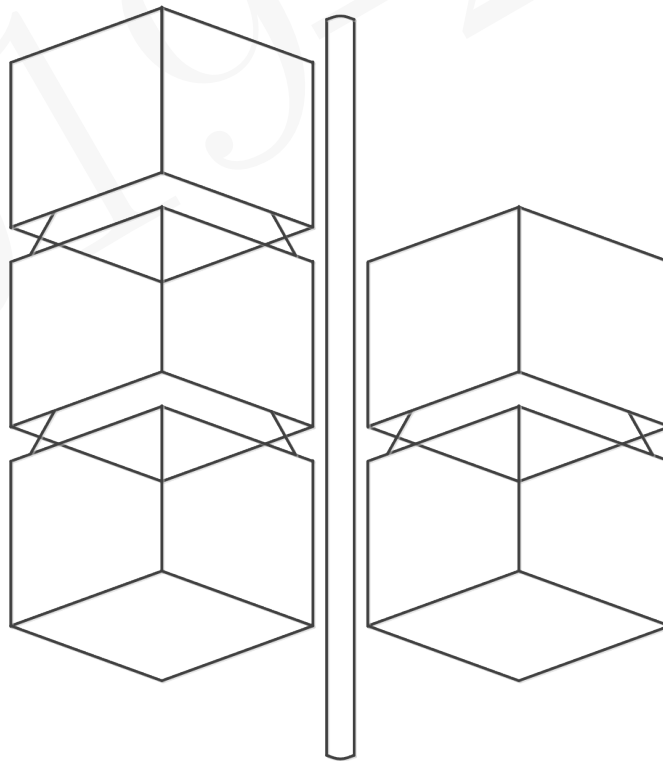


Développement Web - PHP

PHP - Rattrapages

03 juillet 2020

Version 1



Copyright

Ce document est destiné à l'usage interne de Paris 1 - Panthéon Sorbonne.

Copyright © Fabrice BOISSIER - 2020

Ce document est soumis à conditions :

Il est interdit de partager ce document avec d'autres personnes.

Vérifiez que vous disposez de la dernière révision de ce document.

Table des matières

1	Consignes Générales	IV
2	Format de Rendu	V
3	Aide Mémoire	VI
4	Mini Projet	1
5	Palier 0 - Base de Données (Pré-Requis)	2
6	Palier 1 - Présentation des Produits (2 points)	4
7	Palier 2 - Modification des Produits (3 points)	5
8	Palier 3 - Commande (3 points)	6
9	Palier 4 - Panier (5 points)	7
10	Palier 5 - Utilisateurs (4 points)	8
11	Palier 6 - Administrateurs (3 points)	9

1 Consignes Générales

Consigne Générale 0 : Vous devez lire le sujet.

Consigne Générale 1 : Vous devez respecter les consignes.

Consigne Générale 2 : Vous devez rendre le travail dans les délais prévus.

Consigne Générale 3 : Le non-respect des consignes entraînera des pénalités pouvant aller jusqu'à la note non négociable de 0.

Consigne Générale 4 : Le travail doit être rendu dans le format décrit à la section [Format de Rendu](#).

Consigne Générale 5 : Le travail rendu ne doit pas contenir de fichiers binaires, temporaires, ou d'erreurs (`*~`, `*.o`, `*.a`, `*.so`, `*#*`, `*core`, `*.log`, `*.exe`, binaires, ...).

Consigne Générale 6 : Dans l'ensemble de ce document, la casse (caractères majuscules et minuscules) est très importante. Vous devez strictement respecter les majuscules et minuscules imposées dans les messages et noms de fichiers du sujet.

Consigne Générale 7 : Dans l'ensemble de ce document, **nom1-nom2** correspond à la combinaison des deux noms de votre binôme (par exemple pour Fabrice BOISSIER et Ali JAFFAL, cela donnera **boissier-jaffal**).

Consigne Générale 8 : Dans l'ensemble de ce document, le caractère `_` correspond à une espace (s'il vous est demandé d'afficher `___`, vous devez afficher trois espaces consécutifs).

Consigne Générale 9 : Tout retard, même d'une seconde, entraîne la note non négociable de 0.

Consigne Générale 10 : La triche (échange de code, copie de code ou de texte, ...) entraîne **au mieux** la note non négociable de 0.

Consigne Générale 11 : En cas de problème avec le projet, vous devez contacter le plus tôt possible les responsables du sujet aux adresses mail indiquées.

