

Bachelier en Informatique de Gestion

Web : principes de base Projet de Développement Web

Enseignement supérieur économique de type court

Code FWB : 7534 29 U32 D1, 7534 30 U32 D3

Code ISFCE : 4IWPB, 4IPW3



Table des matières

Généralités

- 01. Introduction au web
- 03. Outils
- 05. Format XML
- 06. Format JSON

Front-End

- 12. Structure HTML
- 13. Formulaire HTML
- 14. Mise en forme CSS
- 15. Adaptabilité
- 17. Javascript
- 18. Bibliothèque jQuery
- 19. Composant Vue.js

Back-End

- 21. Middleware PHP
- 22. Traitement du formulaire
- 23. Architecture MVC
- 24. Données SQL
- 25. Données NoSQL
- 27. Requête asynchrone



13. Formulaires HTML

Objectif

La balise INPUT

Transit des données

La balise BUTTON

La balise FORM

Les autres balises



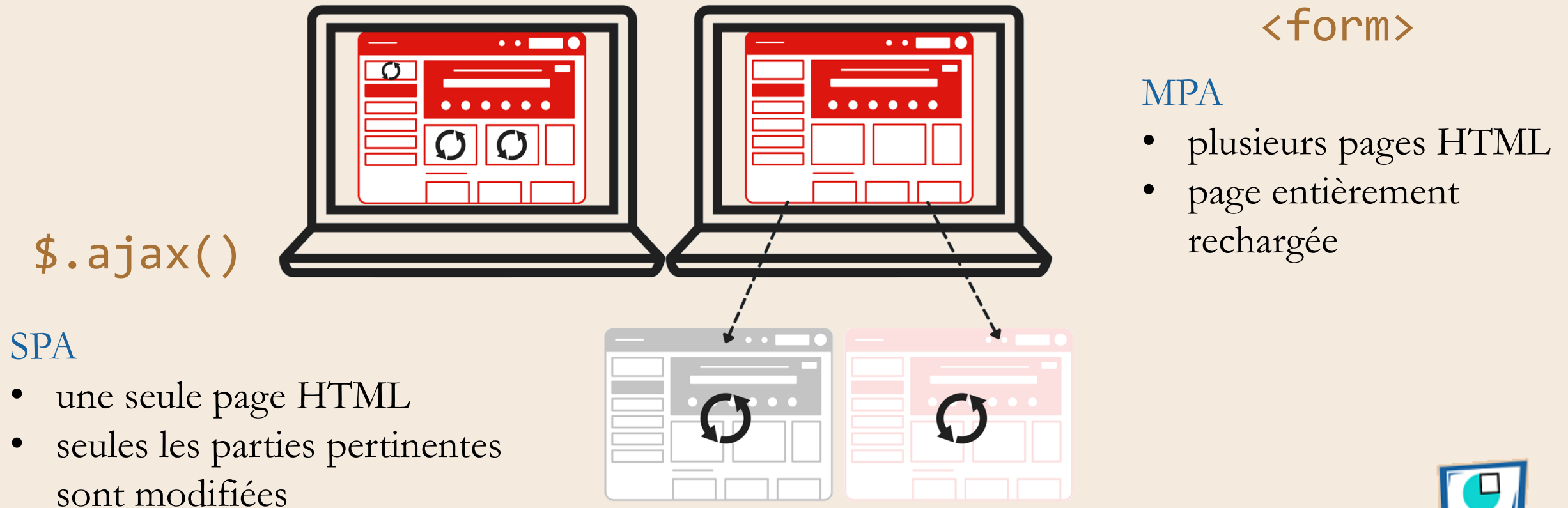
= BUROTIX ()



Transit des données

Single Page App. vs Multiple Page App.

SPA vs MPA



MPA vs SPA, use cases

MPA

- app statique
- services B2B
- e-commerce

SPA

- app dynamique
- social networks
- streaming services
- real-time location services

[discussion sur Reddit](#)



= BUROTIX ()

Objectifs des formulaires

- Approche MPA
 - création d'un formulaire via la balise `<form>`
- Approche SPA
 - création d'un formulaire muni de requêtes AJAX
- Les formulaires sont utilisés pour récolter des informations des utilisateurs.
- Deux problèmes:
 - Comment envoyer les données au serveur ?
 - Comment le serveur traite-t-il les données reçues?



Comment faire transiter les données?

