégorie <strong class="number">catégorie colis</strong>,
      poids <em class="number">poids colis</em> kg
    </li>
    ...
  </ul>
</div>
```

**Écrivez** la fonction `itemColis(c)` qui calcule et renvoie le DOM de l'item `<li>` correspondant au colis `c`.

**Écrivez** la fonction `afficherLivraison(livr, div)` qui affiche les colis de la livraison `livr` dans l'élément `div` représentant un `<div>` qui contient déjà un élément `<p>` suivi d'un élément `ul` : il faut remplacer les contenus des éléments `<p>` et `<ul>` par les contenus correspondant à la livraison `livr` (en minimisant le nombre de ré-affichages).

**Écrivez** la fonction `ajouterAffichageColis(c, div)` qui ajoute l'affichage du colis `c` dans le premier élément `<ul>` (à la fin de celui-ci) de l'élément `div` (représentant un élément `<div>`).

### 1.3 Gestion des colis d'une livraison (8 points)

On a le formulaire suivant qui permet de saisir un colis :

```
<div id="formulaire-colis">
<p>Identifiant : <input name="identifiant" type="text"></p>
<p>Format :
<select name="format">
  <option value="1">Format 1</option>
  <option value="2">Format 2</option>
  <option value="3">Format 3</option>
  <option value="4">Format 4</option>
  <option value="5">Format 5</option>
</select></p>
<p>Poids : <input name="poids" type="text"></p>
<p>
  <button id="bajouter">Ajouter</button>
  <button id="bannuler">Annuler</button>
</p>
</div>
```

**Écrivez** la fonction `colisDuFormulaire()` qui renvoie un colis correspondant aux informations du formulaire de saisie : si une donnée saisie est incorrecte (vide ou pas du bon type) il faudra renvoyer un colis d'identifiant vide, de catégorie 0 et de poids 0.

On suppose écrite la fonction `reinitialiserFormulaire()` qui ré-initialise le formulaire.

**Déclarez** la variable `livraison` qui représente la livraison dont on va gérer les colis, qui a pour identifiant `LIVR123456` et qui n'a pour l'instant aucun colis.

**Écrivez** la fonction `ajouterColisDuFormulaire()` qui si les informations du formulaire de saisie étaient correctes ajoute le colis correspondant au formulaire de saisie à la livraison et met à jour l'affichage de la livraison.

**Écrivez** le gestionnaire d'événements `onReponse` qui sera appelé en réponse à un événement clic dans l'élément `<div id="formulaire-livraison">` : si l'élément cliqué était le bouton `<button id="bajouter">`, il faut ajouter à la livraison le colis correspondant au formulaire ; si l'élément cliqué était le bouton `<button id="bannuler">`, il faut ré-initialiser le formulaire.

**Écrivez** le code qui fait que `onReponse` soit effectivement appelé quand on clique à l'intérieur de `<div id="formulaire-livraison">` en faisant en sorte que ce code soit exécuté une fois la page chargée.