

## Travail pratique : Vitrine commerciale (partie 2)

---

### Énoncé :

Dans ce travail vous aurez à produire un site de commerce électronique simplifié. Précisément, vous devrez concevoir la partie catalogue et le panier d'achat d'un site de commerce électronique. Le site final comprendra un catalogue qui permet à des clients de faire des achats (sans paiement). Afin de faciliter le développement, la gestion des usagers est simplifiée. Le travail est divisé en 2 parties.

### Partie 2 : *Chargement dynamique des données et panier d'achat*

La seconde partie consiste à mettre en place le panier d'achat d'un site de vente en ligne. À partir du code réalisé dans la première partie, vous devez ajouter la gestion du panier d'achat. *[Aussi, vous devez charger la source des données pour les intégrer dynamiquement aux catalogues (Requête XHR/Fetch)].*

Le catalogue devra comprendre les fonctionnalités suivantes :

- Options d'affichage en grille (vignette) ou en liste
- Gestion de l'affichage en diverses pages (12, 24, 36 produits par page ou tous)

*[Les données du catalogue (JSON) devront être chargé à l'aide de l'API XMLHttpRequest ou de Fetch. Les données vous seront fournies ainsi que le format HTML de ceux-ci (annexe). Vous n'êtes pas autorisé à modifier le balisage directement (attributs et balises compris), cependant vous pouvez manipuler ceux-ci par programmation].*

De plus, la gestion du panier d'achat se fera en utilisant le module Pattern en JavaScript. Vous aurez donc à créer des méthodes pour gérer l'ajout des éléments et l'affichage du panier.

### Critères d'évaluation :

- Qualité du design et de l'ergonomie du site (15 %)
- Maîtrise de la programmation orientée objet en JavaScript (30 %)
- Fonctionnement adéquat du site Web dynamique (40 %)
  - Fonctionnement général (10%)
  - Affichage du panier d'achat avec les produits (15 %)
  - Ajout des produits au panier (10 %)
  - Vider le panier (5 %)
- Qualité du code et documentation adéquate (15 %)
- *Bonus : Utilisation correcte du LocalStorage (5 %)*

### Remise :

- Remise sur Léa, le **21 octobre 2018** avant la fin de la journée.

## Cas d'utilisation :

<b>1- Nom : CatalogueProduit</b>
<b>Pré-conditions :</b> Aucune
<b>Description :</b> Au lancement du site web l'utilisateur visualise une page qui présente un catalogue de produits. La page comprend : un bandeau d'entête avec une navigation simple, le catalogue de produit (liste des produits) et un pied de page contenant les informations de droits d'auteur.  Un élément Panier est aussi affiché dans le haut de la page à droite avec le nombre d'item du panier entre parenthèse. (Ex. : Panier (0))
<b>Exception :</b> Aucune
<b>Post-conditions :</b> Le catalogue est affiché selon le cas <b>CatalogueProduit_afficherPage</b> .

<b>2- Nom : CatalogueProduit_afficherPage</b>
<b>Pré-conditions :</b> <ul style="list-style-type: none"><li>• Une page ou un nombre d'article à afficher a été sélectionné</li><li>• Les données du catalogue ont été chargé avec XHR, sinon aller au cas <b>CatalogueProduit_chargerProduits</b></li></ul>
<b>Description :</b> Le catalogue affiche les produits selon : <ul style="list-style-type: none"><li>• la page sélectionné (<b>CatalogueProduit_changePage</b>)</li><li>• le mode d'affichage (<b>CatalogueProduit_changeModeAffichage</b>)</li><li>• le nombre de produit par page (<b>CatalogueProduit_changeNombreProduit</b>)</li></ul> Par défaut : <ul style="list-style-type: none"><li>• Page 1</li><li>• 12 produits/page</li><li>• Affichage en mode vignette</li></ul> Le catalogue affiche le gestionnaire de page en tenant compte du nombre de page total et de la page active. <ul style="list-style-type: none"><li>• Le bouton précédent ne doit pas apparaître sur la première page</li><li>• Le bouton suivant ne doit pas apparaître sur la dernière page</li><li>• Un bouton permet d'aller à chaque page directement</li></ul> Le catalogue affiche les contrôles permettant de changer : <ul style="list-style-type: none"><li>• le mode d'affichage (<b>CatalogueProduit_changeModeAffichage</b>)</li><li>• le nombre de produit par page (<b>CatalogueProduit_changeNombreProduit</b>)</li></ul>
<b>Exception :</b> Aucune
<b>Post-conditions :</b> Aucune

