

BINV-1030

Gestion de données : bases

Partie 2 : Exercices de conception

Cambron I., Damas C., Ferneeuw S., Lecharlier L., Lehmann B.

Bachelier en Informatique de Gestion

2022-2023

Table des matières

Les exercices	3
1 Séance 1.....	3
1.1 Théâtre du TRAC.....	3
1.2 Cinéma de l'angoisse	6
2 Séance 2.....	8
2.1 La gestion de projets	9
2.2 Dix Nez	10
3 Séance 3.....	11
3.1 La superette.....	11
3.2 Gestion des kots	13
4 Séance 4.....	14
4.1 EuroPlastics	14
4.2 Babyboom	16
5 Séance 5.....	20
5.1 Le distributeur de livres et DVD.	20
5.2 Bullox	22
6 Séance 6.....	24
6.1 Grantano.....	24
6.2 Les 3 Cuisses	26
7 Séance 7.....	29
7.1 Courtier d'assurances.....	29
7.2 Bouffe Kot.....	30
8 Séance 8.....	32
8.1 RecetteBook	32
8.2 Lip dub mania	34

Les exercices

1 Séance 1

1.1 Théâtre du TRAC

Partie 1

Le **théâtre du trac** organise une douzaine de spectacles chaque saison. Chaque spectacle est identifié par un code mnémonique de 4 caractères et pour chacun, on mémorise le titre et le prix des places A (proches de scène, les plus chères) et le prix des places B (moins chères). Le prix de ces deux catégories de places varie suivant le spectacle.

Chaque spectacle est joué un certain nombre de fois dans une saison : ce sont les représentations. Il n'y a qu'une représentation par jour par spectacle. Par représentation, on désire mémoriser le nombre de places vendues en distinguant les places A et B. A la fin de chaque représentation, on enregistrera ces nombres. Un certain nombre de places gratuites sont distribuées essentiellement aux journalistes. Un invité doit présenter sa carte d'invitation. Après chaque représentation, on comptera le nombre d'invitations.

- a. On désire connaître le montant des recettes, en distinguant chaque spectacle ainsi que les places A et B. On indiquera également le nombre d'invitations.
- b. On désire également connaître le montant total des recettes par jour de la semaine (du lundi au dimanche). Une fonction SQL permet d'extraire le jour de la semaine d'une date : `date_part1('isodow', une_variable_de_type_date)`. Cette fonction renvoie 1 pour lundi, 2 pour mardi etc.

On vous demande

1. de dessiner le DSD et de spécifier clairement toutes les relations ainsi que les clés primaires et étrangères.
2. d'écrire les CREATE TABLE.
3. d'écrire les deux requêtes répondant aux points a et b ci-dessus (En fin de saison, toutes ces données sont archivées et le programme de la nouvelle saison est introduite dans la DB).

¹Rappel : `date_part` est une fonction qui extrait de l'information dans un attribut de type `timestamp`. Cette fonction prend deux paramètres : le premier pour préciser le type d'information à extraire (`weekday`, `month`, `year`, `hour`, `minute`, ...) et le second est l'attribut du type `timestamp`. Cette fonction renvoie un entier. Supposons une variable `une_date = '2001-02-16 20:38:40'`, `SELECT date_part('hour', une_date)` renvoie 20 ou encore `SELECT date_part('isodow', timestamp '2001-02-16 20:38:40')` renvoie 5 (vendredi).

Partie 2

Très content de votre analyse, le théâtre du trac, vous demande un outil de gestion de ses dépenses. Les dépenses sont réparties par postes : costumes, décors, mise en scène, comédiens, régisseur, etc. Le nombre des postes utilisés peut varier d'un spectacle à l'autre mais la majorité de ces postes sont repris dans chaque spectacle. Voici un exemple de budget : Il est valable pour une saison.

<u>Spectacle Marivaux</u>	
Comédiens	20.000
Mise en scène	2.000
Décors	9.500
<u>Costumes</u>	<u>1.000</u>
Total	32.500
<u>Spectacle Brecht</u>	
Comédiens	5.000
Mise en scène	2.000
Décors	1.500
Costumes	1.000
<u>Eclairages</u>	<u>1.200</u>
Total	10.700

Au début de chaque saison, le programme de la nouvelle saison est mis sur pied. Pour chaque spectacle, le directeur établit un budget, c'est-à-dire une prévision des dépenses, poste par poste. Au cours de la saison, le théâtre reçoit des factures (du costumier, des décorateurs...) paie le metteur en scène, les comédiens. Attention, pour un poste donné d'un spectacle, il peut recevoir plusieurs factures mais une facture ne concerne qu'un seul spectacle. Le directeur enregistre pour chaque dépense à quel spectacle et à quel poste il doit l'attribuer. Une dépense ne peut être affectée qu'à un poste prévu dans le budget. Dans une première version, un champ dépense sera ajouté, à côté de budget, et sera augmenté à chaque ajout de facture.

