

RÉPUBLIQUE TUNISIENNE MINISTÈRE DE L'ÉDUCATION L.S : IBN RACHIK	Devoir de synthèse n :2	15/03/2024
	Épreuve : Systèmes et Technologies de l'Informatique	Section : Sciences de l'informatique
	Durée : 2heures	Coefficient de l'épreuve : 3
Prénom : NOM :		

Partie A HTML (0.25 pt *16 = 4 pts)

Pour chacune des questions ci-dessous, mettre une croix (X) dans la case correspondante à la réponse correcte.

Important : Pour chaque question, toute réponse comportant **plus d'une croix** est considérée **erronée**

1- En HTML5, quels sont les éléments nécessaires à la création d'une liste à puces ?

<input type="checkbox"/>	ul et li
<input type="checkbox"/>	ol et li
<input type="checkbox"/>	ul et ol

2- En HTML5, quelle est la balise à utiliser pour regrouper des éléments liés dans un formulaire ?

<input type="checkbox"/>	<label>
<input type="checkbox"/>	<datalist>
<input type="checkbox"/>	<fieldset>

3- En HTML5, quel est l'attribut à ajouter à un élément checkbox pour qu'il soit coché par défaut ?

<input type="checkbox"/>	checked
<input type="checkbox"/>	disabled
<input type="checkbox"/>	selected

4- En HTML5, quelle est la méthode qui permet d'envoyer au serveur les données d'un Formulaire sans les afficher dans la barre d'adresse d'un navigateur ?

<input type="checkbox"/>	GET
<input type="checkbox"/>	POST
<input type="checkbox"/>	HTTP

5- En HTML5, comment faire l'appel du fichier 'contrôle.js' situé dans le dossier comportant le document HTML appelant ?

<input type="checkbox"/>	<script link="controles.js"></script>
<input type="checkbox"/>	<script src="controles.js"></script>
<input type="checkbox"/>	<script href="controles.js"></script>

6- soit l'aperçu d'un formulaire à créer.

Cocher les structures itératives :		
Selon <input type="checkbox"/>	Pour <input checked="" type="checkbox"/>	Répéter <input checked="" type="checkbox"/>

En HTML5, quel est l'élément qui permet à un utilisateur de sélectionner plusieurs réponses ?

- `<input type="radio"/>`
- `<input type="checkbox"/>`
- `<input type="text"/>`

7-soit une page HTML contenant l'élément suivant :

```
<form .....="verif()">
```

Par quel événement faut-il remplacer les pointillés afin d'exécuter la fonction verif()

- SRC
- Onclick
- onsubmit

8-Quelle est la règle CSS à utiliser pour appliquer la couleur rouge aux textes des deux paragraphes d'identifiants respectifs **p1** et **p2** ?

- `#p1 , #p2 {color : red ;}`
- `#p1 || #p2 {color : red ;}`
- `#p1 & #p2 {color : red ;}`

9-Soit la règle CSS suivante : **p.eleve { color : blue ; }**

A quel(s) paragraphe(s) d'un document HTML5 cette règle est appliquée ?

- au paragraphe défini par : `<p id = "eleve"> ... </p>`
- aux paragraphes définis par : `<p name = "eleve"> ... </p>`
- aux paragraphes définis par : `<p class = "eleve"> ... </p>`

10-Par quelle expression doit-on remplacer les pointillés de la balise ci-dessous pour référencer le fichier mesStyles.css dans une page HTML5 ?

```
<..... = "mesStyles.css" rel = "stylesheet" type = "text/css">
```

- link src
- link href
- a href

11-Soit la déclaration HTML5 suivante :

```
<p class = "bloc" > mon paragraphe </p>
```

Quelle est la règle CSS à utiliser pour aligner au centre le texte « mon paragraphe » ?

```
 bloc { text-align : center; }  
 #bloc { text-align : center;}  
 .bloc { text-align : center; }
```

12-En JavaScript, quelle est l'instruction qui permet d'afficher le message « **Bonjour** » ?

```
 echo("Bonjour");  
 print("Bonjour");  
 alert("Bonjour");
```

13- En JavaScript, quelle est l'instruction qui permet d'exprimer un traitement répétitif 5 fois ?

```
 for (i = 0 , i < 5 , i++) { ...}  
 for (i = 0 ; i < 5 ; i++) { ...}  
 for (i = 0 ; i = 5 ; i++) { ...}
```

