

Questions 1 - Analyse de l'affichage de la liste des restaurants

1.1. A l'aide de la documentation officielle PHP, rappeler le rôle des mots clés suivants :

include_once, include, require, require_once

Importer une seul fois un fichier php (par exemple lors d'une itération), si le fichier contient une erreur le script ne sera pas arrêté.

Importer un fichier. Si le fichier contient une erreur le script ne sera pas arrêté.

Importer un fichier. Si le fichier contient une erreur le script **sera arrêté**.

Importer une seul fois un fichier php (par exemple lors d'une itération), si le fichier contient une erreur le script sera arrêté.

La bonne pratique de code sera donc l'utilisation de ***require***

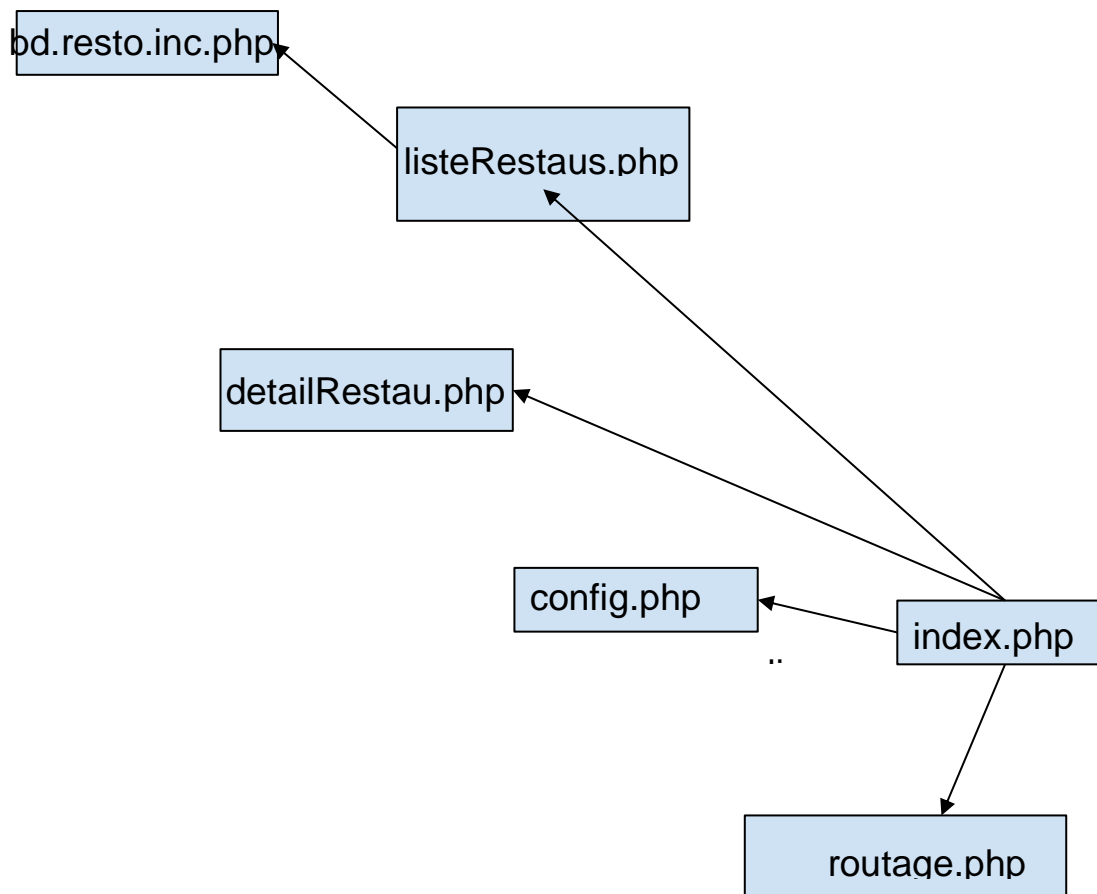
1.2. Déterminer la valeur de la variable \$fichier en affichant son contenu à la ligne précédant la balise de fin de PHP dans index.php.

```
echo print_r($fichier);
```

\$fichier => listeRestaus.php;

1.3. En partant du fichier `index.php`, schématiser l'ensemble des fichiers utilisés (`include` ou `include_once`) pour afficher la liste des restaurants à l'aide de :

- rectangles portant le nom de chaque fichier,
- flèches pointant vers le fichier utilisé (inclus). Sur la flèche, indiquez la ligne qui pointe vers le fichier.



1.4. Relever l'ensemble des requêtes SQL contenues dans les fonctions déclarées dans le fichier bd.resto.inc.php.

| Fonction | Requête |
|--|---|
| <code>function getRestoByIdR (\$idR)</code> | <code>SELECT * FROM `resto` WHERE `nom` like :nomR</code> |
| <code>function getRestos ()</code> | <code>SELECT * FROM `resto`</code> |
| <code>function getRestosByNom R (\$nomR)</code> | <code>SELECT * FROM `resto` WHERE `nomR` like :nomR</code> |
| <code>function getRestosByAdr esse (\$voieAdrR , \$cpR, \$villeR)</code> | <code>SELECT * FROM `resto` WHERE `voieAdrR` AND `cpR` LIKE :cpR AND `villeR` LIKE :villeR</code> |
| | |

1.5. Quels sont les points communs de ces requêtes SQL ?

Elles font toute une requête SELECT pour récupérer des informations précises.