- Deux méthodes :
 - **GET**: envoie les données dans l'URL de la page
 - `https://www.google.com/search?q=developpement+photo`
 - Limité à 255 caractères
 - Paramètres visibles
 - **POST**: envoie les données via la requête HTTP
 - Permet de faire transiter un plus gros nombre de caractères
 - Paramètres invisibles
- Défini avec l'attribut "**method**" de la balise **<form>**
 - `<form method="get" ...`
 - `<form method="post" ...`



Comment traiter les données?

- Il faut envoyer la requête contenant les données du formulaire (via GET ou POST) à un **script** qui pourra les traiter (ex. page contenant du PHP)
- Défini avec l'attribut action

`<form method="get" action="./register.php" ...`



Exemple de formulaire

- Déclaration d'un formulaire

```
<form method="get" action="./register.php">
```

```
...
```

```
</form>
```





Balises HTML



= BUROTIX ()

La balise `<input>`

- L'élément le plus populaire d'un formulaire est la balise `<input>`.
- `<input>` est utilisé pour recueillir l'information de l'utilisateur.
- `<input>` peut être utilisé de nombreuses manières différentes en fonction de la valeur de son attribut `type`.



La balise <input>

- Champs de texte

Prénom:

Jean

```
<input type="text" name="prenom">
```

- Mot de passe

```
<input type="password" name="pwd">
```

- Bouton radio

```
<input type="radio" name="sexe" value="m">Homme
```

```
<input type="radio" name="sexe" value="f">Femme
```

☒ Homme

☐ Femme

ATTENTION : La valeur de "name" doit être la même pour tous les boutons.

La balise <input>

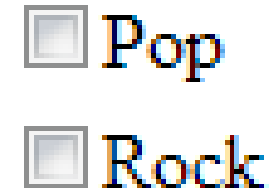
■ Checkbox

```
<input type="checkbox"
       name="type_musique[]"
       value="pop">
```

Pop

```
<input type="checkbox"
       name="type_musique[]"
       value="rock">
```

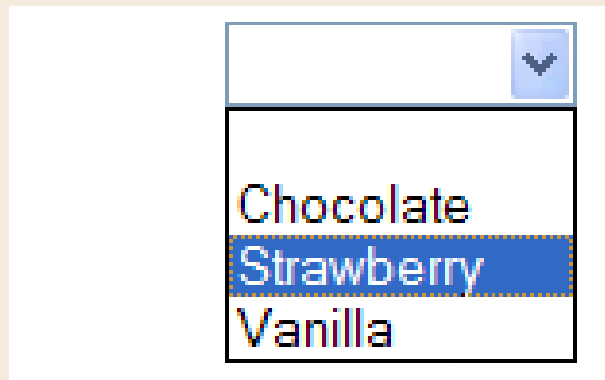
Rock



La balise <select>

■ Select

```
<select name="voiture">  
  <option value=""></option>  
  <option value="c">Chocolate</option>  
  <option value="s" selected>Strawberry</option>  
  <option value="v">Vanilla</option>  
</select>
```



L'attribut “value” de la balise <input>

- Lorsque le formulaire est envoyé, ce sont les valeurs des attributs **value** qui sont envoyés au serveur, liées au nom de l'input (attribut **name**).
- Ecrire dans un **input** de type texte modifie son attribut **value**.

