

Révision théorique

Exercice n°1 :

Pour chacune des questions suivantes, une seule proposition est correcte. Mettre une croix (X) dans la case correspondante à la proposition correcte.

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

1. Soit l'élément h1 suivant : `<h1 id="titre" class="titre">Titre de niveau 1 </h1>`, on veut **doubler** sa taille lorsqu'il est **survolé** par le pointeur de la souris pendant **5s**. Quelle règle CSS doit-on utiliser ?

☐	<code>#titre:hover { font-style : italic ; transition : 5s;}</code>
☐	<code>#titre:hover { font-size : 20px ; transition : 5s;}</code>
☐	<code>#titre:hover { transform : scale(2 , 2) ; transition : 5s;}</code>

2. En PHP, la fonction `mysql_affected_rows`

- ☐ retourner le nombre de lignes affichées par la dernière requête SELECT
- ☐ retourner le nombre de colonnes affichés par la dernière requête SELECT
- ☐ retourner le nombre de lignes supprimées par la dernière requête DELETE

3. Dans une page web, on souhaite créer un **bouton** permettant l'appel de la fonction JavaScript `verif1()`. Quelle ligne d'instructions permet d'effectuer cette tâche ?

- ☐ `<button onclick = "verif1()">Cliquez ici</button>`
- ☐ `<a href = verif1()>Cliquez ici</a>`
- ☐ `<button>Cliquez ici</button = verif1()>`

4. Soit la règle CSS suivante : `p.bli{ color: red; }` permet de colorer en rouge

- ☐ le texte de tous les paragraphes.

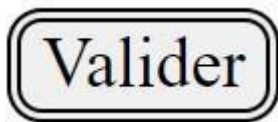
☐ le texte de tous les paragraphes ayant **bli** comme valeur de l'attribut **class**.

☐ le texte du seul paragraphe ayant **bli** comme valeur de l'attribut **id**.

5. Soit la règle CSS suivante appliquée à un élément **button** :

```
button { font-family: Cavolini 20px;  
border: dotted 2px;  
border-radius: 5px; }
```

Quel est le résultat qui correspond à la règle ci-dessus?



9. Pour supprimer une ligne d'une table, quelle commande SQL doit-on utiliser ?

drop line

delete from

drop table



10. Pour modifier le nom et le type d'une colonne d'une table, quelle commande SQL doit-on utiliser ?

Modify

Change

Rename



Exercice n°2

Compléter le tableau ci-dessous tout en précisant l'instruction et/ou le rôle et /ou le langage utilisé (HTML, JAVASCRIPT, PHP, SQL) et/ou le rôle des instructions :

Instruction	Langage	Rôle
select count(*) from client ;		
\$t=strtolower(trim(\$t)) ;		

.....		
.....	JavaScript	Vérifier si une chaîne S est convertible en un nombre
.....	PHP	Importer et inclure le fichier "Fiche.php"
<body onload="Fonc1()">		
.....	JavaScript	Déterminer la position de dernier espace dans une variable S de type chaînes de caractères.
.....	SQL	Calculer le nombre de lignes retournées par la requête SELECT
<script src="Controles.js"></script>		

Exercice n°3

Partie A

Dans le but d'améliorer ses services, une agence de voyage a développé un site web qui offre à ses clients les services suivants :

- Recherche d'un hôtel,
- Paiement des frais relatifs aux vacances.

