

# R1.02

## DÉVELOPPEMENT D'INTERFACES WEB



CSS IS ♥  
DETAILS MAKE PERFECTION, AND  
PERFECTION IS NOT A DETAIL

CSS IS ♥  
DETAILS MAKE PERFECTION, AND  
PERFECTION IS NOT A DETAIL  
WAIT... WHAT?  
PLAGIARISM!

index.html

```
1 <figure>
2   <blockquote cite="http://www...">
3     <p>Details make perfection, and
4       perfection is not a detail.<p>
5   </blockquote>
6   <figcaption>
7     —Leonardo Da Vinci
8     <cite>Trouvée dans une papillote</cite>
9   </figcaption>
10 </figure>
```

# CONSTRUIRE UNE CITATION

*Details make perfection, and perfection is not a detail.*

—Leonardo Da Vinci *Trouvée dans une papillote*



index.html

style.css

```
1  blockquote {  
2    font-style: italic;  
3    margin-left: 32px;  
4    font-family: Georgia, serif;  
5    border-left: 10px solid lightblue;  
6  }  
7  
8  blockquote + figcaption {  
9    font-size: smaller;  
10   margin-left: 32px;  
11 }
```

# CONSTRUIRE UNE CITATION

*Details make perfection, and perfection is not a detail.*

—Leonardo Da Vinci *Trouvée dans une papillote*



**IUT2**   
Université Grenoble Alpes

index.html

style.css

```
1  blockquote {
2    font-style: italic;
3    margin-left: 32px;
4    font-family: Georgia, serif;
5    border-left: 10px solid lightblue;
6  }
7
8  blockquote + figcaption {
9    font-size: smaller;
10   margin-left: 32px;
11 }
```

# CONSTRUIRE UNE CITATION

*Details make perfection, and perfection is not a detail.*

—Leonardo Da Vinci *Trouvée dans une papillote*



**IUT2**   
Université Grenoble Alpes

# EN SAÉ

- Paraphrase du site de l'entreprise interdit (et sanctionné)
- En cas de copie d'un contenu sur un site Web, deux solutions :
  - Reformuler avec vos mots (ça tombe bien, c'est un site à destination du grand public) ;
  - Utiliser l'environnement `<blockquote> . . . </blockquote>`
- En cas d'utilisation d'une image qui ne vous appartient pas, citer la source.



# LES FORMULAIRES HTML



# LES FORMULAIRES HTML

- Omniprésents sur le Web ;
- Permettent la saisie d'information du visiteur de la page à des moyens de traitements informatiques ;



# LES FORMULAIRES HTML

- Omniprésents sur le Web ;
- Permettent la saisie d'information du visiteur de la page à des moyens de traitements informatiques ;
- Nombreux cas applicatifs :
  - Formulaire de prise de contact
  - Gestion de commentaires sur un Blog
  - Gestion du panier électronique sur un site de e-commerce
  - Pop-up de la politique de la gestion des cookies (RGPD)



# L'ÉLÉMENT FORM

- La balise `<form>` permet de définir un formulaire ;
- Plusieurs attributs possibles, les deux plus courants sont :

```
1 <form action="traitement.php" method="post">  
2   <!-- Le contenu du formulaire ici -->  
3 </form>
```

## action

Définit la page Web à appeler pour traiter les informations du formulaire. (lien avec PHP BUT2).

## method

Comment seront envoyées les données à la page de traitement (methode HTTP) :

- **post** données ajoutées au *body* de la requête HTTP (non visibles dans l'URL)
- **get** données transférées en tant que variables dans l'URL (visibles)



# LES ÉLÉMENTS DE FORMULAIRE

## Les éléments HTML classiques :

- Champs de texte simples
- Zones de texte multi-lignes
- Cases à cocher
- Boutons radio
- Liste déroulante
- Les champs pour déposer des fichiers
- Les boutons (soumission, RAZ, etc.)



# LES ÉLÉMENTS DE FORMULAIRE

## Les éléments HTML classiques :

- Champs de texte simples
- Zones de texte multi-lignes
- Cases à cocher
- Boutons radio
- Liste déroulante
- Les champs pour déposer des fichiers
- Les boutons (soumission, RAZ, etc.)

## Les nouveautés HTML5 :

- Les sliders
- Les champs de saisie des dates (*date pickers*)
- Les champs de recherche dans une liste
- Les outils de saisie de couleurs (*color pickers*)
- etc.



# LES ÉLÉMENTS DE FORMULAIRE

## Les éléments HTML classiques :

- Champs de texte simples
- Zones de texte multi-lignes
- Cases à cocher
- Boutons radio
- Liste déroulante
- Les champs pour déposer des fichiers
- Les boutons (soumission, RAZ, etc.)

## Les nouveautés HTML5 :

- Les sliders
- Les champs de saisie des dates (*date pickers*)
- Les champs de recherche dans une liste
- Les outils de saisie de couleurs (*color pickers*)
- etc.

**Note :** on ne les passera pas tous en revue, mais on va voir l'essentiel.



# L'ÉLÉMENT INPUT

Élément polyvalent pour représenter les champs de texte, les boutons radios, les cases à cocher...

Votre prénom :

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

Votre nom :

```
<input type="text" name="nom"><br>
```

Votre age :

```
<input type="radio" name="age"> 10-15
```

```
<input type="radio" name="age"> 15-20
```

```
<input type="radio" name="age"> 20-25<br>
```

```
<input type="checkbox" name="cookies"> Des cookies !
```

Votre prénom :

Votre nom :

Votre age : ☐ 10-15 ☐ 15-20 ☐ 20-25

☐ Des cookies !

- **Name** est utilisé à des fins de traitement informatique ;



# L'ÉLÉMENT INPUT

Éléments polyvalents pour représenter les champs de texte, les boutons radios, les cases à cocher...

