Programmation et projet encadré - L7TI005

mini-projet 2 : préparation au travail en groupe

Yoann Dupont yoann.dupont@sorbonne-nouvelle.fr
Pierre Magistry pierre.magistry@inalco.fr

2024-2025

Université Sorbonne-Nouvelle
INALCO
Université Paris-Nanterre

Les enjeux de cette portion de cours :

- continuer le mini-projet
- utiliser un format plus adapté pour le rendu final en groupe
  - sortie simple tabulaire
  - aujourd'hui : transformation en tableau HTML
- si le temps : ajouter d'autres colonnes au tableau

HTML

HTML, c'est quoi

HTML (**H**yper**T**ext **M**arkup **L**anguage) est un langage de balisage pour représenter des pages web. Format reconnu par tous les navigateurs.

HTML, c'est quoi

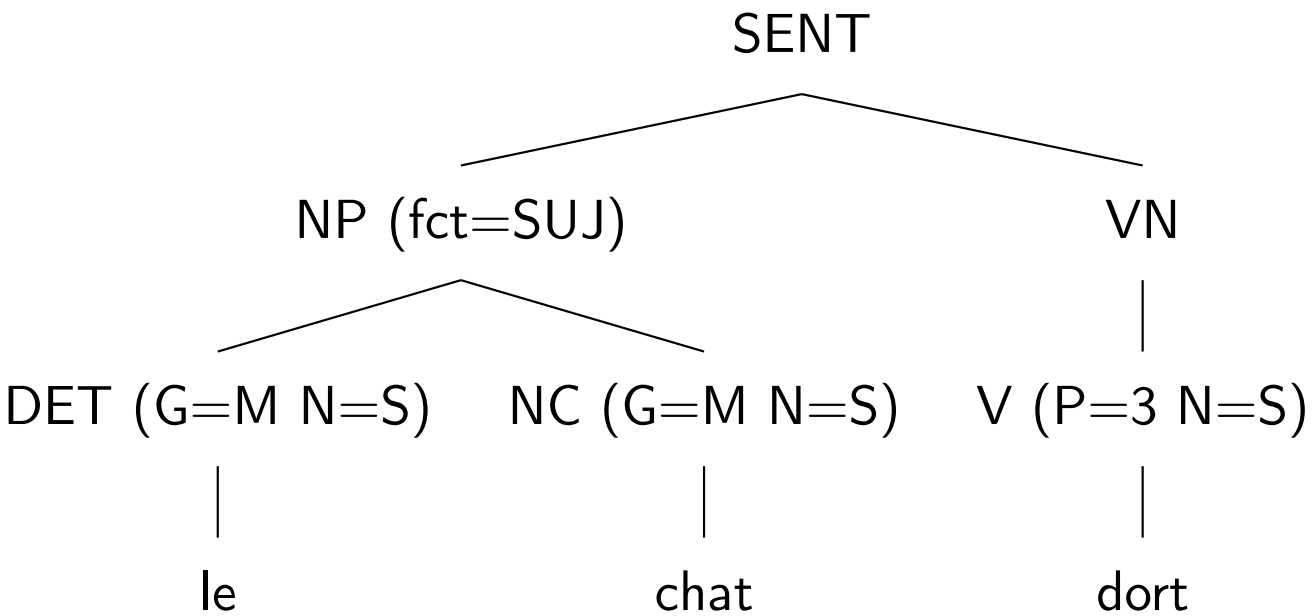
HTML (**H**yper**T**ext **M**arkup **L**anguage) est un langage de balisage pour représenter des pages web. Format reconnu par tous les navigateurs.

Permet de structurer l'information d'un page pour la rendre lisible :

- Dérivé du SGML (**S**tandard **G**eneralized **M**arkup **L**anguage) et "frère" du XML
- Permet de marquer des zones dans du contenu textuel
- Ces zones fournissent structure et enrichissements

Du balisage, à quoi ça ressemble ? I

HTML définit des balises qui marquent explicitement le début et la fin d'une zone. On peut inclure des balises dans d'autres, mais pas de chevauchement : HTML se rapproche donc des constituants syntaxiques :



Du balisage, à quoi ça ressemble ? II

Pour moins d'ambiguïté, les balises sont marquées explicitement. il y en a 3 types :

- Ouvrantes : <balise> → le début d'une zone
- Fermantes : </balise> → la fin d'une zone
- Autofermantes ou vides : <balise/> → position dans le document[1]

