

EXAMEN

TOUS SUPPORTS AUTORISÉS, COMMUNICATION INTERDITE.

BARÈME SUJET 1 (20 POINTS + 2 BONUS)

Partie théorique (10 point)

- 1) Ecrire une interface qui obligera une classe à implémenter la fonction login prenant en paramètre un user et un password (0.5 point)
- 2) Citez 6 méthodes magiques. Expliquez à quelle moment elles sont déclenchées et les paramètres qu'elles prennent. (1.5 points)
- 3) Je suis une méthode magique appelé implicitement à la fin d'un script. Qui suis-je ? (1 point)
- 4) Je suis une classe qui pourra être hérité mais qui ne pourra pas être instancier (impossible de faire new ...). Qui suis-je et comment faire pour me créer ? (1 point)
- 5) Donnez 3 types de visibilité sur un attribut expliquez la différence. (1.5 points)
- 6) Expliquez ce qu'est une exception. Décrire les éléments qui la compose (1 point)
- 7) Dans un modèle MVC quel est le rôle du router. Quelle est la variable super globale qu'il utilise pour accomplir son rôle ? (1 point)
- 8) Expliquez les avantages de l'utilisation d'un pattern MVC. (1.5 points)
- 9) Détaillez les éléments qui composent un pattern MVC (1 point)

Veillez vous référer au modalités de rendu pour cet exercice.

Partie pratique (11 point)

Vous devez réaliser un CRUD afin de pouvoir ajouter, éditer ou supprimer des recettes de ravioles.

Une recette de raviole est composée des éléments suivants :

- Un titre (obligatoire)
- 1 ingrédient principal (obligatoire)
- Un champ de type textarea pour expliquer la recette (optionnel)
- (bonus) 1 image

Le (bonus) = (1 point en plus)

Optionnel = il existe mais on est pas obligé de le saisir. C'est pas « je le fais si je veux »

Pour valider des points dans cette épreuve, vous devrez répondre aux user-stories suivantes. Vous devrez impérativement utiliser des concepts objets et un pattern MVC.

En tant qu'utilisateur je peux ajouter une recette de raviole (2.5 points)

En tant qu'utilisateur je peux avoir un listing de toutes les variantes de ravioles dans un tableau (2.5 points)

En tant qu'utilisateur je peux voir le détail d'une recette de raviole (2.5 points)

En tant qu'utilisateur je peux éditer une recette de raviole.

En tant qu'utilisateur, je peux supprimer une recette de raviole (2.5 points)

En tant qu'utilisateur, je peux ajouter, voir et éditer la photo d'une recette de raviole (1 point)

Rendu (1 point bonus)

Le rendu devra être effectué sur la plateforme github. Le nom du repository devra impérativement comprendre votre nom, votre prénom et le sigle POO. Les réponses aux questions techniques devront se situer dans un fichier qui s'appellera readme.md à la racine de votre projet.

SQL

PROPOSITION SQL pour créer la base de donnée (celle que j'ai utilisé) :

```
/*!40101 SET
@OLD_CHARACTER_SET_CLIENT=@@CHARACTER_SET_CLIENT */;
/*!40101 SET
@OLD_CHARACTER_SET_RESULTS=@@CHARACTER_SET_RESULTS */;
/*!40101 SET
@OLD_COLLATION_CONNECTION=@@COLLATION_CONNECTION */;
/*!50503 SET NAMES utf8mb4 */;
/*!40103 SET @OLD_TIME_ZONE=@@TIME_ZONE */;
/*!40103 SET TIME_ZONE='+00:00' */;
/*!40014 SET @OLD_UNIQUE_CHECKS=@@UNIQUE_CHECKS,
UNIQUE_CHECKS=0 */;
/*!40014 SET @OLD_FOREIGN_KEY_CHECKS=@@FOREIGN_KEY_CHECKS,
FOREIGN_KEY_CHECKS=0 */;
/*!40101 SET @OLD_SQL_MODE=@@SQL_MODE,
SQL_MODE='NO_AUTO_VALUE_ON_ZERO' */;
/*!40111 SET @OLD_SQL_NOTES=@@SQL_NOTES, SQL_NOTES=0 */;
DROP TABLE IF EXISTS `raviole`;
/*!40101 SET @saved_cs_client = @@character_set_client */;
/*!50503 SET character_set_client = utf8mb4 */;
CREATE TABLE `raviole` (
  `id` int(11) NOT NULL AUTO_INCREMENT,
  `ingredient_principal` varchar(250) NOT NULL,
  `titre` varchar(250) NOT NULL,
  `recette` text,
  `file` varchar(250) DEFAULT NULL,
  PRIMARY KEY (`id`),
  UNIQUE KEY `raviole_id_uindex` (`id`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
/*!40101 SET character_set_client = @saved_cs_client */;
```