Votre prénom :

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

Votre nom :

```
<input type="text" name="nom"><br>
```

Votre âge :

```
<input type="radio" name="age"> 10-15
```

```
<input type="radio" name="age"> 15-20
```

```
<input type="radio" name="age"> 20-25<br>
```

```
<input type="checkbox" name="cookies"> Des cookies !
```

Votre prénom :

Votre nom :

Votre âge : ☐ 10-15 ☐ 15-20 ☐ 20-25

☐ Des cookies !

- **Name** est utilisé à des fins de traitement informatique ;

Première observation : sans CSS, c'est moche :(





# CHAMPS DE TEXTE

## (LISTE NON EXHAUSTIVE)

```
<input type="text" name="prenom"  
      placeholder="Votre prénom" required>
```

Votre prénom

```
<input type="number" name="age"  
      placeholder="Votre age">
```

Votre age

```
<input type="email" name="email"  
      placeholder="Votre email">
```

Votre email

- **Type** peut permettre de spécifier le formatage attendu ;
- **Placeholder** permet d'afficher un texte quand le champ est vide ;
- **Required** permet d'indiquer qu'un champ est obligatoire.



# BOUTONS RADIO ET CHECKBOX

```
1 <input type="radio" name="age" value="10-15"> 10-15<br>
2 <input type="radio" name="age" value="16-20"> 16-20<br>
3 <input type="radio" name="age" value="21-25" checked> 21-25
```

☐ 10-15  
☐ 16-20  
☒ 21-25

- **name** permet de grouper les boutons radios (la sélection de l'un entraîne la désélection des autres du même groupe) ;
- **value** permet de savoir quel bouton a été sélectionné (pour traitement en PHP ou JavaScript).
- **checked** permet de cocher par défaut un checkbox ou un bouton radio.



# BOUTONS RADIO ET CHECKBOX

```
1 <input type="radio" name="age" value="10-15"> 10-15<br>
2 <input type="radio" name="age" value="16-20"> 16-20<br>
3 <input type="radio" name="age" value="21-25" checked> 21-25
```

☐ 10-15  
☐ 16-20  
☒ 21-25

- **name** permet de grouper les boutons radios (la sélection de l'un entraîne la désélection des autres du même groupe) ;
- **value** permet de savoir quel bouton a été sélectionné (pour traitement en PHP ou JavaScript).
- **checked** permet de cocher par défaut un checkbox ou un bouton radio.

```
1 <input type="checkbox" name="cookies" checked> Des cookies !
```

☒ Des cookies !



# INPUT VS. TEXTAREA

```
1 <input type="text" name="prenom" value="John">
```

John

```
1 <textarea name="description">
2 Ceci est une description
3 sur plusieurs lignes.
4 </textarea>
```

Ceci est une  
description



# INPUT VS. TEXTAREA

```
1 <input type="text" name="prenom" value="John">
```

John

```
1 <textarea name="description">  
2 Ceci est une description  
3 sur plusieurs lignes.  
4 </textarea>
```

Ceci est une  
description

**Note :** dans un champs de texte simple, la valeur est définie par l'attribut *value*. Dans un champs de texte multilignes, la valeur se trouve entre les balises `<textarea> . . . </textarea>`.



# COMBO BOX (BOÎTE COMBINÉE)

```
1 <select name="voiture">
2   <option value="peugeot">Peugeot</option>
3   <option value="renault" selected>Renault</option>
4   <option value="citroen">Citroën</option>
5 </select>
```

Renault ▾

```
1 <select name="voiture" multiple>
2   <option value="peugeot">Peugeot</option>
3   <option value="renault" selected>Renault</option>
4   <option value="citroen">Citroën</option>
5 </select>
```

Peugeot  
Renault  
Citröen



# COMBO BOX (BOÎTE COMBINÉE)

```
1 <select name="voiture">
2   <option value="peugeot">Peugeot</option>
3   <option value="renault" selected>Renault</option>
4   <option value="citroen">Citroën</option>
5 </select>
```

Renault ▾

```
1 <select name="voiture" multiple>
2   <option value="peugeot">Peugeot</option>
3   <option value="renault" selected>Renault</option>
4   <option value="citroen">Citroën</option>
5 </select>
```

Peugeot  
Renault  
Citröen

**Note :** différence entre la valeur affichée (entre les balises `<option> . . . </option>`) et la valeur qui sera traitée informatiquement (dans l'attribut *value*).





# LES BOUTONS

```
<form action="." method="GET">
  <input type="text" name="nom" placeholder="nom">
  <br>
  <button>Click me</button>
  <input type="submit" value="Envoyer">
  <input type="reset" value="Réinitialiser">
</form>
<input type="submit" value="Envoyer (hors form)">
```

Nom...

Click me Envoyer Réinitialiser

Envoyer (hors form)

**Note :** Aucun effet si les boutons ne sont pas dans un formulaire `<form> . . . </form>`.





# ÉLÉMENT LABEL

Représente une légende pour un élément de l'interface, et plus particulièrement d'un formulaire.

```
1 <label for="prenom">Votre prénom :</label>
2 <input type="text" value="John" name="prenom" id="prenom">
3 <label for="cookies">Des cookies !</label>
4 <input type="checkbox" name="cookies" id="cookies">
```

Votre prénom :

John

Des cookies ! ☐

## Avantages

- Focus sur le label = focus sur l'élément de formulaire ;
- Donne des indications aux lecteurs d'écran pour les personnes malvoyantes (accessibilité).

## Inconvénient

- Duplication : deux attributs *name* et *id*.



# ÉLÉMENT LABEL (SUITE)

Possibilité d'enrober l'élément HTML par le label (l'attribut *for* devient inutile):

```
1 <label>Votre prénom :  
2   <input type="text" value="John" name="prenom">  
3 </label>  
4 <label>Des cookies !  
5   <input type="checkbox" name="cookies">  
6 </label>
```

Votre prénom :

Des cookies ! ☐

## Avantages

- Supprime la nécessité d'avoir un identifiant

## Inconvénient

- Contraint l'application de style



# ÉLÉMENT LABEL (SUITE)

Possibilité d'enrober l'élément HTML par le label (l'attribut *for* devient inutile):

```
1 <label>Votre prénom :  
2   <input type="text" value="John" name="prenom">  
3 </label>  
4 <label>Des cookies !  
5   <input type="checkbox" name="cookies">  
6 </label>
```

Votre prénom :

Des cookies ! ☐

## Avantages

- Supprime la nécessité d'avoir un identifiant

## Inconvénient

- Contraint l'application de style

Quid des boutons radios ?