[1] Techniquement, cette syntaxe n'est nécessaire qu'en XML. En HTML, il est préférable de s'en passer, comme pour
 par exemple. Source: https://developer.mozilla.org/fr/docs/Glossary/Void_element

Du balisage, à quoi ça ressemble ? II

Pour moins d'ambiguïté, les balises sont marquées explicitement. il y en a 3 types :

- Ouvrantes : <balise> → le début d'une zone
- Fermantes : </balise> → la fin d'une zone
- Autofermantes ou vides : <balise/> → position dans le document[1]

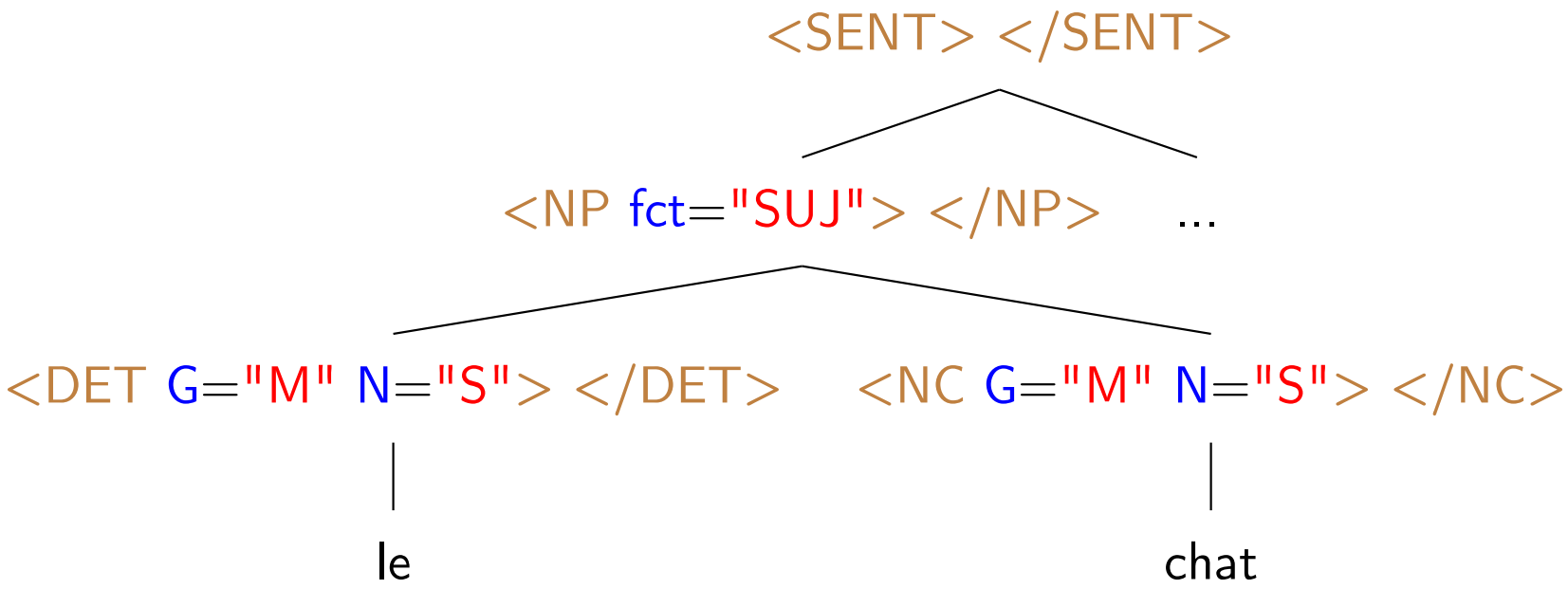
Les attributs d'une balise (donc du nœud) sont des couples clé/valeur renseignés sur la balise ouvrante ou autofermante :

- <NP fct="SUJ">
- <DET G="M" N="S">

[1] Techniquement, cette syntaxe n'est nécessaire qu'en XML. En HTML, il est préférable de s'en passer, comme pour
 par exemple. Source: https://developer.mozilla.org/fr/docs/Glossary/Void_element

Du balisage, à quoi ça ressemble ? III

Si on reprend l'exemple mais en XML :



Du balisage, à quoi ça ressemble ? IV

Le fichier tel qu'il sera écrit au format texte :

```
<SENT>
  <NP fct="SUJ">
    <DET G="M" N="S">Le</DET>
    <NC G="M" N="S">chat</NC>
  </NP>
  <VN>
    <V P="3" N="S">dort</V>
  </VN>
</SENT>
```

