

CORRIGE - exercice 4 - HTML_CSS_TypeScript

Architecture

Se reporter au fichier « **Application HTML_CSS_TS préparation et exécution.pdf** » - **étape 1** pour les explications détaillées.

Sur votre « devhom », emplacement où se trouve vos applications web sur le serveur « devweb » de l'IUT de Metz, dans le dossier créé ou à créer IHM,

- créer le dossier TP4
- copier le contenu du dossier « squelette » (fichier compressé « squelette » à télécharger sur Arche), dans le dossier TP4
 - le fichier « tsconfig.json » et les dossiers « vue », « src/contrôleur » et « src/modele » apparaissent
 - le TP4 n'utilise pas de données extraites d'une base de données → supprimer le dossier « modele »

Travail à faire

- écrire le fichier « **tp4.html** » dans le dossier « **vue** » et le fichier **tp4.css** associé dans le sous-dossier « **css** » du dossier « vue » à partir de la maquette ci-contre.

Formulaire de saisie de la fiche de renseignements

Formulaire de saisie de la fiche de renseignements. Le formulaire est divisé en quatre sections : IDENTIFICATION, COMPETENCES, COMMENTAIRE et un bouton Valider. Dans la section IDENTIFICATION, il y a des boutons radio pour Madame et Monsieur, des champs de texte pour Nom (nom) et Prénom (prénom), et un champ de date pour Né le (jj/mm/aaaa). Dans la section COMPETENCES, il y a un menu déroulant pour Diplôme obtenu le plus élevé, un champ de texte pour Expérience professionnelle (en années), et des cases à cocher pour Systèmes d'exploitation (Linux, Windows, Autre). Dans la section COMMENTAIRE, il y a un grand champ de texte.

Affichage du récapitulatif après validation

Affichage du récapitulatif après validation. Le formulaire est divisé en quatre sections : IDENTIFICATION, COMPETENCES, COMMENTAIRE et RECAPITULATIF. Dans la section IDENTIFICATION, Madame est sélectionnée, Nom est Martin, Prénom est Albertine, et Né le est 09/05/1990. Dans la section COMPETENCES, Diplôme obtenu le plus élevé est BUT, LP, Licence, Expérience professionnelle (en années) est 11, et Systèmes d'exploitation sont Linux et Autre (android). Dans la section COMMENTAIRE, le texte est "Travaille actuellement dans une société de services et d'ingénierie en Informatique". Dans la section RECAPITULATIF, le résumé est : Mme Albertine Martin, 09/05/1990, linux, android, 11 ans d'expérience, diplôme le plus élevé : BUT, LP, Licence. Un bouton Corriger est visible en bas à droite.

CORRIGE - exercice 4 - HTML_CSS_TypeScript

Conception HTML/CSS

Se reporter au fichier « **Application HTML_CSS_TS préparation et exécution.pdf** » - **étape 2**

Créer le fichier « vue/tp4.html »

1) Définir la structure globale du fichier « tp4.html » avec les parties

- <head> avec le lien vers le fichier « tp4.css »
- <body>
- <script> avec l'appel du fichier « tp4.js »

```
<!DOCTYPE html>
<html>
  <head>
    <meta http-equiv="content-type" content="text/html; charset=UTF-8">
    <title> Fiche de renseignements </title>
    <link type="text/css" rel="stylesheet"
          href= "../css/tp4.css">
  </head>
  <body>
  </body>
  <script type="module" src = "../controleur/tp4.js">
  </script>
</html>
```

2) Définir une zone identifiée « div_tp4 » dans <body>

```
<!DOCTYPE html>
<html>
  <head>
    <meta http-equiv="content-type" content="text/html; charset=UTF-8">
    <title> Fiche de renseignements </title>
    <link type="text/css" rel="stylesheet"
          href= "../css/tp4.css">
  </head>
  <body>
    <div id='div_tp4'>
    </div>
  </body>
  <script type="module" src = "../controleur/tp4.js">
  </script>
</html>
```

3) Est-ce utile que la zone occupe entièrement l'espace écran ? **pas nécessairement, on choisit tout de même d'occuper l'intégralité de l'espace écran.**

4) Définir le style associé à la zone ?

tp4.css

```
#div_tp4 {  
  width      : 100%;  
  height     : auto;  
  border     : 1px solid #808080;  
  font-size  : 1.1rem;  
  background-color : lightgrey;  
}
```

CORRIGE - exercice 4 - HTML_CSS_TypeScript

- 5) Déterminer le nombre de zones horizontales y compris le titre et définir les zones sans définir leur contenu.