A la fin de la saison, le directeur, voudrait imprimer le résultat suivant (une présentation moins élégante sera acceptée)

<u>Spectacle</u>	<u>Poste</u>	<u>Budget</u>	<u>Dépense</u>	<u>Différence</u>
Spectacle Marivaux	Comédiens	20.000	18.000	+2.000
	Mise en scène	2.000	1.800	+200
	Décors	9.500	12.000	-2.500
	Costumes	1.000	2.000	-1.000
Spectacle Brecht				
	Comédiens	5.000	7.000	-2.000
	Mise en scène	2.000	2.000	0
	Décors	1.500	1.500	0
	Costumes	1.000	800	+200
	Eclairages	1.200	1.250	-50

On vous demande d'adapter votre DSD et d'écrire la requête qui permet de réaliser le rapport ci-dessus.

Partie 3

Le directeur voudrait maintenant pouvoir mémoriser toutes les factures cachées pour le moment derrière l'attribut dépense. Pour chaque facture, on mémorisera son numéro, la date de facturation, la date d'échéance ainsi que le montant de la facture. On enregistre les factures liées à des budgets de dépense prévu pour un poste d'un spectacle.

On vous demande de modifier le DSD.

Finalement, écrivez la requête qui permet d'afficher la liste des factures reçues (le numéro de la facture, le titre du spectacle, la date de la facture ainsi que le libelle du poste, le montant de la facture) triées par poste de dépense et puis par date.

Bonus

Que faut-il changer à votre DSD pour permettre l'encodage de factures non liées à un budget prévu ?

Devoir : requête à faire sur le DSD de la **partie 2**

Donnez, pour chaque poste dont le budget a été dépassé dans plusieurs films, son identifiant, son libellé ainsi que le nombre de films dans lesquels son budget a été dépassé. Les postes dont le budget a été dépassé dans le plus de films doivent apparaître en premier.

1.2 Cinéma de l'angoisse

Partie 1

Le **Cinéma de l'angoisse** organise un festival des plus grands films d'horreur primés partout dans le monde.

Chaque film est identifié par un code de 5 caractères composé de la première lettre du nom du réalisateur, la première lettre de son prénom et de 3 chiffres incrémentaux.

Il est possible de réserver deux types de places :

- la place simple comprend une entrée pour un film
- la place VIP comprend un apéritif avant le film, une entrée pour un film et un souper après celui-ci.

Le prix de ces deux types de places varie pour chaque film.

Par exemple :

- Pour le film « Le javelot infernal », le prix des places simples est de 5€ et celui des places VIP de 27 €
- Pour le film « Le retour de la tronçonneuse », le prix des places simples est de 7€ et celui des places VIP de 31€

Chaque film est joué un certain nombre de fois pendant le festival : ce sont les séances. En fonction de la demande, il peut y avoir un nombre de séances plus ou moins grand d'un même film.

Pour chaque séance, il faut mémoriser la date et l'heure auxquelles elle a lieu.

Par séance, on désire mémoriser le nombre de places vendues en distinguant les places simples et VIP.

- a. On désire connaître le montant des recettes, en distinguant chaque film ainsi que les places simples et VIP.
- b. On désire également connaître le montant total des recettes par jour de la semaine (du lundi au dimanche). Pour rappel, une fonction SQL permet d'extraire le jour de la semaine d'une date : `date_part2('isodow', une_variable_de_type_date)` . Cette fonction renvoie 1 pour lundi, 2 pour mardi etc.

On vous demande

1. de dessiner le DSD et de spécifier clairement toutes les relations ainsi que les clés primaires et étrangères.
2. d'écrire les CREATE TABLE.
3. d'écrire les deux requêtes répondant aux points a et b ci-dessus.

² Supposons une variable `une_date = '2001-02-16 20:38:40'`, `SELECT date_part('hour', une_date)` renvoie 20 ou encore `SELECT date_part('isodow', timestamp '2001-02-16 20:38:40')` renvoie 5 (vendredi).

Partie 2

Très content de votre analyse, le Cinéma de l'angoisse, vous demande un outil de gestion de ses dépenses. Les dépenses sont réparties par postes : invités spéciaux (réalisateur, acteurs...), droits de diffusion, location de la salle, frais pour les soupers, frais d'apéritif, etc. Le nombre des postes utilisés peut varier d'un film à l'autre mais la majorité de ces postes sont repris dans chaque film. Voici un exemple de budget : Il est valable pour un festival.

<u>Film « Le retour de la tronçonneuse »</u>	
Invités spéciaux	20.000
Droits de diffusion	10.000
Location salle	2.000
Soupers	5.500
<u>Apéritifs</u>	<u>1.000</u>
Total	38.500
<u>Film « Un tueur parmi les profs »</u>	
Droits de diffusion	10.000
Location salle	2.000
Soupers	4.500
<u>Apéritifs</u>	<u>800</u>
Total	17.300

Au début de chaque année, le programme du nouveau festival est mis sur pied. Pour chaque film, le directeur établit un budget, c'est-à-dire une prévision des dépenses, poste par poste.

Pendant le festival, le cinéma reçoit des factures (du réalisateur, des gestionnaires des salles...). Attention, pour un poste donné d'un film, il peut recevoir plusieurs factures mais une facture ne concerne qu'un seul film.

Le directeur enregistre pour chaque dépense à quel film et à quel poste il doit l'attribuer. Une dépense ne peut être affectée qu'à un poste prévu dans le budget. Dans une première version, un champ dépense sera ajouté, à côté de budget, et sera augmenté à chaque ajout de facture.

