

Fondamentaux du Web

HTML, CSS, JavaScript

Lab 2 — HTML structurant

Sémantique, formulaires, navigation entre 5 pages

Licence 1 DUT1-Informatique — 2025–2026

Dr. El Hadji Bassirou TOURÉ
École Supérieure Polytechnique
Département Génie Informatique
Université Cheikh Anta Diop de Dakar

Informations pratiques

Type	Lab guidé
Séance	S3 — deuxième Lab du semestre
Prérequis	Lab 1 terminé (page d'accueil HTML poussée avec tag v0.1)
Validation	5 pages HTML liées + formulaire fonctionnel + tag v0.2
Rapport	1-2 pages : ce que j'ai fait + difficultés + lien repo

Objectifs du Lab

À la fin de ce Lab, vous aurez :

1. Compris et utilisé les **balises sémantiques** HTML5 (`header`, `nav`, `main`, `section`, `footer`, `aside`)
2. Refondu votre `index.html` avec une structure sémantique propre
3. Créé les **4 autres pages** de Jàng Afrig : `quiz.html`, `resultat.html`, `scores.html`, `apropos.html`
4. Construit un **formulaire** (saisie du pseudo + sélection du mode)
5. Lié les 5 pages entre elles avec une **navigation cohérente**
6. **Validé** votre code HTML sur validator.w3.org
7. Posé le tag v0.2 sur GitHub
8. Rédigé votre rapport de Lab

Table des matières

1 Avant de coder	2
1.1 Là où on en est	2
1.2 Le problème de la page unique	2
1.3 Ce qu'on va construire aujourd'hui	2

2	Les balises sémantiques HTML5	3
2.1	Pourquoi « sémantique » ?	3
2.2	Les 6 balises à connaître	3
2.3	Sémantique vs. anonyme : le même rendu, du sens en plus	3
3	Étape 1 : Refondre index.html	5
3.1	Le nouveau squelette	5
3.2	L'entité ©	6
4	Étape 2 : La page Quiz avec un formulaire	7
4.1	Code complet de quiz.html	7
4.2	Décortiquer le formulaire	8
4.3	Pourquoi <label> est important	8
5	Étape 3 : Les 3 autres pages	9
5.1	Page resultat.html	9
5.2	Page scores.html	9
5.3	Page apropos.html	10
6	Étape 4 : Tester la navigation	12
6.1	Recharger la page d'accueil	12
6.2	Vérification : la nav doit être identique partout	12
7	Valider votre code HTML	13
7.1	Comment l'utiliser	13
7.2	Lire le rapport	13
8	Exercices d'application	14
9	Aide-mémoire	15
9.1	Les balises sémantiques	15
9.2	Les balises de formulaire	15
9.3	Attributs utiles	15
10	Routine Git de fin de Lab	16
11	Rapport à produire	16
12	Résumé et transition	16
12.1	Ce que vous avez accompli	16
12.2	Le bloc HTML est terminé	16
12.3	Ce qu'on fera au CM2 et au Lab 3	16

1 Avant de coder

1.1 Là où on en est

Au Lab 1, vous avez créé une **page unique** `index.html` avec toute l'information : titre, modes, palmarès, image, liens. Tout sur une seule page.

1.2 Le problème de la page unique

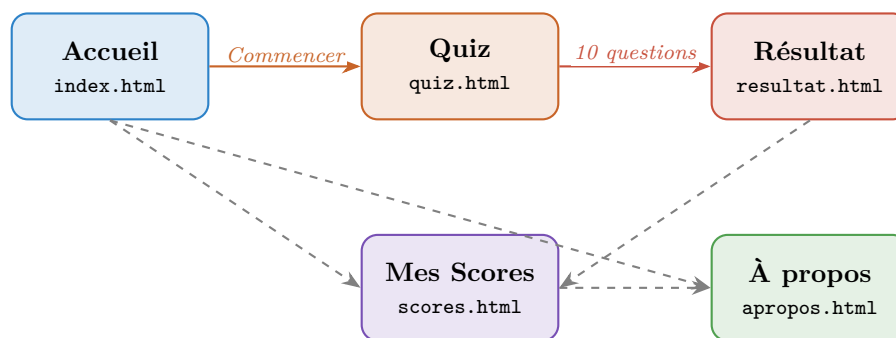
Une vraie application a **plusieurs pages**. Imaginez Jàng Afrig quand l'app sera complète :

- L'utilisateur arrive sur l'accueil
- Il choisit un mode et son pseudo
- Il clique « Commencer » → il va sur la page *quiz*
- Il répond à 10 questions
- Il arrive sur la page *résultat* avec son score
- Il peut aller voir la page *mes scores* (classement)
- Il peut consulter la page *à propos* (sources, équipe)

5 pages distinctes, donc **5 fichiers HTML** dans votre repo.