14-Pour accéder à un élément HTML, JavaScript peut utiliser la méthode.

```
 alert()  
 document.write()  
 document.getElementById(id).
```

15- En JavaScript, la méthode charAt(i).

```
 Renvoie la longueur d'une chaîne de caractères.  
 Renvoie le caractère i d'une chaîne de caractères.  
 Renvoie la première position d'une sous chaîne ch1 dans une chaîne ch.
```

16- En JavaScript, pour Convertir la chaîne x en minuscule on peut utiliser l'expression.

```
 toLowerCase(x)  
 x.toLowerCase  
 x.toLowerCase()
```

Partie B CSS +javascript)(3+2=5 pts))

Ci-après, on présente deux éléments relatifs à un site web permettant la gestion des clients.

- **1^{er} élément** : un aperçu d'une page nommée **ajoutClient.html**, contenant un formulaire pour l'ajout d'un client à la base de données. Sachant que :
 - ✓ la largeur des zones de saisie des champs **Identifiant** et **Téléphone** est égale à **120 px**,
 - ✓ la largeur des zones de saisie des champs **Nom-client** et **E-mail** est égale à **250 px**, □
 - le style de bordure de toutes les zones de saisie est **solid**.
- **2^{ème} élément** : un contenu d'une feuille de style CSS, nommée **Styles.css**, liée à la page **ajoutClient.html** (les deux fichiers sont situés dans un même dossier).

Ajout d'un client

Identifiant

Nom-client

Téléphone

E-mail

1er élément

```

1  .bordure{ border-style: solid;}
2
3  .saisie{ border-style: solid;
4  width: 250px;}
5
6  #saisieNom{ width: 250px;}
7
8  #idclient,#tel{ width: 120px;}
9

```

2ème élément

Aperçu de la page « Ajoutclient.Html »

Contenu de la feuille « style.css »

A. Pour chacune des situations suivantes répondre par **V** si la déclaration HTML5 proposée répond à la situation donnée, ou **F** dans le cas contraire. (0.75 pt *4=3pts)

Important : Toute case vide ou comportant une réponse autre que **V** ou **F** est considérée **erronée**.

1) Mettre en forme la zone de saisie relative au champ **Identifiant** conformément à l'aperçu.

<input type="text"/>
<input type="text"/>
<input type="text"/>
<input type="text"/>

```

<input type = "text" id = "idclient" name = "idclient" class = "bordure"/>
<input type = "text" id = "idclient" name = "idclient" class = "saisie"/>
<input type = "text" id = "idclient" name = "idclient"/>

```

2) Mettre en forme la zone de saisie relative au champ **Nom-Client** conformément à l'aperçu.

<input type="text"/>
<input type="text"/>
<input type="text"/>

```

<input type = "text" id = "nomCl" name = "nomClient" class = "bordure"/>
<input type = "text" id = "saisieNom" name = "nomClient" class = "bordure"/>
<input type = "text" id = "nomCl" name = "nomClient" class = "saisie"/>

```

3) Mettre en forme la zone de saisie relative au champ **Téléphone** conformément à l'aperçu.

<input type="text"/>
<input type="text"/>
<input type="text"/>

```

<input type = "tel" id = "telephone" name = "telephone" class = "bordure"/>
<input type = "tel" id = "telephone" name = "telephone" class = "saisie"/>
<input type = "tel" id = "telephone" name = "telephone"/>

```

4) Attacher à la page **ajoutClient.html** le fichier **controles.js**, se trouvant dans le même dossier.

```

 <script src = "controles.js"></script>
 <script link = "controles.js"></script>
 <script ref = "controles.js"></script>

```

B. On s'intéresse au formulaire de la page ajoutClient.html. (2 pt)

Ci-après, on présente un script incomplet d'une fonction intitulée **verifIdentifiant** permettant de vérifier que la valeur saisie, via le formulaire, de l'identifiant d'un client est une chaîne alphanumérique.

