

Exercice 1 :



Mise en forme d'un texte

Options de Mise en forme

Police

Taille

Couleur

Appliquer à l'arrière plan

☐

Appliquer au Texte

☐

Texte à transformer

Transformer

Nombre de caractères :

```

1. <!DOCTYPE html>
2. <html lang="en">
3. <head>
4. <meta charset="UTF-8">
5. <meta name="viewport" content="width=device-width, initial-scale=1.0">
6. <link rel="stylesheet" href="tp2.css">
7. <script src="tp2.js"></script>
8. <title>Document</title> </head>
9. <body onload="salut()" <header>
10. 
11. <h2>Mise en forme d'un texte</h2>
12. </header>
13. <form>
14. <fieldset>
15. <legend>Options de Mise en forme</legend>
16. <label for="police">Police</label>
17. <input type="text" list="polices" name="police" id="police">
18. <datalist id="polices">
19. <option value="Verdana"></option> <option value="Courier New"></option>
20. <option value="Georgia"></option> </datalist>
21. <label for="taille">Taille</label>
22. <input type="range" name="taille" id="taille" min="8" max="24">
23. <label for="couleur">Couleur</label>
24. <select name="couleur" id="couleur">
25. <option value="Red">Red</option> <option value="Blue">Blue</option>
26. <option value="Green">Green</option> <option value="Yellow">Yellow</option>
27. <option value="Black">Black</option></select>
28. <label for="Arrière plan">Appliquer à l'arrière plan</label>
29. <input type="radio" name="zone" id="arriere_plan" value="a">
30. <label for="Texte">Appliquer au Texte</label>
31. <input type="radio" name="zone" id="couleur_texte" value="t" </fieldset>
32. </fieldset> <legend>Texte à transformer</legend>

```

```

33. <textarea name="texte1" id="texte1" value="" onkeyup="longueur()" > </textarea><br>
34. <input type="button" value="Transformer" onclick="transformer()" > <br>
35. <p>Nombre de caractères : <span id="longueur" ></span> </p>
36. </fieldset></form></body></html>

```

Dans une école primaire l'enseignante de discipline informatique veut faire une application qui fait une **mise à jour du texte** pour mieux expliquer aux élèves les différentes modifications peuvent être appliquer sur un texte, si joint l'interface crée : Soit l'extrait du code HTML :

- 1- Indiquer devant chaque rôle présent dans le tableau ci-dessous le numéro de la ligne contenant le code HTML

Rôle	N°Ligne
Fournir les informations supplémentaires sur la page web qui seront utilisées par les moteurs de recherche	5
Spécifier l'encodage des caractères de la page web	4
Lier une feuille de style CSS externe à la page Web	6
Exécuter des fonctions JavaScript au moment où la page web est entièrement chargée	9
Créer une dataliste	15
Créer une liste déroulante	22
Regrouper un ou plusieurs éléments d'un formulaire	14
Créer un bouton permettant d'appliquer la fonction de modification	34
Un type d'input qui sert à sélectionner une valeur dans une plage définie	22

- 2- Terminer les lignes suivants extrait du code HTML en se basant sur la liste suivante :

Src, onkeyup, onmouseover, span, onclick, onmouseout

- 1 : source d'image , 2 : quand on passe la souris ; 3 : quand la souris sort du cadre ; 4 : au clic sur l'un des boutons du clavier ; 5 : au clic sur le bouton ; 6 : un conteur en ligne permer d'afficher un contenu

- a-
- b- <textarea name="texte1" id="texte1" value="" ⁽⁴⁾ = "longueur()">
- c- <input type="button" value="Transformer" ⁽⁵⁾ = "transformer()">
- d- <p>Nombre de caractères : < ⁽⁶⁾ id="longueur" ></.....> </p>

3- Liste d'instructions

margin: 5px; position:relative; font-family: d'Arial Narrow', Arial, sans-serif ; font-size: 18px; font-weight: bold; left:200px; display: block;	
Mise en forme	Code CSS
Tous les éléments doivent être : ● Police du texte : arial ● Style de caractère gras ● Taille de caractère égal à 18px ● Une marge : 5px ;	<pre>*{font-family: d'Arial Narrow', Arial, sans-serif; font-weight: bold; font-size: 18px; margin: 5px;}</pre>
Les zones de texte et la liste déroulante ont les mises en formes suivantes: ● Un positionnement relatif ● Un comportement d’affichage bloc ● La position par rapport au gauche est de 200px	<pre>input, select{ left:200px; position:relative; display: block; }</pre>
Le bouton a une marge gauche de 70px	<pre>input[type=submit] { margin-left: 70px; }</pre>
Les boutons radios ont la marge gauche de 220px	<pre>input[type=radio]{ left: 220px; }</pre>
La zone de saisie de texte multilignes : ● Largeur et hauteur de 100% ● Une couleur rouge ● Taille de caractère de 20px ● Le contenu se défile lorsqu’il déborde la zone	<pre>textarea { width: 100%; height:100%; color:red; font-size: 20px; overflow:scroll; }</pre>

