

Conception Web 1 - TP3

Formulaire, Javascript

Dans ce TP, vous développerez la page de réservation à l'aide d'un formulaire et mettrez en place des vérifications testant les données entrées par l'utilisateur.

L'objectif est de ce TP est de :

- Développer un formulaire.
- Maîtriser les options de validation des entrées utilisateurs (en HTML).
- Utiliser du Javascript pour améliorer la validation des entrées utilisateurs.

Exercice 1 – Formulaire

Le but de cette question est de compléter la page **reservation.html** (créée au TP2) à l'aide d'un formulaire.

The screenshot shows a web browser window with the address bar displaying a file path. The page has a yellow background with a green rectangular area containing the reservation form. At the top of the form is a mountain icon and the title 'PAS LOIN PAS CHER !'. Below the title is a navigation menu with links: 'Notre agence', 'Nos voyages', 'Informations', and 'Réservation'. The main heading for the form is 'Veuillez entrer vos coordonnées pour réserver votre voyage.'.

The form fields and their current values are:

- Nombre de personnes : Ex: 2 (*)
- Date de réservation : jj / mm / aaaa (*)
- Destination choisie : Foix (*)
- Vérifier la disponibilité (button)
- Appellation : Mme (*)
- Nom : Entrez votre nom (*)
- Numéro de téléphone : Ex: 0123456789 (*)
- Adresse e-mail : Ex: prenom.nom@fai.com (*)
- Moyen de paiement : Chèques vacances (*)
- Commentaires : Entrez votre commentaire (text area)
- ☒ Je souhaite recevoir des offres promotionnelles.
- ☐ J'ai lu les conditions générales de vente. (*)
- Envoi (button)
- Reset (button)

Below the form, it states: 'Les informations suivies d'un astérisque sont obligatoires.'

At the bottom, there is a footer with copyright information and contact details for two locations:

- Toulouse : 21 rue de Metz, 31000 Toulouse. Tel : 012 98 76 54 32
- Bordeaux : 134 Quai de Bacalan, 33000 Bordeaux. Tel : 012 98 76 54 31

Figure 1. Formulaire de la page **reservation.html**.

1. Créez le formulaire (cf. **Figure 1**) en respectant les consignes suivantes :
 - a. Le formulaire utilise la méthode GET et son action est d'ouvrir la page **confirmation.html** (elle sera créée à la question suivante).
 - b. L'input pour nombre de personnes n'affiche que des chiffres.
 - c. L'input pour date de réservation n'affiche que des dates.
 - d. L'input pour le choix de destination propose à l'utilisateur de choisir parmi les choix suivants : "Foix", "Sarlat", "Gorge de l'Ardèche" et "Canal du Midi".
 - e. "Vérifier la disponibilité" est un input de type "button".
 - f. L'input pour Appellation permet de choisir entre "M." et "Mme".
 - g. Les inputs de nom et numéro de téléphone permettent d'afficher du texte sur 1 ligne.
 - h. L'input pour l'adresse email limite le type d'entrée à des emails.
 - i. L'input pour moyen de paiement est un radio Button (note : d'autres choix seront ajoutés lors d'une question suivante).
 - j. L'input pour Commentaires permet d'écrire plusieurs lignes de texte.
 - k. La case correspondant à « Je souhaite recevoir des offres promotionnelles » est cochée par défaut et celle pour « J'ai lu les conditions initiales » est non cochée.
 - l. Chaque input aura des attributs name et id (uniques, sauf pour les radio buttons).
2. Le formulaire contiendra 2 boutons : "Envoi" (qui soumettra le formulaire) et "Reset" (qui réinitialisera le formulaire).
3. Créer la page **confirmation.html** (sur le modèle de la **Figure 2**). Elle affichera un message prédéfini "Nous confirmons avoir bien reçu votre réservation le XX/XX/XXXX à X:XX. A bientôt !".
4. Tester les boutons "Envoi" et "Reset".

