**PHP**

**1.Quel est la différence entre un langage de programmation et un langage de description ? Citez un exemple pour chacun d'entre d'eux.**

**2. PHP s'exécute côté serveur, qu'est-ce que cela veut dire ? Décrivez comment se déroule une communication entre un navigateur et ces éléments côté serveur : PHP, Apache, MySQL.**

**3. Citez un langage de programmation qui s'exécute coté client.**

JavaScript est un langage de programmation coté client

**4. Citez quelques types primitifs de variable. Est-il possible en PHP de comparer des variables de type primitif différent ? Si oui, que se passe-t-il si par exemple on compare une chaine avec un entier ?**

Voici quelques types de variable primitif : Boolean, Int , Float

Si on compare une chaine avec un entier ça retourne False car leur type est diiférent

**5. Comment peut-on parcourir en PHP une variable de type tableau (array) ?**

**6. Quels sont les opérateurs PHP de comparaison stricte ? Ecrivez une ligne de code qui compare strictement deux variables PHP.**

**7. Ecrivez un script PHP : - Codez une fonction multiplication() avec deux paramètres, qui réalise la multiplication de ces paramètres puis renvoie le résultat - Utilisez cette fonction dans le script puis affichez l'opération elle-même ainsi que son résultat.**

*function* Multiplication(*$number1*, *$number2*){

$total = (*$number1*\**$number2*);

*return* 'le produit de '.*$number1*.' et de '.*$number2*.' est : '.$total;  
}

*echo* Multiplication('5','6');

**Dans le Navigateur :**

le produit de 5 et de 6 est : 30

**PHP Base de données**

**1. A quoi sert MySQL ?**

MySQL est un gestionnaire de base de données , c’est-à-dire qu’il sert à stocker des données pour pouvoir ensuite les utiliser comme on le souhaite , les trier , et les appeler où nous voulons

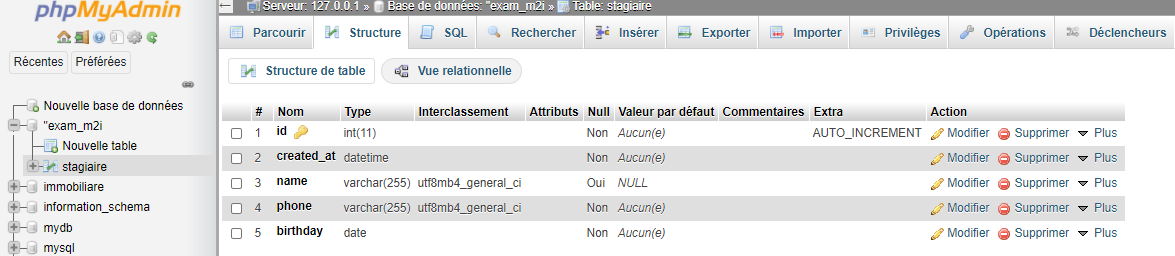
**2. Quelle est la différence entre une base de données, des cookies et des variables de session ? Précisez à quel moment chacune de ces données sont supprimées.**

La **base de donnée** sert à afficher les données sur le site grâce a des requêtes , ces données peuvent être supprimé également par des requêtes

Les **cookies** sont stockées coté client , on peut les supprimés en suppriment le cache du navigateur

Les **variables de session** sont stocker coté serveur , les données sont supprimées une fois que la session est fermée

**3. Concevez une base de données "exam\_m2i" via PhpMyAdmin en créant une table stagiaire, qui contient 5 colonnes en faisant le meilleur choix pour le type : id, created\_at, name, phone, birthday.**

**Php bin**

**4. Ecrivez un script PHP qui se connecte à cette base de données "esv\_m2i", grâce à PDO.**

try{

    $db = new PDO('mysql:host=localhost;port=3306;dbname="exam\_m2i','root','',[

        PDO::ATTR\_ERRMODE => PDO::ERRMODE\_EXCEPTION, *//On active les erreurs SQL*

    ]);

    }catch(Exception $exception){

        echo $exception->getMessage();  *//Affiche le message de l'erreur quand ça detecte pas la base*

        exit(); *// Arrete le script PHP*

    }

**5. Ecrivez le script PHP, qui une fois connecté à cette base, récupère tous les enregistrements de la table stagiaire et les affiche dans la page. Affichez l'ensemble des informations de chaque stagiaire séparées par des tirets, avec un retour à la ligne entre chaque stagiaire. Pour tester ce script, vous pouvez ajouter manuellement des enregistrements grâce à PhpMyAdmin.**

function getAllStagiaires(){

*global* $db;

    $query = $db->query(('SELECT \* FROM `stagiaire`'));

    return $query->fetchAll();

}

<?php

    require 'routedb.php';

?>

<!DOCTYPE *html*>

<html *lang*="en">

<head>

    <meta *charset*="UTF-8">

    <meta *name*="viewport" *content*="width=device-width, initial-scale=1.0">

    <title>Document</title>

</head>

<body>

<?php

foreach (getAllStagiaires() as $Onestagiaire){

?>

<p>

    <?php

        echo $Onestagiaire['id'].'-'.$Onestagiaire['name'].