Indiquez votre âge :

- ☐ 10-15 ☐ 16-20 ☒ 21-25



# TRAITEMENT DES FORMULAIRES

- **Côté client (JavaScript) :**
  - "Contrôler" les données entrées par l'utilisateur
  - Affichage des messages d'erreurs avant envoi
- **Côté serveur (PHP, Java) :**
  - "Contrôler" les données entrées par l'utilisateur
  - Enregistrement en base
  - Création de fichiers, etc.



Traitement client et serveur.

Source de l'image: [javascript-coder.com](http://javascript-coder.com)



**IUT2**   
Université Grenoble Alpes

# TRAITEMENT DES FORMULAIRES

- Côté client (JavaScript) :
  - "Contrôler" les données entrées par l'utilisateur
  - Affichage des messages d'erreurs avant envoi
- Côté serveur (PHP, Java) :
  - "Contrôler" les données entrées par l'utilisateur
  - Enregistrement en base
  - Création de fichiers, etc.



Traitement client et serveur.

**Règle d'or :** ne jamais faire confiance à l'utilisateur. Toujours s'assurer que les informations entrées sont correctes avant un traitement informatique (e.g., enregistrement en base de données).

Source de l'image: [javascript-coder.com](http://javascript-coder.com)



**IUT2**   
Université Grenoble Alpes



# TRAITEMENT DES FORMULAIRES



Attaque par injection SQL.

Source de l'image: [hitek.fr](http://hitek.fr)



**IUT2**   
Université Grenoble Alpes

# LES STYLES ET LES FORMULAIRES

# STYLES SIMPLES

```
input {  
  border: none; /* Supprime la bordure par défaut */  
  border-bottom: thick solid gray; /* Bordure basse uniquement */  
}  
  
input:focus { /* Uniquement lorsque le champs a le focus */  
  border-bottom-color: blue;  
  color: blue;  
}
```

Votre prénom





# STYLES SIMPLES

```
input {  
  border: none; /* Supprime la bordure par défaut */  
  border-bottom: thick solid gray; /* Bordure basse uniquement */  
}  
  
input:focus { /* Uniquement lorsque le champs a le focus */  
  border-bottom-color: blue;  
  color: blue;  
}
```

Votre prénom

**Note :** présence d'une fine bordure lorsque le champs est sélectionné alors que `border: none;`  
Pour s'en débarrasser : `input { outline: none; }`



# STYLES SIMPLES

```
input {  
  border: none; /* Supprime la bordure par défaut */  
  border-bottom: thick solid gray; /* Bordure basse uniquement */  
}  
  
input:focus { /* Uniquement lorsque le champs a le focus */  
  border-bottom-color: blue;  
  color: blue;  
}
```

Votre prénom

---

**Note :** présence d'une fine bordure lorsque le champs est sélectionné alors que `border: none;`  
Pour s'en débarrasser : `input { outline: none; }`

**Note 2 :** changement de couleur dès que le focus est donné. Peut-on améliorer cela ?



# MANIPULER LE PLACEHOLDER (1/2)

```
input {  
  outline: none;  
  border: none;  
  border-bottom: thick solid gray;  
}  
  
input:not(:placeholder-shown) { /* Si non vide */  
  border-bottom-color: blue;  
  color: blue;  
}
```

Votre prénom

---



# MANIPULER LE PLACEHOLDER (1/2)

```
input {  
  outline: none;  
  border: none;  
  border-bottom: thick solid gray;  
}  
  
input:not(:placeholder-shown) { /* Si non vide */  
  border-bottom-color: blue;  
  color: blue;  
}
```

Votre prénom

**:placeholder-shown** : meta-classe CSS pour vérifier si le placeholder est actuellement affiché. Ici, nous utilisons la négation (**:not**) pour vérifier qu'il est caché (sous-entendu, il y a du contenu).



# MANIPULER LE PLACEHOLDER (2/2)

```
input {  
  outline: none;  
  border: none;  
  border-bottom: thick solid green;  
}
```

```
input::placeholder { /* Modifie l'élément placeholder */  
  color: green;  
}
```

Votre prénom



# MANIPULER LE PLACEHOLDER (2/2)

```
input {  
  outline: none;  
  border: none;  
  border-bottom: thick solid green;  
}
```

```
input::placeholder { /* Modifie l'élément placeholder */  
  color: green;  
}
```

Votre prénom

**::placeholder** : meta-élément CSS pour modifier le style du placeholder.

Attention : seulement des styles simples.



**IUT2**   
Université Grenoble Alpes

# STYLES CHAMPS VALIDES ET INVALIDES

```
input:invalid {  
  background-color: pink;  
}  
  
input:valid {  
  background-color: lightgreen;  
}
```