1.3 Ce qu'on va construire aujourd'hui

Les **5 pages HTML** de Jàng Afrig, sans CSS ni JavaScript. Toutes liées entre elles avec une **barre de navigation** commune.



(la nav présente sur toutes les pages permet d'aller partout)

Fig. 1 : Les 5 pages de Jàng Afrig et leur navigation.

2 Les balises sémantiques HTML5

Au Lab 1, vous n'avez utilisé que des balises de *contenu* : `<h1>`, `<p>`, `<ul>`, etc. Mais HTML5 propose aussi des balises de **structure**, dites *sémantiques*.

2.1 Pourquoi « sémantique » ?

Sémantique signifie « qui a du sens ». Au lieu de découper sa page en boîtes anonymes (`<div>`, `<div>`, `<div>`), on utilise des balises qui **disent ce que la zone contient** : header, navigation, contenu principal, footer...

L'analogie du journal

Pensez à un journal papier comme *Le Soleil* ou *L'Observateur* :

- En haut : la **une** (`<header>`) avec le logo et la date
- Sous la une : un **sommaire** (`<nav>`) qui dit où trouver quoi
- Au milieu : les **articles** (`<main>` contenant des `<article>`)
- En bas : la **mention légale** (`<footer>`) avec contact et imprimeur

On pourrait tout mettre dans des boîtes anonymes, mais ce serait beaucoup moins clair.

2.2 Les 6 balises à connaître

Balise	Rôle
<code><header></code>	En-tête de la page (logo, titre principal, parfois la nav)
<code><nav></code>	Barre de navigation (les liens vers les autres pages)
<code><main></code>	Contenu principal de la page. Une seule par page.
<code><section></code>	Une section thématique du contenu (<i>ex.</i> « Modes de jeu »)
<code><article></code>	Un contenu <i>autonome</i> qui pourrait être réutilisé ailleurs (un article de blog, une carte produit, un quiz...)
<code><footer></code>	Pied de page (mentions, copyright, liens secondaires)
<code><aside></code>	Contenu lié mais pas essentiel (encart, pub, citation latérale)

2.3 Sémantique vs. anonyme : le même rendu, du sens en plus

Visuellement, il n'y a aucune différence entre :

```
<div>
  <div>Mon site</div>
  <div>Lien1 Lien2</div>
</div>
```

et :

```
<header>
```

```
<h1>Mon site</h1>
<nav>Lien1 Lien2</nav>
</header>
```

Mais le second est compris par les moteurs de recherche, les lecteurs d'écran, les outils d'analyse. **Un code sémantique est un code respectueux de l'utilisateur et professionnel.**

3 Étape 1 : Refondre index.html

On reprend votre `index.html` du Lab 1 et on l'**enveloppe** dans une structure sémantique. Le contenu reste le même, on ajoute juste des balises autour.

3.1 Le nouveau squelette

Remplacez tout le `<body>` de votre `index.html` par cette structure :

```
<body>

  <header>
    <h1>Jàng Afrig</h1>
    <p>Apprendre l'Afrique, un pays à la fois</p>
  </header>

  <nav>
    <a href="index.html">Accueil</a> |
    <a href="quiz.html">Jouer</a> |
    <a href="scores.html">Mes Scores</a> |
    <a href="apropos.html">À propos</a>
  </nav>

  <main>

    <section>
      <h2>Bienvenue</h2>
      <p>
        Bienvenue sur <strong>Jàng Afrig</strong>, un quiz
        géographique sur les
        <strong>54 pays africains</strong>. Choisissez un mode de
        jeu et battez
        votre meilleur score !
      </p>
    </section>

    <section>
      <h2>Les modes de jeu</h2>
      <ul>
        <li><strong>Capitales</strong> : « Quelle est la capitale
          du Mali ? »</li>
        <li><strong>Drapeaux</strong> : « À quel pays appartient
          ce drapeau ? »</li>
        <li><strong>Langues</strong> : « Dans quel pays
          parle-t-on le swahili ? »</li>
        <li><strong>Voisins</strong> : « Combien de voisins a
          l'Algérie ? »</li>
        <li><strong>Chrono</strong> : maximum de bonnes réponses
          en 60 secondes</li>
      </ul>
    </section>

    <section>
      <h2>Palmarès des champions</h2>
      <p><a href="scores.html">Voir le classement complet</a></p>
    </section>

  </main>
```

```

<footer>
  <p>
    &copy; 2025--2026 Jàng Afrig --- Projet pédagogique L1
    DUT1-Info, ESP/UCAD.
    <a href="apropos.html">En savoir plus</a>.
  </p>
</footer>

</body>