Fiche de renseignements

IDENTIFICATION

☐ Madame ☐ Monsieur

Nom Prénom

Né le

COMPETENCES

Diplôme obtenu le plus élevé

Expérience professionnelle (en années)

Systemes d'exploitation

☐ Linux ☐ Windows ☐ Autre

COMMENTAIRE

Valider

RECAPITULATIF

Mme Albertine Martin
09/05/1990
linux, android
11 ans d'expérience
diplôme le plus élevé : BUT, LP, Licence

Corriger

```
<div id='div_tp4'>
  <header>
    Fiche de renseignements
  </header>
  <div id="div_formulaire">
    <div class="labeltitre">IDENTIFICATION</div>
    <div class="div_identification">
    </div>
    <div class="labeltitre">COMPETENCES</div>
    <div class="div_competence">
    </div>
    <div class="labeltitre">COMMENTAIRE</div>
    <div class="div_commentaire">
    </div>
    <div id="div_formulaire_boutons" class="boutons">
    </div>
  </div>
  <div id="div_recap">
    <div class="labeltitre">RECAPITULATIF</div>
    <div class="div_commentaire">
    </div>
    <div id="div_recap_boutons" class="boutons">
    </div>
  </div>
</div>
```

CORRIGE - exercice 4 - HTML_CSS_TypeScript

```
tp4.css
header {
  background-color : #808080;
  color           : #ffffff;
  margin-bottom  : 1rem;
  text-align     : center;
}
.labeltitre {
  color           : #ffffff;
  background-color : #5b5b5b;
  font-weight     : bold;
  padding         : 0.2em;
  width           : 12rem;
  margin-top      : 1rem;
  margin-bottom   : 0.5rem;
}
```

CORRIGE - exercice 4 - HTML_CSS_TypeScript

- 6) Déterminer pour chaque zone,
- si un découpage en colonnes est nécessaire ?
 - définir le contenu et ajouter les styles nécessaires

6.1) *div_identification* : **OUI (2 colonnes tel que la seconde colonne contient deux blocs et chaque bloc est divisé en 4 colonnes)**

```
<div class="div_identification">
  <span>
    <input type="radio" id="radio_madame" name="civilite" value="Mme">
    <label for="madame">Madame</label>
    <br>
    <input type="radio" id="radio_monsieur" name="civilite" value="M.">
    <label for="monsieur">Monsieur</label>
  </span>
  <span>
    <div class="div_identite">
      <label for="edt_nom">Nom</label>
      <span>
        <input type="text" id="edt_nom" placeholder="nom" size="25" >
      </span>
      <label for="edt_prenom">Prénom</label>
      <span>
        <input type="text" id="edt_prenom" placeholder="prénom" size="25" >
      </span>
    </div>
    <br>
    <div class="div_identite">
      <label for="edt_dtnais">Né le</label>
      <span>
        <input type="date" id="edt_dtnais" size="10">
      </span>
    </div>
  </span>
</div>
```

tp4.css

```
.div_identification {
  display      : grid;
  grid-template-columns : 2fr 8fr; /* total à 10 */
  margin-left   : 1rem;
}

.div_identite {
  display      : grid;
  grid-template-columns : 1fr 3fr 1fr 3fr;
}

label {
  text-align   : right;
  margin-right : 1rem;
}

input {
  border      : 1px solid #5b5b5b;
  color       : #000000;
  font        : inherit;
  text-align  : left;
}

input[type=radio] {
  cursor      : pointer;
  width       : 1.1rem;
  height      : 1.1rem;
}
```