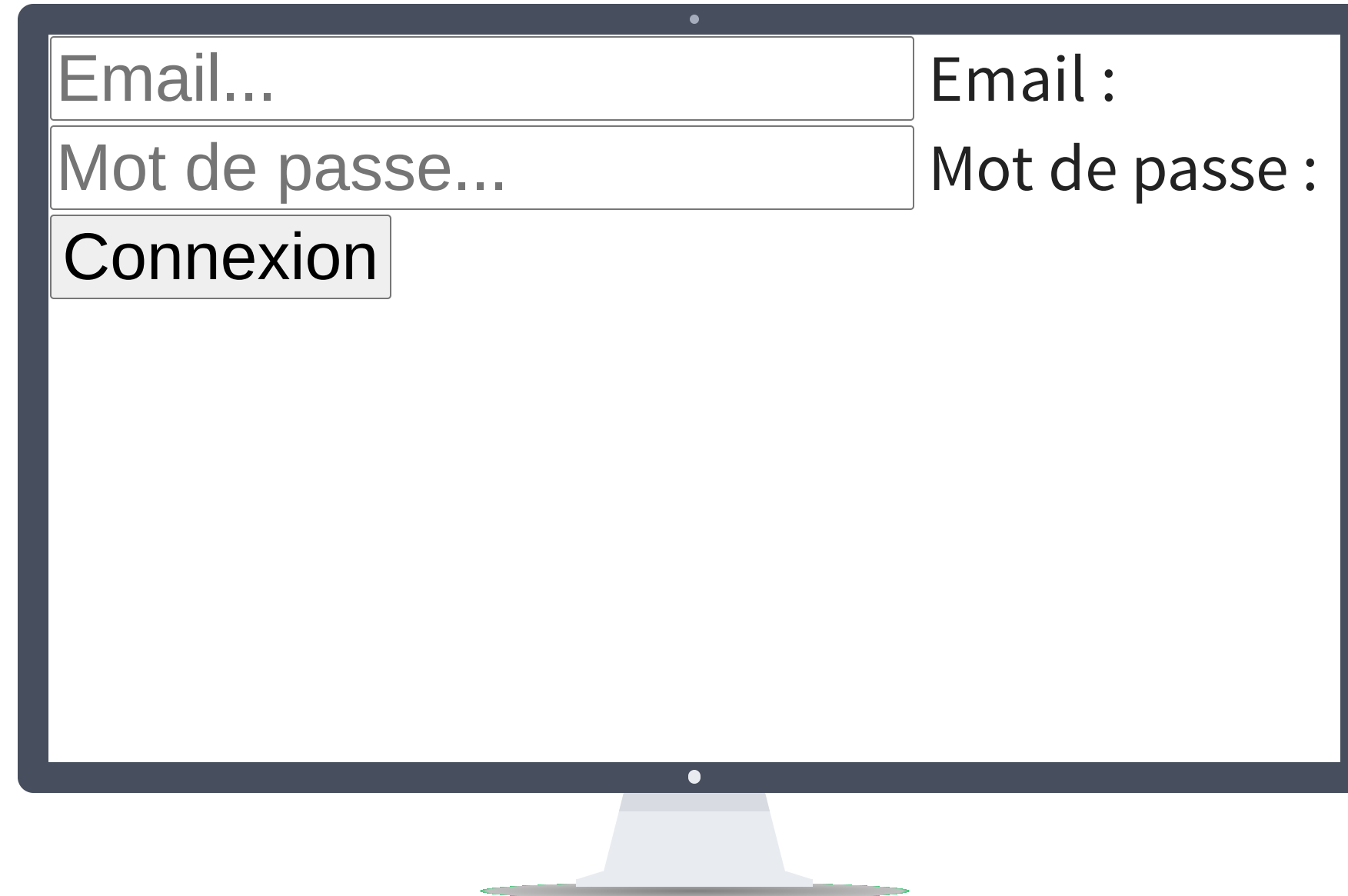
Votre prénom



index.html

```
1 <form>
2   <div class="field">
3     <input type="email" id="email"
4         placeholder="Email...">
5     <label for="email">Email :</label>
6   </div>
7
8   <div class="field">
9     <input type="password" id="mdp"
10        placeholder="Mot de passe...">
11     <label for="mdp">Mot de passe :</label>
12   </div>
13   <input type="submit" value="Connexion">
14 </form>
```

## STYLER LE LABEL



Email...

Mot de passe...

Connexion

Email :

Mot de passe :