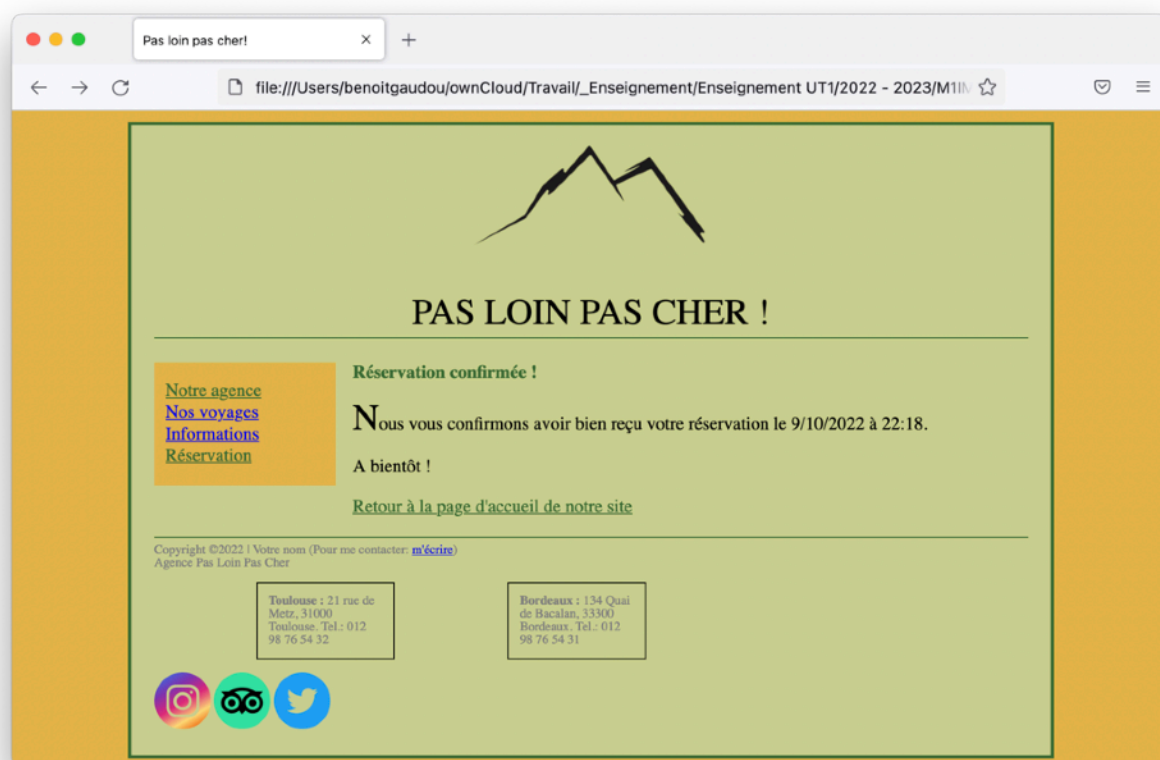


Figure 2. Page de confirmation

5. Aide à l'utilisateur : ajouter pour chaque input (lorsque c'est pertinent) les attributs **title** et **placeholder**, pour aider l'utilisateur à remplir le formulaire.
6. Validation des entrées utilisateur (en HTML) :
 - a. Tous les inputs suivis d'un astérisque doivent être remplis.
 - b. Le numéro de téléphone doit être composé de 10 chiffres (à vérifier avec l'attribut **pattern**).

Exercice 2 – Javascript

L'objectif de cet exercice est d'ajouter l'appel à plusieurs fonctions Javascript afin d'améliorer la vérification des entrées utilisateurs.

Confirmer l'envoi d'un formulaire.

1. En utilisant la fonction Javascript `confirm`, lorsque l'utilisateur clique sur envoi, un pop-up s'affiche demandant confirmation. Si l'utilisateur annule l'envoi, il ne doit pas être amené sur la page de confirmation. (Cf. Cours).

Valider un formulaire.

