

# JavaScript

## TD 1

Tom Mafille

27 novembre 2024

### PARTIE 1

1. Aller dans le navigateur, inspecter, console pour voir le résultat.

```
$( document ).ready(function() {  
    console.log('JAJ');  
});
```

### PARTIE 2

1. Définir les variables :

```
$(document).ready(function() {  
    // variables  
    var $ulElement = $('ul');  
    var $addButton = $('#addButton');  
  
    // pour visualiser  
    console.log($ulElement);  
    console.log($addButton);  
});
```

### PARTIE 3

1. Event bouton cliqué :

```
$(document).ready(function() {  
    var $ulElement = $('ul');  
    var $addButton = $('button');  
  
    $addButton.on('click', function() {  
        console.log('CLICK');
```

```
});  
});
```

## PARTIE 4

2. Print la valeur de l'input :

```
$(document).ready(function() {  
    var $ulElement = $('ul');  
    var $addButton = $('button');  
    var $inputElement = $('#item'); // # <=> by id  
  
    $addButton.on('click', function() {  
        var inputValue = $inputElement.val();  
        console.log('valeur de la sauce :', inputValue)  
    });  
});
```

## PARTIE 5

3. Créer un élément li avec cette valeur et l'ajouter à l'élément ul :

```
$(document).ready(function() {  
    var $ulElement = $('ul');  
    var $addButton = $('button');  
    var $inputElement = $('#item');  
  
    $addButton.on('click', function() {  
        var inputValue = $inputElement.val();  
  
        if (inputValue.trim() !== '') { //vérifier que c'est non NUL (comme Antoine)  
            var $liElement = $('<li></li>');  
            $liElement.text(inputValue);  
            $ulElement.append($liElement);  
            $inputElement.val('');  
        } else {  
            console.log('ya rien.');        }  
    });  
});
```

## PARTIE 6

1. Ajout d'un bouton Delete :

```
$(document).ready(function() {
    var $ulElement = $('ul');
    var $addButton = $('button');
    var $inputElement = $('#item');

    $addButton.on('click', function() {
        var inputValue = $inputElement.val();

        if (inputValue.trim() !== '') { //vérifier que c'est non NUL (comme Antoine
            var $liElement = $('<li></li>');
            $liElement.text(inputValue);
            $ulElement.append($liElement);
            $inputElement.val('');

            // Bouton DELETE
            var $deleteButton = $('<button>Delete</button>');
            // Ajout du bouton dans le <li>
            $liElement.append($deleteButton);

            // Ajout du bouton dans le <ul>
            $ulElement.append($liElement);
            // Supp
            $inputElement.val('');
        } else {
            console.log('ya rien. ');
        }
    });
});
```

## PARTIE 7

2. Suppression lorsqu'on presse sur le bouton Delete :

```
$(document).ready(function() {
    var $ulElement = $('ul');
    var $addButton = $('button');
    var $inputElement = $('#item');

    $addButton.on('click', function() {
```

```

var inputValue = $inputElement.val();

if (inputValue.trim() !== '') { //vérifier que c'est non NUL (comme Antoine)
    var $liElement = $('<li></li>');
    $liElement.text(inputValue);
    $ulElement.append($liElement);
    $inputElement.val('');

    // Bouton DELETE
    var $deleteButton = $('<button>Delete</button>');
    $deleteButton.on('click', function() {
        $liElement.remove(); //Quand click -> suppression
    });

    // Ajout du bouton dans le <li>
    $liElement.append($deleteButton);

    // Ajout du bouton dans le <ul>
    $ulElement.append($liElement);
    // Supp
    $inputElement.val('');
} else {
    console.log('ya rien.');
```

## BONUS

3. Pas d'ajout si champ texte vide, message d'erreur si champ vide (+ masquer si ok) :

```

$(document).ready(function() {
    var $ulElement = $('ul');
    var $addButton = $('button');
    var $inputElement = $('#item');

    var $errorMessage = $('<span style="color: red; display: none;">Le champ ne peut pas être vide</span>');
    $inputElement.after($errorMessage);

    $addButton.on('click', function() {
        var inputValue = $inputElement.val();
```

```

    if (inputValue.trim() !== '') { //vérifier que c'est non NUL (comme Antoine)
        $errorMessage.hide();

        // Création d'un <li>
        var $liElement = $('<li></li>');
        $liElement.text(inputValue);

        // Bouton DELETE
        var $deleteButton = $('<button>Delete</button>');
        $deleteButton.on('click', function() {
            $liElement.remove(); // Supprimer le <li> correspondant
        });

        // Ajouter le bouton au <li> et le <li> au <ul>
        $liElement.append($deleteButton);
        $ulElement.append($liElement);
        $inputElement.val('');
    } else {
        $errorMessage.show();
    }
});
});

```

4. Focus sur le champ texte après clic ajout + vider champ texte après clic ajout + Valider si utilisation de la touche entrée.

```

$(document).ready(function() {
    var $ulElement = $('ul');
    var $addButton = $('button');
    var $inputElement = $('#item');
    var $errorMessage = $('<span style="color: red; display: none;">Le champ ne peut pas être vide</span>');
    $inputElement.after($errorMessage);

    function addItem() {
        var inputValue = $inputElement.val();

        if (inputValue.trim() !== '') { //vérifier que c'est non NUL (comme Antoine)
            $errorMessage.hide();

            // Création d'un <li>
            var $liElement = $('<li></li>');
            $liElement.text(inputValue);