A la fin de la saison, le directeur, voudrait imprimer le résultat suivant (une présentation moins élégante sera acceptée)

Film	Poste	Budget	Dépense	Différence
« Le retour de la tronçonneuse »				
	Invités spéciaux	20.000	25.000	-5.000
	Droits diffusion	10.000	10.000	0
	Location salle	2.000	2.000	0
	Soupers	4.500	2.000	+2.500
	Apéritifs	1.000	1.100	-100
« Un tueur parmi les profs »				
	Droits diffusion	10.000	15.000	-5.000
	Location salle	2.000	1.800	+200
	Soupers	4.500	3.700	+800
	Apéritifs	800	1.250	-450

On vous demande d'adapter votre DSD et d'écrire la requête qui permet de réaliser le rapport ci-dessus.

Partie 3

Le directeur voudrait maintenant pouvoir mémoriser toutes les factures cachées pour le moment derrière l'attribut dépense. Pour chaque facture, on mémorisera son numéro, la date de facturation, la date d'échéance ainsi que le montant de la facture. On enregistre les factures liées à des budgets de dépense prévu pour un poste d'un film.

On vous demande de modifier le DSD et d'adapter la requête écrite à la partie 2.

Finalement, écrivez la requête qui permet d'afficher la liste des factures reçues (le numéro de la facture, le titre du film, la date de la facture ainsi que le libellé du poste, le montant de la facture) triées par poste de dépense et puis par date.

Bonus

Que faut-il changer à votre DSD pour permettre l'encodage de factures non liées à un budget prévu ?

Devoir : requête à faire sur le DSD de la **partie 2**

Donnez, pour chaque poste dont le budget a été dépassé dans plusieurs films, son identifiant, son libellé ainsi que le nombre de films dans lesquels son budget a été dépassé. Les postes dont le budget a été dépassé dans le plus de films doivent apparaître en premier.

2 Séance 2

2.1 La gestion de projets

Une société de software voudrait un outil pour planifier les projets qu'elle doit développer et s'adresse à vous.

Un projet est identifié par un numéro. On mémorise pour chaque projet le titre (ex : filmothèque), le nom du client, une date d'échéance et un montant. Un projet peut aussi être identifié par son titre.

La société se compose d'une vingtaine d'analystes-programmeurs. Pour chacun d'eux, on mémorise son matricule, son nom, son prénom, son GSM et son email. Cette dernière information doit également permettre de retrouver les analystes-programmeurs. Le GSM et l'email peuvent être modifiés.

En début de projet, une découpe en composants est effectuée. Chaque composant a un descriptif et passe par une série d'étapes successives : l'étape d'**analyse**, l'étape de **conception**, l'étape de **programmation**, l'étape de **tests** et l'étape de **maintenance**. Chaque composant est sous la responsabilité d'un seul analyste-programmeur. Chaque composant comprendra également un temps présumé qui sera fonction de la difficulté de celui-ci et de l'expérience de l'analyste-programmeur qui en est responsable. Pour chaque composant, on pourra également préciser une date de début et une date de fin des tests.

A la fin de chaque journée, chaque analyste-programmeur note son emploi du temps c'est-à-dire le temps qu'il a consacré à chacun des composants dont il est responsable. Un analyste-programmeur ne peut consacrer du temps qu'à un composant répertorié : il n'y a pas de composant détaché d'un projet (du style « boire du café »). Quand un analyste-programmeur a terminé une des étapes de développement d'un composant, il le fait passer à l'étape suivante. Quand l'étape d'analyse est finie, le composant passe à l'étape de conception ; une fois cette dernière terminée, le composant passe à l'étape de programmation. Enfin, quand les tests sont achevés, le composant passe à l'étape de maintenance.

1. Réaliser le diagramme de structure de données de l'application.
2. Ecrivez les requêtes permettant de répondre aux points ci-dessous :
 - a. Pour chaque projet et pour chaque composant, donnez le nombre d'heures prévues et le nombre d'heures déjà réalisées et l'étape en cours.
 - b. Donnez, pour chaque analyste-programmeur, son nom, son matricule et le nombre d'heures travaillées durant le mois de janvier 2016.
 - c. Donnez, pour chaque projet, son numéro, son titre, le nombre de composants qu'il comporte et le nombre de composants qui sont à l'étape maintenance.

BONUS : Parfois un analyste-programmeur consacre du temps à un projet mais ce temps n'entre en compte dans aucun composant de celui-ci. Comment adapter le DSD pour répondre à ce problème ?

2.2 Dix Nez

La société **Dix Nez** engage des artistes qui exécutent un certain nombre de numéros, seul ou à plusieurs. La société souhaite enregistrer le temps consacré par chaque artiste pour la préparation d'un numéro.

Un artiste possède un nom, un prénom, un numéro de téléphone ainsi qu'une adresse. Pour identifier un artiste on utilise un code constitué de la première lettre du nom suivi de la première lettre du prénom suivi éventuellement d'un numéro. Le numéro de téléphone permet également de retrouver un artiste de manière unique.

Un numéro possède un intitulé, une durée et une catégorie parmi les suivantes {CLOWN, MAGIE, FORCE}. L'intitulé du numéro est unique, toutefois il peut être modifié.

Certains numéros sont exécutés par plusieurs artistes.