CORRIGE - exercice 4 - HTML_CSS_TypeScript

6.2) *div_competence* : **OUI (2 colonnes tel que la première colonne contient deux blocs et chaque bloc est divisé en 2 colonnes)**

Diplôme obtenu le plus élevé	<input type="text"/>
Expérience professionnelle (en années)	<input type="text"/>
Systèmes d'exploitation	
<input type="checkbox"/> Linux	
<input type="checkbox"/> Windows	
<input type="checkbox"/> Autre <input type="text" value="à préciser"/>	

```
<div class="div_competence">
  <span>
    <div class="div_complement">
      <label for="select_diplome">Diplôme obtenu le plus élevé</label>
      <span>
        <select id="select_diplome">
          <option value=""></option>
          <option value="Aucun">Aucun</option>
          <option value="Bac">Bac</option>
          <option value="BTS, DUT">BTS, DUT</option>
          <option value="BUT, LP, Licence">BUT, LP, Licence</option>
          <option value="Master">Master</option>
          <option value="Doctorat">Doctorat</option>
        </select>
      </span>
    </div>
    <br>
    <div class="div_complement">
      <label for="edt_experience">Expérience professionnelle<br>
        <i>(en années)</i></label>
      <span>
        <input type="number" id="edt_experience" size="3">
      </span>
    </div>
  </span>
  <span>
    <label>Systèmes d'exploitation</label>
    <br>
    <input type="checkbox" id="chk_linux" name="systeme" value="linux">
    <label for="linux">Linux</label>
    <br>
    <input type="checkbox" id="chk_windows" name="systeme" value="windows">
    <label for="windows">Windows</label>
    <br>
    <input type="checkbox" id="chk_autre" name="systeme" value="autre">
    <label for="autre">Autre</label>
    <input type="text" id="edt_autre" placeholder="à préciser" size="20"
  </span>
</div>
```


CORRIGE - exercice 4 - HTML_CSS_TypeScript

```
tp4.css
.div_competence {
  display      : grid;
  grid-template-columns : 6fr 4fr;
  margin-left   : 1rem;
}
.div_complement {
  display      : grid;
  grid-template-columns : 1fr 1fr;
}

input[type=checkbox] {
  cursor      : pointer;
  width       : 1.1rem;
  height      : 1.1rem;
}
select {
  font        : inherit;
  background-color : white;
  border      : none;
}
```

6.3) *div_commentaire* : **NON**

```
<div class="div_commentaire">
  <textarea id="edt_commentaire" rows="3" ></textarea>
</div>
```

```
tp4.css
.div_commentaire {
  display      : grid;
  grid-template-columns : 1fr;    /* toute la largeur ==>
le textarea occupera la totalité de la largeur */
  margin-left   : 1rem;
  margin-right  : 1rem;
}
textarea[id=edt_commentaire] {
  resize : vertical;
}
```

CORRIGE - exercice 4 - HTML_CSS_TypeScript

6.4) *div_formulaire_boutons* : **NON**

Valider

```
<div id="div_formulaire_boutons" class="boutons">  
  <input id="btn_valider" type="button" value="Valider">  
</div>
```

```
tp4.css  
.boutons {  
  text-align : right;  
  margin-right: 1rem;  
  margin-top : 1rem;  
}  
  
input[type=button] {  
  cursor      : pointer;  
  border      : none;  
  background-color : #00b0ff;  
  color       : #000000;  
  height      : 2rem;  
  vertical-align : middle;  
  min-width   : 8rem;  
  text-align   : center;  
}  
  
input[type=button]:hover {  
  background-color : #0000ff;  
  border          : 1px solid #ffffff;  
  color           : #ffffff;  
}
```

CORRIGE - exercice 4 - HTML_CSS_TypeScript

6.5) *div_commentaire* : **NON**

Mme Albertine Martin
09/05/1990
linux, android
11 ans d'expérience
diplôme le plus élevé : BUT, LP, Licence

```
<div class="div_commentaire">  
  <select id="select_recap" size="5"> </select>  
</div>
```