```

```

    // Bouton DELETE
    var $deleteButton = $('<button>Delete</button>');
    $deleteButton.on('click', function() {
        $liElement.remove(); // Supprimer le <li> correspondant
    });

    // Ajouter le bouton au <li> et le <li> au <ul>
    $liElement.append($deleteButton);
    $ulElement.append($liElement);

    // Réinitialiser + focus
    $inputElement.val('').focus();
} else {
    $errorMessage.show();
}
}

$addButton.on('click', function() {
    addItem();
});

// touche ENTREEEEEEEE
$inputElement.on('keypress', function(event) {
    if (event.key === 'Enter') {
        event.preventDefault(); // Empêcher le comportement par défaut
        addItem();
    }
});
});
});

```

5. Quantité d'articles dans le panier + message de confirmation pour supprimer un article

```

$(document).ready(function() {
    var $ulElement = $('ul');
    var $addButton = $('button');
    var $inputElement = $('#item');

    // Créer un message d'erreur
    var $errorMessage = $('<span style="color: red; display: none;">Le champ ne peut pas être vide</span>');
    $inputElement.after($errorMessage);

```

```

// Afficher la quantité d'articles dans le panier
var $itemCount = $('<p>Articles dans le panier : <span id="itemCount">0</span></p>');
$ulElement.after($itemCount);
var $itemCountSpan = $('#itemCount');

// Fonction pour mettre à jour la quantité d'articles
function updateItemCount() {
    var count = $ulElement.children('li').length;
    $itemCountSpan.text(count);
}

// Fonction d'ajout d'un élément à la liste
function addItem() {
    var inputValue = $inputElement.val();

    if (inputValue.trim() !== '') { // Vérifier que l'entrée n'est pas vide
        $errorMessage.hide();

        // Création d'un <li>
        var $liElement = $('<li></li>');
        $liElement.text(inputValue);

        // Bouton DELETE
        var $deleteButton = $('<button>Delete</button>');
        $deleteButton.on('click', function() {
            // Demander confirmation avant de supprimer
            var confirmDelete = confirm('Voulez-vous vraiment supprimer cet article?');
            if (confirmDelete) {
                $liElement.remove(); // Supprimer le <li> correspondant
                updateItemCount(); // Mettre à jour le compteur
            }
        });

        // Ajouter le bouton au <li> et le <li> au <ul>
        $liElement.append($deleteButton);
        $ulElement.append($liElement);

        // Réinitialiser et focus sur le champ input
        $inputElement.val('').focus();

        // Mettre à jour le compteur
        updateItemCount();
    }
}

```

```

    } else {
        $errorMessage.show();
    }
}

// Gestion du clic sur le bouton "Add"
$addButton.on('click', function() {
    addItem();
});

// Gestion de la touche "Entrée" dans le champ input
$inputElement.on('keypress', function(event) {
    if (event.key === 'Enter') {
        event.preventDefault(); // Empêcher le comportement par défaut
        addItem();
    }
});
});

```

6. Ajouter un texte quantité + un variable quantité de valeur 1. Ajouter un menu déroulant pour changement de quantité + bouton ok pour valider la nouvelle quantité Pour chaque ajout, générer une quantité stock max aléatoire et l'afficher pour chaque article Mettre une alerte “vous ne pouvez pas commander plus de x exemplaires” si la valeur précédente est dépassée.

```

$(document).ready(function() {
    var $ulElement = $('ul');
    var $addButton = $('button');
    var $inputElement = $('#item');
    var $errorMessage = $('<span style="color: red; display: none;">Le champ ne peut pas dépasser</span>');
    $inputElement.after($errorMessage);

    // Afficher la quantité d'articles dans le panier
    var $itemCount = $('<p>Articles dans le panier : <span id="itemCount">0</span></p>');
    $ulElement.after($itemCount);
    var $itemCountSpan = $('#itemCount');

    // Fonction pour mettre à jour la quantité d'articles
    function updateItemCount() {
        var count = $ulElement.children('li').length;
        $itemCountSpan.text(count);
    }
}

```



```

// Fonction d'ajout d'un élément à la liste
function addItem() {
    var inputValue = $inputElement.val();

    if (inputValue.trim() !== '') { // Vérifier que l'entrée n'est pas vide
        $errorMessage.hide();

        // Création d'un <li>
        var $liElement = $('<li></li>');
        $liElement.text(inputValue);

        // Bouton DELETE
        var $deleteButton = $('<button>Delete</button>');
        $deleteButton.on('click', function() {
            // Demander confirmation avant de supprimer
            var confirmDelete = confirm('Voulez-vous vraiment supprimer cet art');
            if (confirmDelete) {
                $liElement.remove(); // Supprimer le <li> correspondant
                updateItemCount(); // Mettre à jour le compteur
            }
        });

        // Ajouter le bouton au <li> et le <li> au <ul>
        $liElement.append($deleteButton);
        $ulElement.append($liElement);

        // Réinitialiser et focus sur le champ input
        $inputElement.val('').focus();

        // Mettre à jour le compteur
        updateItemCount();
    } else {
        $errorMessage.show();
    }
}

// Gestion du clic sur le bouton "Add"
$addButton.on('click', function() {
    addItem();
});

// Gestion de la touche "Entrée" dans le champ input