```

Enregistrez. Visuellement la page est presque identique à celle du Lab 1, mais elle est maintenant **structurée sémantiquement**. ✓

Ce qui a changé

- Le bloc titre est dans un `<header>`
- Les liens vers les pages sont dans un `<nav>`
- Tout le contenu central est dans un `<main>`, divisé en 3 `<section>`
- Les mentions sont dans un `<footer>`
- Le tableau du palmarès est remplacé par un lien vers `scores.html` (qu'on va créer)

3.2 L'entité `©`

Vous avez vu `©` dans le footer. C'est une **entité HTML** : elle s'affiche ©. Quelques entités utiles :

Entité	Affiche	Quand l'utiliser
<code>&copy;</code>	©	Copyright
<code>&nbsp;</code>	espace	Espace insécable (avant ::!?)
<code>&amp;</code>	&	Le caractère « et commercial »
<code>&lt;</code>	<	Le chevron ouvrant (sans démarrer une balise)
<code>&gt;</code>	>	Le chevron fermant
<code>&hellip;</code>	...	Points de suspension

4 Étape 2 : La page Quiz avec un formulaire

Créez un nouveau fichier `quiz.html` dans votre repo. C'est la page où l'utilisateur saisit son pseudo et choisit son mode de jeu.

4.1 Code complet de `quiz.html`

```
<!DOCTYPE html>
<html lang="fr">
  <head>
    <meta charset="UTF-8">
    <title>Quiz --- Jàng Afrig</title>
  </head>
  <body>

    <header>
      <h1>Jàng Afrig</h1>
      <p>Apprendre l'Afrique, un pays à la fois</p>
    </header>

    <nav>
      <a href="index.html">Accueil</a> |
      <a href="quiz.html">Jouer</a> |
      <a href="scores.html">Mes Scores</a> |
      <a href="apropos.html">À propos</a>
    </nav>

    <main>
      <section>
        <h2>Avant de jouer</h2>

        <form>
          <p>
            <label for="pseudo">Votre pseudo :</label>
            <input type="text" id="pseudo" name="pseudo"
              placeholder="Ex. Fatou" required>
          </p>

          <p>
            <label for="mode">Mode de jeu :</label>
            <select id="mode" name="mode" required>
              <option value="">-- Choisir un mode
                --</option>
              <option value="capitales">Capitales</option>
              <option value="drapeaux">Drapeaux</option>
              <option value="langues">Langues</option>
              <option value="voisins">Voisins</option>
              <option value="chrono">Chrono</option>
            </select>
          </p>

          <p>
            <button type="submit">Commencer le quiz</button>
          </p>
        </form>
      </section>
    </main>
```

```

    <footer>
      <p>&copy; 2025--2026 Jàng Afrig --- L1 DUT1-Info,
        ESP/UCAD.</p>
    </footer>

  </body>
</html>

```

Enregistrez. Live Server affiche la page (cliquez sur « Jouer » depuis l'accueil). Vous voyez un formulaire avec un champ texte, un menu déroulant, et un bouton. ✓

4.2 Décortiquer le formulaire

Élément	Rôle
<form>	Conteneur du formulaire (englobe tous les champs)
<label for="X">	Étiquette d'un champ. <code>for="X"</code> se lie à <code>id="X"</code> .
<input type="text">	Champ de texte sur une ligne
placeholder="..."	Texte gris d'aide qui disparaît à la frappe
required	Le champ est obligatoire (validation HTML5 native)
<select> + <option>	Menu déroulant à options fixes
<button type="submit">	Bouton qui soumet le formulaire

Test du formulaire

Cliquez sur « Commencer le quiz » sans rien remplir : le navigateur **bloque** l'envoi et affiche un message « Veuillez remplir ce champ ». C'est la *validation HTML5 native*, gratuite, sans JavaScript.