Par exemple, Mick Hey exécute 3 numéros : « poulpe friction », « l'art triste » et « l'alarme fatale ». Les deux premiers numéros sont dans la catégorie CLOWN, tandis que le troisième est dans la catégorie FORCE. Pour le numéro « poulpe friction » il est accompagné par Doc Nald et Dinh Go.

Chaque artiste doit pouvoir encoder le temps consacré à la mise en place d'un numéro. Il doit pouvoir clairement enregistrer ce temps.

Par exemple Mick Hey doit pouvoir enregistrer qu'il a consacré, pour le spectacle « poulpe friction » :

- 2h le lundi 6 août,
- 5 h le mardi 7 août
- 1h le mercredi 8 août

Il doit être possible d'accompagner ces emplois du temps d'un petit texte justificatif (par exemple : « répétition de l'intro »).

1. Réalisez le diagramme de structure de données de l'application.
2. Ecrivez les requêtes permettant de répondre aux points ci-dessous :
 - a. Donnez pour chaque numéro, son « identifiant », son intitulé et la liste des artistes (code, nom et prénom) qui y participent. Triez par ordre croissant d'identifiant
 - b. Donnez, pour chaque numéro et pour chaque artiste l'exécutant, le temps total consacré par cet artiste pour ce numéro.
 - c. Donnez le nombre de numéros par catégorie.

BONUS : Parfois un artiste peut consacrer du temps à la mise en place d'un numéro qu'il n'exécute pas. Comment adapter le DSD pour répondre à ce problème ?

3 Séance 3

3.1 La superette

Dessinez le diagramme de structure des données pour le problème suivant :

Une petite superette indépendante veut informatiser la gestion de ses ventes et de son stock.

Les membres du personnel occupent indifféremment les postes de réassortisseurs ou de caissiers. Il y a trois caisses enregistreuses qui sont ouvertes en fonction des besoins (en cas d'affluence, les trois caisses sont ouvertes).

Chaque employé a un nom, un prénom et un code personnel. Chaque caisse a une marque et une date de dernier entretien.

Chaque caissier introduit son code personnel lors de l'ouverture de la caisse ou lorsqu'il remplace une autre personne. Il ferme sa session lorsqu'il quitte la caisse. Une caisse qui n'est pas « occupée » par un membre du personnel n'est donc pas capable d'enregistrer une vente.

Lors d'une vente, le client reçoit un ticket reprenant un numéro de ticket (auto-incrémenté), la date, le nom du caissier qui occupe la caisse et la liste des articles (avec pour chacun une brève description, la quantité vendue et le prix) et enfin le montant total du ticket et le type de paiement (liquide ou Bancontact). Vous trouverez des exemplaires de tickets en fin d'énoncé.

Chaque article a un code barre unique, une description et un prix unitaire.

A chaque vente, le stock de chaque article vendu sera décrémenté de la quantité vendue. Quand un caissier enregistre les articles d'un client à la caisse, rien ne garantit que les articles défilent dans un ordre quelconque. Dans cette « liste » le même article peut revenir plusieurs fois.

Certains clients ont une carte de fidélité. Ces cartes sont nominatives et seul le propriétaire peut l'utiliser. Lors de son passage à la caisse, le client montre sa carte, la caissière enregistre le numéro de la carte. Le montant total du ticket sera ajouté à un total des achats cumulés du client. Lorsque ce montant atteint 500 €, le client reçoit un bon de 5 €. On mémorise le nombre de bons distribués à chaque client (pour envoyer des publicités aux clients les plus fidèles). Tous les clients n'ont pas de carte (il faut la demander). Pour un client sans carte, on n'enregistre pas sa référence « client ».

Le but de l'application est de connaître rapidement le chiffre d'affaires global, le chiffre d'affaires par article, la situation du stock ou plus précisément les articles pour lesquels le stock est descendu sous le seuil de sécurité.

Examinez attentivement les tickets de caisse fournis à la page suivante.

On vous demande de :

1. réaliser le DSD (diagramme de structure des données) ;
2. écrire les requêtes donnant le chiffre d'affaires total et le chiffre d'affaires total par article ;
3. écrire la requête donnant le nombre et le montant des paiements en espèces ;
4. écrire les `CREATE TABLE`.

superette

No : 25896 Date : 25-03-2012 10 : 23
Caisse : 2 - Caissier : Jean (JB)
Bonjour Monsieur Gillard,

Référence Qté * P.U. (€) Montant

BIC crayon graphite HB N°2 2 x 10 pièces
1 * 3.65 = 3.65
ROSYPACK Milan gomme 8 pièces
1 * 1.15 = 1.15

Art : 2 TOTAL A PAYER : 4.80 €

Mode Paiement : Bancontact
Total fidélité après visite : 36€
Merci, à bientôt

superette

No : 25903 Date : 25-03-2012 10 : 23
Caisse : 1 - Caissier : Jean (JC)
Bonjour,

Référence Qté * P.U. (€) Montant

BIC crayon graphite HB N°2 2 x 10 pièces
1 * 3.65 = 3.65

Art : 1 TOTAL A PAYER : 3.65 €

Mode Paiement : Liquide

Merci, à bientôt

superette

No : 25956 Date : 26-03-2012 16 : 56
Caisse : 1 - Caissier : Jean (JB)