1.6. En observant le nom des fonctions, le résultat affiché, et la requête contenue dans la fonction, expliquer d'une manière très générale ce que fait chacune des fonctions présente dans le script bd.resto.inc.php.

Cette fonction permet donc de récupérer les informations d'un restaurant à partir de l'identifiant passé en paramètre.

getResto() ⇒ Cette fonction permet de récupérer toutes les informations de la base de données resto, c'est-à-dire tous les restaurants et le détail de chacun.

getRestoByNomR() ⇒ Cette fonction permet de récupérer les restaurants par leur nom.

getRestoByAdress() ⇒ Cette fonction permet de récupérer les restaurants par adresse.

1.7. Quelle fonction définie dans le script bd.resto.inc.php est utilisée dans listeRestos.php ?

-getRestos()

Questions 2 - Début d'analyse de la structure du site

2.1. Trouve-t-on des éléments de CSS ou de HTML dans les fichiers des dossiers controleur et modele ?

Non

2.2. Dans quel dossier sont contenus les fichiers gérant les éléments de HTML et de CSS ?

Dans le dossier : vue.

2.3. Consulter le code source de la page lors d'une visite de l'URL "http://server/MVCTP1/index.php?action=liste". Retrouve-t-on le même contenu que celui des scripts du dossier vue ?

Non.

2.4. Le code PHP contenu dans vueListeRestos.php est-il toujours visible après réception par le navigateur ? Par quoi est-il remplacé ?

Une partie du code reste la même (html) toutes les variables sont remplacées par leur valeur et les fonctions sont exécutées.

2.4. Rappeler le point commun entre toutes les fonctions définies dans le script bd.resto.inc.php contenu dans le dossier modèle.

Le point commun est leur action, GET, de récupérer l'information.

2.5. Trouve-t-on dans d'autres fichiers que ceux du dossier modèle des références à la base de données ?

Si le projet est bien organisé, non.

Synthèse : expliquer globalement le rôle des scripts contenus dans les dossiers suivants

| |
|---|
| modèle: Les fonctions de requêtage et de connexion à la base de données |
| vue: Les outils permettant d'afficher le contenu des requêtes |

Questions 3 – Début d'analyse de la partie contrôleur

3.1. Quelle fonction d'accès aux données est utilisée dans ce contrôleur ? Que permet-elle de récupérer dans la base de données ?

getRestos() ⇒ Récupérer une liste des restaurants et leur détails

3.2. Les données récupérées sont-elles affichées à l'écran ? L'affichage est-il fait dans le script contrôleur ?

Un tableau contenant titre du restau et le détail
Oui il se fait à travers le contrôleur

3.3. Quels scripts sont inclus dans les dernières lignes du contrôleur ? Quels sont leurs rôles ?

```
include RACINE . "/vue/entete.html.php";  
include RACINE . "/vue/vueListeRestos.php";  
include RACINE . "/vue/pied.html.php";
```

Le rôle de ces scripts est de construire la page html.

3.4. Quelle donnée est transmise au contrôleur en méthode GET ?

La valeur de \$_GET dans ce cas correspond à la destination du chemin

3.5. Rappeler le rôle de la fonction getRestoByIdR(). Pourquoi cette méthode a-t-elle besoin d'un paramètre ?

Elle permet de récupérer le restaurant via le nom, elle a besoin d'un paramètre car pour aller chercher le restaurant il faut transmettre l'ID du restaurant.

La variable \$unResto est créée lors de l'appel à la fonction getRestoByIdR().

3.6. Explorer les fichiers "vue" inclus à la fin du fichier contrôleur et repérer les lignes où sont utilisées cette variable

LIGNE 33

LIGNE 34

LIGNE 35

3.7. Deux autres variables créées dans le contrôleur sont affichées dans la vue. Lesquelles ?

\$menuBurger ⇒ entete.html.php

\$titre ⇒ head.html.php

3.8. Le contrôleur gère-t-il directement l'accès aux données ?

Non le contrôleur ne gère pas directement l'accès au donné.

Synthèse contrôleur

Où sont affichées les données créées ou récupérées dans le contrôleur ?

Elles sont affichées dans la vue.

Comment ces données sont transmises au contrôleur :

- de l'utilisateur vers le contrôleur ?

Les données sont transmises de l'utilisateur au contrôleur via des requêtes HTTP, qui peuvent contenir des paramètres d'URL, des données dans le corps de la requête, ou des en-têtes HTTP.

- de la base de données vers le contrôleur ?

Les données sont transmises de la base de données au contrôleur via le modèle.