

Projet : « Gestion d'une entreprise de pizzas à domicile »

Pierre Lefebvre

Vous pouvez réaliser ce projet par équipe de deux à quatre étudiants. Il faudra déposer votre projet sous Blackboard. La date limite de dépôt vous sera donnée ultérieurement. Le format du document déposé doit être au format pdf. Il est conseillé d'utiliser Looping pour la conception de la base de données. Ce projet représentera la moitié de la note de l'unité, la seconde moitié sera le devoir sur table.

On veut modéliser la gestion d'une entreprise de fabrication et de livraison de pizzas à domicile : la société RaPizz. Il s'agit d'une société en franchise qui utilise des formats et des compositions de pizzas normalisées à partir d'un ensemble d'ingrédients déterminés. En d'autres termes, le client n'a pas la liberté de composer lui-même une pizza personnalisée ; il doit choisir dans le catalogue proposé.

Produits

Les produits vendus sont... des pizzas. Une pizza est caractérisée par son nom, les ingrédients qui la composent et son prix de base. Pour chaque pizza, il existe trois tailles : « naine », « humaine » et « ogresse ». La « naine » est 1/3 moins chère que le prix de base, c'est-à-dire la taille « humaine », et l'« ogresse » est 1/3 plus chère.

Mode de distribution

Les pizzas sont livrées par des livreurs qui circulent en voiture ou à moto et qui n'ont pas de véhicules attitrés. La base de données doit également permettre le suivi de l'activité des livreurs et des véhicules qu'ils utilisent.

Modalités de vente

Le mode de vente est du type prépayé : préalablement à toute commande, les clients doivent s'abonner au service et approvisionner leur compte. On vérifie le solde du compte avant de préparer et de livrer la commande. Il existe deux systèmes de bonification :

- Une pizza gratuite est offerte au bout de 10 pizzas achetées.
- Toute pizza livrée en plus de trente minutes est gratuite.

Objectifs du système

Le but de cette base de données est de gérer l'activité quotidienne de vente et de livraison de pizzas :

- vérification du solde du compte et facturation aux clients ;
- suivi du chiffre d'affaires ;
- refus d'honorer les commandes pour lesquelles le solde du compte client est insuffisant ;
- non-facturation des pizzas gratuites (retard ou fidélité).

On veut également effectuer des statistiques diverses sur l'activité :

- identification du meilleur client ;
- identification du plus mauvais livreur (nombre de retards dans la livraison)
- net du véhicule utilisé ;
- identification du plus mauvais livreur (nombre de retards dans la livraison)
- véhicule utilisé ;
- identification de la pizza la plus ou la moins demandée ;
- identification de l'ingrédient favori...

Afin de simplifier le problème, on considère que l'opération de base à modéliser dans cette activité est la vente d'une unique pizza. La notion de commande qui peut contenir plusieurs pizzas n'est pas prise en compte. On pourra faire évoluer le système plus tard si besoin est pour intégrer cet aspect.

Travail à réaliser

Conception de la base de données

- 1) Elaboration du modèle entité-association
- 2) Passage au modèle relationnel
- 3) Script de la création des tables
- 4) Script d'insertion des données dans la base de données

Interrogation de la base de données

1) Menu

On veut extraire les données qui servent à imprimer la carte, ce qui signifie que l'on veut disposer du nom de chaque pizza, de son prix et des ingrédients qui la composent.

2) Fiche de livraison

On veut imprimer une fiche de livraison qui mentionne le nom du livreur, le type du véhicule utilisé, le nom du client, la date de la commande, le retard éventuel, le nom et le prix de base de la pizza.

3) Questions diverses

- Quels sont les véhicules n'ayant jamais servi ?
- Calcul du nombre de commandes par client ?
- Calcul de la moyenne des commandes ?
- Extraction des clients ayant commandé plus que la moyenne ?

4) Programmation

Cette dernière partie est entièrement libre quant à ce que vous souhaitez développer. Elle peut exploiter l'API JDBC et Java, les procédures stockées, les triggers.