```

```

$inputElement.on('keypress', function(event) {
    if (event.key === 'Enter') {
        event.preventDefault(); // Empêcher le comportement par défaut
        addItem();
    }
});
});

```

7. Générer un prix HT et TTC pour chaque article. Générer le prix total du panier, sur un bouton ou de manière automatique.

```

$(document).ready(function() {
    var $ulElement = $('ul');
    var $addButton = $('button');
    var $inputElement = $('#item');

    // Créer un message d'erreur
    var $errorMessage = $('<span style="color: red; display: none;">Le champ ne peut pas être vide</span>');
    $inputElement.after($errorMessage);

    // Afficher la quantité d'articles dans le panier
    var $itemCount = $('<p>Articles dans le panier : <span id="itemCount">0</span></p>');
    $ulElement.after($itemCount);
    var $itemCountSpan = $('#itemCount');

    // Afficher le prix total
    var $totalPrice = $('<p>Total du panier : <span id="totalPrice">0.00</span> €</p>');
    $ulElement.after($totalPrice);
    var $totalPriceSpan = $('#totalPrice');

    // Fonction pour mettre à jour la quantité d'articles
    function updateItemCount() {
        var count = $ulElement.children('li').length;
        $itemCountSpan.text(count);
    }

    // Fonction pour calculer le prix total
    function updateTotalPrice() {
        var total = 0;
        $ulElement.children('li').each(function() {
            var priceTTC = parseFloat($(this).data('priceTTC'));
            total += priceTTC;
        });
    }

```

```

$totalPriceSpan.text(total.toFixed(2) + ' €');
}

// Fonction d'ajout d'un élément à la liste
function addItem() {
    var inputValue = $inputElement.val();

    if (inputValue.trim() !== '') {
        $errorMessage.hide();

        // Générer une quantité de stock maximum aléatoire
        var maxStock = Math.floor(Math.random() * 10) + 1; // Entre 1 et 10

        // Générer un prix HT aléatoire entre 5 et 50 €
        var priceHT = (Math.random() * (50 - 5) + 5).toFixed(2); // Entre 5 et
        var tvaRate = 0.20; // 20% de TVA
        var priceTTC = (priceHT * (1 + tvaRate)).toFixed(2); // Calcul du prix

        // Création d'un <li>
        var $liElement = $('<li></li>');
        $liElement.text(inputValue + ' - Quantité : ');

        // Afficher le prix HT et TTC
        var $priceInfo = $('<span>Prix HT : ' + priceHT + ' € | Prix TTC : ' +
        $liElement.append($priceInfo);

        // Ajout de la quantité initiale
        var $quantitySpan = $('<span> 1</span>');
        $liElement.append($quantitySpan);

        // Ajout d'une indication du stock maximum
        var $stockInfo = $('<span style="margin-left: 10px; color: gray;">(Stoc
        $liElement.append($stockInfo);

        // Menu déroulant pour changer la quantité
        var $quantitySelect = $('<select></select>');
        for (var i = 1; i <= maxStock; i++) {
            $quantitySelect.append('<option value="' + i + '">' + i + '</option>
        }

        // Bouton pour valider la nouvelle quantité
        var $validateButton = $('<button style="margin-left: 5px;">OK</button>');

```

```

$validateButton.on('click', function() {
    var selectedQuantity = parseInt($quantitySelect.val());
    if (selectedQuantity <= maxStock) {
        $quantitySpan.text(selectedQuantity);

        // Mettre à jour le prix TTC en fonction de la quantité
        var updatedPriceTTC = (selectedQuantity * priceTTC).toFixed(2);
        $liElement.data('priceTTC', updatedPriceTTC);
        updateTotalPrice(); // Recalculer le total
    } else {
        alert('Vous ne pouvez pas commander plus de ' + maxStock + ' ex');
    }
});

$liElement.append($quantitySelect);
$liElement.append($validateButton);

// Bouton DELETE
var $deleteButton = $('<button style="margin-left: 5px;">Delete</button>');
$deleteButton.on('click', function() {
    var confirmDelete = confirm('Voulez-vous vraiment supprimer cet article?');
    if (confirmDelete) {
        $liElement.remove();
        updateItemCount();
        updateTotalPrice(); // Recalculer le total après suppression
    }
});

// Ajouter le bouton au <li> et le <li> au <ul>
$liElement.append($deleteButton);
$ulElement.append($liElement);

// Stocker le prix TTC dans l'élément <li>
$liElement.data('priceTTC', priceTTC);

// Réinitialiser et focus sur le champ input
$inputElement.val('').focus();

// Mettre à jour le compteur et le prix total
updateItemCount();
updateTotalPrice();
} else {

