

Bases de données 2

TD 1 - Interrogation de Base de Données

Matthieu Nicolas

Exercice 1 - Gestion de stocks

Soit la base de données relationnelle décrite par le schéma relationnel suivant :

Produits(prod_id, nom, pu)

Depots(depot_id, adr, volume)

Stocks(prod_id, depot_id, qte)

et possédant le contenu suivant :

prod_id	nom	pu
p1	A3	10.0
p2	crayon	9
p3	stylo	15
p4	A4	10.0

depot_id	adr	volume
1	Nancy	100
2	Laxou	200
3	Vandoeuvre	115
4	Nancy	220
5	Nancy	1000

prod_id	depot_id	qte
p1	1	0
p3	2	9
p1	3	15
p2	4	20
p3	5	0
p1	4	5
p2	5	2
p3	3	30
p1	2	10

Figure 1: Snapshot (instantané) de la base de données *Gestion de stocks*

Construire et peupler la base de données correspondante, nommée *gestion_stock*, à l'aide de *phpMyAdmin*. Puis construire les requêtes SQL répondant aux questions suivantes :

1. Nom, prix unitaire et quantité des produits stockés dans le dépôt 2
2. Volume du dépôt ayant la capacité de stockage maximale
3. Prix moyen des produits présents dans le dépôt 3
4. Produits stockés dans les dépôts qui sont en rupture de stock dans ce dépôt (< 10)
5. Identifiant et adresse du dépôt ayant la capacité de stockage maximale

6. Coût total de la marchandise par dépôt par produit.

Pour cette requête, notez qu'il est d'ajouter possible dans la clause *SELECT* des champs calculés. Dans le cas présent, le coût de la marchandise pour un produit stocké dans un dépôt s'obtient en multipliant son prix unitaire par la quantité.

Exercice 2 - Bibliothèque

Soit la base de données relationnelle décrite par le schéma relationnel suivant :

Pays(id_pays, nom)

Auteurs(id_auteur, nom, prenom, #nationalite)

Livres(id_livre, titre, date_parution, #id_auteur)

Tags(id_tag, libelle)

TagsParLivre(#id_livre, #id_tag)

Lecteurs(id_lecteur, nom, prenom, adresse)

Emprunts(#id_livre, #id_lecteur, date_emprunt, est_rendu)

Importer cette base de données dans *phpMyAdmin* à l'aide du script *bibliotheque.sql* fourni. Puis construire les requêtes SQL répondant aux questions suivantes :

1. Liste des livres de Michel Bussy
2. Libellé des tags associés au livre "L'homme qui mit fin à l'histoire"
3. Liste des emprunts de Clayson Berkowitz
4. Liste des lecteurs n'ayant pas rendu un livre
5. Nom et prénom des auteurs sud-coréens ou japonais
6. Titre, date de parution, noms et prénoms des auteurs de livres parus en 2020.

Pour manipuler les dates "facilement", SQL met à disposition un ensemble de fonctions (présentées ici : <https://sql.sh/fonctions/date-heure>), notamment *YEAR* que je vous laisse découvrir.

7. Liste des livres possédant le tag "Humour" et "Fantasy".

Afin de décomposer une requête compliquée en plusieurs étapes simples, il est possible de stocker le résultat d'un *SELECT* dans une relation temporaire à l'aide de la notation suivante : *CREATE TEMPORARY TABLE nom_relation SELECT ...*

Cette relation temporaire devient ensuite utilisable dans une nouvelle requête comme n'importe quelle autre table.