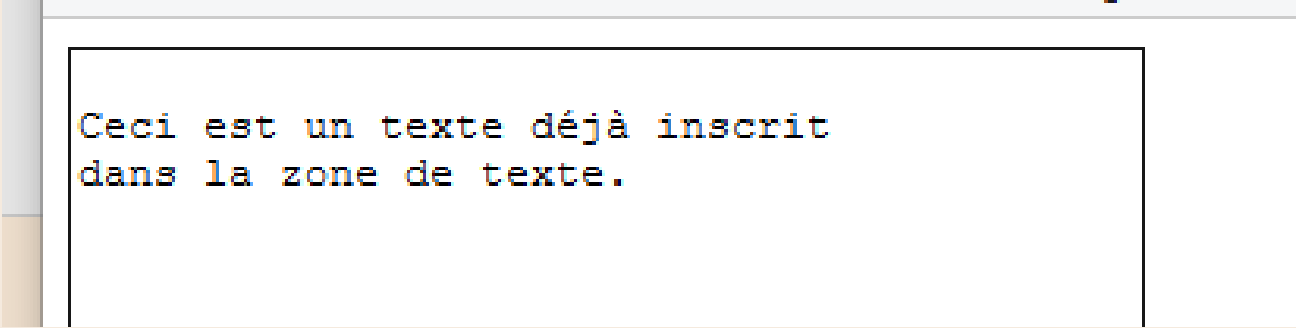
La balise <textarea>

- Zone de texte

```
<textarea rows="10" cols="30">
```

Ceci est un texte déjà inscrit
dans la zone de texte.

```
</textarea>
```



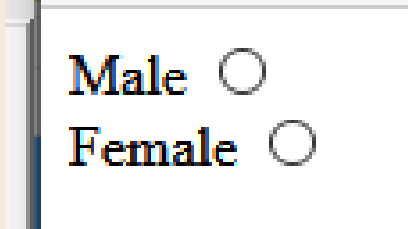
Ceci est un texte déjà inscrit
dans la zone de texte.



La balise <label>, 1^{ère} manière

- Cherchez son utilité !

```
<label>  
  Male  
  <input type="radio"  
    name="sex"  
    value="male" >  
</label>  
<label>  
  Female  
  <input type="radio"  
    name="sex"  
    value="female">  
</label>
```



Male ☒
Female ☐



La balise <label>, 2^{ème} manière

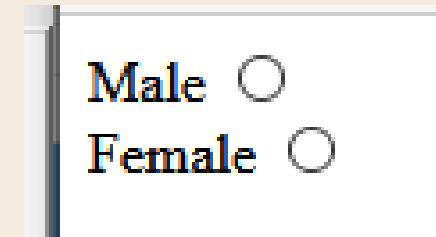
- Cherchez son utilité !

```
<label for="male">Male</label>
```

```
<input type="radio"  
       name="sex"  
       id="male"  
       value="male" >
```

```
<label for="female">Female</label>
```

```
<input type="radio" name="sex"  
       id="female" value="female">
```



Male ☐
Female ☐



La balise <datalist>

- Cherchez son utilité !

```
<input list="browsers">
```

```
<datalist id="browsers">
```

```
  <option value="Internet Explorer">
```

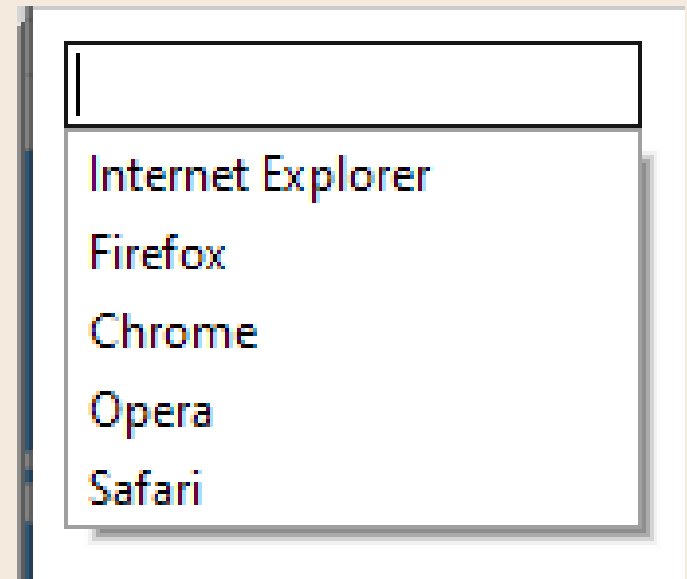
```
  <option value="Firefox">
```

```
  <option value="Chrome">
```

```
  <option value="Opera">
```

```
  <option value="Safari">
```

```
</datalist>
```



A screenshot of a web browser's address bar. The address bar is white with a black border. Below it, a dropdown menu is open, displaying five browser names: Internet Explorer, Firefox, Chrome, Opera, and Safari. The text is in a standard sans-serif font.