```

```

        $errorMessage.show();
    }
}

// Gestion du clic sur le bouton "Add"
$addButton.on('click', function() {
    addItem();
});

// Gestion de la touche "Entrée" dans le champ input
$inputElement.on('keypress', function(event) {
    if (event.key === 'Enter') {
        event.preventDefault();
        addItem();
    }
});
});

```

8. Pas d'article identique en double Bouton de remise à zéro de la liste (au lieu de recharger la page)

```

$(document).ready(function() {
    var $ulElement = $('ul');
    var $addButton = $('button');
    var $inputElement = $('#item');

    // Créer un message d'erreur
    var $errorMessage = $('<span style="color: red; display: none;">Le champ ne peut pas
    $inputElement.after($errorMessage);

    // Afficher la quantité d'articles dans le panier
    var $itemCount = $('<p>Articles dans le panier : <span id="itemCount">0</span></p>');
    $ulElement.after($itemCount);
    var $itemCountSpan = $('#itemCount');

    // Afficher le prix total
    var $totalPrice = $('<p>Total du panier : <span id="totalPrice">0.00</span> €</p>');
    $ulElement.after($totalPrice);
    var $totalPriceSpan = $('#totalPrice');

    // Fonction pour mettre à jour la quantité d'articles
    function updateItemCount() {
        var count = $ulElement.children('li').length;
    }
}

```

```

    $itemCountSpan.text(count);
}

// Fonction pour calculer le prix total
function updateTotalPrice() {
    var total = 0;
    $ulElement.children('li').each(function() {
        var priceTTC = parseFloat($(this).data('priceTTC'));
        total += priceTTC;
    });
    $totalPriceSpan.text(total.toFixed(2) + ' €');
}

// Fonction d'ajout d'un élément à la liste
function addItem() {
    var inputValue = $inputElement.val().trim();

    if (inputValue === '') {
        $errorMessage.text("Le champ ne peut pas être vide.").show();
        return;
    }

    // Vérification des doublons
    var duplicate = false;
    $ulElement.children('li').each(function() {
        // Comparer le nom de l'article sans espaces supplémentaires et en ignorant
        var itemName = $(this).find('.itemName').text().trim().toLowerCase();
        if (itemName === inputValue.toLowerCase()) {
            duplicate = true;
        }
    });

    if (duplicate) {
        $errorMessage.text("Cet article existe déjà dans le panier.").show();
        return;
    }

    $errorMessage.hide();

    // Générer une quantité de stock maximum aléatoire
    var maxStock = Math.floor(Math.random() * 10) + 1; // Entre 1 et 10

```

```

// Générer un prix HT aléatoire entre 5 et 50 €
var priceHT = (Math.random() * (50 - 5) + 5).toFixed(2); // Entre 5 et 50
var tvaRate = 0.20; // 20% de TVA
var priceTTC = (priceHT * (1 + tvaRate)).toFixed(2); // Calcul du prix TTC

// Création d'un <li>
var $liElement = $('<li></li>');

// Créer un span pour le nom de l'article avec une classe pour la référence
var $itemNameSpan = $('<span class="itemName"></span>');
$itemNameSpan.text(inputValue);
$liElement.append($itemNameSpan);

// Afficher le prix HT et TTC
var $priceInfo = $('<span> Prix HT : ' + priceHT + ' € | Prix TTC : ' + priceTTC);
$liElement.append($priceInfo);

// Ajout de la quantité initiale
var $quantitySpan = $('<span> 1</span>');
$liElement.append($quantitySpan);

// Ajout d'une indication du stock maximum
var $stockInfo = $('<span style="margin-left: 10px; color: gray;">(Stock max : ' + maxStock);
$liElement.append($stockInfo);

// Menu déroulant pour changer la quantité
var $quantitySelect = $('<select></select>');
for (var i = 1; i <= maxStock; i++) {
    $quantitySelect.append('<option value="' + i + '">' + i + '</option>');
}

// Bouton pour valider la nouvelle quantité
var $validateButton = $('<button style="margin-left: 5px;">OK</button>');
$validateButton.on('click', function() {
    var selectedQuantity = parseInt($quantitySelect.val());
    if (selectedQuantity <= maxStock) {
        $quantitySpan.text(selectedQuantity);

        // Mettre à jour le prix TTC en fonction de la quantité
        var updatedPriceTTC = (selectedQuantity * priceTTC).toFixed(2);
        $liElement.data('priceTTC', updatedPriceTTC);
        updateTotalPrice(); // Recalculer le total
    }
});

```

```

        } else {
            alert('Vous ne pouvez pas commander plus de ' + maxStock + ' exemplaires');
        }
    });

    $liElement.append($quantitySelect);
    $liElement.append($validateButton);

    // Bouton DELETE
    var $deleteButton = $('<button style="margin-left: 5px;">Delete</button>');
    $deleteButton.on('click', function() {
        var confirmDelete = confirm('Voulez-vous vraiment supprimer cet article ?');
        if (confirmDelete) {
            $liElement.remove();
            updateItemCount();
            updateTotalPrice(); // Recalculer le total après suppression
        }
    });

    // Ajouter le bouton au <li> et le <li> au <ul>
    $liElement.append($deleteButton);
    $ulElement.append($liElement);

    // Stocker le prix TTC dans l'élément <li>
    $liElement.data('priceTTC', priceTTC);

    // Réinitialiser et focus sur le champ input
    $inputElement.val('').focus();

    // Mettre à jour le compteur et le prix total
    updateItemCount();
    updateTotalPrice();
}

// Fonction pour réinitialiser la liste
function resetList() {
    var confirmReset = confirm('Voulez-vous vraiment réinitialiser la liste ?');
    if (confirmReset) {
        $ulElement.empty();
        updateItemCount();
        updateTotalPrice();
    }
}

