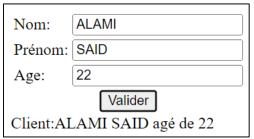
## Développement front-end TP 1



Formateur : LAMOURI Najib

## Exercice 1:

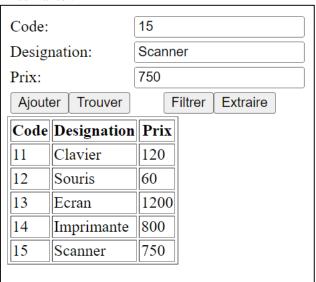
- 1) Définir une classe Client caractérisée par les attributs nom, prenom et age.
- 2) Ajouter la méthode info() permettant de retourner une chaine de caractères de la forme : Client:ALAMI Said agé de 22.
- 3) Créer la page HTML suivante :



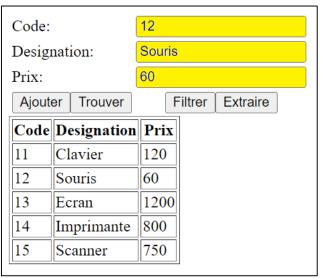
Ecrire le code du gestionnaire d'événement du bouton « Valider » permettant de créer un objet de type Client à partir des champs renseignés par l'utilisateur et de l'afficher en bas en utilisant la méthode info() de la classe Client.

## **Exercice 2:**

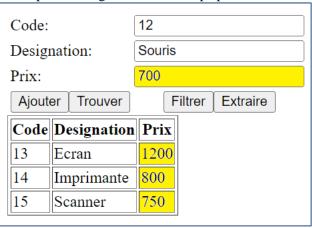
- 1) Définir une classe Produit caractérisé par les attributs code, désignation et prix.
- 2) Créer la page HTML suivante :



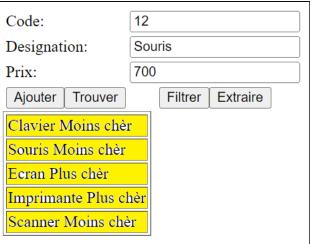
- 3) Déclarer un tableau global dans lequel nous allons stocker la liste des produits.
- 4) Ecrire la fonction Ajouter permettant d'ajouter un produit à la liste des produits et mettre à jour le div affichant la table html ci-dessus. Un clic sur le bouton « Ajouter » fait appel à cette fonction.
- 5) Ecrire la fonction Trouver en réponse au clic sur le bouton « Trouver » permettant d'afficher la désignation et le prix du produit recherché par code dans les champs correspondants en utilisant la méthode find :



6) En utilisant la méthode filter, écrire la méthode Filtrer permettant d'afficher les produits avec le prix supérieur au prix renseigné dans le champs prix :



7) En utilisant la méthode map et l'opérateur conditionnel ternaire, en réponse au clic sur le bouton « Extraire » afficher à partir du tableau (liste des produits) la table html suivante contenant les désignations des produits et les mention « Plus cher » ou « Moins cher » selon le prix du produit s'il est inférieur ou non à 800 :



## **Exercice 3:**

```
Considérons la liste suivante : modules=[
{'code' :'M101','Intitulé' :'Métier et formation','MasseHoraire' : {'pre':15,'dis':0}},
{'code' :'M102','Intitulé' :'Algorithmique','MasseHoraire' : {'pre':90,'dis':20}},
{'code' :'M103' 'Intitulé' :'Sites Web Statigues' 'MasseHoraire' : {'pre':75 'dis':20}}
```

{'code':'M103','Intitulé':'Sites Web Statiques','MasseHoraire': {'pre':75,'dis':20}}];

- 1) Ecrire une Arrow fonction permettant de calculer et retourner la masse totale d'un module (masse présentielle+masse à distance).
- 2) En utilisant la méthode map, afficher la masse totale (pre+dis) de chaque module.

- 3) En utilisant la méthode reduce, calculer et afficher la masse totale cumulée de tous les modules.
- 4) Ecrire une pure fonction qui ajoute l'attribut filière avec comme valeur 'DEV' à tous les modules du tableau ci-dessus sans le modifier. Afficher les deux tableaux (avec filière, et sans filières).
- 5) Appliquer la destruction d'objet au module dont le code est 'M101' puis l'afficher.