Conseil : N'attendez pas la dernière minute pour commencer à travailler sur le sujet.

2 Format de Rendu

Responsable(s) du projet :	Fabrice BOISSIER <fabrice.boissier@univ-paris1.fr>
Balise(s) du projet :	[PHP] [RATT]
Nombre d'étudiant(s) par rendu :	1
Procédure de rendu :	Envoi par mail
Nom du répertoire :	nom-Rattrapages
Nom de l'archive :	nom-Rattrapages.zip
Date maximale de rendu :	03/07/2020 23h42
Durée du projet :	5 jours
Architecture/OS :	WAMP ou MAMP
Langage(s) :	PHP & SQL
Compilateur/Interpréteur :	Apache/PHP & MySQL/MariaDB
Options du compilateur/interpréteur :	

L'arborescence attendue pour le projet est la suivante :

```
nom-Rattrapages/  
nom-Rattrapages/AUTHORS.txt  
nom-Rattrapages/BDD.sql.gz  
nom-Rattrapages/MiniProjet/  
nom-Rattrapages/MiniProjet/index.php  
nom-Rattrapages/MiniProjet/admin_simple.php  
nom-Rattrapages/MiniProjet/commande.php  
nom-Rattrapages/MiniProjet/panier.php  
nom-Rattrapages/MiniProjet/login.php  
nom-Rattrapages/MiniProjet/profil.php  
nom-Rattrapages/MiniProjet/admin.php
```

Les fichiers suivants sont requis :

AUTHORS.txt contient les nom(s) et prénom(s) du (ou des) auteur(s).

3 Aide Mémoire

Le travail doit être rendu au format **.zip**, c'est-à-dire une archive **zip** compressée avec un outil adapté (les logiciels *7zip* ou *Keka* sont gratuits et adaptés).

Tout autre format d'archive (rar, 7zip, gz, gzip, bzip, ...) ne sera pas pris en compte, et votre travail ne sera pas corrigé (entraînant la note de 0).

Dans ce sujet précis, vous aurez à construire des tables sur MySQL avec PhpMyAdmin. Vous devrez donc exporter vos tables depuis PhpMyAdmin et joindre le fichier. Pour exporter une base de données depuis PhpMyAdmin : cliquez sur le nom de la base de données dans le menu de gauche, puis dans le menu en haut à droite, cliquez sur **Export**. Sélectionnez **Custom** pour personnaliser l'export. Choisissez le format **SQL**. Puis cliquez sur **Save output to a file**, et sélectionnez dans **Compression** : *gzipped*. Puis cliquez tout en bas sur **Go**.

4 Mini Projet

Objectif : Le but de l'exercice est de faire une maquette minimaliste d'un site de vente en ligne.

Ce mini projet, à réaliser **seul(e)**, vous amènera à développer une interface minimaliste pour gérer des produits d'un magasin en ligne. Les notions abordées dedans impliquent de coder en PHP, de gérer des formulaires (avec GET et POST), et de créer des tables SQL sous PhpMyAdmin avec des clés étrangères pour faire une jointure dans des requêtes SQL.

La qualité graphique du site n'est absolument pas évaluée ! Seule la *logique métier* est testée (néanmoins, un site inutilisable à cause de l'interface graphique inexistante ou inutilisable vaut 0). Vous devrez donc uniquement utiliser quelques balises HTML telles que :

- `<html>`, `<header>`, `<title>`, `<body>`
- `<center>`, `<h1>`, `<h2>`, `<h3>`, ``, `<i>`, `<p>`, `
`
- ``
- `<table>`, `<tr>`, `<th>`
- `<form action=... method=...>`, `<input type=... name=...>`

Ce projet n'est absolument pas complexe si vous le réalisez pas à pas, mais il nécessite plusieurs jours de travail.

Vous devez suivre l'ordre des paliers pour obtenir les points : tant que le palier 1 n'est pas fonctionnel, il est inutile de tenter le palier 2, car il s'appuiera inévitablement sur le palier précédent (de plus, chaque palier peut vous aider à déboguer les suivants, et beaucoup de code peut être dupliqué de page en page). Se concentrer exclusivement sur les paliers 5 et/ou 6 ne vous octroiera aucun point : vous **devez** réaliser les paliers précédents.

N'oubliez pas de tester les valeurs de retour du connecteur SQL et surtout d'afficher les messages d'erreur lorsque vous développez ! Ceux-ci indiquent explicitement ce qui ne fonctionne pas.

