ECOLE MAROCAINE DES SCIENCES DE L'INGENIEUR
Membre de HONORIS UNITED UNIVERSITIES

Examen - Année universitaire 2022-2023

Nom : ……Prénom: ……Classe:.....

Matière :	**Langages de scripts**	Filière : IIR Niveau : 3	
Date de l'épreuve :	07/02/2023 11:00	Durée de l'épreuve : 2H	

— La clarté de la feuille et du code est prise en considération.
— **Documents NON autorisés**

1e partie : Questions de compréhension (10 points)

Cochez la/les bonne(s) réponse(s) : *(4pt)*

1. Comment peut-on ajouter un élément au début d'un tableau en javascript:
   - ☐ push
   - ☐ shift
   - ☐ pop
   - ☐ slice
   - ☑ aucune de ces fonctions

2. Comment peut-on accéder à un élément HTML en utilisant JavaScript?
   - ☑ En utilisant la méthode getElementById()
   - ☑ En utilisant la méthode querySelector()
   - ☑ En utilisant la méthode getElementsByClassName()
   - ☐ Aucune des réponses n'est correcte.

3. Dans quelle situation le code ci-dessous affiche le message suivant « Ceci est un piège »

```
<?php session_start();
$_SESSION = array();
$_SESSION["lespieges"] = array();
if(! isset($_SESSION["piege"]))
echo "Ceci est un piège"; ?>
```

   - ☐ S'il existe déjà une session, et qu'elle contient une variable "piege".
   - ☐ S'il existe déjà une session, et qu'elle ne contient pas de variable "piege".
   - ☐ Jamais
   - ☑ Tout le temps

4. Parmi les variables suivantes, entourez celles qui ont un nom valide en PHP :

`mavar`, `$mavar`, `$var5`, `$_mavar`, `$_5var`, `$__élément1`, `$hotel4*`

5. Comment peut-on déclarer un tableau en JavaScript?
   - ☑ var monTableau = new Array()
   - ☐ var monTableau = array()
   - ☐ var monTableau = {}
   - ☑ var monTableau = []

6. Comment peut-on ajouter un élément à la fin d'un tableau en PHP ?
   - ☑ array_push()
   - ☐ array_add()
   - ☐ array_append()
   - ☐ array_insert()

7. Quelle structure permettra de parcourir tout le tableau array en JavaScript ?
   - ☐ for(let i = 0; i < array.size; i++) {...}
   - ☐ for(let i = 1 ; i < array.length; i++) {...}
   - ☑ for(let i = array.length - 1; i >= 0; i--) {...}
   - ☐ for(let i = 0; i <= array.length; i++) {...}

8. Quelle sera la sortie du code PHP suivant?

```
<?php
$total = "5 personne";
$autre = 5;
$total = $autre + $total;
echo "$total";
?>
```

   - ☐ Erreur
   - ☐ 10 personne
   - ☑ 10
   - ☐ 55 personne

9. Quelle est la différence entre == et === en JavaScript ? *(0,5pt)*

== compare juste les valeurs alors que === compare aussi les types

.....
.....

10. Quelle est la différence entre **`getElementById(.....)`** et **`querySelector(.....)`** en JavaScript? *(0,5pt)*

getElementById permet de chercher uniquement par ID, alors que querySelector utiliser la syntaxe des sélecteurs CSS qui utilisent plusieurs informations pour la recherche

.....

11. Quelle est la principale différence entre les sessions et les cookies? *(0,5pt)*

les sessions stockent les données du côté serveur alors que les cookies stockent les données du côté client (navigateur)

.....
.....

12. On souhaite écrire le code équivalent ci-dessous en javascript discret. Entourer les codes erronés *(1,5pt)*

```
<body onload="start()">
<div id="test"
onclick="tester()"></div>
<div id="essai"
onclick="essai()"></div>
</body>
```

```
window.onload=function(){
document.getElementById("test").onclick=tester();
document.querySelector(".test").onclick=tester;
document.querySelector("#essai").onclick=essai;
let divs=document.querySelectorAll("div");
divs[0].onclick=tester;
divs[1].onclick=essai();
function tester(){
alert("vous avez cliqué sur tester");
}
function essai(){
console.log("vous avez cliqué sur essai");
}}
```

13. Soit le fichier "exam.js" ci-dessous, lié à un fichier exam.php contenant un DIV avec id="nbessaie", un label avec id=indice", un bouton qui fait appel à la fonction deviner() au clic, et un input de type "number" avec id="propo".

```
var num = Math.floor((Math.random * 90) + 1)+10;
var nbEssais = 0;
function deviner() {
 if (nbEssais == 10){
   alert("Désolé, c'est fini. Le nombre correct était : " + num);
   return;
 }
 document.getElementbyid('nbessaie').innerHTML =10-++nbEssais;
 var x = document.getElementById('indice');
 var propo = parseInt(document.getElementById("propo").value);
 if (propo < num){
   x.color="red";
   x.innerHTML="Non, le nombre est plus grand";
 }else if (propo > num){
   x.color="green";
   x.innerHTML ="Non, le nombre est plus petit";
 }else  if (propo == num)
   alert("Correct ! Vous avez trouvé en " + nbEssais + " essais.");}
```

a. Soulignez deux erreurs différentes commises dans le code et corrigez-les: *(1pt)*

Math.random().....

document.getElementById('nbessaie').....

x.style.color .....