Vous allez maintenant créer une fonction JavaScript permettant de tester la possibilité du voyage demandé en fonction du nombre de voyageur.

2. Vous allez maintenant créer un nouveau fichier, `script.js`, qui contiendra les scripts Javascript et la lier avec la page HTML `reservation.html`.
3. Créez une fonction (`testDisponibilite`) qui vérifie le nombre de personnes pour le voyage.
 - a. Dans le fichier `script.js`, créez une fonction qui prend l'id d'un élément en paramètre. Cette fonction va d'abord récupérer l'élément correspondant à cet id, puis tester si la valeur entrée par l'utilisateur est plus petite que 0. Si c'est le cas, la fonction affichera un pop-up alertant que le nombre de personnes doit être positif. (Cf. Cours)
 - b. Dans le fichier `reservation.html`, appelez cette fonction (avec en paramètre l'id de la zone de texte où le nombre de participants est entré) lorsque l'utilisateur clique sur le bouton « Vérifier la disponibilité » (attribut **onClick**).
 - c. Le nombre de participants à un voyage ne peut pas dépasser 20 personnes. Modifiez la fonction précédente pour qu'elle affiche un message lorsque le nombre de participants dépasse 20.
 - d. Cette fonction retourne `true` si le nombre de participants est correct (entre 0 et 20) et `false` sinon (cf. Cours).
 - e. Dans le fichier `reservation.html`, appelez cette fonction lorsque l'utilisateur clique sur le bouton Envoi du formulaire (cf. Cours).

Modifier l'affichage de la page en fonction d'actions de l'utilisateur.

Un nouveau moyen de paiement peut maintenant être proposé pour les commandes : le paiement par virement. La page `reservation.html` doit être modifiée de manière à ce que :

- L'utilisateur puisse choisir par le biais de bouton radio le nouveau mode de paiements : virement (en plus de chèque voyage)
- Si c'est le paiement par virement qui est choisi, l'utilisateur verra apparaître un code IBAN, et un code BIC.
- Par défaut, c'est le paiement par carte bancaire qui est sélectionné.

Moyen de paiement :	<input checked="" type="radio"/> Virement	<input type="radio"/> Chèques vacances	<input type="radio"/> (*)
	IBAN :	<input type="text" value="FR07 XXXX XXXX XXXX XXXX"/>	

4. Dans la page HTML `reservation.html`, ajoutez un second radio Button pour le mode de paiement.
5. Dans la même cellule du tableau, rajoutez une balise `div` contenant le texte « IBAN : » et l'input permettant d'entrer l'IBAN. Ce `div` sera identifié par un `id` « `divIBAN` ».
6. Lorsque l'utilisateur clique sur le radio Button `Virement`, la fonction « `afficherIBAN()` » est appelée. Lorsque l'utilisateur clique sur le radio Button `Chèques vacances`, la fonction « `cacheIBAN()` » est appelée.
7. Dans le fichier `script.js`, écrire les 2 fonction `afficherIBAN()` et `cacheIBAN()` :
 - a. Dans chaque fonction, commencez par récupérer l'élément ayant l'identifiant `divIBAN`. Vous pouvez maintenant changer son attribut `display` de style (la valeur « `none` » permet de le cacher, « `block` » permet de l'afficher). Exemple :
`elt.style.display = "block";`

Retour sur la validation

Nous souhaitons maintenant que la vérification du nombre de personnes dépendent aussi des autres choix de l'utilisateur :

- Le voyage à Foix n'est disponible que pour 5 personnes ou plus.
 - Le mode de paiement « Chèques vacances » n'est possible que pour 1 ou 2 personnes.
8. Modifiez la fonction `testDisponibilite` définie précédemment pour ajouter ces 2 règles¹².

¹ Dans une fonction JavaScript, si la variable **moyen_paiement** contient un input de type radio button, alors **moyen_paiement.checked** renverra `true` si le button est sélectionné, `false` sinon.

² En JS, la conjonction de 2 conditions (i.e. `and`) est notée `&&`.