

Note Technique Explicative

Organisation de la Base de Données

La base de données "Fleurs" est conçue pour gérer les informations relatives aux clients, aux produits (fleurs et accessoires), aux bouquets, aux magasins, aux stocks et aux commandes dans un magasin de fleurs.

La base de données est conçue en utilisant le modèle entité association, après plusieurs essais le schéma en annexe est le plus fonctionnel lors de l'implémentation des tables en langage SQL le choix principal est le fait de séparer les bouquets dit commande standard des produits et de les constituer comme somme de produits.

Schéma logique d'une Commande Standard : Commande -> Commande_bouquet (id_bouquet) -> Bouquet (Id_bouquet) -> Somme des Produits à l'aide de Bouquet Produit qui fait le lien entre (Id_bouquet et les Id_produit).

Cela nous permet donc de tout ramener à la table produit qui permet de gérer le stock après chaque commande sans devoir créer des stocks de bouquet au préalable.

Tables de la Base de Données

1. **Table Client** : Cette table contient des informations sur les clients, y compris leur nom, prénom, numéro de téléphone, email, mot de passe, carte de crédit, le niveau de fidélité et la date de création du compte. L'attribut "e_mail" est déclaré unique pour éviter les doublons et garantir l'intégrité des données lors de l'inscription sur l'application d'un nouveau client.
2. **Table Produit** : Cette table contient des informations sur les produits, y compris leur ID, nom, type (fleur ou accessoire), prix et disponibilité.
3. **Table Bouquet** : Cette table contient des informations sur les bouquets, y compris leur ID, nom, description, prix et catégorie.
4. **Table Bouquet Produit** : Cette table de liaison représente la relation entre les bouquets et les produits. Elle contient les IDs des bouquets et des produits, et utilise ces IDs comme clés étrangères pour référencer les tables "Bouquet" et "Produit".
5. **Table Magasin** : Cette table contient des informations sur les magasins, y compris leur ID, nom et adresse.
6. **Table Stock** : Cette table contient des informations sur le stock de chaque produit dans chaque magasin, y compris l'ID du magasin, l'ID du produit, la quantité de produit en stock et le seuil d'alerte pour ce produit. Elle utilise les IDs du magasin et du produit comme clés étrangères pour référencer les tables "Magasin" et "Produit".
7. **Table Commande** : Cette table contient des informations sur les commandes, y compris leur ID, l'ID du client qui a passé la commande, l'ID du magasin où la commande a été passée, la date de la commande, l'adresse de livraison, le message accompagnant la commande, la date de livraison souhaitée, l'état de la commande, le type de commande, le prix total, la réduction et le prix maximum. Elle utilise l'ID du client et l'ID du magasin comme clés étrangères pour référencer les tables "Client" et "Magasin".
8. **Table Commande Produit** : Cette table de liaison représente la relation entre les commandes et les produits. Elle contient les IDs des commandes et des produits, la quantité de chaque produit commandé et utilise ces IDs comme clés étrangères pour référencer les tables "Commande" et "Produit".

9. **Table Commande Bouquet** : Cette table de liaison représente la relation entre les commandes et les bouquets. Elle contient les IDs des commandes et des bouquets, la quantité de chaque bouquet commandé et utilise ces IDs comme clés étrangères pour référencer les tables "Commande" et "Bouquet".

Explication des requêtes importantes principales

Module client :

Requête Connexion Client / Procédure Vérification Email : Cette procédure stockée est utilisée pour vérifier si un email est déjà utilisé dans la base de données par un autre client. Elle prend en entrée une adresse e-mail et retourne un message indiquant si l'adresse e-mail est disponible ou déjà prise. Familiarisation avec les procédures stockées compréhension des DELEMITER // car utilisation de la ; dans la procédure, In et Out comme paramètre d'entrée et de sortie.

Requête Fidélité : Cette requête est utilisée pour calculer le nombre de commandes par mois pour un client spécifique et mettre à jour son statut de fidélité en conséquence.

Historique des Commandes : Cette requête est utilisée pour afficher l'historique des commandes d'un client spécifique. Elle retourne l'ID du client, l'ID de la commande, la date de la commande, le prix total, l'ID du produit, le nom du produit, la quantité de produit, l'ID du bouquet, le nom du bouquet et la quantité de bouquet pour chaque commande passée par le client. Les résultats sont triés par date de commande décroissante, puis par ID de commande, ID de produit et ID de bouquet. Cette requête permet de retracer tous les détails des commandes d'un client. Utilisation du LEFT pour garder les attributs nuls et avoir une meilleure vision des habitudes du client.

Module Commande et stock :

Ces procédures stockées facilitent la création de nouvelles commandes en automatisant le calcul de la réduction, la détermination de l'état de la commande, et l'insertion des informations de la commande dans les tables appropriées.

`CreateStandardOrder1` : Une nouvelle ligne est insérée dans la table `Commande_Bouquet` pour lier la commande au bouquet commandé.

`CreatePersonalizedOrder` : Cette procédure stockée permet de créer une nouvelle commande personnalisée. A noter que cette procédure ne récupère pas le string du message.

Mise à jour du stock pour une commande personnalisée : Cette requête met à jour le stock de produits pour une commande personnalisée. Elle décrémente la quantité en stock pour chaque produit dans la commande, en utilisant une jointure entre les tables Stock, Commande_Produit et Commande. Pour la mise à jour du stock pour une commande il s'agit d'une jointure entre les tables Stock, Bouquet_Produit, Commande_Bouquet et Commande.

Vérification du seuil d'alerte : Cette requête vérifie si la quantité en stock pour chaque produit dans le magasin concerné est tombée en dessous du seuil d'alerte à la suite de la commande.

Modules statistiques :

Meilleurs clients du mois/de l'année : Ces deux requêtes sont des exemples de requêtes synchronisées (ou auto-sync en anglais). Elles utilisent une sous-requête pour filtrer les commandes selon le mois ou l'année actuelle, puis elles regroupent les résultats par Id_client et somment le Prix_total de chaque groupe pour obtenir le total mensuel ou annuel dépensé par chaque client. Les résultats sont ensuite triés par Total_mensuel ou Total_annuel en ordre décroissant.

5 bouquets les plus vendus / 5 produits les plus vendus : Cette requête est un exemple d'une requête UNION. Elle combine les résultats de deux requêtes SELECT distinctes. La première sous-requête sélectionne les 5 bouquets les plus vendus et la deuxième sous-requête sélectionne les 5 produits les plus vendus. Les résultats sont ensuite combinés et triés par Quantite_vendue en ordre décroissant.