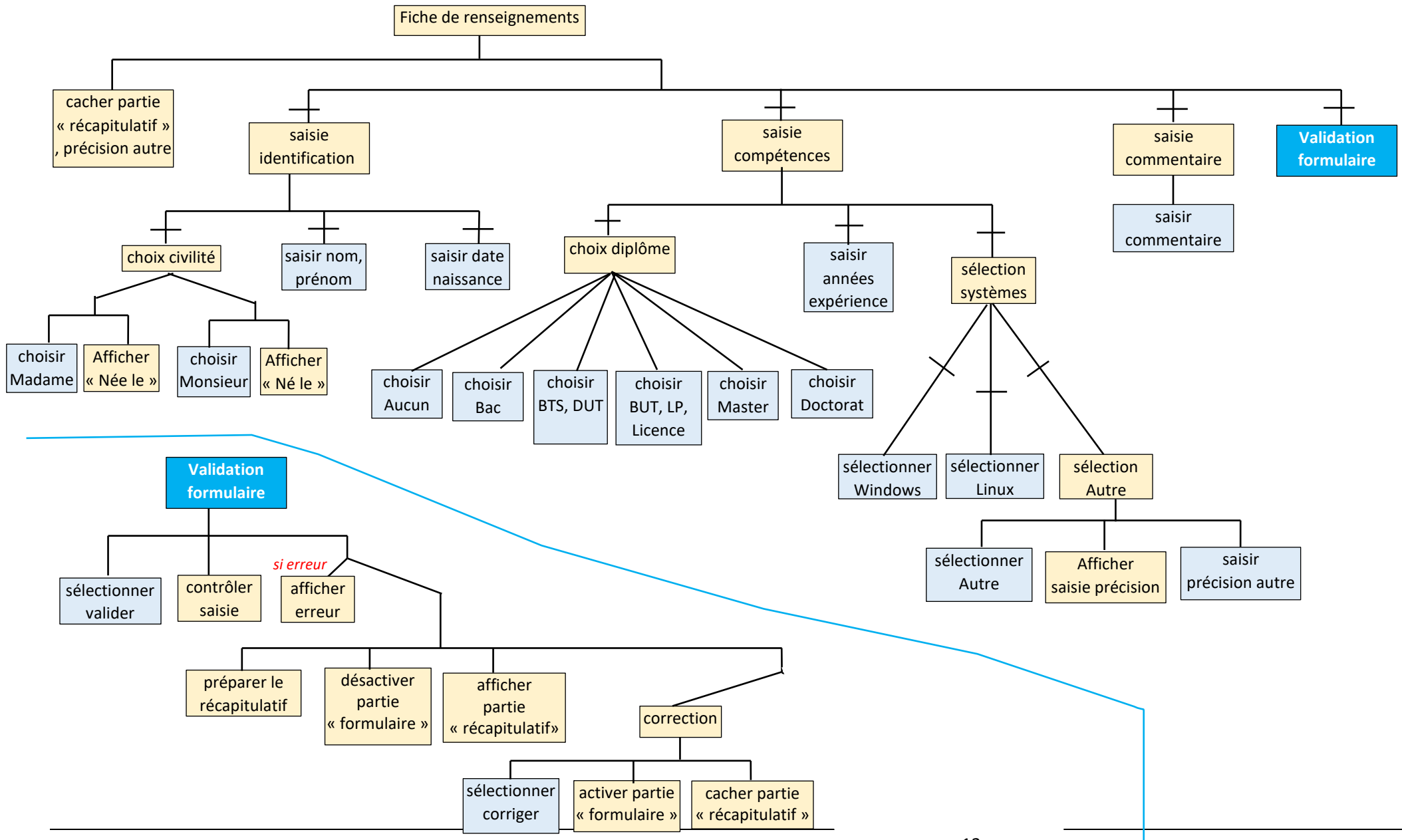
6.6) *div_recap_boutons* : **NON**

Corriger

```
<div id="div_recap_boutons" class="boutons">  
  <input id="btn_corriger" type="button" value="Corriger">  
</div>
```