N.B. : On suppose que la valeur saisie de l'identifiant d'un client est non vide.

```

function verifIdentifiant()
{
    x = document.getElementById("idclient").value
    x=..... // Tâche_1 : Convertir x en minuscule

    for(i=0;i<.....;i++) // Tâche_2 : Récupérer la longueur de x
    {
        c=..... // Tâche_3 : Récupérer le ième caractère de x

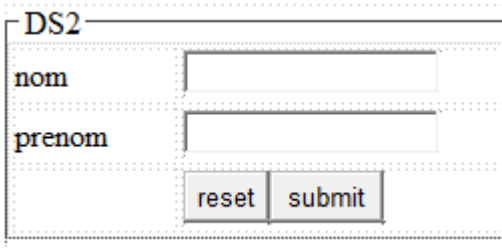
        if (.....( (c >='a' && c <='z') || (c >='0' && c <='9') ) ) // Tâche_4 : Utiliser l'opérateur de négation
        {
            alert('verifier Identifiant')
            return false
        }
    }
}

```

Pour chaque ligne, contenant des pointillés, du script de la fonction **verifIdentifiant**, Compléter le tableau suivant, en écrivant l'expression correcte correspondante à chaque « **Tâche** » indiquée dans la fonction **verifIdentifiant**.

Tâche	Tâche_1	Tâche_2	Tâche_3	Tâche_4
Expression

Partie C PHP (3 pts)

Soit le formulaire suivant :	Code HTML
	<pre><form action="test.php" method="get"> <fieldset > <legend>DS2</legend> <table > <tr><td>nom</td> <td><input name="..." type="text" /></td> </tr><tr> <td>prenom</td> <td><input name="T2" type="text" /></td> </tr> <tr><td></td><td> <input type="reset" value="reset" /> <input type="submit" value="submit" /></td> </tr> </table> </fieldset></form></pre>

On vous propose un script php [test.php](#) permettant de récupérer les données du formulaire ci-dessus, puis se connecter à la base **gestion_eleve** pour vérifier l'existence d'un élève dans la table registre enfin l'insérer.

Sachant que la représentation textuelle de la table registre est

Registre(num,nom,prenom)

Voici le code SQL de la table registre

```
Create table registre(
Num int(2) auto_increment primary key,
Nom varchar(10),
Prenom varchar(20) not null) ;
```

Pour cet effet on vous propose la fonction suivante **test()** :

```
<?php
mysql_connect("localhost","root,");
mysql_select_db("....."); //tache1 nom de la base
$a=.....[“T1”]; //tache2 récupération de donnée
$b=.....[“T2”];
$r=mysql_query("select * from registre where nom='$a' and prenom='$b'")
if (mysql_num_rows($r)==1)
    echo("élève déjà existant",.....) //tache6 afficher le numero de l'eleve
else{
    $r=mysql_query("insert into registre values(.....,'$a','$b')") //tache3 valeur du champs num
    if(.....!=0) //tache4 tester le résultat d'une requête d'insertion
        echo("bienvenue dans notre classe");
    }
?>
```

Pour chaque ligne, contenant des pointillés du script de la fonction **test**, on donne trois propositions d'expressions dont une seule est correcte pour réaliser la « **Tâche** » indiquée.

Question : Compléter le tableau suivant, en écrivant le numéro de l'expression correcte correspondante à chaque « **Tâche** » indiquée dans la fonction **test**.

Tâche_1		Tâche_2		Tâche_3		Tâche_4			
N°	Expression	N°	Expression	N°	Expression	N°	Expression	N°	Expression
1	élève	1	\$ _POST	1	1	1	mysql_fetch_row(\$r)		
2	Gestion_eleve	2	\$ _Post	2	'	2	mysql_fetch_array(\$r)		
3	registre	3	\$ _GET	3	null	3	mysql_affected_rows(\$r)		

Tâche	Tâche_1	Tâche_2	Tâche_3	Tâche_4
N° Expression

Tache6 : écrire le code php permettant d'afficher le numéro de l'élève :

.....

Partie D SQL(11.25 pts)

Exercice n°1 (6 pts)

Soit la base de données BANQUE contenant les tables suivantes :

- **AGENCE** (Num_Agence, Nom, Ville, Actif)
- **CLIENT** (Num_Client, Nom, Prenom, Ville)
- **COMPTE** (Num_Compte, Num_Agence#, Num_Client#, Solde)
- **EMPRUNT** (Num_Emprunt, Num_Agence#, Num_Client#, Montant)

1. Sans utiliser **DISTINCT**, donnez une requête équivalente en SQL :

SELECT DISTINCT Num_Client FROM COMPTE WHERE solde< 1000 OR solde>100000;

.....

2. Les clients(**Num_Client**) n'ayant pas de compte dans la même agence que **Liliane Touati**.

.....

3. Les agences (**Num_Agence**) ayant un **actif** plus élevé que toutes les agences de **BebBhar**.

.....
.....
.....
.....
.....
.....