Bonjour Monsieur De Wolf,

Référence Qté * P.U. (€) Montant

farde 3 rabats 32 x 24 cm
10 * 3.45 = 34.5

Art : 10 TOTAL A PAYER : 34.5 €

Mode Paiement : Bancontact

Total fidélité après visite : 489€
Merci, à bientôt

superette

No : 25986 Date : 27-03-2012 14 : 23
Caisse : 3 - Caissier : Denis (DD)

Bonjour Madame Gherschon,

Référence Qté * P.U. (€) Montant

KANGOUROU crayons de couleur 12 pièces
1 * 1.99 = 1.99

Art : 1 TOTAL A PAYER : 1.99 €

Mode Paiement : Liquide

Total fidélité après visite : 1.99€
Merci, à bientôt

3.2 Gestion des kots

Le service logement de l'IPL désire informatiser les différentes possibilités de logement offertes aux étudiants de l'école. A cette fin, on fait appel à vos talents d'analyse pour le design des données. Le système permet d'enregistrer les différentes visites effectuées par les étudiants dans les kots proposés et de contracter éventuellement un bail entre le propriétaire du bien et l'étudiant.

Un kot est détaillé par sa superficie, son loyer, sa durée de contrat (10 ou 12 mois), sa mixité (féminin, masculin ou mixte), sa caution et son adresse ; cette adresse est retenue sous forme de simple chaîne de caractères.

On connaît également le propriétaire du kot. Un propriétaire peut posséder plusieurs kots en location. On connaît le nom, le prénom du propriétaire ainsi que ses coordonnées, c.-à-d. son adresse, son email et son numéro de téléphone.

Les étudiants peuvent visiter plusieurs kots. Un étudiant peut effectuer plusieurs visites du même kot. Toutes les visites sont enregistrées dans le système. Chaque visite est caractérisée par la date à laquelle elle a lieu. Il se peut que plusieurs visites d'un même kot soient planifiées à une même date. On connaît de l'étudiant son numéro d'inscription à l'école, son adresse, son email et son numéro de téléphone.

Suite à une visite, un bail peut être signé entre le propriétaire et l'étudiant. Ce bail référence la visite qui a débouché sur la signature. Dans ce bail, on indique la date de signature, la date de début du bail et la date de l'état des lieux d'entrée. La date de l'état des lieux de sortie sera complétée ultérieurement. On indique également le numéro du compte bloqué contenant la caution qui devra également être complété ultérieurement.

Par exemple, un propriétaire Emile Darbot possède 2 kots en location à la rue clos Chapelle aux Champs, 45 à 1200 Bruxelles.

- Le premier kot, 20 m², est loué 245€ par mois pour une durée de 10 mois mais uniquement à des filles. La caution est de deux mois.
- Le deuxième kot, 14 m², est loué 199€ par mois pour une durée de 12 mois mais uniquement à des garçons. La caution s'élève à 600€
- Un étudiant d'informatique Lucien Dumay visite le deuxième kot une première fois le 23 mars 2011. Une autre visite du même kot a eu lieu le 1 avril avec ses parents

Suite à cette deuxième visite, Emile Darbot et Lucien Dumay signent un bail le 1 avril. Ce bail débute le 1 septembre 2021. L'état des lieux d'entrée est fixé le 31 août. Le numéro du compte bloqué de la caution sera indiqué avant le 1 septembre.

On vous demande de :

1. réaliser le diagramme de structure de données
2. écrire les requêtes donnant le nombre de kots total et le nombre de kots par propriétaire
3. écrire la requête donnant la liste des kots n'ayant jamais été loués
4. écrire les `CREATE TABLE`

BONUS: Si un étudiant ne peut réaliser qu'une seule visite d'un kot donné, pourrait-on changer quelque chose dans votre DSD afin que le DBMS vérifie cette contrainte ? Expliquez en vous appuyant sur un exemple.

4 Séance 4

4.1 EuroPlastics

EuroPlastics fabrique des gadgets publicitaires (porte-clefs, petits jouets pour enfants...). Elle ne fabrique que sur commande. Elle n'a aucun stock de produits finis, excepté ce qui doit être livré aux clients.

Chaque client a un numéro, un nom et une adresse. Chaque article a une référence (qui permet de l'identifier), une description et un prix ; nous ne prenons pas en compte ses autres propriétés dans cet exercice.

L'atelier comprend plusieurs machines numérotées de 1 à 8. Pour chacune d'elles, on connaît sa marque et sa date d'achat. On doit pouvoir enregistrer la date de son dernier entretien. Chaque machine est spécialisée dans la fabrication d'un certain nombre d'articles du catalogue. Cependant un article est susceptible d'être fabriqué sur différentes machines mais le temps de fabrication varie selon la machine. Par exemple l'article de référence SMURF (« porte-clef Stroumpf ») est fabriqué soit sur la machine 1 en 30 secondes soit sur la machine 8 en 45 secondes.

Les clients passent des commandes. Chaque commande reprend les informations suivantes : date de la commande, date d'échéance (date promise de livraison), la référence du client. Une commande peut concerner plusieurs articles différents. Une même commande ne peut contenir deux lignes qui concernent le même article. Pour chaque article commandé, on enregistre la quantité désirée par le client.

Lorsque la commande a été enregistrée, il faudra affecter chaque ligne de cette commande dans le planning. Cette opération est effectuée après l'enregistrement de la commande, en ajoutant à la ligne de commande les informations suivantes : date et heure de fabrication, et n° de machine. Lors de l'affectation, il faut évidemment vérifier que la machine puisse fabriquer l'article précisé et que la plage horaire de la machine affectée à cet article soit libre.

