



Cégep **André-Laurendeau**

Cours 420-345-AL-A22  
Programmation web avancée  
Lundi 17 octobre 2022  
**Examen intra**

Prénom & Nom : \_\_\_\_\_

Matricule : \_\_\_\_\_

Signature : \_\_\_\_\_

**/ 100**

**Consignes de l'examen :**

- *La partie 1 de cet examen doit être réalisée sur ordinateur*
- *La partie 2 doit être répondue sur ce formulaire*
- *Vous ne devez pas communiquer en utilisant votre ordinateur durant l'examen*
- *Vous n'avez pas le droit d'utiliser de téléphone mobile*
- *La durée de cet examen est de 2 heures*

**Vous devez remettre :**

- *La copie papier signée.*
- *Le code source soumis dans Github pour le projet Classrooms qui sera créé en utilisant le lien suivant :*
  - *<https://classroom.github.com/a/iFPALSor>*

## Partie 01 Angular :

Pour cette partie vous allez créer une application Angular permettant de faire une petite interface pour acheter des moniteurs. Vous trouverez un GIF animé dans le répertoire images vous permettant de voir les éléments visuels que vous devez créer.

Un répertoire `models` contient un fichier `articles.ts`. Vous devez mettre ce répertoire dans votre projet Angular à l'intérieur du répertoire app. Vous n'avez pas à modifier ce fichier. Ce fichier export deux éléments :

- `IArticle` :
  - Contiens l'interface pour les éléments que vous devez utiliser. Vous pourrez importer cette définition dans les fichiers requis
- `articlesGlobaux`
  - Le **seul fichier** qui à la permission d'importer ce vecteur est `app.component.ts`. Les autres composants devront recevoir les données de ce vecteur par l'entremise de `app.component.ts`.

Vous devez suivre les directives suivantes pour effectuer votre travail.

- Vous devez ajouter trois composantes
  - `Liste-produits`
  - `Description`
  - `Panier`
- Structure
  - Vous n'avez pas à vous soucier de la vue mobile pour ce travail
  - La liste de produit doit être à gauche
  - La vue de détail d'un produit et le panier doivent être affichés à droite et demeurer dans la vue même si l'utilisateur défile la liste de produit
  - Chaque composant doit être bordé par une bordure 2
    - Rouge pour `liste-produits`, bleu pour `description` et jaune pour `panier`.
- Composant `liste-produits`
  - Affiche une carte pour chaque produit
  - Afficher l'image du produit
  - Afficher le nom du produit comme titre de la carte
  - Prix de vente
    - Si le produit est en vente, afficher le prix de vente en gras et rouge et rayer le prix normal.
      - ~~\$379.99~~ **\$279.99**
    - Si l'article n'est pas en vente, afficher le prix en gras.
      - **\$319.99**

- Mettre deux boutons, un « ajoute » (vert) et un « retire » (rouge) au bas de la carte.
  - Il doit y avoir un espace entre les deux boutons.
  - Les deux boutons ensemble doivent prendre la pleine largeur de la carte.
- Si l'utilisateur clique sur la carte vous devez :
  - Mettre la description du produit cliqué dans le composant **description**.
  - La carte sélectionnée doit présenter une bordure jaune de 2 pixels d'épaisseur.
- Si l'utilisateur clique sur le bouton ajoute
  - Vous devez ajouter l'article dans une liste d'achat.
  - Si la carte n'est pas sélectionnée, cliquer sur ce bouton ne devrait pas la sélectionner.
  - Chaque clique ajoute l'élément, il n'est pas requis de faire un total, vous pouvez simplement ajouter l'élément plusieurs fois dans le panier.
- Si l'utilisateur clique sur le bouton retire
  - Vous devez retirer l'article de la liste d'achat.
  - Si la carte n'est pas sélectionnée, cliquer sur ce bouton ne devrait pas la sélectionner.
  - Si l'article existe plusieurs fois dans le panier, vous devez les retirer tous.
- Composant **description**
  - Affiche une boîte intitulée Description qui contient la description de l'article sélectionné.
  - S'il n'y a pas d'article sélectionné, cet élément ne doit pas apparaître dans la page.

Description:

Conçu pour offrir des expériences de jeu immersives, ce moniteur de jeu FreeSync de 32 po de Samsung est doté d'un écran incurvé 1500R avec résolution HD intégrale 1920 x 1080, pour des images cristallines. Son temps de réponse gris à gris de 4 ms,

- Composant **panier**
  - Affiche une boîte intitulée Panier :
    - Afficher le total du panier dans cette boîte à droite.
  - Affiche la liste des articles contenus dans le panier
    - Afficher le SKU à gauche
    - Le montant de l'article en vente à droite
  - S'il n'y a pas d'article dans le panier, vous devez mettre une mention « Panier vide »
- Utilisez le pipe « currency » pour afficher vos prix

Panier:	\$2,769.90
14611307	\$279.99
14611307	\$279.99
14933347	\$169.99
14933347	\$169.99

Liste d'articles

Description de  
l'article



Moniteur jeu DEL incurvé VA HD int. GTG 4 ms  
FreeSync 75 Hz 32 po Samsung (LC32R500)-Bleu  
gris foncé

~~\$379.99~~ **\$279.99**

Ajoute

Retire

Description:

Conçu pour offrir des expériences de jeu immersives, ce moniteur de jeu FreeSync de 32 po de Samsung est doté d'un écran incurvé 1500R avec résolution HD intégrale 1920 x 1080, pour des images cristallines. Son temps de réponse gris à gris de 4 ms, son taux de rafraîchissement de 75 Hz et la technologie Radeon FreeSync d'AMD assurent l'exécution fluide et sans décalage des jeux rapides, sans décalage ou saccade.

Panier:

\$2,769.90

14611307  
14611307  
14933347  
14933347  
15732100  
14763270  
14763270  
14763270  
14763270  
16233319

\$279.99  
\$279.99  
\$169.99  
\$169.99  
\$149.99  
\$399.99  
\$399.99  
\$399.99  
\$399.99  
\$119.99

Panier d'achats

Bordure sélectionnée

Bordure non sélectionnée



## **Partie 02 Typescript : (20 points)**

Expliquer la différence pour chaque expression de « return » présenté plus bas :

- Sont-ils différents?
- Quelle est la différence?
- Quels sont leurs avantages et/ou inconvénients?

```
interface UnElement {
  param_1: string,
  param_2: number,
  param_3: boolean,
  param_4: string
}

@Component({
  selector: 'app-maclass',
  template: '',
  styles: []
})
export class MaClasse {
  elements: UnElement[] = [];

  envoieElements(): UnElement[] {
    /* 1 */ return this.elements.map(e => ({... e}));
    /* 2 */ return this.elements.map(e => e);
    /* 3 */ return [...this.elements];
    /* 4 */ return JSON.parse(JSON.stringify(this.elements));
    /* 5 */ return this.elements;
  }
}
```

NOTE : Vous pouvez donner votre explication avec du code, un dessin ou une description.

1	
2	
3	
4	
5	