```



```

}

// Gestion du clic sur le bouton "Add"
$addButton.on('click', function() {
    addItem();
});

// Gestion de la touche "Entrée" dans le champ input
$inputElement.on('keypress', function(event) {
    if (event.key === 'Enter') {
        event.preventDefault();
        addItem();
    }
});

// Ajouter un bouton de remise à zéro de la liste
var $resetButton = $('<button>Réinitialiser la liste</button>');
$resetButton.on('click', function() {
    resetList();
});
$ulElement.after($resetButton);
});

```

7. Bouton imprimer la liste Bouton télécharger la liste au format PDF Bouton télécharger la liste en CSV (Intitulé, quantité, quantité max, prix ht, tva, prix ttc) Bouton importer une liste au format CSV

```

$(document).ready(function() {
    var $ulElement = $('ul');
    var $addButton = $('button');
    var $inputElement = $('#item');

    // Créer un message d'erreur
    var $errorMessage = $('<span style="color: red; display: none;">Le champ ne peut pas
    $inputElement.after($errorMessage);

    // Afficher la quantité d'articles dans le panier
    var $itemCount = $('<p>Articles dans le panier : <span id="itemCount">0</span></p>');
    $ulElement.after($itemCount);
    var $itemCountSpan = $('#itemCount');

    // Afficher le prix total
    var $totalPrice = $('<p>Total du panier : <span id="totalPrice">0.00</span> €</p>');

```

```

$ulElement.after($totalPrice);
var $totalPriceSpan = $('#totalPrice');

// Fonction pour mettre à jour la quantité d'articles
function updateItemCount() {
    var count = $ulElement.children('li').length;
    $itemCountSpan.text(count);
}

// Fonction pour calculer le prix total
function updateTotalPrice() {
    var total = 0;
    $ulElement.children('li').each(function() {
        var priceTTC = parseFloat($(this).data('priceTTC'));
        total += priceTTC;
    });
    $totalPriceSpan.text(total.toFixed(2) + ' €');
}

// Fonction d'ajout d'un élément à la liste
function addItem() {
    var inputValue = $inputElement.val().trim();

    if (inputValue === '') {
        $errorMessage.text("Le champ ne peut pas être vide.").show();
        return;
    }

    // Vérification des doublons
    var duplicate = false;
    $ulElement.children('li').each(function() {
        // Comparer le nom de l'article sans espaces supplémentaires et en ignorant
        var itemName = $(this).find('.itemName').text().trim().toLowerCase();
        if (itemName === inputValue.toLowerCase()) {
            duplicate = true;
        }
    });

    if (duplicate) {
        $errorMessage.text("Cet article existe déjà dans le panier.").show();
        return;
    }
}

```

```

$errorMessage.hide();

// Générer une quantité de stock maximum aléatoire
var maxStock = Math.floor(Math.random() * 10) + 1; // Entre 1 et 10

// Générer un prix HT aléatoire entre 5 et 50 €
var priceHT = (Math.random() * (50 - 5) + 5).toFixed(2); // Entre 5 et 50
var tvaRate = 0.20; // 20% de TVA
var priceTTC = (priceHT * (1 + tvaRate)).toFixed(2); // Calcul du prix TTC

// Création d'un <li>
var $liElement = $('<li></li>');

// Créer un span pour le nom de l'article avec une classe pour la référence
var $itemNameSpan = $('<span class="itemName"></span>');
$itemNameSpan.text(inputValue);
$liElement.append($itemNameSpan);

// Afficher le prix HT et TTC
var $priceInfo = $('<span> Prix HT : ' + priceHT + ' € | Prix TTC : ' + priceTTC + '</span>');
$liElement.append($priceInfo);

// Ajout de la quantité initiale
var $quantitySpan = $('<span> 1</span>');
$liElement.append($quantitySpan);

// Ajout d'une indication du stock maximum
var $stockInfo = $('<span style="margin-left: 10px; color: gray;">(Stock max : ' + maxStock + '</span>');
$liElement.append($stockInfo);

// Menu déroulant pour changer la quantité
var $quantitySelect = $('<select></select>');
for (var i = 1; i <= maxStock; i++) {
    $quantitySelect.append('<option value="' + i + '">' + i + '</option>');
}

// Bouton pour valider la nouvelle quantité
var $validateButton = $('<button style="margin-left: 5px;">OK</button>');
$validateButton.on('click', function() {
    var selectedQuantity = parseInt($quantitySelect.val());
    if (selectedQuantity <= maxStock) {