Le formulaire ne mène nulle part pour l'instant

Cliquer sur « Commencer » va recharger la page. C'est normal : il manque le JavaScript pour traiter les données. On s'en occupera au **Lab 7**. Pour aujourd'hui, l'objectif est juste d'avoir le formulaire *visible et structuré*.

4.3 Pourquoi <label> est important

L'attribut `for` d'un `<label>` relie l'étiquette au champ. Cliquer sur « Votre pseudo : » met le curseur dans le champ. Pour les utilisateurs malvoyants, le lecteur d'écran annonce « Votre pseudo, champ texte » au lieu d'un anonyme « champ texte ». **Toujours mettre un <label>.**

5 Étape 3 : Les 3 autres pages

On va créer rapidement les 3 pages restantes. Elles ont toutes la **même structure** (header + nav + main + footer), seul le contenu du `<main>` change.

5.1 Page resultat.html

Créez `resultat.html` avec ce contenu (la partie `<main>` seulement est différente, le reste est identique aux autres pages) :

```
<main>
  <section>
    <h2>Résultat de votre partie</h2>

    <p>Bravo <strong>Fatou</strong> !</p>
    <p>Mode : <em>Capitales</em></p>
    <p>Score : <strong>8 / 10</strong></p>

    <p>
      <a href="quiz.html">Rejouer</a> |
      <a href="scores.html">Voir mes scores</a> |
      <a href="index.html">Retour à l'accueil</a>
    </p>
  </section>
</main>
```

Note : les valeurs « Fatou », « Capitales », « 8/10 » sont *hardcodées* pour l'instant. Au Lab 7, JavaScript les remplacera par les vraies valeurs de la partie.

5.2 Page scores.html

Créez `scores.html`. Le `<main>` contient le tableau du palmarès (récupéré du Lab 1) :

```
<main>
  <section>
    <h2>Palmarès des champions</h2>

    <p>Les meilleurs scores enregistrés (données fictives pour
      l'instant) :</p>

    <table>
      <tr>
        <th>Rang</th>
        <th>Pseudo</th>
        <th>Mode</th>
        <th>Score</th>
        <th>Date</th>
      </tr>
      <tr>
        <td>1</td>
        <td>Fatou</td>
        <td>Capitales</td>
        <td>10/10</td>
        <td>15 mars 2026</td>
      </tr>
      <tr>
        <td>2</td>
        <td>Moussa</td>
```

```

        <td>Drapeaux</td>
        <td>9/10</td>
        <td>12 mars 2026</td>
    </tr>
    <tr>
        <td>3</td>
        <td>Aminata</td>
        <td>Capitales</td>
        <td>9/10</td>
        <td>10 mars 2026</td>
    </tr>
    <tr>
        <td>4</td>
        <td>Cheikh</td>
        <td>Voisins</td>
        <td>8/10</td>
        <td>8 mars 2026</td>
    </tr>
    <tr>
        <td>5</td>
        <td>Awa</td>
        <td>Langues</td>
        <td>8/10</td>
        <td>5 mars 2026</td>
    </tr>
</table>
</section>
</main>

```

5.3 Page apropos.html

Créez apropos.html. Le <main> :

```

<main>
  <section>
    <h2>À propos de Jàng Afrig</h2>
    <p>
      Jàng Afrig est une application web de quiz géographique sur
      les
      <strong>54 pays africains</strong>. Le mot <em>jàng</em>
      signifie
      « apprendre, étudier » en wolof.
    </p>
    <p>
      Ce projet est développé dans le cadre du cours
      <strong>Fondamentaux du Web</strong> en L1 DUT1-Informatique
      à l'École Supérieure Polytechnique (ESP/UCAD).
    </p>
  </section>

  <section>
    <h2>Sources des données</h2>
    <p>
      Les données sur les pays sont fournies par
      <a href="https://restcountries.com">REST Countries v3.1</a>,
      une API libre et gratuite.
    </p>