CORRIGE - exercice 4 - HTML_CSS_TypeScript

le modèle des tâches (en gris bleu : les composants à placer dans la maquette)



Se reporter au fichier « **Application HTML_CSS_TS préparation et exécution.pdf** » - **étape 3**

Créer le fichier « **src/controleur/class_tp4.ts** »

1) Définir le type « TTp4Form »

```
type TTp4Form = {  
  divFormulaire : HTMLElement  
  , divRecap : HTMLElement  
  , divFormBoutons : HTMLElement  
  , radioMadame : HTMLInputElement  
  , radioMonsieur : HTMLInputElement  
  , edtNom : HTMLInputElement  
  , edtPrenom : HTMLInputElement  
  , edtDtNais : HTMLInputElement  
  , listeDiplome : HTMLSelectElement  
  , edtExperience : HTMLInputElement  
  , chkWindows : HTMLInputElement  
  , chkLinux : HTMLInputElement  
  , chkAutre : HTMLInputElement  
  , edtAutre : HTMLInputElement  
  , edtCommentaire : HTMLInputElement  
  , listeRecap : HTMLSelectElement  
  , btnValider : HTMLInputElement  
  , btnCorriger : HTMLInputElement  
}
```

2) Créer la classe « VueTp3 » et définir l'attribut privé « _form », la fonction « init », le getter associé à l'attribut privé

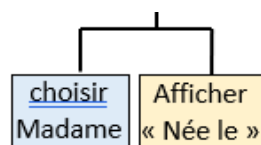
```
class VueTp4 {  
  private _form : TTp4Form  
  
  init(form : TTp4Form) : void {  
    this._form = form;  
    // cacher la partie "récapitulatif"  
    // cacher la zone de saisie à renseigner quand la case "autre" est décochée  
    this.form.divRecap.hidden = true;  
    this.form.edtAutre.hidden = true;  
  }  
  
  get form() : TTp4Form { return this._form }  
}
```

Fiche de renseignements

cacher partie
« récapitulatif »
, précision autre

3) Quels sont les événements/actions déclenchés par l'utilisateur ?

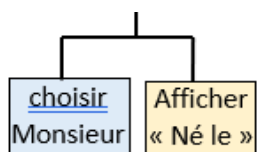
A partir du modèle des tâches



« **click** » sur la civilité « radio_madame »

→ « Née le »

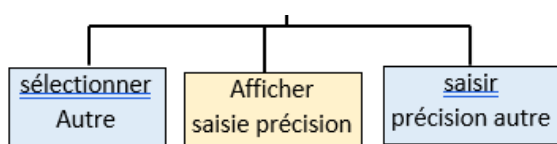
→ appel de la méthode **changeCivilité()**



« **click** » sur la civilité « radio_monsieur »

→ « Né le »

→ appel de la méthode **changeCivilité()**

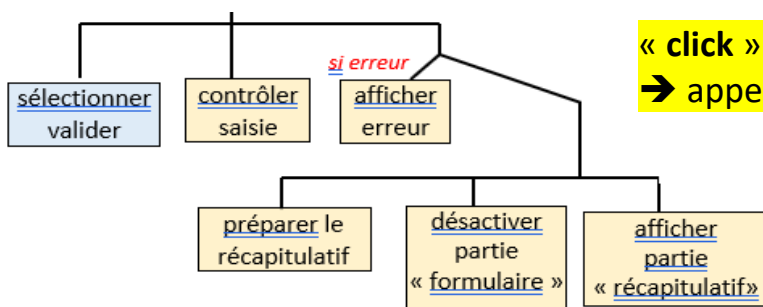


« **change** »

sur la sélection du système « chk_autre »