Nous supposons, dans la suite, que les erreurs ont été corrigées et que le script est opérationnel.

b. Quelles sont les valeurs MIN et MAX de la variable "num" ? *(1pt)*

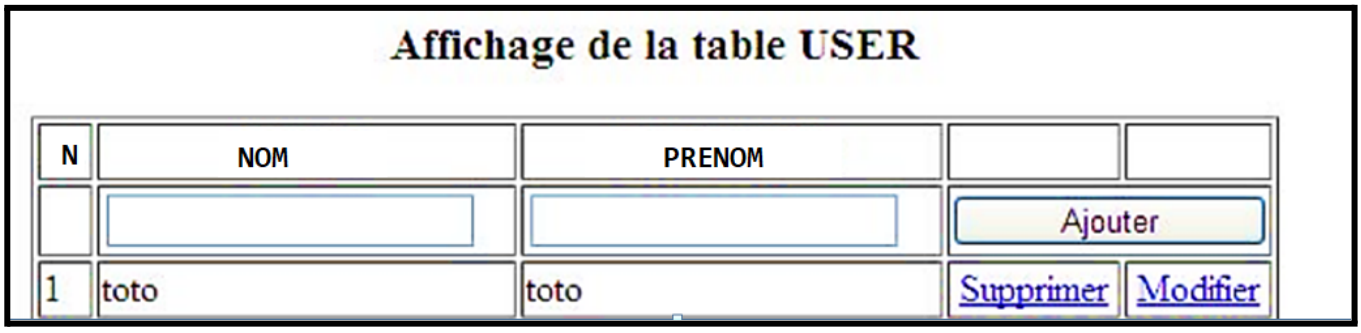
..... *11* ≤ num ≤ *100*.....

c. Quelle fonction permettrait de vérifier si la variable "propo" est numérique? *(1pt)*

isNaN(propo).....

2e partie : Production de scripts (10 points)

A. Nous souhaitons mettre en œuvre la gestion des utilisateurs d'une application web (voir l'apparence de la page "index.php" dans la figure ci-dessous).



1. Le bouton "Ajouter" ainsi que les deux `input` sont inclus dans un formulaire dont voici la balise d'ouverture :

```
<form method="POST" action="ajouter.php" onsubmit="return verifierform()">
```

Sachant que le premier input a un `id="nom"` et le deuxième un `id="prenom"`. Écrire la fonction JavaScript **`verifierform()`** qui bloquera l'envoi du formulaire et affiche une alerte d'erreur dans les cas suivants : *(2pt)*

- si le nom est égale au prénom
- si le nom ou le prénom saisi est composé uniquement de chiffres.

```
function verifierform(){
   let nom = document.getElementById("nom").value;
   let prenom = document.getElementById("prenom").value;
   return (nom!=prenom && isNaN(nom) && isNaN(prenom));
}
```

.....

2. Écrire le code PHP qui permettra la connexion à la base de données "*monappli" sur la machine locale, en utilisant le login "utilisateur" et le mot de passe "1234":* *(1pt)*

```php
try{
  $db = new PDO("mysql:host=localhost;dbname=monappli", "utilisateur", "1234");
  $db->setAttribute(PDO::ATTR_ERRMODE,PDO::ERRMODE_EXCEPTION);
} catch(PDOException $e){ die( "Erreur: ".$e->getMessage());}
```

3. Écrire le code PHP du script "ajouter.php" qui ajoute les valeurs envoyées dans la table USER puis redirige vers la page "index.php" *(2pt)*

```php
<?php
require_once("db.inc.php");
$db->exec("INSERT INTO USER (nom,prenom) VALUES ('{$_POST['nom']}','{$_POST['prenom']}')");
header('Location: index.php');
?>
```

.....
.....

4. Le lien « supprimer » envoie vers un script `delete.php?id=<?=$id?>` qui supprime l'enregistrement correspondant à la ligne et redirige vers la page « index.php ». Donnez le code source de ce script. *(2pt)*

```php
<?php
require_once("db.inc.php");
$db->exec("DELETE FROM USER WHERE id='{$_GET['id']}'");
header('Location: index.php');
?>
```

.....
.....

B. Soit le tableau suivant:

```
var semaine = [ 'lun', 'mar', 'mer', 'jeu', 'ven', 'sam', 'ddi' ];
```

Ecrire le code javascript qui permet de :

a. Retirer la dernière valeur du tableau semaine *(1pt)*

```
semaine.pop();
```

b. Ajouter la valeur 'dim' à la fin du tableau *(1pt)*

```
semaine.push('dim');
```

c. Trier le tableau semaine par ordre alphabétique croissant *(1pt)*

```
semaine.sort();
```