Une ligne de commande sera lors de la commande dans l'état E(nregistrée) ; elle passera à l'état P(lanifiée) lorsque les informations du planning auront été remplies (date de fabrication, heure et n° de machine) ; elle passera à l'état "*Cloturée*" lorsqu'elle sera achevée.

Lorsque toutes les lignes d'une commande sont fabriquées, il faut faire passer la commande à l'état "*Réalisée*".

Pour aider la personne qui s'occupe du planning de la production, il faudrait lui fournir des listes (avec date et heure d'édition) : liste des lignes de commande non affectées au planning, liste d'occupation des machines, liste des commandes prêtes à être expédiées aux clients, etc.

Voici le dessin de la liste d'occupation des machines. Ne seront repris dans cette liste que les lignes à l'état P :

<u>Occupation des machines</u>							
Machine	N° Ligne COM	Reference Article	Nom-client	Échéance	Qté	Date-heure Début	Date-heure Fin
1	118	SMURF	Moreau	25/6/2003	20	12/5/03 12h00	12/5/03 12h17
	128	TIGRE	ESSO	28/6/2003	250	12/5/03 13h00	12/5/03 15h00
	Nombre d'articles 270						
2	117	SPIROU	Moreau	24/6/2003	30	13/5/03 12h00	13/5/03 12h55
	129	LION	TEXACO	26/6/2003	450	13/5/03 14h00	13/5/03 18h00
	Nombre d'articles 480						
3	116	HUITRE	OSTENDE	20/6/2003	20	12/5/03 12h00	12/5/03 12h25
	130	STYLO	NIAS	21/6/2003	250	12/5/03 13h00	12/5/03 15h00
	78	SPIROU	pax	18/5/2003	100	13/5/03 8H00	13/5/03 10h10
	Nombre d'articles 370						
4	115	PCLEF	BMW	25/6/2003	20	12/5/03 12h00	12/5/03 12h17
	134	PSTAR	MERCEDES	22/6/2003	250	12/5/03 13h00	12/5/03 15h00
	Nombre d'articles 270						
Nombre total d'articles					1.290		

Vous devez

1. Dessiner le diagramme de structure des données.
2. Ecrire le(s) `SELECT`(s) SQL nécessaire(s) pour établir la liste « Occupation des machines » (sans les sommes représentant le nombre d'articles et le nombre total d'articles).

4.2 Babyboom

Le magasin BBB (BaByBoom) souhaite compléter son application web en permettant aux futurs parents de créer une liste de naissance directement sur le site internet et de la gérer eux-mêmes. Cette application permettra également d'offrir des cadeaux appartenant à la liste.

Le magasin possède déjà une application dans laquelle les clients peuvent faire des achats en ligne. Une table des articles du magasin existe donc déjà et est définie comme suit :

articles	
PK	<u>reference</u>
	denomination NN prix NN

Dans l'application existante, les clients peuvent effectuer des achats en s'identifiant par le biais d'un login et d'un mot de passe. Le système comprend donc déjà la table CLIENTS telle que présentée ci-après :

clients	
PK	<u>login</u>
	nom NN prenom NN email NN adresse mdp NN

Dans la nouvelle application, un client pourra créer (et gérer) une liste de naissance pour son bébé ou alors offrir un cadeau sur une liste existante.

Une liste de naissance est une liste d'articles sélectionnés par les futurs parents. Il s'agit des cadeaux qu'ils souhaitent éventuellement recevoir à la naissance de leur bébé.

Lorsqu'un parent désire créer une liste, il doit s'identifier sur l'application et donc devenir client du magasin. On peut créer plusieurs listes de naissance mais une liste n'appartient qu'à un seul client.

Lorsque la liste est créée, on lui attribue un numéro identifiant ainsi que deux mots de passe ; celui pour la gestion de la liste et celui pour effectuer un cadeau sur celle-ci.

- Le client (parent) utilisera donc le numéro de la liste et le mot de passe de gestion pour gérer sa liste. Ce mot de passe est le **même** que celui de son compte client du magasin !
- Lorsqu'on désire offrir un cadeau sur une liste, on devra connaître le numéro de la liste ainsi que le mot de passe pour offrir un cadeau.

Le client (parent) reçoit d'ailleurs le courrier suivant à la création de la liste :

Cher client,

Félicitation pour l'heureux événement qui vous attend et merci d'avoir choisi BaByBoom pour votre liste de naissance !

Vos amis et votre famille peuvent dorénavant consulter votre liste de naissance sur Internet, y choisir un petit cadeau et le réserver.

Un mot de passe leur a été spécialement attribué à cet effet :

- Numéro de liste : 285909
- Leur mot de passe : HYG8956

Vous trouverez ci-dessous le numéro de la liste et le mot de passe vous permettant de gérer votre liste de naissance (pour voir l'évolution de la liste sur Internet, ajouter ou supprimer des articles, etc.) :

- Numéro de liste : 285909
- Le mot de passe pour gérer votre liste correspond à votre mot de passe client habituel.

Vous avez encore des questions ? N'hésitez pas à contacter la Ligne Conseil de BaByBoom au 02 383 52 85 (du lundi au vendredi pendant les heures de bureau et le samedi jusqu'à midi). Vous pouvez également envoyer un e-mail à ligne.conseil@bbb.be.