4. Le **solde moyen** des comptes clients, pour chaque agence dont le solde moyen est supérieur à 10000.
(Tableau résultat : **Num_Agence,** **Solde_Moyen**).

.....
.....
.....
.....
.....
.....

5. Le **nombre** de clients de l'agence de nom "**Biz_Ag**" dont la **ville** n'est pas renseignée dans la relation **CLIENT**.

.....
.....
.....
.....
.....

6. Les clients (**Num_Client**) ayant un compte dont le **solde** est supérieur à la **somme totale** de tous les **actifs** des agences de **Tunis**

.....
.....
.....+.....
.....
.....
.....

Exercice n°2 (5.25 pts)

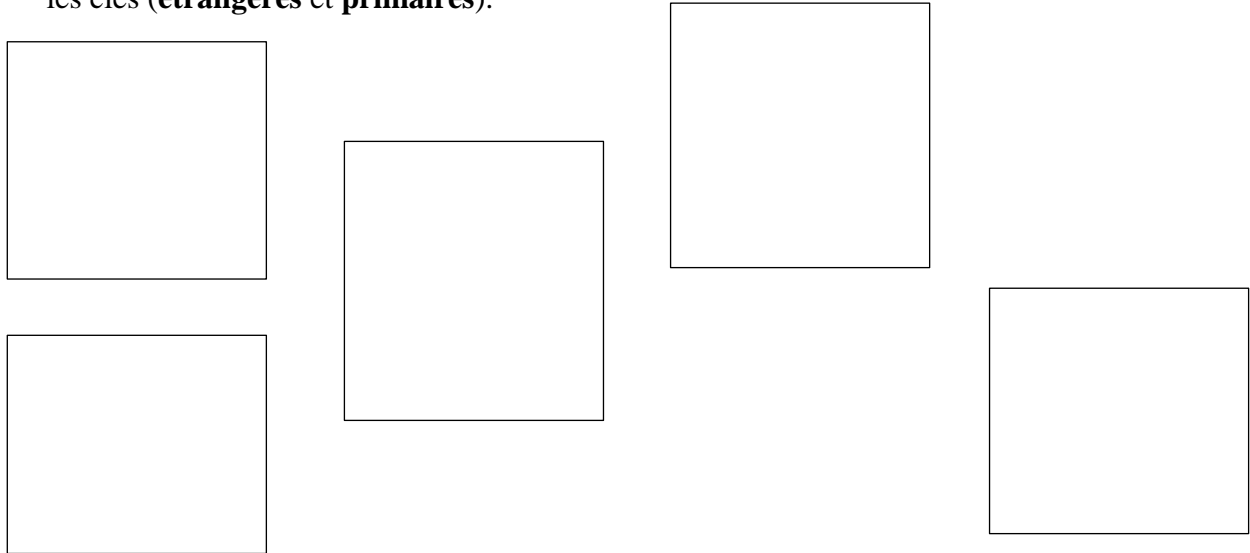
Une société veut modéliser son système de gestion des ventes, pour cela elle élabore le modèle relationnel suivant :

- **Client** (*clt_num*, clt_nom, clt_prenom, clt_pays, clt_loc, clt_type)



- **Commandes** (*cmd_num*, cmd_date, clt_num#, mag_num#)
- **Magasin**(*mag_num*, mag_loc, mag_gerant)
- **Ligne_cmd**(*cmd_num#*, *art_num#*, lcd_qte, lcd_liv, lvd_pu, lcd_date_liv)
- **Article**(*art_num*, art_nom, art_poids, art_stock, art_pa, art_pv)

1. Donner la représentation graphique de la présentation textuelle précédente en précisant **seulement** les clés (**étrangères** et **primaires**).



2. Donner le rôle de la requête SQL suivante :

```
Select art_num, AVG(lcd_pu), count(*) from Ligne_cmd group by art_num
```

.....

3. Donner par magasin les statistiques de vente en 2019, à savoir le **nombre de vente**, le **nombre de clients différents**, et le **nombre de jours** ou il y a eu des ventes.

.....

4. Donner le rôle de la requête SQL suivante :

```
select clt_num from Commandes where cmd_date between "2022-06-01" and "2022-06-30"
```

.....

.....

.....

.....

.....

.....

.....

.....

5. Classer les localités en fonction du nombre décroissant de clients qui y habitent

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....