

The logo of the University of Bordeaux is displayed against a background with a blue diagonal stripe in the top left and a dark grey diagonal stripe in the bottom right. The text 'université' is in a dark brown sans-serif font, with the 'u' and 'e' featuring blue highlights. Below it, 'de' is in a smaller dark brown font, and 'BORDEAUX' is in a larger, bold, dark brown sans-serif font.

université  
de **BORDEAUX**

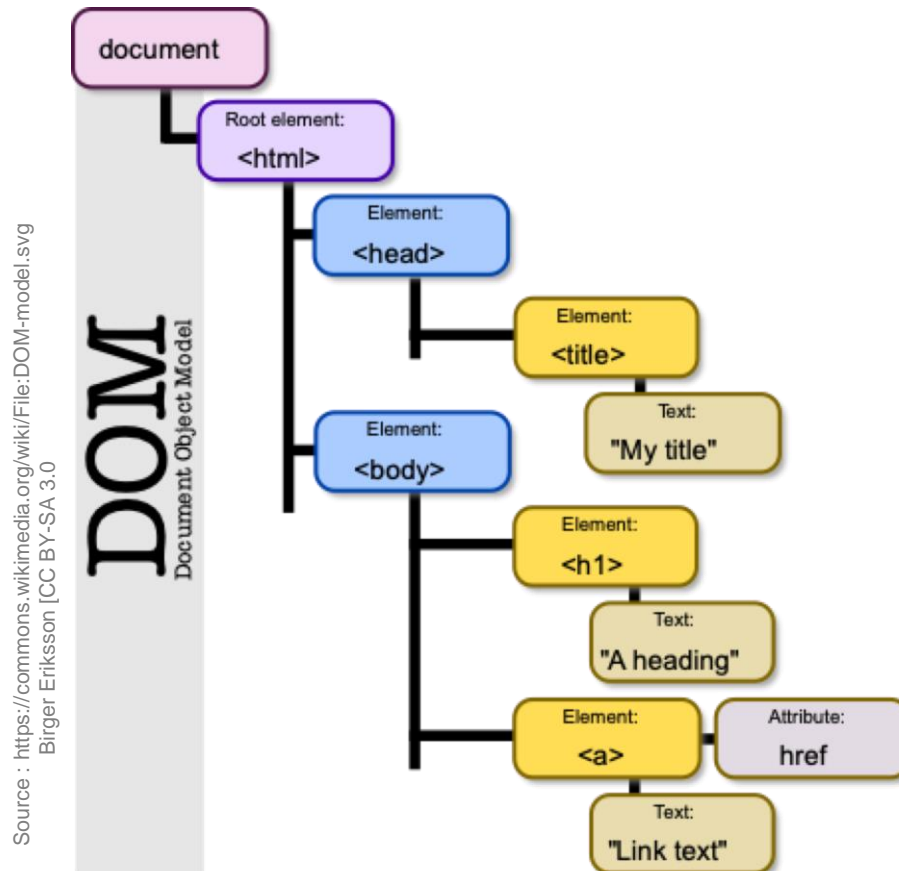
# Bloc 1

JavaScript

- JavaScript
  - › 1995 (début), ECMA standard (1997), ECMAScript (2018)
- Exécuté dans un navigateur web
  - › Chrome, Firefox, Safari, etc.
- Programme le comportement des pages web
  - › Ajouter, changer et retirer tous les éléments et les attributs HTML.
  - › Ajouter, changer et retirer tous les styles CSS.
  - › Ajouter, changer, retirer et réagir aux événements HTML.
- Utilisé dans d'autres contextes
  - › Programmes de bureau et serveur

# HTML et JavaScript (1/3)

- Lorsqu'une page Web est chargée, le navigateur crée un Document Object Model (DOM)
- Le code JavaScript s'exécute sur le DOM



# HTML et JavaScript (2/3)

- Ajouter le code JavaScript dans la page HTML à l'intérieur d'une balise **<script>** :

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8" />
  </head>
  <body>
    <script>
      window.addEventListener('load', function () {
        console.log('Cette fonction est exécutée une fois
        quand la page est chargée.');
      });
    </script>
  </body>
</html>
```

# HTML et JavaScript (3/3)

→ Mettre le code JavaScript dans un fichier externe (.js)

```
console.log('Cette fonction est exécutée une fois  
quand la page est chargée.');
```

et pointer ce fichier depuis le HTML :

```
<!DOCTYPE html>  
<html>  
  <head>  
    <meta charset="utf-8" />  
  </head>  
  <body>  
    <script src="./script.js">  
    </script>  
  </body>  
</html>
```

# DOM Element

- Les éléments DOM sont des objets JavaScript qui proposent une API.
- Pour manipuler des éléments HTML il faut d'abord les trouver:
  - › id
  - › nom de balise
  - › nom de classe
  - › sélecteurs CSS

```
var target = document.getElementById("monId");  
var img = document.createElement('img');  
img.src = './img/02.BMP';  
target.appendChild(img);
```

# DOM Event

- Le DOM émet des événements (DOM Event) lorsque ses éléments (DOM Element) subissent des interactions.
  - › Lorsqu'un utilisateur clique sur la souris
  - › Quand une page Web/ image est chargée
  - › Quand la souris passe sur un élément
  - › Lorsqu'un champ de saisie est modifié
  - › Lorsqu'un formulaire HTML est soumis
- JavaScript permet d'ajouter des traitements (fonctions *callbacks*) qui seront exécutés lorsqu'un événement sera émis.

```
clickAjoutCarte() {  
  let img = document.createElement('img');  
  img.src = './img/01.BMP';  
  document.getElementById("mes-cartes").appendChild(img)  
}  
document.getElementById("ajout-carte").onclick(clickAjoutCarte);
```



# Glisser-Déposer (*Drag and Drop*) (1/3)

- Depuis la version 5 d'HTML, tout élément peut devenir déplaçable en mettant son attribut `draggable` à `true`.
- Trois *callbacks* doivent être spécifiés:
  - › **ondragstart**: émis lorsque d' clique sur un élément à déplacer .
  - › **ondragover** : émis lorsque l'élément déplacé survole un autre éléments.
  - › **ondrop**: émis lorsque l'élément déplacé est déposé sur un autre élément.

# Glisser-Déposer (*Drag and Drop*) (2/3)

```
<!DOCTYPE HTML>
<html>
  <head>
    <script>
function allowDrop(ev) {
  ev.preventDefault();
}

function drag(ev) {
  ev.dataTransfer.setData("text", ev.target.id);
}

function drop(ev) {
  ev.preventDefault();
  var data = ev.dataTransfer.getData("text");
  ev.target.appendChild(document.getElementById(data));
}
    </script>
  </head>
```

# Glisser-Déposer (*Drag and Drop*) (3/3)

```
<body>
```

```
<div id="div1" ondrop="drop(event)" ondragover="allowDrop(event)"></div>
```

```

```

```
</body>
```

```
</html>
```

# Résumé

- JavaScript est un langage interprété
- JavaScript s'exécute sur l'**arbre d'éléments** HTML (DOM tree) crée par le navigateur
- Exploité par les **navigateurs** pour manipuler le comportement des pages web
- Utilisé dans d'autres contextes  
(Programmes de bureau et serveur)