Nous vous répondrons dès que possible.

Cordiales salutations,

Dès qu'il reçoit ces informations, le client (parent) peut donc gérer sa liste.

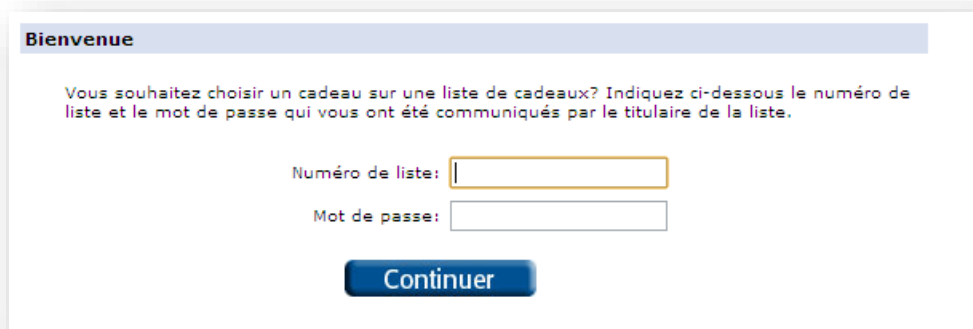
Voici un exemple de liste de naissance :

Référence	Dénomination	Quantité	Prix unitaire
906446	Maiko Thermomètre de bain	1	5,50
780696	Tetra gants de toilette de couleurs vives	1	7,95
906569	Bébé Jou Biberon Pooh Bear 125 ml	2	4,50
872018	Tomy Babyphone TA100 Classic Monitor	1	106,9
818838	Little Company Sac couch. Solo Sleeper 80 cm	1	39,95
856170	Textile Poussin Tour de lit	1	49,95
721039	Bavoir Jungle coton bio	5	6,29

Attention, dans la liste certains articles peuvent se trouver en quantité supérieure à 1. Dans l'exemple, le biberon "Pooh Bear" s'y trouve 2 fois et le bavoir "Jungle" 5 fois.

L'objectif de la nouvelle application est de permettre d'offrir des cadeaux sur une liste de naissance. Une personne qui désire offrir un cadeau peut être cliente du magasin mais pas nécessairement. L'inscription n'est pas obligatoire !

Si une personne souhaite faire un cadeau, elle doit se connecter sur le site : www.babyboom.be. Ensuite, elle doit encoder le numéro de la liste et le mot de passe adéquat (dans notre exemple num liste : 285909 et mot de passe : HYG8956) :



Bienvenue

Vous souhaitez choisir un cadeau sur une liste de cadeaux? Indiquez ci-dessous le numéro de liste et le mot de passe qui vous ont été communiqués par le titulaire de la liste.

Numéro de liste:

Mot de passe:

Continuer

Une fois connectée sur la liste, la personne visualise la liste et peut sélectionner les articles de son choix pour son cadeau. Evidemment dans la liste qu'il visualise, ne sont visibles que les articles qui n'ont pas encore été offerts.

Revenons sur la liste précédente (supposant que le baby phone ait déjà été offert ainsi que 2 bavoirs).

Référence	Dénomination	Prix unitaire	
906446	Maiko Thermomètre de bain	5,50	x
780696	Tetra gants de toilette de couleurs vives	7,95	
906569	Bébé Jou Biberon Pooh Bear 125 ml	4,50	
906569	Bébé Jou Biberon Pooh Bear 125 ml	4,50	
818838	Little Company Sac couch. Solo Sleeper 80 cm	39,95	
856170	Textile Poussin Tour de lit	49,95	
721039	Bavoir Jungle coton bio	6,29	x
721039	Bavoir Jungle coton bio	6,29	x
721039	Bavoir Jungle coton bio	6,29	

Remarquez que les articles présents en plus d'un exemplaire sont dupliqués autant de fois que nécessaire ; ceci uniquement par soucis de clarté de l'affichage !

La personne qui désire faire un cadeau cochera ce qui lui plait dans les lignes correspondantes. Dans l'exemple, elle a sélectionné deux bavoirs et le thermomètre. Elle effectue donc un cadeau contenant 3 articles : le « Maiko Thermomètre de bain » à 5,5€ et deux « Bavoir Jungle coton bio » à 6,29€.

Lorsque la personne arrête son choix, elle doit

- soit s'identifier car elle est déjà cliente du magasin.
- soit elle n'est pas cliente et doit donc remplir les informations nom et prénom.

Ensuite, elle peut laisser un message pour les futurs parents. Enfin, elle termine en effectuant le paiement en ligne de son cadeau.

Concrètement, elle complète le formulaire suivant :

Liste numéro 285909 :

Vous avez choisi d'offrir les articles :

- Maiko Thermomètre de bain 5,50€ * 1
- Bavoir Jungle coton bio 6,29€ * 2

Pour un total de **18,08€**

<p><u>Vous êtes déjà client ?</u></p> <p>Login : <input type="text"/></p> <p>Mot de passe : <input type="password"/></p>	<p><u>Vous n'êtes pas client ?</u></p> <p>Prénom et Nom : <input type="text"/></p>
--	--

Laissez un message aux parents :

L'application conservera également la date du cadeau.