```

```
</section>

<section>
  <h2>Technologies utilisées</h2>
  <ul>
    <li>HTML5 pour la structure</li>
    <li>CSS3 pour la mise en forme</li>
    <li>JavaScript moderne pour l'interactivité</li>
    <li>Git et GitHub pour le versioning et l'hébergement</li>
  </ul>
</section>

<section>
  <h2>Code source</h2>
  <p>
    Le code est disponible sur GitHub :
    <a href="https://github.com/votre-username/jang-afrig">
      github.com/votre-username/jang-afrig
    </a>
  </p>
</section>
</main>
```

Astuce de productivité

Au lieu de retaper le squelette à chaque fois, **copiez** `quiz.html` puis **renommez** la copie en `resultat.html`, et changez seulement le contenu du `<main>`. Pareil pour `scores.html` et `apropos.html`. C'est ainsi qu'on travaille en vrai.

6 Étape 4 : Tester la navigation

6.1 Recharger la page d'accueil

Dans le navigateur (où Live Server tourne) :

1. Allez sur <http://127.0.0.1:5500/index.html>
 2. Cliquez sur « Jouer » dans la nav → vous arrivez sur `quiz.html`
 3. Cliquez sur « Mes Scores » → `scores.html`
 4. Cliquez sur « À propos » → `apropos.html`
 5. Depuis n'importe quelle page, cliquez sur « Accueil » → retour à `index.html`
- Toutes les pages sont accessibles depuis n'importe quelle autre. ✓

6.2 Vérification : la nav doit être identique partout

Le piège du copier-coller

Si vous changez le menu sur une page, vous devez le changer sur toutes les autres. Sinon, l'utilisateur a une nav différente selon où il est : incohérent et frustrant.

Aujourd'hui, on accepte cette duplication (4 fois le même `<nav>`). **Plus tard** (en L2 avec un framework, ou même au Lab 7 avec JS), on apprendra à ne définir la nav qu'*une seule fois*.

7 Valider votre code HTML

Le W3C (World Wide Web Consortium, l'organisme qui définit HTML) propose un **validateur en ligne gratuit** qui détecte les erreurs de syntaxe dans votre HTML.

7.1 Comment l'utiliser

1. Allez sur <https://validator.w3.org/nu/>
2. Onglet **Validate by Direct Input**
3. Ouvrez `index.html` dans VS Code, sélectionnez **tout** (Ctrl+A), copiez (Ctrl+C)
4. Collez dans la zone de texte du validateur
5. Cliquez **Check**

7.2 Lire le rapport

- Bandeau **vert** : « Document checking completed. No errors or warnings to show. » → votre HTML est parfait ✓
- Bandeau **rouge** : il y a des erreurs. Le rapport indique la ligne et la nature du problème.
Faites tourner le validateur sur les 5 pages. Corrigez les erreurs au fur et à mesure.

Erreurs courantes

- Balise non fermée : `<p>texte sans </p>`
- Balise mal imbriquée : `<p><strong>text</p></strong>`
- Attribut `alt` manquant sur `<img>`
- Plusieurs `<h1>` dans la même page
- Caractères spéciaux non échappés (`<` au lieu de `<` dans du texte)

8 Exercices d'application

Trois mini-exercices

Exercice 1 — Ajouter un `<aside>` sur l'accueil

Sur `index.html`, ajoutez un `<aside>` (encart latéral) après le `<main>` avec un « saviez-vous ? » géographique. Exemple :

```
<aside>
  <h3>Le saviez-vous ?</h3>
  <p>
    L'Afrique compte <strong>54 pays</strong> reconnus par l'ONU,
    avec plus de <strong>2000 langues</strong> parlées.
  </p>
</aside>
```

Exercice 2 — Enrichir le formulaire

Sur `quiz.html`, ajoutez dans le `<form>` :

- Un champ `<input type="number">` pour choisir le **nombre de questions** (min 5, max 20, valeur par défaut 10)
- 3 `<input type="radio">` pour la **difficulté** : Facile, Moyen, Difficile
- N'oubliez pas les `<label>` associés

Exercice 3 — Page formulaire de contact (bonus)

Créez une nouvelle page `contact.html` avec un formulaire complet :

- Nom (`input type="text", required`)
- Email (`input type="email", required`)
- Sujet (`select` avec 3 options : Question, Bug, Suggestion)
- Message (`<textarea>` de 5 lignes, required)
- Bouton « Envoyer »

Ajoutez le lien vers cette page dans le `<nav>` de toutes les pages.