Partie JavaScript :

1- Le code de la fonction « activer », dont l’appel figure à la ligne 10 du code HTML, est le suivant

```
function activer()
{ document.getElementById("img1").src="img2.jpg"; }
```

Donner le rôle de cette fonction : l'image se change en passant la souris

.....

2- Le code incomplet de la fonction ‘longueur’, dont l’appel figure au ligne 33 :

a- Terminer le code en se basant sur les taches mentionnées

```
function transformer() {
    resultat=document.getElementById("texte1");
    police2=document.getElementById("police").value;
    taille2=document.getElementById("taille").value;
    couleur2=document.getElementById("couleur").value;
    zon=document.getElementsByName("zone");
    if(zon[0].checked) { // tache1 : signifie que l'arrière plan est coché
    // tache2 : changer la couleur de l'arrière plan
    resultat.....=couleur2; }
```

b- Terminer le code pour donner un style au texte par les styles choisis

```
else if(zon[1].checked) {
    resultat.....; style.fontSize=taille2
    resultat.....; style.fontFamily=police2
    resultat.....; } style.color=couleur2
```

Exercice N°3

Dans le cadre de la gestion des ressources hydrauliques, on dispose d’un site web qui diffuse les données des barrages des différentes régions de la Tunisie. Ces informations sont disponibles sur une cartographie interactive (voir **Figure1** et **Figure2** de la page 13) qui permet d’afficher pour chaque barrage choisi son nom, sa superficie et éventuellement d’autres informations supplémentaires enregistrées dans la base de données «**Gestion_Barrage**».

Généralement, un barrage est équipé de plusieurs vannes qui sont contrôlées par des capteurs permettant de collecter des informations telles que le débit, la pression et la température.

1ère Partie : Gestion de la base de données

La base de données « **Gestion_Barrage** » est décrite par la représentation textuelle simplifiée ci-dessous :

Barrage (idBar, nomBar, volume, hauteur, superficie)

NiveauEau (dateNiv, idBar#, niveau)

Vanne (idVan, etatVan, idBar#)

Capteur (idCapt, refCapt, idVan#)

Mesure (idMes, valMes, dateMes, idCapt#)

Alerte (idAlert, degAlert, statAlert, idMes#)

Les champs des tables sont décrits dans le tableau suivant :

Champ	Description et type
idBar	Identifiant d'un barrage, de type entier.
nomBar	Nom d'un barrage, de type chaîne de caractères.
volume	Capacité maximale d'un barrage exprimée en million m ³ , de type réel.
hauteur	Hauteur d'un barrage exprimée en mètre, de type réel.
superficie	Superficie d'un barrage exprimée en km ² , de type réel.
dateNiv	Date et heure de prélèvement du niveau d'eau d'un barrage.
niveau	Niveau d'eau exprimé en mètre, de type réel.
idVan	Identifiant d'une vanne, de type entier.
etatVan	Etat d'une vanne, de type entier (0 = fermé, 1 = ouvert).
idCapt	Identifiant d'un capteur, de type entier.
refCapt	Référence d'un capteur, de type chaîne de caractères.
idMes	Identifiant d'une mesure enregistrée par un capteur, de type entier.
valMes	Valeur de la mesure enregistrée par un capteur, de type réel.
dateMes	Date et heure d'une mesure enregistrée par un capteur.
idAlert	Identifiant d'une alerte déclenchée suite à une mesure, de type entier.
degAlert	Degré d'une alerte, de type entier (1 : avertissement, 2 : grave, 3 : critique).
statAlert	Statut d'une alerte, de type caractère (" R " : résolu, " A " : actif, " E " : en attente).