<b>3- Nom : CatalogueProduit_chargerProduits</b>
<b>Pré-conditions :</b> <ul style="list-style-type: none"><li>• Les données du catalogue n'ont pas été chargé</li></ul>
<b>Description :</b> <p>Le catalogue envoie une requête XHR sur le fichier ./data/produits.json et récupère le contenu dans le catalogue.</p>
<b>Exception :</b> <p>Aucune</p>
<b>Post-conditions :</b> <p>Si les données du catalogue sont correctement chargées aller au cas <b>CatalogueProduit_afficherPage</b></p>

<b>4- Nom : CatalogueProduit_changeModeAffichage</b>
<b>Pré-conditions :</b> <p>L'utilisateur change le mode d'affichage</p>
<b>Description :</b> <p>Deux modes d'affichage :</p>
<b>Vignette :</b> <ul style="list-style-type: none"><li>• Affichage de 4 produits par ligne avec une grande image.</li></ul>
<b>Liste :</b> <ul style="list-style-type: none"><li>• Affichage d'un tableau de produits avec une petite image.</li></ul> <p>L'affichage de chaque produit comprend une image, une description, le nom de la catégorie, le prix et un bouton "Ajouter au panier".</p>
<b>Exception :</b> <p>Aucune</p>
<b>Post-conditions :</b> <p>Aucune</p>

<b>5- Nom : CatalogueProduit_changeNombreProduit</b>
<b>Pré-conditions :</b> L'utilisateur change le nombre de produit par page
<b>Description :</b> L'utilisateur peut choisir entre : <ul style="list-style-type: none"><li>• 12 produits/page</li><li>• 24 produits/page</li><li>• 36 produits/page</li><li>• tous</li></ul>
<b>Exception :</b> Aucune
<b>Post-conditions :</b> Si la page active n'est pas la première page, la page active est changée pour la page 1 (CatalogueProduit_changePage)

  

<b>6- Nom : AfficherCatalogueProduit_changePage</b>
<b>Pré-conditions :</b> L'utilisateur choisi de changer la page qui est affiché.
<b>Description :</b> <ul style="list-style-type: none"><li>• Si l'utilisateur appuie sur le bouton suivant la page est incrémentée</li><li>• Si l'utilisateur appuie sur le bouton précédent, la page est décréementée</li><li>• L'utilisateur peut choisir directement une page en cliquant sur le bouton correspondant</li></ul>
<b>Exception :</b> <ul style="list-style-type: none"><li>• Si la page choisie est inexistante, ne pas effectuer de changement et afficher un message d'erreur à l'utilisateur.</li></ul>
<b>Post-conditions :</b> Le catalogue est affiché selon le cas CatalogueProduit_afficherPage.

  

<b>7- Nom : PanierAchat_ajouterProduit</b>
<b>Pré-conditions :</b> L'utilisateur clique sur le bouton « Ajouter au panier » d'un produit
<b>Description :</b> <ul style="list-style-type: none"><li>• Le produit est envoyé vers le panier.</li><li>• Un indicateur permet de voir le nombre d'item unique dans le panier. La quantité de chaque item n'est pas affiché.</li></ul>
<b>Exception :</b> <ul style="list-style-type: none"><li>• Si l'item est déjà dans le panier, il n'est pas ajouté. Aucune erreur n'est affichée.</li></ul>
<b>Post-conditions :</b> Aucune

<b>8- Nom : PanierAchat_gestionPanier</b>
<b>Pré-conditions :</b> L'utilisateur clique sur le bouton du panier d'achat
<b>Description :</b> <ul style="list-style-type: none"><li>• Les produits dans le panier s'affichent (image, titre, prix)</li><li>• L'utilisateur a l'option de retourner magasiner ou de vider son panier.</li></ul>
<b>Exception :</b> Aucune
<b>Post-conditions :</b> Aucune

## Annexes

### Structure des données de produit

Les produits sont structurés selon le code HTML ci-après. En aucun cas, vous pouvez modifier le balisage ou les attributs des produits directement dans le code source. Une autre liste de produits (non fournie) sera utilisée pour la correction.

**Code :**

```
<article class="produit">
  <header class="nom">Nom du produit</header>
  <section class="image"><img src=""></section>
  <section class="description">Une description assez courte</section>
  <section class="prix">
    <span class="prix-valeur">valeur</span>
    <span class="prix-unite">$can</span>
  </section>
  <section class="categorie">NomCategorie</section>
</article>
```

Toutes les images utilisées doivent avoir la même dimension (300 x 300 pixels).

### Structure des objets

Vous devez utiliser la structure suivante pour créer vos objets (Méthodes minimales)

MonApp
Afficher() ChangerPage(numPage) getNombreProduit() setNombreParPage(nombre) setModeAffichage(mode)  AjouterProduit(produit) ViderPanier() AfficherPanier()  [ChargerProduit ()] getInfoProduit(item)

Panier
Dessiner () Afficher () Fermer () Vider () Ajouter (item) getNombreItem()