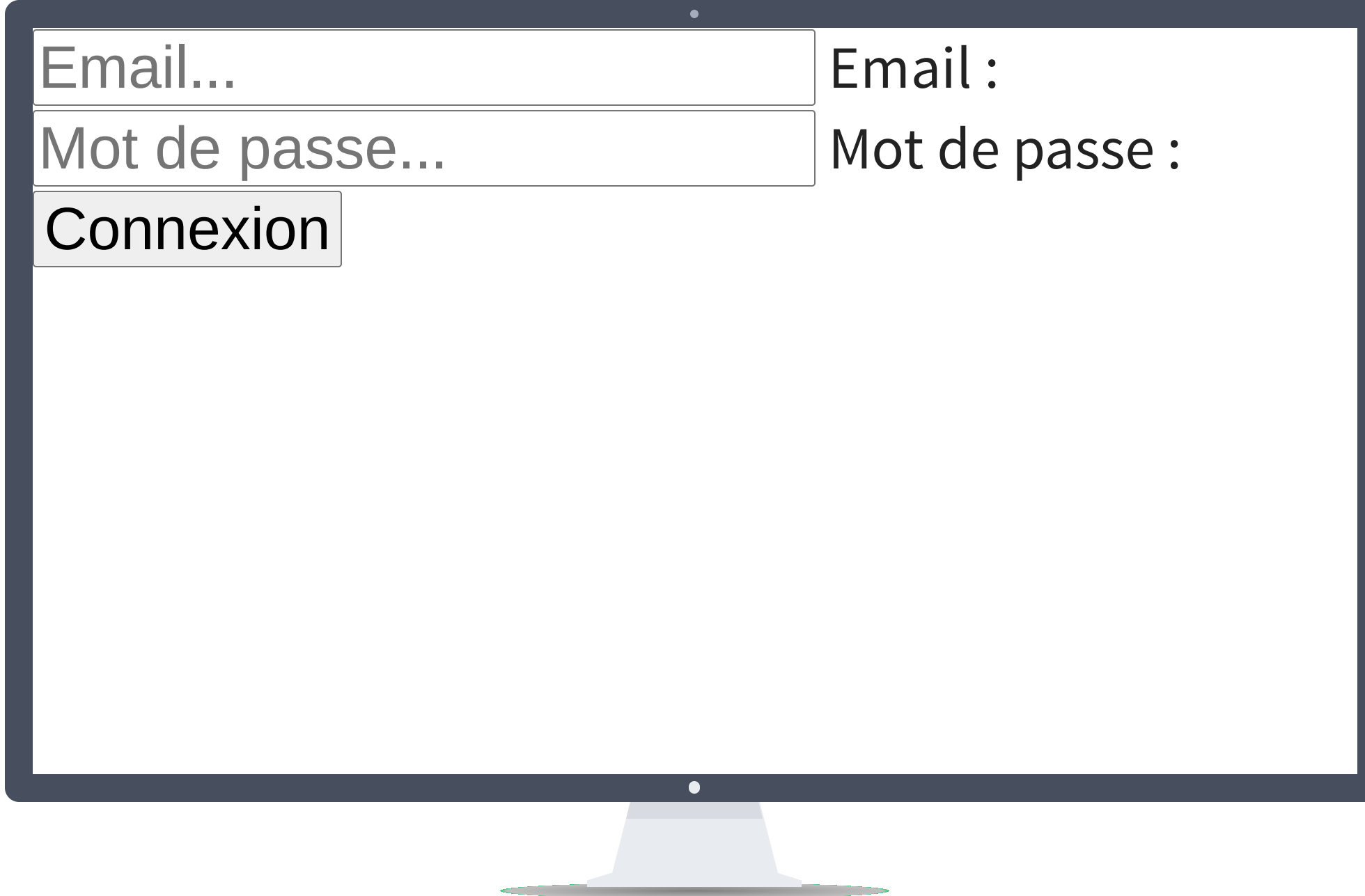




index.html style.css

```
1 form {
2   display: flex;
3   gap: 20px;
4   justify-content: center;
5   align-items: center; }
6
7 input {
8   outline: none;
9   border: none;
10  border-bottom: thick solid gray; }
11
12 input[type="submit"] {
13   border-color: #6666aa;
14   background-color: lightblue;
15   padding: .2em 1em;
16   border-radius: 20px; }
```

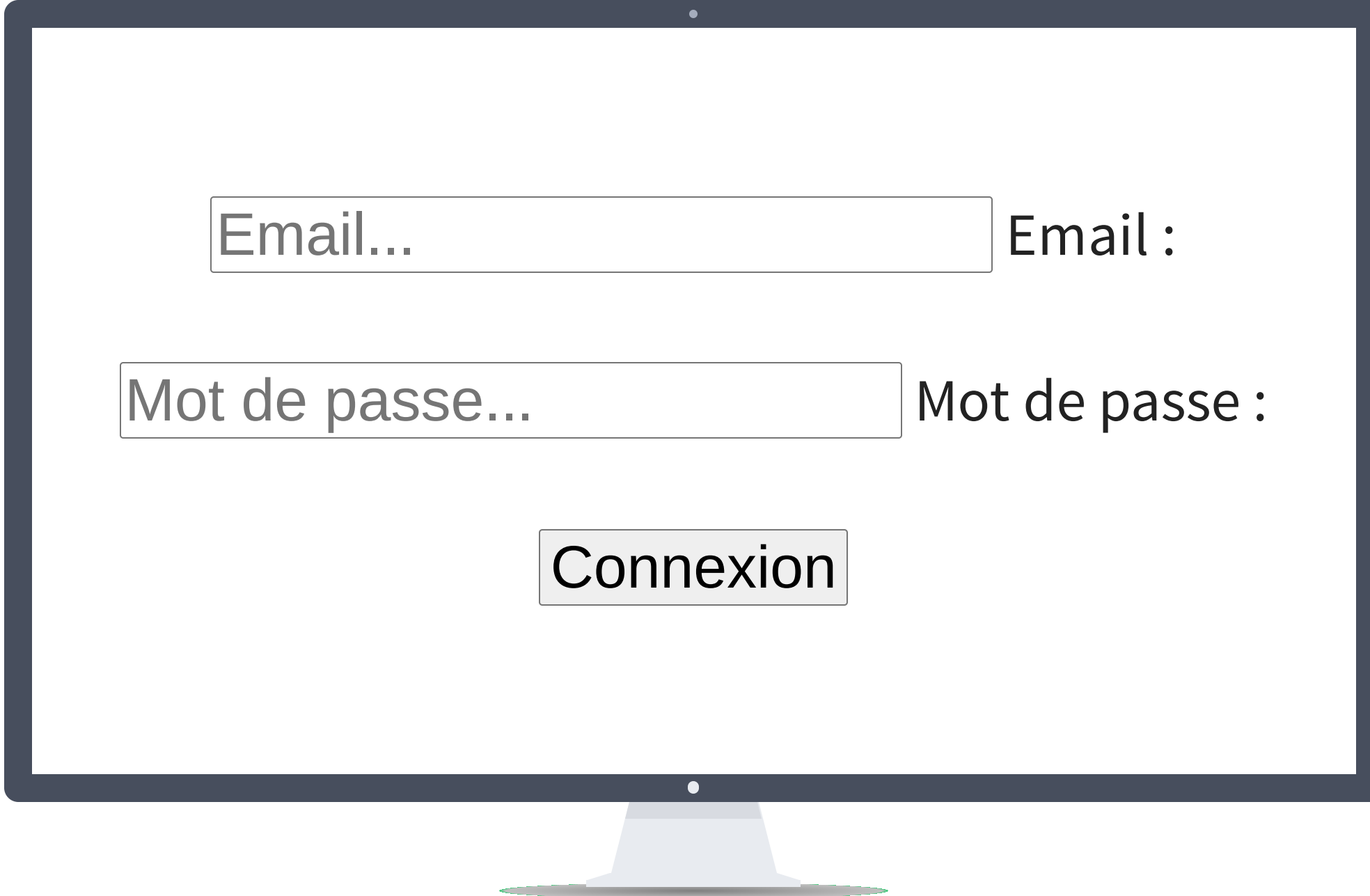
## STYLISER LE LABEL



index.html style.css

```
1 form {
2   display: flex;
3   gap: 20px;
4   justify-content: center;
5   align-items: center; }
6
7 input {
8   outline: none;
9   border: none;
10  border-bottom: thick solid gray; }
11
12 input[type="submit"] {
13   border-color: #6666aa;
14   background-color: lightblue;
15   padding: .2em 1em;
16   border-radius: 20px; }
```