Ci- après le code relatif à la page "Index.html" qui représente la page d'accueil du site :

```
<html>
<head>
<meta charset="utf-8">
```

```

<title>Page d'accueil</title>
</head>
<body>
<header>
<center><h2>Agence du voyage : South Travel</h2></center></header>
<nav>
<ul>
<li><a href="intro.html" target="conteneur">A propos</a></li>
<li><a href="Reserver.html"
target="conteneur">Réservation</a></li><li><a href="Payer.html"
target="conteneur">Payement</a></li>
</ul>
</nav>
<main><iframe></iframe></main>
<footer> Email : Webmaster@Site.com - Tel : 75123456 </footer>
</body></html>

```

1. Suite à une étape de validation, le validateur nous affiche l'état suivant qui comporte deux anomalies : un avertissement (**Warning**) et un erreur (**Error**)

- Warning** Consider adding a `lang` attribute to the `html` start tag to declare the language of this document.
[From line 1, column 16 to line 2, column 6](#)

```
type html>=><html>=><head
```


For further guidance, consult [Declaring the overall language of a page](#) and [Choosing language tags](#).
If the HTML checker has misidentified the language of this document, please [file an issue report](#) or [send e-mail to report the problem](#).
- Error** The `center` element is obsolete. [Use CSS instead](#).
[From line 8, column 13 to line 8, column 20](#)

```
<header><center><h2> A
```

Compléter le tableau suivant par la cause de l'anomalie et proposer une correction.

N° anomalie	Type	Cause	Solution
1	Warning		

2	Error		
---	-------	---	---

2. Lors du clic sur chacun des 3 liens, le contenu de la page Web cible ne s'affichera pas dans la fenêtre en ligne <iframe> appropriée.

Résoudre ce problème en complétant l'élément **iframe** par les valeurs convenables des deux attributs **name** et **src** ; sachant que l'élément **iframe** comporte par défaut la page "intro.html".

```
<iframe src="....."
name="....."></iframe>
```

3. Afin de mettre en forme le contenu de cette page Web, ajouter l'élément html pour attacher cette page avec le fichier "**MesStyles.CSS**" comportant les règles de mises en forme demandées.

```
<title>Page d'accueil</title>
```

```
.....
```

```
.....
```

```
</head>
```

4. Soit l'imprime écran suivant relatif à la page **index.html** :

Agence du voyage : South Travel

- [A propos](#)
- [Réservation](#)
- [Payement](#)

▼ South travel

Notre agence vous offre divers destinations et des prix acceptables

Email : Webmaster@Site.com - Tel : 75123456

Compléter le tableau suivant en indiquant, pour chaque n° de zone, le nom de la balise qui convient : **footer**, **nav**, **main** et **header**,

N° Zone	Balise
①	
②	
③	

④	
---	--

5. On se propose d'ajouter une règle **CSS** au fichier **MesStyles.CSS** afin l'éliminer les puces relatives à la liste comportant les liens hypertextes. Compléter la règle suivante en ajoutant la déclaration convenable.

<pre>ul {.....:..... ; }</pre>

- 6.Ecrire la règle **CSS** qui permet d'appliquer, à chaque ligne de liste : une bordure bleu, en trait continu, d'épaisseur 3pixels et des coins arrondis de 10%.

```
li {border : .....;
..... :10%;}
```

- 7.Pour l'entête de la page, écrire la règle **CSS** qui permet de changer, lors de son survol avec le pointeur de la souris, sa couleur d'arrière en "Orange"

<pre>header:.....{.....:.....;}</pre>

Partie B (4 points)

Soit la base de données simplifiée intitulée **Gest_Cartes** relative à la gestion des cartes bancaires.

Cette base est décrite par la représentation textuelle suivante :

proprietaire (cin, nomprenom, genre, tel)
carteBancaire(numcarte, cin#, dateExp, cvc)

Les champs des tables sont décrits dans le tableau ci-dessous :

champ	Description et type
Cin	Numéro de la CIN d'un propriétaire, de type chaine de 8 chiffres
nomprenom	Nom et prénom et d'un propriétaire, de type chaine de 50 caractère au maximum
Genre	Gentre d'un propriétaire, de type caractère("M" : Masculin ou "F" : Féminin)
Tél	Numéro du téléphone d'un propriétaire, de type chaine de 8 chiffres
numcarte	Numéro d'une carte bancaire, de type chaine de 10 chiffres
Dateexp	Date d'expiration d'une carte bancaire
Cvc	Le code de sécurité d'une carte, de type entier

1. Ecrire en SQL les requêtes permettant de répondre aux questions suivantes : a)Créer la table proprietaire :

b) Ajouter la contrainte d'unicité (unique**) sur la colonne nomprenom.**

c) Appliquer la contrainte d'intégrité référentielle entre les deux tables

[illegible]

Enregistrement d'une carte bancaire

N° de la carte :

N° CIN :

Date d'Expiration :

Code CVC :

a) Indiquer le type de la zone de saisie **a)** relative à la date d'expiration de la carte.