```

```

    $quantitySpan.text(selectedQuantity);

    // Mettre à jour le prix TTC en fonction de la quantité
    var updatedPriceTTC = (selectedQuantity * priceTTC).toFixed(2);
    $liElement.data('priceTTC', updatedPriceTTC);
    updateTotalPrice(); // Recalculer le total
  } else {
    alert('Vous ne pouvez pas commander plus de ' + maxStock + ' exemplaires');
  }
});

$liElement.append($quantitySelect);
$liElement.append($validateButton);

// Bouton DELETE
var $deleteButton = $('<button style="margin-left: 5px;">Delete</button>');
$deleteButton.on('click', function() {
  var confirmDelete = confirm('Voulez-vous vraiment supprimer cet article ?');
  if (confirmDelete) {
    $liElement.remove();
    updateItemCount();
    updateTotalPrice(); // Recalculer le total après suppression
  }
});

// Ajouter le bouton au <li> et le <li> au <ul>
$liElement.append($deleteButton);
$ulElement.append($liElement);

// Stocker le prix TTC dans l'élément <li>
$liElement.data('priceTTC', priceTTC);

// Réinitialiser et focus sur le champ input
$inputElement.val('').focus();

// Mettre à jour le compteur et le prix total
updateItemCount();
updateTotalPrice();
}

// Fonction pour réinitialiser la liste
function resetList() {

```

```

    var confirmReset = confirm('Voulez-vous vraiment réinitialiser la liste ?');
    if (confirmReset) {
        $ulElement.empty();
        updateItemCount();
        updateTotalPrice();
    }
}

// Fonction pour imprimer la liste
function printList() {
    window.print();
}

// Fonction pour télécharger en PDF
function downloadPDF() {
    var doc = new jsPDF();

    var content = [];
    $ulElement.children('li').each(function() {
        var itemName = $(this).find('.itemName').text();
        var priceTTC = $(this).data('priceTTC');
        content.push(itemName + ' | ' + priceTTC + ' €');
    });

    doc.text(content.join('\n'), 10, 10);
    doc.save('panier.pdf');
}

// Fonction pour télécharger en CSV
function downloadCSV() {
    var csvContent = "Item,Quantity,Max Stock,Price HT,TVA,Price TTC\n";
    $ulElement.children('li').each(function() {
        var itemName = $(this).find('.itemName').text();
        var quantity = $(this).find('span').eq(1).text().trim();
        var maxStock = $(this).find('span').eq(2).text().match(/\d+/)[0];
        var priceHT = $(this).find('span').eq(1).text().split(' | ')[0].split(': ');
        var tva = 0.20; // 20% TVA
        var priceTTC = $(this).data('priceTTC');
        csvContent += itemName + ',' + quantity + ',' + maxStock + ',' + priceHT + ','
    });

    var blob = new Blob([csvContent], { type: 'text/csv;charset=utf-8;' });

```

```

    var link = document.createElement('a');
    link.href = URL.createObjectURL(blob);
    link.download = "panier.csv";
    link.click();
}

// Fonction pour importer une liste CSV
function importCSV(event) {
    var file = event.target.files[0];
    var reader = new FileReader();
    reader.onload = function(e) {
        var lines = e.target.result.split('\n');
        for (var i = 1; i < lines.length; i++) {
            var data = lines[i].split(',');
            if (data.length >= 6) {
                var itemName = data[0];
                var quantity = parseInt(data[1]);
                var maxStock = parseInt(data[2]);
                var priceHT = parseFloat(data[3]);
                var priceTTC = parseFloat(data[5]);

                // Ajouter l'article à la liste
                var $liElement = $('<li></li>');
                var $itemNameSpan = $('<span class="itemName"></span>').text(itemName);
                $liElement.append($itemNameSpan);

                var $priceInfo = $('<span> Prix HT : ' + priceHT + ' € | Prix TTC : ');
                $liElement.append($priceInfo);

                var $quantitySpan = $('<span>' + quantity + '</span>');
                $liElement.append($quantitySpan);

                var $stockInfo = $('<span style="margin-left: 10px; color: gray;">(S)');
                $liElement.append($stockInfo);

                $liElement.data('priceTTC', priceTTC);
                $ulElement.append($liElement);

                updateItemCount();
                updateTotalPrice();
            }
        }
    }
}

```

```

    };
    reader.readAsText(file);
}

// Ajout des boutons supplémentaires
var $printButton = $('<button>Imprimer la liste</button>');
$printButton.on('click', printList);
$ulElement.after($printButton);

var $pdfButton = $('<button>Télécharger en PDF</button>');
$pdfButton.on('click', downloadPDF);
$ulElement.after($pdfButton);

var $csvButton = $('<button>Télécharger en CSV</button>');
$csvButton.on('click', downloadCSV);
$ulElement.after($csvButton);

var $importButton = $('<button>Importer une liste CSV</button>');
var $importInput = $('<input type="file" accept=".csv" style="display:none;">');
$importButton.on('click', function() {
    $importInput.click();
});
$importInput.on('change', importCSV);
$ulElement.after($importButton);
$ulElement.after($importInput);

// Gestion du clic sur le bouton "Add"
$addButton.on('click', function() {
    addItem();
});

// Gestion de la touche "Entrée" dans le champ input
$inputElement.on('keypress', function(event) {
    if (event.key === 'Enter') {
        event.preventDefault();
        addItem();
    }
});

// Ajouter un bouton de remise à zéro de la liste
var $resetButton = $('<button>Réinitialiser la liste</button>');
$resetButton.on('click', function() {