## STYLISER LE LABEL



Email... Email :

Mot de passe... Mot de passe :

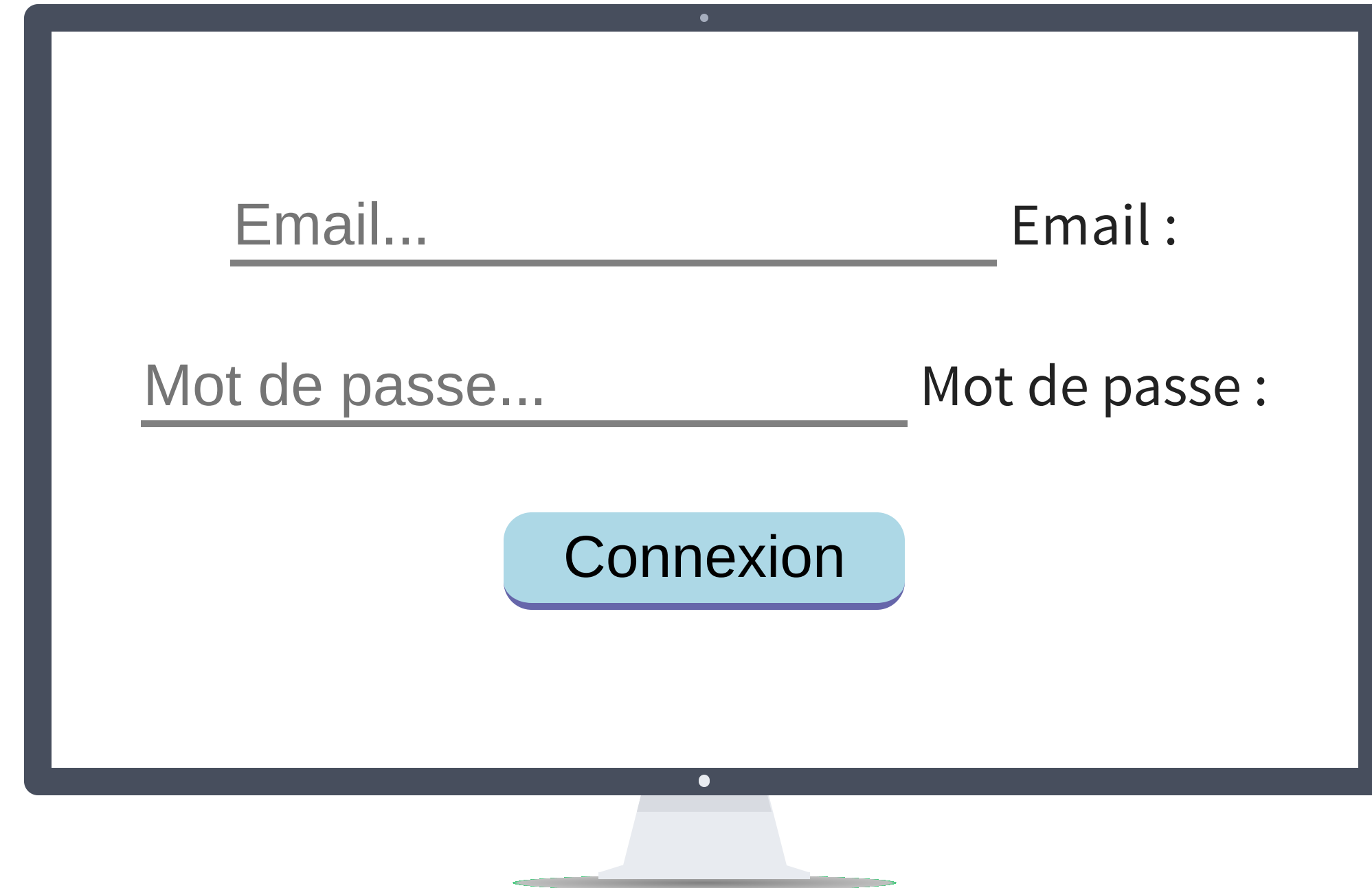
Connexion



index.html style.css

```
1 form {
2   display: flex;
3   gap: 20px;
4   justify-content: center;
5   align-items: center; }
6
7 input {
8   outline: none;
9   border: none;
10  border-bottom: thick solid gray; }
11
12 input[type="submit"] {
13   border-color: #6666aa;
14   background-color: lightblue;
15   padding: .2em 1em;
16   border-radius: 20px; }
```