HTML : les spécificités

HTML reprend la construction globale du balisage, mais a sa propre syntaxe :

```
<html>
  <head>...</head>
  <body>...</body>
</html>
```

HTML : les spécificités

HTML reprend la construction globale du balisage, mais a sa propre syntaxe :

```
<html>
  <head>...</head>
  <body>...</body>
</html>
```

Où :

- head : l'entête du fichier (avec les métadonnées)
- body : le corps du fichier (avec le contenu textuel et la structure)

Bonne ressource pour approfondir HTML : https://www.w3schools.com/html/default.asp

HTML : l'entête

L'entête `head`[2] contient beaucoup d'informations intéressantes. On reviendra plus en détail plus tard sur certaines.

Une métadonnée particulière nous sera intéressante ici : l'encodage (charset)[3].

[2] Documentation entête HTML : https://www.w3schools.com/html/html_head.asp
[3] Documentation charset HTML : https://www.w3schools.com/html/html_charset.asp

HTML : l'entête

L'entête `head`[2] contient beaucoup d'informations intéressantes. On reviendra plus en détail plus tard sur certaines.

Une métadonnée particulière nous sera intéressante ici : l'encodage (charset)[3].

```
<html>
  <head>
    [...]
    <meta  charset="UTF-8" />
    [...]
  </head>
  <body>...</body>
</html>
```

[2] Documentation entête HTML : https://www.w3schools.com/html/html_head.asp
[3] Documentation charset HTML : https://www.w3schools.com/html/html_charset.asp

HTML : Créer un tableau

Pour créer un tableau en HTML, nous avons besoin de 4 balises :

- `table` : la balise racine du tableau
- `tr` : ***t****able* ***r****ow*, une ligne (se place dans table)
- `th` : ***t****able* ***h****eader*, une cellule d'entête (seulement la première ligne)
- `td` : ***t****able* ***d****ata*, une cellule classique (toutes les lignes pas entête)

---

https://www.w3schools.com/html//html_tables.asp

HTML : Créer un tableau

Pour créer un tableau en HTML, nous avons besoin de 4 balises :

- `table` : la balise racine du tableau
- `tr` : ***t**able **r**ow*, une ligne (se place dans table)
- `th` : ***t**able **h**eader*, une cellule d'entête (seulement la première ligne)
- `td` : ***t**able **d**ata*, une cellule classique (toutes les lignes pas entête)

```
<table>
  <tr><th>livre</th><th>taille</th></tr>
  <tr>
    <td>Du côté de chez Swann</td><td>1.0Mo</td>
  </tr>
  <tr>
    <td>L'Assommoir</td><td>990 ko</td>
  </tr>
</table>
```

---

https://www.w3schools.com/html//html_tables.asp

Exercice : transformer la sortie tabulaire en HTML

En partant du miniprojet de la semaine dernière, nous voulons à présent transformer la sortie tabulaire en sortie au format HTML. La page produite devra contenir :

1. un entête
2. un corps
    - devra contenir au moins le tableau des données récupérées, avec une ligne d'entête et les résultats pour chaque URL.

Ce code HTML devra être écrit dans un fichier `.html` qui devra être lisible par un navigateur web quelconque.

Structure attendue pour le mini-projet

En supposant que votre dépôt s'appelle **PPE1-2024**, l'arborescence de vos fichiers devra ressembler à cela une fois tous les exercices finis :

```
PPE1-2024
├── journal.md
├── exercices
│   └── ...
└── miniprojet
    ├── programmes
    │   └── miniprojet.sh
    ├── urls
    │   └── fr.txt
    └── tableaux
        └── tableau-fr.html
```

Consignes de rendu

- corriger votre code au besoin ;
  - ajouter le tag **miniprojet-1-revu** après correction (si votre **miniprojet-1** est déjà bon, créez simplement un tag sur le même commit) ;
- faire les exercices du miniprojet sur son dépôt individuel ;
  - transformer la sortie TSV en HTML, supprimer le TSV de votre dépôt ;
  - (bonus) faire la feuille d'exercices sur les comptages de mots/bigrammes ;
- créer le tag **miniprojet-2** à la fin du travail et le push sur Github ;
- créer un fichier texte avec le lien github vers le tag **miniprojet-2** ;
- au plus tard : le lundi 10/11 à 23h59 ;
- au besoin : continuer à réfléchir au mot à étudier et nous contacter si vous avez des idées.