II. Manipulation des données et de la structure de la base

N.B. : - 0.25 par type d'erreur syntaxique et - 0.25 par erreur sémantique

A- Ecrire les requêtes SQL permettant de répondre aux questions suivantes :

- 1) Quelles sont les vannes ouvertes du barrage d'identifiant 5 ? on affichera le champ idVan.

```
SELECT idVan
FROM Vanne
WHERE etatVan = 1
AND idBar = 5
```

1.5 Point

- 2) Quels sont les capteurs qui ont générés des alertes ayant un degré critique et dont le statut est non résolu ? On affichera les champs idCapt et refCapt.

```
SELECT C.idCapt , refCapt
FROM Capteur C, Mesure M, Alerte A
WHERE degAlert = 3
AND statAlert != 'R'
AND A.idMes = M.idMes
AND M.idCapt = C.idCapt
```

1.5 Point

// on acceptera aussi statAlert IN ('A', 'E')

- 3) Quel est le nombre de vannes par barrage ? On affichera les champs idBar, nomBar ainsi que le nombre de vannes.

```
SELECT B.idBar, nomBar, COUNT(V.idVan)
FROM Barrage B, Vanne V
WHERE B.idBar = V.idBar
GROUP BY V.idBar
```

1.5 Point



13 JUN 2024

- 4) Quels sont les barrages susceptibles de subir une crue (le niveau d'eau est supérieur à 90% de la hauteur du barrage), pendant le mois en cours de l'année 2024 ? On affichera, sans répétition, les champs idBar et nomBar.

```
SELECT DISTINCT B.idBar, nomBar
FROM Barrage B, NiveauEau N
WHERE niveau > hauteur*0.9
AND MONTH(dateNiv) = MONTH(NOW())
AND YEAR(dateNiv) = 2024
AND N.idBar = B.idBar
```

1.5 Point

// on acceptera aussi MONTH(dateNiv) = 6

// on acceptera aussi YEAR(dateNiv) = YEAR(NOW())

- 5) Quels sont les capteurs dont la dernière date de mesure est effectuée avant le **15 février 2024** ? On affichera les champs *idCapt*, *refCapt* et la dernière date de mesure effectuée.

```
SELECT M.idCapt, refCapt, MAX (dateMes)
FROM Capteur C, Mesure M
WHERE M.idCapt = C.idCapt
GROUP BY M.idCapt
HAVING MAX(dateMes) < '2024-02-15'
```

1.5 Point

B- Ecrire les requêtes SQL permettant de répondre aux situations suivantes :

- 1) Pour alléger la base de données, on se propose de supprimer les alertes dont le statut est "R".

```
DELETE FROM Alerte
WHERE statAlert = 'R'
```

1 Point

- 2) En raison d'un dysfonctionnement du capteur d'identifiant 4, on se propose de diminuer de 2 unités les mesures enregistrées entre le **29 janvier 2024** et le **02 février 2024**.

```
UPDATE Mesure
SET valMes = valMes -2
WHERE idCapt = 4
AND dateMes BETWEEN '2024-01-29' AND '2024-02-03'
```

1 Point

// On acceptera aussi la condition : dateMes BETWEEN '2024-01-29' AND '2024-02-02'

- 3) Pour classer les capteurs selon leurs catégories, on se propose d'ajouter une nouvelle table nommée **Categorie** définie comme suit :

Champ	Description et type
idCat	Identifiant d'une catégorie, de type entier (clé primaire).
nomCat	Nom d'une catégorie, de type chaîne de 50 caractères au maximum.

CREATE TABLE Categorie (idCat INT PRIMARY KEY, nomCat VARCHAR(50)) **1 Point**

- 4) On se propose de restreindre les valeurs permises pour le champ **nomCat** de la table **Categorie**, déjà créée, aux valeurs suivantes : "débit", "pression" et "température".

```
ALTER TABLE Categorie
ADD CONSTRAINT
CHECK (nomCat IN ( "débit", "pression", "température"))
```

1 Point

- 5) On se propose d'établir la relation entre les tables **Capteur** et **Categorie** en ajoutant le champ **idCat** comme clé étrangère dans la table **Capteur**

```
ALTER TABLE Capteur
ADD idCat INT REFERENCES Categorie(idCat)
```

1 Point

// On acceptera aussi :

```
ALTER TABLE Capteur ADD idCat INT
ALTER TABLE Capteur FOREIGN KEY (idCat) REFERENCES Categorie(idCat)
```

// On acceptera aussi :

```
ALTER TABLE Capteur ADD idCat INT,
ADD FOREIGN KEY (idCat) REFERENCES Categorie(idCat)
```



13 JUIN 2024