'-'.$Onestagiaire['created\_at'].'-'.$Onestagiaire['phone'].

'-'.$Onestagiaire['birthday'];

    ?>

</p>

<?php

}

?>

</body>

</html>

**PHP Objet**

**1. Quelles sont les différences entre le langage orienté objet et le style procédural ? Donnez les avantages et les inconvénients de chacun d'entre d'eux.**

**2. Comment déclare-t-on et instancie-t-on un objet en PHP ?**

**3. A quoi sert un constructeur ?**

Le constructeur sert a instancier un objet php

**4. Quelles sont les rôles des getters / setters ?**

On utilise le get pour aller chercher la valeur d’une variable

On utilise le set pour changer la valeur d’une variable

**5. Quelles sont les différences entre des propriétés/méthodes privées et publiques ?**

La propriété public fait en sorte que la variable soit accessible partout

La propriété private fait en sorte que la variable soit accessible seulement à l’intérieur de la classe

**6. Qu'est-ce que l'héritage ? Quel est le mot-clé utilisé en PHP pour l'héritage ?**

L’héritage sert a étendre une classe sur une autre , l’enfant hérite des méthodes protected et public du parent le mot clé utilisé en PHP pour l’héritage est extends

**7. Qu'est-ce qu'une interface ? Avec quel mot-clé une classe PHP peut-elle implémenter une interface ?**

Une interface est une classe où on déclare des méthodes

Une classe qui implémente une interface est obligé d’implémenter chaque méthodes de l’interface

Le mot clé est ***implements***

**8. Qu'est-ce qu'une classe abstraite ?**

Une classe abstraite est une classe qui n’a pas toutes les méthodes implémentées

**9. Qu'est-ce qu'une méthode magique en PHP ? Donnez quelques exemples et expliquez leur utilité.**

**10. Quel opérateur en PHP permet de savoir si une variable objet est bien une instance d'une classe en particulier ?**

**11. Ecrivez une classe PHP qui modélise un stagiaire : id, created\_at, name, phone, birthday. Cette classe doit contenir les propriétés, le constructeur et les getters/setters correspondants.**

**12. Utilisez cette classe dans un script PHP, en l'important grâce à un autoload PHP. Instancier un stagiaire et donner une valeur à chacune de ses propriétés.**

**PHP Symfony (Le projet sans le dossier vendor) Les étapes 6 à 13 peuvent être réalisées dans votre TP existant ou dans un nouveau projet Symfony si vous préférez. L’objectif est de voir si vous êtes capable de refaire une entité, un formulaire, une page et une relation entre des entités.**

1. **Qu'est-ce qu'un Framework ? Donnez différents avantages et inconvénients.**

Un framework (cadre de travail ) est un outil conçu pour aider les développeurs dans leur travail,

L’avantage est qu’un framework fourni comme son nom l’indique un cadre de travail , une standardisation de la façon de développer , de ce fait si un développeur utilise bien symfony par exemple il n’aura pas de mal a se faire relire par un autre utilisateur de symfony qui saura où trouver les fichiers qu’il l’intéresse ,

Un framework permet aussi de coder beaucoup plus rapidement car il permet de faire énormément de chose avec beaucoup moins de manipulation a condition de savoir l’utiliser

L’inconvénient qu’il peut y avoir c’est qu’au détriment de la vitesse de programmation et le rangement des dossiers , c’est que l’apprentissage de certains framework nécessite beaucoup de temps , et que l’on soit bridé par le framework pour certaine choses

En résumé au détriment d’un peu de liberté pour coder , le framework , si on le maitrise permet de coder plus facilement et plus rapidement .

**2. Qu'est qu'un ORM et à quoi cela sert-il ? Quel est l'ORM par défaut utilisé avec symfony ?**

Un ORM est un programme qui permet de communiquer avec une base de données

L’ORM de Symfony est Doctrine

**3. A quoi sert Composer ?**

Composer permet d’accéder à une large librairie de service que l’on peut appeler si nous en avons besoin dans notre projet

**4. Décrivez le fonctionnement de Symfony au niveau d’une route.**

Gérer les routes avec symfony est très simple , exemple si nous faisons php bin/console make:controller suivant du nom que l’on souhaite pour le controller

Nous allons générer un controller et une page twig

Ces 2 éléments sont liés grâce au controller où l’on peut trouver la route pour accéder a cette page

Par exemple si dans le controller nous trouvons :

*/\*\*  
 \** ***@Route****("/homepage", name="home")  
 \*/*

Et que nous sommes dans une autre page twig que celle de l’accueil nous pouvons appeler cette page grâce au name de la route du controller comme ceci