# STYLISER LE LABEL



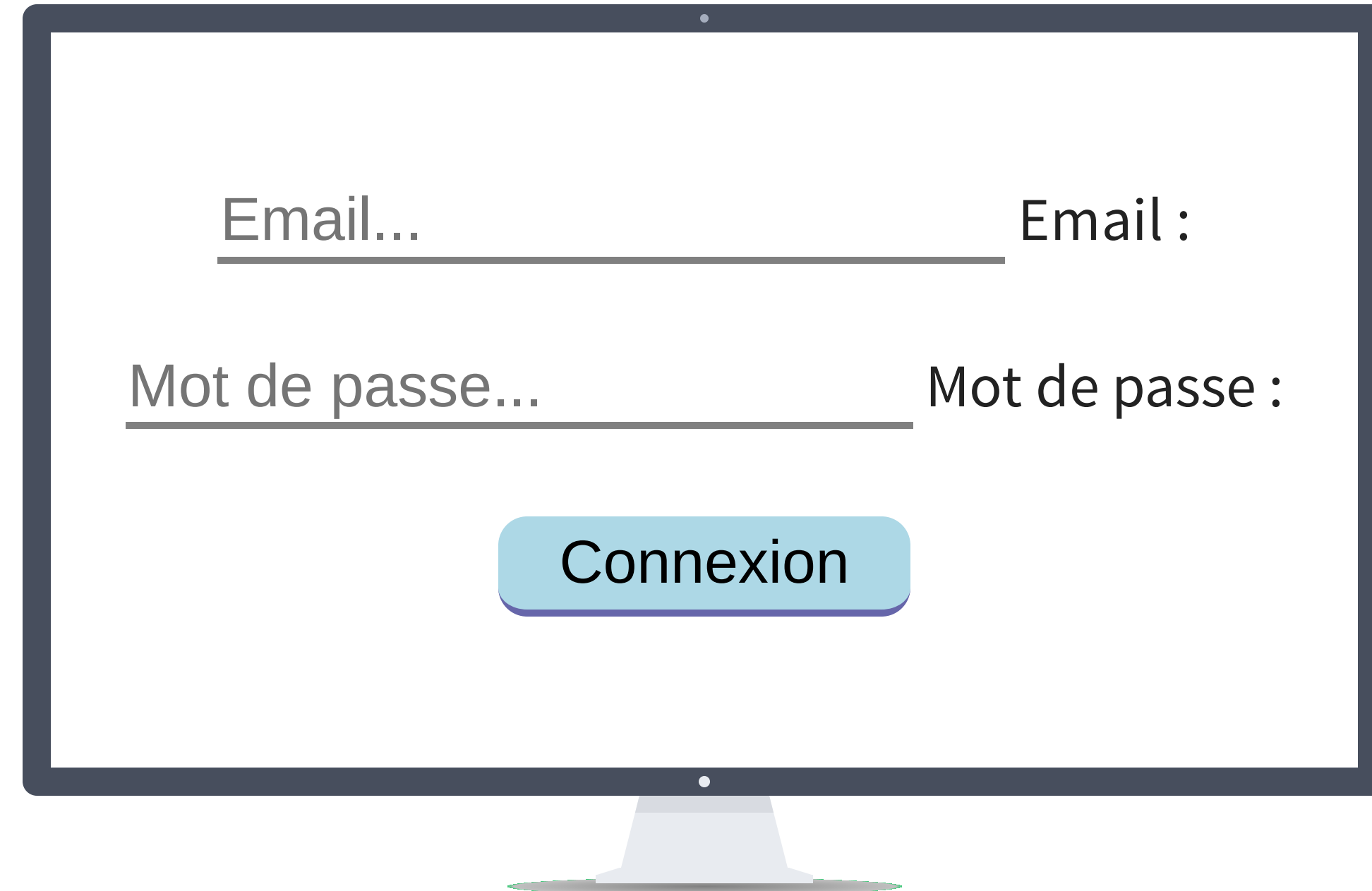
index.html

style.css

style.css (suite)

```
1 .field {
2   position: relative;
3 }
4
5 label {
6   position: absolute;
7   display: none; /* cache le label */
8   top: 0;
9   left: 0;
10  font-size: smaller;
11  font-weight: bold;
12 }
13
14 input:not(:placeholder-shown) + label {
15   display: block;
16 }
```