Validez vos pages au W3C avant de passer à la suite.

9 Aide-mémoire

9.1 Les balises sémantiques

Balise	Rôle
<code><header></code>	En-tête (logo, titre principal)
<code><nav></code>	Barre de navigation
<code><main></code>	Contenu principal (un seul par page)
<code><section></code>	Section thématique
<code><article></code>	Contenu autonome réutilisable
<code><aside></code>	Contenu lié mais secondaire
<code><footer></code>	Pied de page

9.2 Les balises de formulaire

Balise	Rôle
<code><form></code>	Conteneur d'un formulaire
<code><label for="X"></code>	Étiquette d'un champ
<code><input type="text"></code>	Champ texte
<code><input type="email"></code>	Champ email (validation auto)
<code><input type="number"></code>	Champ numérique
<code><input type="radio"></code>	Bouton radio (choix unique)
<code><input type="checkbox"></code>	Case à cocher (choix multiple)
<code><select> + <option></code>	Menu déroulant
<code><textarea></code>	Champ texte multi-lignes
<code><button type="submit"></code>	Bouton de soumission

9.3 Attributs utiles

Attribut	Rôle
<code>id="..."</code>	Identifiant unique sur la page (référéncé par <code>label for</code>)
<code>name="..."</code>	Nom du champ (utilisé côté serveur, plus tard)
<code>placeholder="..."</code>	Texte d'aide gris dans un champ vide
<code>required</code>	Champ obligatoire
<code>value="..."</code>	Valeur par défaut d'un champ

10 Routine Git de fin de Lab

Dans le terminal intégré de VS Code :

```
git status
git add .
git commit -m "v0.2 : 5 pages HTML sémantiques + formulaire"
git push
git tag v0.2
git push --tags
```

Vérifiez sur <https://github.com/votre-username/jang-afrig> que :

- Les 5 fichiers HTML sont là (index.html, quiz.html, resultat.html, scores.html, apropos.html)
- Le tag v0.2 est visible dans l'onglet **Tags**

11 Rapport à produire

Rapport du Lab 2 — obligatoire

Format : 1-2 pages, déposé dans rapports/rapport-lab-2-NOM-Prenom.pdf de votre repo.

Contenu (4 sections) :

1. **Ce que j'ai fait** : 5 pages créées, balises sémantiques utilisées, formulaire construit, exercices d'application complétés, validation W3C passée.
2. **Difficultés rencontrées** : navigation cassée, validateur W3C qui rouspète, formulaire qui se comporte bizarrement...
3. **Comment j'ai résolu ces difficultés** : recherche, aide d'un camarade, lecture de la doc.
4. **Lien vers mon repo GitHub avec le tag v0.2** :
ex. <https://github.com/fatou-diop/jang-afrig/releases/tag/v0.2>

12 Résumé et transition

12.1 Ce que vous avez accompli

- Maîtrisé les balises sémantiques HTML5 ✓
- Construit les 5 pages de Jàng Afrig ✓
- Construit votre premier formulaire fonctionnel ✓
- Lié les pages avec une navigation cohérente ✓
- Validé votre code au W3C ✓
- Posé le tag v0.2 ✓

12.2 Le bloc HTML est terminé

Vous avez vu **tout ce qu'il faut savoir en HTML** pour ce semestre : structure, contenu, sémantique, formulaires, tableaux, liens, images.

12.3 Ce qu'on fera au CM2 et au Lab 3

CM2 (S4) : le CSS arrive. Vous découvrirez les sélecteurs, la cascade, le box model, et Flexbox.

Lab 3 (S5) : vous habillerez Jàng Afrig avec des couleurs panafricaines, des polices Google Fonts, et de jolis boutons. Tag v0.3.

À ce stade du semestre

Votre application a une structure complète et propre. Elle est encore moche (texte noir sur fond blanc), mais **toutes les fondations HTML sont posées**. Aux Labs 3 et 4, on transforme ça en application visuellement attractive.

Lab 2 — Terminé ✓

Vous avez construit l'ossature complète de Jàng Afrig.
Place à la décoration au CM2 et au Lab 3.

Rendez-vous au CM2 !