<li><a *class*="dropdown-item" *href*="{{ path ('home')}} ">Admin</a></li>

Ainsi dans l’url sera afficher /homepage

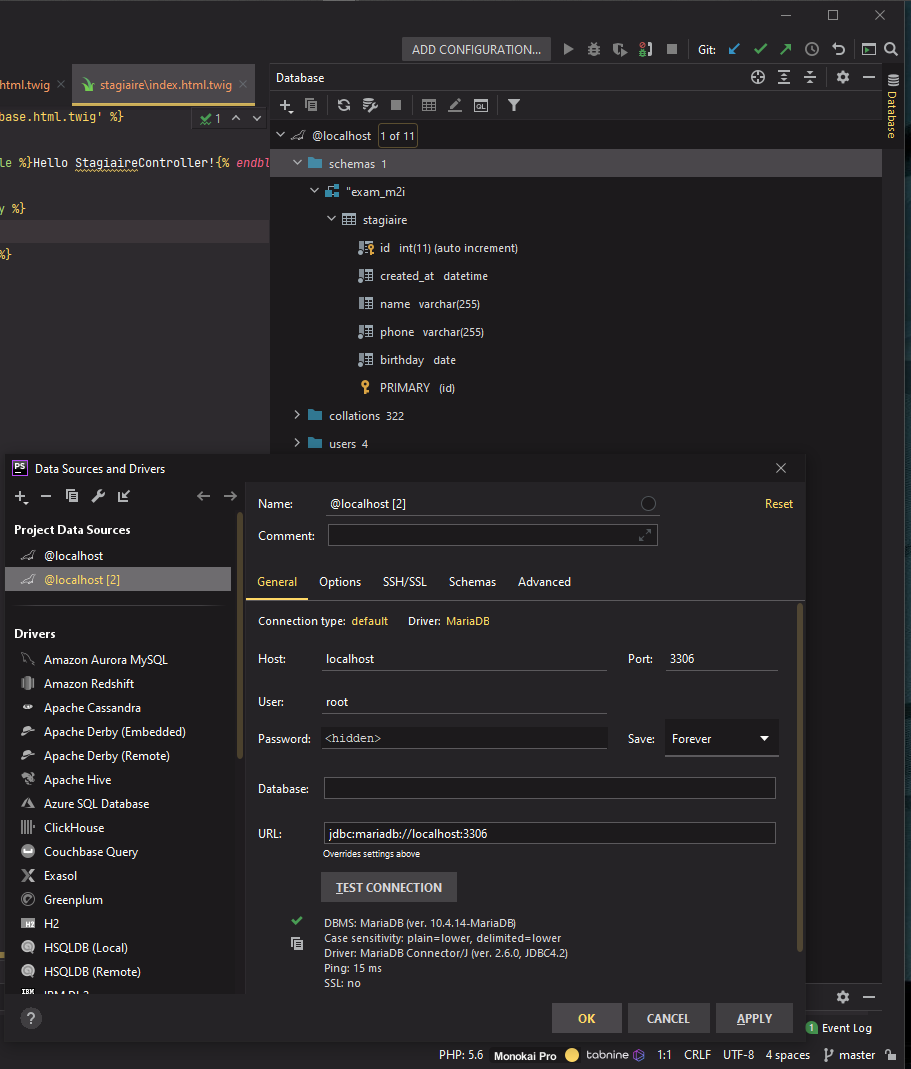
**5. Qu’est-ce qu’un service ? A quoi cela sert-il ?**

Un service est une classe avec une fonction bien précise , on peu créer nos propres services de façon à les réutiliser dans un autre projet ou même l’appeler a divers endroit du meme projet

, exemple nous créons un service pour un calcul de TVA , si nous voulons l’utiliser dans plusieurs pages/controller , il suffit d’appeler ce service via la controller dédié à la page que nous voulons

**6. Utiliser le TP existant ou créer un projet Symfony**

DATABASE\_URL="mysql://root:@127.0.0.1:3306/"exam\_m2i?serverVersion=mariadb-10.4.14"

****

**7. Modélisez à nouveau la classe stagiaire des exercices précédents, cette fois-ci en passant par la ligne de commande.**

**8. Modélisez une autre classe "Competence", et la lier à la classe stagiaire en tenant compte de ceci : un stagiaire peut avoir plusieurs compétences, et une compétence peut être liée à plusieurs stagiaires.**

**9. Créez le formulaire qui permet d'ajouter dans la base une compétence.**

**10. Créez également un formulaire qui permet d'ajouter dans la base un stagiaire. Ce formulaire doit proposer de lier des compétences à ce stagiaire.**

**11. Créez les pages pour voir ces formulaires ainsi que la liste de ces éléments dans la BDD.**

**12. Gérez les validations grâce aux annotations (Assert) en forçant tous les champs à être obligatoires, le code postal à 5 chiffres et le téléphone à 10 chiffres.**

**13. Sécurisez ces pages de formulaire pour un ROLE\_ADMIN. Vous devrez implémenter la partie sécurité avec un formulaire de login.**