```

```

        resetList();
    });
    $ulElement.after($resetButton);
});

```

8. Alternance de couleurs dans la liste Bouton supprimer en rouge

```

$(document).ready(function() {
    var $ulElement = $('ul');
    var $addButton = $('button');
    var $inputElement = $('#item');

    // Créer un message d'erreur
    var $errorMessage = $('<span style="color: red; display: none;">Le champ ne peut pas
    $inputElement.after($errorMessage);

    // Afficher la quantité d'articles dans le panier
    var $itemCount = $('<p>Articles dans le panier : <span id="itemCount">0</span></p>');
    $ulElement.after($itemCount);
    var $itemCountSpan = $('#itemCount');

    // Afficher le prix total
    var $totalPrice = $('<p>Total du panier : <span id="totalPrice">0.00</span> €</p>');
    $ulElement.after($totalPrice);
    var $totalPriceSpan = $('#totalPrice');

    // Fonction pour mettre à jour la quantité d'articles
    function updateItemCount() {
        var count = $ulElement.children('li').length;
        $itemCountSpan.text(count);
    }

    // Fonction pour calculer le prix total
    function updateTotalPrice() {
        var total = 0;
        $ulElement.children('li').each(function() {
            var priceTTC = parseFloat($(this).data('priceTTC'));
            total += priceTTC;
        });
        $totalPriceSpan.text(total.toFixed(2) + ' €');
    }

    // Fonction d'ajout d'un élément à la liste

```



```

function addItem() {
    var inputValue = $inputElement.val().trim();

    if (inputValue === '') {
        $errorMessage.text("Le champ ne peut pas être vide.").show();
        return;
    }

    // Vérification des doublons
    var duplicate = false;
    $ulElement.children('li').each(function() {
        // Comparer le nom de l'article sans espaces supplémentaires et en ignorant
        var itemName = $(this).find('.itemName').text().trim().toLowerCase();
        if (itemName === inputValue.toLowerCase()) {
            duplicate = true;
        }
    });

    if (duplicate) {
        $errorMessage.text("Cet article existe déjà dans le panier.").show();
        return;
    }

    $errorMessage.hide();

    // Générer une quantité de stock maximum aléatoire
    var maxStock = Math.floor(Math.random() * 10) + 1; // Entre 1 et 10

    // Générer un prix HT aléatoire entre 5 et 50 €
    var priceHT = (Math.random() * (50 - 5) + 5).toFixed(2); // Entre 5 et 50
    var tvaRate = 0.20; // 20% de TVA
    var priceTTC = (priceHT * (1 + tvaRate)).toFixed(2); // Calcul du prix TTC

    // Création d'un <li>
    var $liElement = $('<li></li>');
    var isOdd = $ulElement.children('li').length % 2 === 0;
    $liElement.css('background-color', isOdd ? '#f9f9f8' : '#ffff2');

    // Créer un span pour le nom de l'article avec une classe pour la référence
    var $itemNameSpan = $('<span class="itemName"></span>');
    $itemNameSpan.text(inputValue);
    $liElement.append($itemNameSpan);

```

```

// Afficher le prix HT et TTC
var $priceInfo = $('<span> Prix HT : ' + priceHT + ' € | Prix TTC : ' + priceTTC);
$liElement.append($priceInfo);

// Ajout de la quantité initiale
var $quantitySpan = $('<span> 1</span>');
$liElement.append($quantitySpan);

// Ajout d'une indication du stock maximum
var $stockInfo = $('<span style="margin-left: 10px; color: gray;">(Stock max : ');
$liElement.append($stockInfo);

// Menu déroulant pour changer la quantité
var $quantitySelect = $('<select></select>');
for (var i = 1; i <= maxStock; i++) {
    $quantitySelect.append('<option value="' + i + '">' + i + '</option>');
}

// Bouton pour valider la nouvelle quantité
var $validateButton = $('<button style="margin-left: 5px;">OK</button>');
$validateButton.on('click', function() {
    var selectedQuantity = parseInt($quantitySelect.val());
    if (selectedQuantity <= maxStock) {
        $quantitySpan.text(selectedQuantity);

        // Mettre à jour le prix TTC en fonction de la quantité
        var updatedPriceTTC = (selectedQuantity * priceTTC).toFixed(2);
        $liElement.data('priceTTC', updatedPriceTTC);
        updateTotalPrice(); // Recalculer le total
    } else {
        alert('Vous ne pouvez pas commander plus de ' + maxStock + ' exemplaires');
    }
});

$liElement.append($quantitySelect);
$liElement.append($validateButton);

// Bouton DELETE
var $deleteButton = $('<button style="margin-left: 5px; color: white; background-color: #f00;">DELETE</button>');

$deleteButton.on('click', function() {