# STYLISER LE LABEL



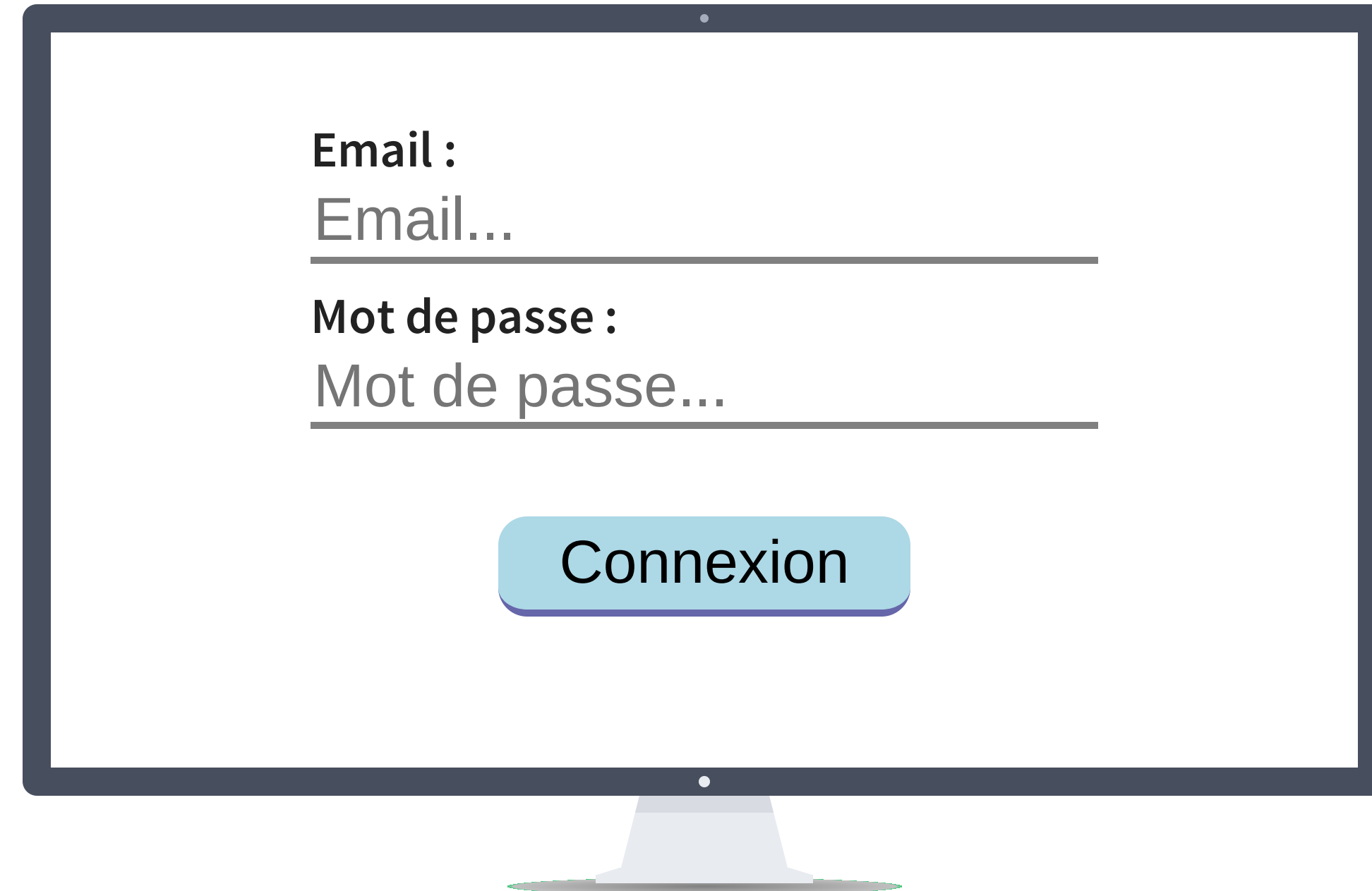
index.html

style.css

style.css (suite)

```
1 .field {
2   position: relative;
3 }
4
5 label {
6   position: absolute;
7   display: none; /* cache le label */
8   top: 0;
9   left: 0;
10  font-size: smaller;
11  font-weight: bold;
12 }
13
14 input:not(:placeholder-shown) + label {
15   display: block;
16 }
```

# STYLISER LE LABEL



**IUT2**   
Université Grenoble Alpes

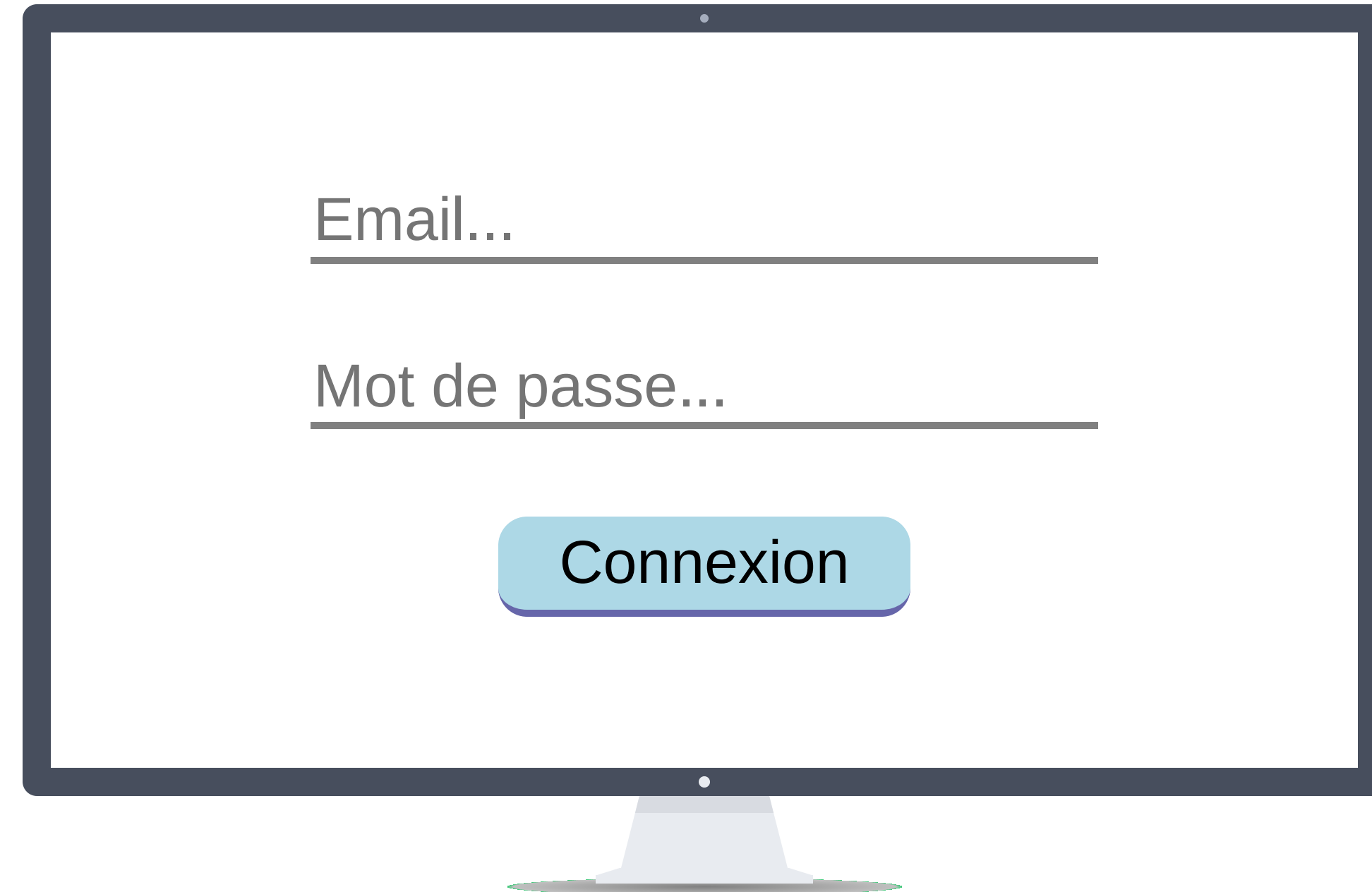
index.html

style.css

style.css (suite)

```
1 .field {
2   position: relative;
3 }
4
5 label {
6   position: absolute;
7   display: none; /* cache le label */
8   top: 0;
9   left: 0;
10  font-size: smaller;
11  font-weight: bold;
12 }
13
14 input:not(:placeholder-shown) + label {
15   display: block;
16 }
```

# STYLISER LE LABEL



**IUT2**   
Université Grenoble Alpes

# EN SAÉ (ET PORTFOLIO)

- Réalisation de formulaires pour la page de contact, formulaire de recherche, etc.
- Mettre en place le formatage avec CSS (pensez au Flex !)
- La partie serveur (et vérification cliente) ne sera pas fonctionnel, c'est normal !
  - BUT2 programmation côté serveur avec PHP
  - BUT2 (parcours A) programmation côté client avec JavaScript