Balises HTML 5

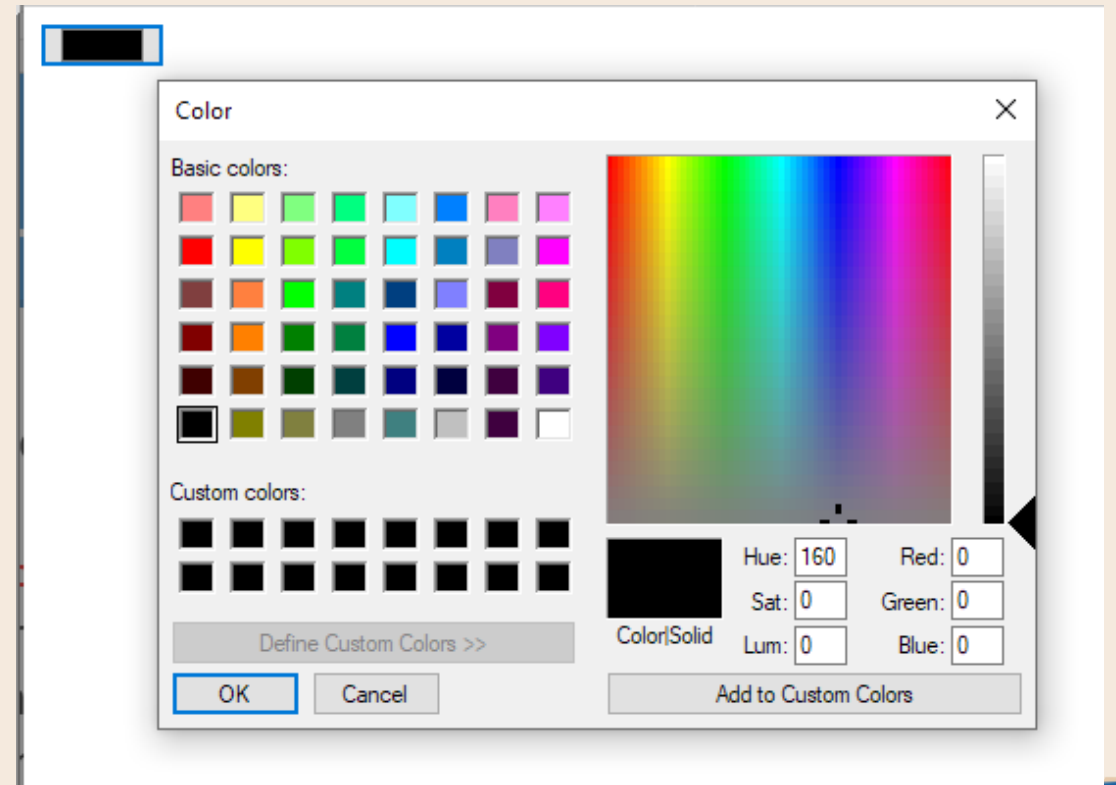


= BUROTIX ()

Input Types introduits en HTML5

- color
- date
- datetime
- datetime-local
- email
- month
- number
- range
- search
- tel (mobile)
- time
- url
- week

`<input type="color">`



Attributs - HTML(5)

■ Quelques attributs utiles pour **<input>** :

- **checked** (checkbox, radio)
- **disabled**
- **height** et **width**
- **min** et **max** (nombre, date)
- **maxlength** (nombre

caractère max.)

- **multiple** (email, file)
- **pattern** (regexp)
- **placeholder** (exemple d'input de l'élément)
- **required**
- **value**



Soumettre un formulaire



- Une fois rempli, le formulaire est "soumis au serveur", c  d que les donn  es y sont envoy  es.
- Bouton de type “submit”
 - `<button type="submit">S 'inscrire</button>`
 - remarque : il existe un codage ancien
 - `<input type="submit" value="S 'inscrire">`
- Bouton de type “button”
 - appel d'un code JavaScript = programmation !
 - formulaire non envoy   automatiquement au serveur
 - exemple : un tel bouton peut valider les input, envoyer le form au serveur et afficher "veuillez patienter".



Exo 01 : Sample "Atomic" Form

- Télécharger et placer dans le WAMP
 - `exo06-01_sample_atomic_form.html`
- Exemple de
 - Input type text
 - Input type password
 - Input type radio
 - Input type checkbox
 - Select option
 - Input + Datalist
 - Input type color
 - Input type range
 - Button type submit
 - Button type button



Exo 03 : Sample "Molecular" Form

- Télécharger et placer dans le WAMP les deux fichiers
 - `exo06-03_sample_atomic_form.html`
- Exemple de
 - Image Upload
 - File Upload



Exo 05 : Catalogue + Form Input

- Créer un formulaire qui permet d'ajouter des éléments dans le tableau HTML.
- Ci-contre un exemple de formulaire.

Catalogue	
produit	prix
bière	10
sandwich	3
verre	5
assiette	32
<input type="text"/>	<input type="text"/>
<input type="button" value="add"/>	



Exo 06 : commande de sandwich

- Créez un formulaire permettant de commander un sandwich
 - votre nom
 - baguette : blanc ou gris
 - garniture principale : poulet curry, thon piquant, boulette
 - garniture secondaire : carotte, salade, concombre
 - avec ou sans mayonnaise