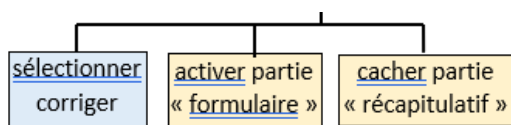
(coché → afficher « edt_autre »
, non coché → cacher « edt_autre »)

→ appel de la méthode **systemeAutreChange()**



« **click** » sur le bouton « btn_valider »

→ appel de la méthode **valideSaisie()**



« **click** » sur le bouton « btn_corriger »

→ appel de la méthode **corrigeSasie()**

- 4) Définir les méthodes de la classe `changeCivillite()`, `systemeAutreChange()`, `valideSaisie()`, `corrigeSaisie()`

```
class VueTp4 {
```

```
    • • •
```

```
    changeCivillite() :void {
```

```
        // à compléter
```

```
    }
```

```
    systemeAutreChange() :void {
```

```
        // à compléter
```

```
    }
```

```
    valideSaisie() :void {
```

```
        // à compléter
```

```
    }
```

```
    corrigeSaisie() :void {
```

```
        // à compléter
```

```
    }
```

```
}
```

```
let vueTp4 = new VueTp4;
```

5) Créer l'objet « vueTp4 », instanciation de la classe

```
export { vueTp4 }
```

6) Écrire la commande d'exportation de l'objet

- 5) Écrire le corps des méthodes de la classe

```
changeCivillite() : void {
```

```
    // Madame ==> "Née le", Monsieur ==> "Né le"
```

```
    let chaine : string;
```

```
    if (this.form.radioMadame.checked) { chaine = "Née le" }
```

```
    else { chaine = "Né le" }
```

```
    // changement de la balise <label> liée à la zone de saisie "edt_nais"
```

```
    this.form.edtDtNais.labels[0].textContent = chaine;
```

```
}
```

```
systemeAutreChange() : void {
```

```
    // cacher la zone de saisie "edt_autre" si la case "autre" est décochée
```

```
    this.form.edtAutre.hidden = !this.form.chkAutre.checked;
```

```
    this.form.edtAutre.value = '';
```

```
    // placer le focus sur la zone de saisie
```

```
    this.form.edtAutre.focus();
```

```
}
```

```

valideSaisie() : void {
    // récupération des données du formulaire dans des variables
    let civilite = "";
    if (this.form.radioMadame.checked) {
        civilite = this.form.radioMadame.value
    }
    else if (this.form.radioMonsieur.checked) {
        civilite = this.form.radioMonsieur.value
    }

    const nom      : string    = this.form.edtNom.value.trim();
    const prenom   : string    = this.form.edtPrenom.value.trim();
    const dtuais   : Date      = this.form.edtDtNais.valueAsDate;
    const experience : string  = this.form.edtExperience.value;
    const diplome   : string    = this.form.listeDiplome.value;
    const autre     : string    = this.form.edtAutre.value.trim();

    let systeme = "";
    // récupérer toutes les cases à cocher de "système"
    if (this.form.chkWindows.checked) {
        systeme += this.form.chkWindows.value + ', '
    }
    if (this.form.chkLinux.checked) {
        systeme += this.form.chkLinux.value + ', '
    }
    systeme += autre;

    // traitement des erreurs
    let erreur = "";
    if (civilite.length === 0) { erreur += "Civilité à sélectionner<br>"; }
    if (nom.length === 0) { erreur += "Nom à renseigner<br>"; }
    if (prenom.length === 0) { erreur += "Prénom à renseigner<br>"; }
    if (dtuais === null) { erreur += "Date de naissance à renseigner<br>"; }
    if (diplome.length === 0) { erreur += "Diplôme à renseigner<br>"; }

    const nb = Number(experience); // si la valeur n'est pas un nombre, nb ← 0
    if (nb < 1) { erreur += "Expérience doit être supérieure à 1 année<br>"; }

    if (systeme.length === 0) { erreur += "Au moins un système à cocher<br>"; }
    else if (!this.form.edtAutre.hidden) {
        // si système "autre" est coché alors "autre" ne doit pas être vide
        if (autre.length === 0) { erreur += 'Système "autre" à préciser<br>'; }
    }
}