```

```

        var confirmDelete = confirm('Voulez-vous vraiment supprimer cet article ?');
        if (confirmDelete) {
            $liElement.remove();
            updateItemCount();
            updateTotalPrice(); // Recalculer le total après suppression
        }
    });

    // Ajouter le bouton au <li> et le <li> au <ul>
    $liElement.append($deleteButton);
    $ulElement.append($liElement);

    // Stocker le prix TTC dans l'élément <li>
    $liElement.data('priceTTC', priceTTC);

    // Réinitialiser et focus sur le champ input
    $inputElement.val('').focus();

    // Mettre à jour le compteur et le prix total
    updateItemCount();
    updateTotalPrice();
}

// Fonction pour réinitialiser la liste
function resetList() {
    var confirmReset = confirm('Voulez-vous vraiment réinitialiser la liste ?');
    if (confirmReset) {
        $ulElement.empty();
        updateItemCount();
        updateTotalPrice();
    }
}

// Fonction pour imprimer la liste
function printList() {
    window.print();
}

// Fonction pour télécharger en PDF
function downloadPDF() {
    var doc = new jsPDF();

```

```

    var content = [];
    $ulElement.children('li').each(function() {
        var itemName = $(this).find('.itemName').text();
        var priceTTC = $(this).data('priceTTC');
        content.push(itemName + ' | ' + priceTTC + ' €');
    });

    doc.text(content.join('\n'), 10, 10);
    doc.save('panier.pdf');
}

// Fonction pour télécharger en CSV
function downloadCSV() {
    var csvContent = "Item,Quantity,Max Stock,Price HT,TVA,Price TTC\n";
    $ulElement.children('li').each(function() {
        var itemName = $(this).find('.itemName').text();
        var quantity = $(this).find('span').eq(1).text().trim();
        var maxStock = $(this).find('span').eq(2).text().match(/\d+/)[0];
        var priceHT = $(this).find('span').eq(1).text().split(' | ')[0].split(': ');
        var tva = 0.20; // 20% TVA
        var priceTTC = $(this).data('priceTTC');
        csvContent += itemName + ',' + quantity + ',' + maxStock + ',' + priceHT + ',' + priceTTC + '\n';
    });

    var blob = new Blob([csvContent], { type: 'text/csv;charset=utf-8;' });
    var link = document.createElement('a');
    link.href = URL.createObjectURL(blob);
    link.download = "panier.csv";
    link.click();
}

// Fonction pour importer une liste CSV
function importCSV(event) {
    var file = event.target.files[0];
    var reader = new FileReader();
    reader.onload = function(e) {
        var lines = e.target.result.split('\n');
        for (var i = 1; i < lines.length; i++) {
            var data = lines[i].split(',');
            if (data.length >= 6) {
                var itemName = data[0];
                var quantity = parseInt(data[1]);
            }
        }
    };
    reader.readAsText(file);
}

```

```

    var maxStock = parseInt(data[2]);
    var priceHT = parseFloat(data[3]);
    var priceTTC = parseFloat(data[5]);

    // Ajouter l'article à la liste
    var $liElement = $('<li></li>');
    var $itemNameSpan = $('<span class="itemName"></span>').text(itemName);
    $liElement.append($itemNameSpan);

    var $priceInfo = $('<span> Prix HT : ' + priceHT + ' € | Prix TTC : ');
    $liElement.append($priceInfo);

    var $quantitySpan = $('<span>' + quantity + '</span>');
    $liElement.append($quantitySpan);

    var $stockInfo = $('<span style="margin-left: 10px; color: gray;">(S)');
    $liElement.append($stockInfo);

    $liElement.data('priceTTC', priceTTC);
    $ulElement.append($liElement);

    updateItemCount();
    updateTotalPrice();
  }
}

};
reader.readAsText(file);
}

// Ajout des boutons supplémentaires
var $printButton = $('<button>Imprimer la liste</button>');
$printButton.on('click', printList);
$ulElement.after($printButton);

var $pdfButton = $('<button>Télécharger en PDF</button>');
$pdfButton.on('click', downloadPDF);
$ulElement.after($pdfButton);

var $csvButton = $('<button>Télécharger en CSV</button>');
$csvButton.on('click', downloadCSV);
$ulElement.after($csvButton);

```

```

var $importButton = $('<button>Importer une liste CSV</button>');
var $importInput = $('<input type="file" accept=".csv" style="display:none;">');
$importButton.on('click', function() {
    $importInput.click();
});
$importInput.on('change', importCSV);
$ulElement.after($importButton);
$ulElement.after($importInput);

// Gestion du clic sur le bouton "Add"
$addButton.on('click', function() {
    addItem();
});

// Gestion de la touche "Entrée" dans le champ input
$inputElement.on('keypress', function(event) {
    if (event.key === 'Enter') {
        event.preventDefault();
        addItem();
    }
});

// Ajouter un bouton de remise à zéro de la liste
var $resetButton = $('<button>Réinitialiser la liste</button>');
$resetButton.on('click', function() {
    resetList();
});
$ulElement.after($resetButton);
});

```

9. Titre de page web, couleurs variées, image/texture de fond de site ou d'item Boutons arrondis Animations