`<input type="....." Id="datExp"
name="datExp">`

b) Indiquer le type de la zone de saisie **b)** relative au Code CVC ainsi qu'au nom de l'attribut permettant de définir le texte apparent "xxx"

`<input type="....." Id="cvc"
name="cvc".....="xxx">`

c) Le clic sur le bouton **Enregistrer** fait appel à la fonction JavaScript intitulée **Verif()** qui permet de vérifier le remplissage des champs du formulaire en respectant les contraintes suivantes :

Champ	Contrainte
N° de la carte	Une chaine de 10 chiffres et comportant exactement une seule occurrence du chiffre "5"
N° CIN	Une chaine de 8 chiffres dont le premier est "0" ou "1"
Date expiration	Une date non nulle (Sélection obligatoire d'une date)
Code CVC	Une entiere de 3 chiffres

d) Soit le contenu troué de la fonction **Verif()**.

```
function Verif()
{
    var a=document.getElementById("num").value;
var b=document.getElementById("cin").value;    var
c=document.getElementById("datExp").value;    var
d=document.getElementById("cvc").value;
test=true;
    if (..... || a.lastIndexOf("5")>
a.indexOf("5") || a.length!=10)
        {alert("Num carte ?"); test=false;}
    else if (b.charAt(0)<"0" || ..... || b.length!=8)
        {alert("Cin ?"); test=false;}
    else if (.....)
        {alert("Date ?"); test=false;}
    else if (.....)
        {alert("cvc"); test=false;}
    else .....;
}
```

Compléter les pointillés par la conditions convenables afin que la fonction Verif() réussisse à vérifier les conditions présentées par l'énoncé.

Condition
<code>c.length==0</code>
<code>d<100 d>999</code>
<code>a.indexOf("5")<0</code>
<code>b.charAt(0)>"1"</code>
<code>return test</code>

e) Une fois les conditions sur le remplissage du formulaire sont vérifiées, le clic sur le bouton **Enregistrer** fait appel au fichier "**EnregistrerCarte.php**" qui permet de :

- Afficher le message "*Vérifier la date d'expiration de la carte*", si la valeur saisie dans la zone
Date d'expiration ne dépasse pas la date système
Ou bien
- Afficher le message "*Propriétaire non inscrit*", si la valeur saisie dans la zone cin n'existe pas dans la table Propriétaire Ou bien
- Afficher le message "*Carte déjà exploitée*" si le numéro de la carte existe dans la table carte Ou bien
- Insérer les données nécessaire à la table Carte et afficher le message "*Ajout fait avec succès*»

Q1 : Compléter l'élément **form** du fichier "**EnregistrerCarte.html**" par les informations nécessaires :

`<form action="....."="POST" onsubmit=".....
.....">`

Q2 : Compléter le tableau suivant par les instructions nécessaires selon la consigne indiquée.

N°	Consignes	Instructions
1.	Récupération des données envoyées par le formulaire.	<code>\$num= S-POST["num"] ;</code> <code>\$cin =</code> <code>\$_POST["....."];</code> <code>\$dexp =</code> <code>..... ;</code> <code>\$cvc =</code> <code>..... ;</code>

2.	Récupération de la date système	\$dact = ;
3.	Appel du fichier connexion.php	("connexion.php") ;
4.	Afficher le message "Vérifier la date d'expiration de la carte", si la date système dépasse la date d'expiration de la carte.	if (..... >) echo("vérifier la date d'expiration de la carte") ;
5.	Insérer les données nécessaire à la table Carte et afficher le message "Ajout fait avec succès"	mysql_query("..... into values (.....,,,)") and ("ajout fait avec succès"):

Exercice n°4 (07 points)

Un **Sniffer** est un programme qui permet de capturer toutes les informations envoyées sur un réseau et de récupérer les données de ses utilisateurs.