5 Palier 0 - Base de Données (Pré-Requis)

Nom du(es) fichier(s) : **BDD.sql.gz**
Répertoire : **nom-Rattrapages/**
Droits sur le répertoire :
Droits sur le(s) fichier(s) :
Fonctions autorisées :

Objectif : Le but du palier est de faire une base de données sur PhpMyAdmin pour la suite du mini projet.

Vous devez créer une base de données simple qui contiendra des produits, des numéros de commande, et des liens entre les commandes et les produits. Vous devrez donc créer 3 tables dans MySQL en utilisant PhpMyAdmin, et vous devrez fournir l'extraction de la table et des données de test que vous insèrerez. Le fichier contenant l'export doit être nommé : **BDD.sql.gz**.

Étant donné que les paliers suivants impliquent d'ajouter d'autres tables, vous enverrez l'export de la base de données avec l'ensemble des tables requises pour le fonctionnement du site (et pas uniquement les 3 premières tables décrites ici, sauf si vous n'avez pas atteint les paliers en question)

Les 3 tables à construire pour ce palier sont : *articles*, *commandes*, *paniers*.

La table *articles* contiendra les produits disponibles dans le magasin. Chaque article dispose d'un ID unique, d'un nom, d'une quantité en stock, et d'un prix unitaire de vente.

IDArticle	NomArticle	QuantiteStock	PrixUnitaire
1	Pomme	42	1
2	Carotte	36	1
3	Chocolat	21	5
...

FIGURE 1 – Table : articles

La table *commandes* contiendra les commandes passées. Chaque commande dispose d'un ID unique, ainsi que du prix total de vente.

IDCommande	PrixTotal
1	7
2	1
3	38
...	...

FIGURE 2 – Table : commandes

La table *paniers* contiendra l'ensemble des produits achetés lors des commandes. Cette table fait la jointure entre la table contenant les articles et les commandes : c'est ici que chaque article du panier d'un utilisateur est lié à une commande. Chaque ligne dans cette table correspond donc à un article dans un panier, et elle contient logiquement : l'ID d'une commande, l'ID d'un article, la quantité de cet article dans le panier.

IDCommande	IDArticle	QuantiteCommande
1	1 (<i>pomme</i>)	2
1	3 (<i>chocolat</i>)	1
2	2 (<i>carotte</i>)	1
...

FIGURE 3 – Table : paniers

6 Palier 1 - Présentation des Produits (2 points)

Nom du(es) fichier(s) : **index.php**
Répertoire : **nom-Rattrapages/MiniProjet/**
Droits sur le répertoire :
Droits sur le(s) fichier(s) :
Fonctions autorisées :

Objectif : Le but du palier est d'afficher les produits en vente.

Vous devez écrire un fichier **index.php** permettant de lire le contenu de la table **articles** et d'en afficher le contenu. Chaque produit contenu dans la base de données doit être indiqué avec son nom, son prix, et les quantités restantes.

Vous aurez besoin de faire une requête SQL contenant un **SELECT**.

7 Palier 2 - Modification des Produits (3 points)

Nom du(es) fichier(s) : **admin_simple.php**
Répertoire : **nom-Rattrapages/MiniProjet/**
Droits sur le répertoire :
Droits sur le(s) fichier(s) :
Fonctions autorisées :

Objectif : Le but du palier est de créer une micro interface de gestion.

Vous devez créer une page **admin_simple.php** dans laquelle un utilisateur pourra modifier les propriétés de chaque produit existant, et éventuellement le supprimer.

L'utilisateur disposera également d'un formulaire lui permettant d'ajouter un produit.

Vous aurez besoin de faire des requêtes SQL contenant un **SELECT**, un **UPDATE**, un **DELETE**, ou un **INSERT INTO**.

8 Palier 3 - Commande (3 points)

Nom du(es) fichier(s) : **commande.php**
Répertoire : **nom-Rattrapages/MiniProjet/**
Droits sur le répertoire :
Droits sur le(s) fichier(s) :
Fonctions autorisées :

Objectif : Le but du palier est d'afficher les commandes précédentes.

Vous devez créer une page **commande.php** qui affichera les commandes précédentes et leur contenu. Il est fortement conseillé d'afficher un tableau HTML par commande, et de les espacer de quelques lignes.