```



```

// affichage des informations dans la partie récapitulatif
// sinon affichage message d'erreur
if (erreur.length === 0) {
// rendre inactif la partie "formulaire"
// rendre visible la partie "récapitulatif"
// cacher les boutons du formulaire
this.form.divFormulaire.style.pointerEvents = 'none';
this.form.divRecap.hidden = false;
this.form.divFormBoutons.hidden = true;

const liste = this.form.listeRecap;
// vider la liste avant de la remplir à nouveau
liste.length = 0;

liste.options.add(new Option(civilite + ' ' + prenom + ' ' + nom));
liste.options.add(new Option(dtnais.toLocaleDateString('fr')));
liste.options.add(new Option(systeme));
let an = 'an';
// gestion du pluriel du nombre d'années d'expérience
if (experience > '1') { an = an + 's'; }
liste.options.add(new Option(experience + ' ' + an + " d'expérience"));
liste.options.add(new Option('diplôme le plus élevé : ' + diplome ));
}
else { alert('Erreur dans le formulaire ' + erreur); }
}

corrigeSaisie() : void {
// demande de correction de la saisie
// ==> rendre actif la partie "formulaire", cacher la partie "récapitulatif",
// rendre visible les boutons du formulaire
this.form.divFormulaire.style.pointerEvents = 'auto';
this.form.divRecap.hidden = true;
this.form.divFormBoutons.hidden = false;
}

```

8) Ajouter la définition des événements dans la fonction « init »

```
init(form :Ttp4Form):void {  
    ...  
  
    this.form.radioMadame.onclick  
        = function ():void { vueTp4.changeCivilite(); }  
    this.form.radioMonsieur.onclick  
        = function ():void { vueTp4.changeCivilite(); }  
    this.form.chkAutre.onChange  
        = function ():void { vueTp4.systemeAutreChange(); }  
    this.form.btnValider.onclick  
        = function ():void { vueTp4.valideSaisie(); }  
    this.form.btnCorriger.onclick  
        = function ():void { vueTp4.corrigeSaisie(); }  
}
```

Créer le fichier « src/controleur/tp4.ts »

- 1) Écrire la commande d'importation de l'objet créé précédemment

```
import {vueTp4} from "../controleur/class_tp4"
```

- 2) Appeler la méthode « init » avec les paramètres

```
vueTp4.init (  
{   divFormulaire :document.querySelector('[id=div_formulaire]')  
    ,divRecap      :document.querySelector('[id=div_recap]')  
    ,divFormBoutons:document.querySelector('[id=div_formulaire_boutons]')  
    ,radioMadame   :document.querySelector('[id=radio_madame]')  
    ,radioMonsieur :document.querySelector('[id=radio_monsieur]')  
    ,edtNom        :document.querySelector('[id=edt_nom]')  
    ,edtPrenom     :document.querySelector('[id=edt_prenom]')  
    ,edtDtNais     :document.querySelector('[id=edt_dtnais]')  
    ,listeDiplome  :document.querySelector('[id=select_diplome]')  
    ,edtExperience :document.querySelector('[id=edt_experience]')  
    ,chkWindows   :document.querySelector('[id=chk_windows]')  
    ,chkLinux      :document.querySelector('[id=chk_linux]')  
    ,chkAutre      :document.querySelector('[id=chk_autre]')  
    ,edtAutre      :document.querySelector('[id=edt_autre]')  
    ,edtCommentaire:document.querySelector('[id=edt_commentaire]')  
    ,listeRecap    :document.querySelector('[id=select_recap]')  
    ,btnValider    :document.querySelector('[id=btn_valider]')  
    ,btnCorriger   :document.querySelector('[id=btn_corriger]')  
} );
```

Transpiler vos fichiers « ts » et tester l'application

Se reporter au fichier « **Application HTML_CSS_TS préparation et exécution.pdf** » - **étape 4**