Une société se propose de gérer les flux de données transférées par ses employés via son réseau en se basant sur la technique du **Sniffer**.

Pour ce faire, l'administrateur du réseau utilise la base de données "**Gestion_Transfert**" dont la représentation textuelle est la suivante :

service (numservice, nomservice) **employe** (matricule,
nomprenom, taillemax, numservice#)

transfert (numtransfert, dateheuretransfert, duree, tailedonnees, matricule#)

Description des champs des tables

Champ	Description et type
numservice	Numéro d'un service, de type entier.
nomservice	Nom d'un service, de type chaine de caractères.
matricule	Matricule d'un employé, de ce type chaine de caractères.
nomprenom	Nom et prénom d'un employé, de type chaine de caractères.
taillemax	Taille maximale des données à transférer en Giga bits, de type entier.
numtransfert	Numéro d'un transfert, de type entier.
dateheuretransfert	Date et heure d'un transfert, de type date et heure.
duree	Durée d'un transfert en secondes, de type entier.
tailedonnees	Taille de données transféréed en Giga octets, de type entier.

1) Ecrire en SQL les requêtes permettant de répondre aux questions suivantes :

a. Modifier la taille du champ nomservice à 30 caractères.

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

b. Afficher les transferts dont la durée dépasse 100.

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

c. Effacer les transferts dont la durée est égale à 0.

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

d. Afficher le nombre d'employés par services

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

e. Afficher le nombre des transferts effectués à la date du jour courant.

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

2) Valider chacune des propositions suivantes en mettant dans la case correspondante la lettre **V** si elle est correcte ou la lettre **F** dans le cas contraire. ☐ Un employé peut appartenir à plusieurs services.

☐ Dans un service, on peut trouver un ou plusieurs employé(s).

☐ Un employé peut faire plusieurs transferts de données.

☐ Un transfert de données peut être effectué par plusieurs employés.

3) Lors de l'insertion des données, l'administrateur de la base de données a détecté des anomalies. En effet, le champ "**duree**" accepte des valeurs **négatives** ou **nulles**.

a. Donner le nom de la contrainte d'intégrité non respectée.

.....
.....

b. Écrire la requête **SQL** permettant de prendre en compte cette contrainte.

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

4) Lors de l'exécution d'une requête SQL permettant la suppression d'un employé ayant précédemment transféré des données, un message d'erreur de violation de la contrainte d'intégrité référentielle s'affiche lui interdisant cette suppression.

Écrire la requête SQL permettant de modifier la structure de la base de données afin de permettre à l'administrateur la suppression des employés.

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

5) Afin de différencier les différents employés par leurs fonctions, on se propose de créer une nouvelle table nommée "**fonction**" ayant la description suivante :

Champ	Type	Contrainte
Codefonct	Numérique	Clé primaire
libellefonct	Chaîne de 20 caractères	Valeur obligatoire

a. Écrire la requête **SQL** permettant de créer cette table.

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

.....
.....
b. On veut appliquer l'intégrité référentielle entre les tables "**employe**" et "**fonction**" en respectant les deux règles de gestion suivantes :

R1 : Un employé possède une seule fonction.

R2 : Plusieurs employés peuvent avoir la même fonction.

Écrire la requête **SQL** permettant d'assurer cette règle d'intégrité.

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

..... c)Le champ "**taillemax**" de la table "**employe**" ne dépend pas de l'employé mais dépend de sa fonction. Écrire les requêtes **SQL** permettant de prendre en charge cette règle de gestion.

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