Pour simplifier le développement (et pour déboguer), n'hésitez pas à d'abord remplir votre base de données de quelques lignes dans la table **commandes**, et quelques lignes dans la table **paniers**. (*Dans l'exemple donné dans le palier 0, la commande 1 contient 2 pommes et 1 chocolat, tandis que la commande 2 contient 1 carotte*)

Vous aurez besoin de faire une requête SQL contenant un **SELECT** et un **JOIN** (ou équivalent).

9 Palier 4 - Panier (5 points)

Nom du(es) fichier(s) : **panier.php**
Répertoire : **nom-Rattrapages/MiniProjet/**
Droits sur le répertoire :
Droits sur le(s) fichier(s) :
Fonctions autorisées :

Objectif : Le but du palier est de créer un panier, d'y ajouter des produits, et enregistrer les commandes.

Vous devez tout d'abord créer une page **panier.php** dans laquelle le contenu de la commande courante est affiché (en utilisant l'ID de la commande en cours). Chaque article ajouté dans le panier est enregistré dans la table *paniers*.

L'ID de la commande en cours s'obtiendra de 2 façons différentes selon le cas :

1. Si la table **commandes** est vide, alors la page **panier.php** doit créer une ligne dedans avec l'ID **1**
2. Si la table n'est pas vide, alors on utilise le dernier ID existant (l'ID le plus grand) de la table **commandes** pour créer une nouvelle ligne

Un bouton permettra à l'utilisateur de supprimer un article de son panier.

Cette page contiendra également un bouton **commander** qui permettra de passer une commande. C'est-à-dire, dans le cadre de ce mini-projet et de ce palier, de fixer le montant total de la commande en cours, puis de créer une nouvelle ligne vide dans la table **commandes** avec un nouvel ID.

N'oubliez pas d'ajouter des boutons sur la page **index.php** pour pouvoir ajouter des articles au panier.

Vous aurez besoin de faire des requêtes SQL contenant un **SELECT**, un **UPDATE**, et un **INSERT INTO**.

10 Palier 5 - Utilisateurs (4 points)

Nom du(es) fichier(s) :	login.php profil.php
Répertoire :	nom-Rattrapages/MiniProjet/
Droits sur le répertoire :	
Droits sur le(s) fichier(s) :	
Fonctions autorisées :	

Objectif : Le but du palier est de créer une gestion des sessions utilisateurs.

Vous devez tout d'abord créer une page **login.php** dans laquelle un utilisateur peut créer un compte, s'identifier, ou se délogger. Sur cette page, plusieurs boutons permettront de gérer chaque cas (authentification, fin de session, ou création de compte). Vous devez créer une table **utilisateurs** dans la base de données qui contient au moins un nom d'utilisateur et un mot de passe, mais vous êtes libres d'ajouter d'éventuelles colonnes supplémentaires.

Ensuite, vous devrez créer une page **profil.php** dans laquelle l'utilisateur peut modifier son mot de passe et éventuellement les autres informations.

La gestion des utilisateurs authentifiés implique que les commandes soient gérées séparément pour chacun. À vous d'ajouter les colonnes nécessaires dans les autres tables de la base de données, ainsi que le code nécessaire dans les autres pages PHP (*index.php*, *commande.php*, *panier.php* principalement) pour pouvoir gérer les sessions utilisateur.

11 Palier 6 - Administrateurs (3 points)

Nom du(es) fichier(s) : **admin.php**
Répertoire : **nom-Rattrapages/MiniProjet/**
Droits sur le répertoire :
Droits sur le(s) fichier(s) :
Fonctions autorisées :

Objectif : Le but du palier est de créer une interface complète de gestion d'un site de vente.

Vous devez créer une page **admin.php** qui permet de suivre les commandes et administrer l'ensemble du site. Seuls les utilisateurs considérés comme des *administrateurs* pourront y accéder (à vous d'ajouter une colonne dans la base de données pour le faire).

Les commandes disposeront maintenant d'un état : elles seront soit en mode *panier* (non validée par l'utilisateur), soit en mode *validée* (l'utilisateur a validé la commande), soit en mode *expédiée* (un administrateur a confirmé sur la page *admin.php* qu'une commande *validée* était expédiée). À vous d'ajouter une colonne dans la base de données pour le faire.

Les fonctionnalités devront permettre de :

1. lister les utilisateurs inscrits sur le site
2. ajouter/retirer le droit administrateur aux utilisateurs
3. lister les articles et modifier leurs informations (nom, quantité, prix)
4. lister les commandes validées
5. faire passer une commande de l'état *validée* à l'état *expédiée* avec un bouton
6. lister les commandes expédiées