Vous devez :

1. Réaliser le diagramme de structures de données.
2. Écrire le `SELECT` qui permet de lister les articles (référence + dénomination + prix unitaire) de la liste n° 285909 en précisant à chaque fois la quantité voulue.
3. Écrire le `SELECT` qui permet de lister les articles de la liste n° 285909 qui ont déjà offerts en précisant, pour chaque article, la quantité déjà offerte.

5 Séance 5

5.1 Le distributeur de livres et DVD.

IPL-Loisirs, un distributeur de livres et de DVDs, désire informatiser la gestion de ses membres et des achats réalisés par ceux-ci. Celui-ci fait appel à vos compétences d'analyste de données pour établir le design des données.

IPL-Loisirs offre la possibilité à ses membres d'acheter des nouveautés en livres et DVDs (dans des boutiques ou sur internet) à un prix moins cher que dans le commerce habituel et variant en fonction de l'ancienneté du membre. En contrepartie, un membre s'engage à effectuer au moins un achat par période. Dans le cas où un membre n'aurait pas effectué d'achat dans une période donnée, il devra obligatoirement acheter le livre sélectionné pour cette période, ce dernier lui étant automatiquement envoyé. Il est donc important de pouvoir retrouver les achats effectués par un membre dans une période donnée.

Lors de l'inscription, le nouveau membre doit fournir les informations suivantes : son nom, son prénom et son adresse. S'il s'inscrit par internet, il doit également fournir un login et un mot de passe. Une fois inscrit, un membre se voit attribuer un numéro d'identification.

Pour chaque produit (livre ou dvd) vendu par le distributeur, il faut retenir sa référence, son prix de base et son titre. Pour chaque livre, on connaît aussi son auteur, son nombre de pages et un résumé est parfois fourni. Pour un DVD, on connaît sa durée, son genre et l'année de sa sortie.

À chaque période correspond une date de début, une date de fin et un livre qui est la sélection liée à cette période. On ne peut pas sélectionner deux fois le même livre pour des périodes différentes.

Pour chaque achat, il faut retenir le membre qui l'a effectué, la période pendant laquelle il a été effectué, le détail des produits achetés et le type de paiement choisi parmi les suivants : en liquide, par Bancontact, par virement ou par carte de crédit. Une fois que le paiement aura été effectué, il faut aussi garder la date du paiement.

Achat effectué dans la période du 12/12/2016 au 11/03/2017

Membre : Albert Moreau

<i>Référence</i>	<i>Titre</i>	<i>Quantité</i>	<i>Prix unitaire</i>
848793	Temps glaciaire	1	19.90
879065	Cosmos	1	22.90

Total : 42.80

Type de paiement : Bancontact

Paiement effectué le 25/02/2017

Achat effectué dans la période du 12/12/2016 au 11/03/2017

Membre : Jean Dubois

<i>Référence</i>	<i>Titre</i>	<i>Quantité</i>	<i>Prix unitaire</i>
<i>150062</i>	<i>Samba</i>	<i>2</i>	<i>16.90</i>

Total : 33.80

Type de paiement : virement

Paielement non effectué

On vous demande :

- 1) De réaliser le diagramme de structures de données.
- 2) D'écrire les requêtes suivantes :
 - a. Donner la liste des livres (référence, titre, auteur) ayant déjà été sélectionnés pour une période.
 - b. Donner, pour chaque achat, le nom et le numéro du membre l'ayant effectué, la période pendant laquelle il a été effectué, la date du paiement et le montant total de l'achat. Classez le tout par période et, pour les achats effectués lors d'une même période, classez-les par numéro de client.
 - c. Donner la liste des membres (numéro + nom) n'ayant effectué aucun achat lors de la période du 12/12/2016 au 11/03/2017.

5.2 Bullox

Une société de meubles en kit, Bullox*, vous contacte afin d'analyser le processus de **création de meubles**.

Il existe déjà un système de commande et de facturation, on ne vous demande pas de développer cela.

Cette société se trouve dans un grand entrepôt dans lequel se trouvent actuellement 4 lignes d'assemblage ainsi que le magasin. Les clients peuvent observer les ouvriers qui assemblent les commandes par la grande fenêtre qui sépare l'atelier du magasin.

Lorsqu'un client arrive au magasin, il est accueilli par un vendeur (il y en a deux pour le moment). Le vendeur lui explique le principe de création des meubles et lui propose ensuite de créer des meubles sur mesure.

Si le client n'existe pas déjà dans le système, le vendeur lui demande les informations suivantes : son nom, son prénom, sa date de naissance, son adresse, son tel et son email. Les deux dernières informations sont facultatives ; les autres sont obligatoires.

Enfin, en discutant, le client et le vendeur constituent les meubles. Un meuble se compose de diverses pièces : des traverses, des casiers, des tiroirs, ... chacune de ses pièces peut composer des meubles différents.

Par exemple, un meuble de 4 casiers de 90 cm de large et de 37 cm de haut en MDF naturel (code CS 4C-9Z/G) se décompose en 5 tablettes en MDF naturel de 7mm, 10 traverses frontales, 10 traverses latérales, etc.

Ensuite, on peut ajouter des composantes (ex: tringles, attache pour tringle etc.) et en enlever (ex: -1 tablette pour faire une longue penderie avec la tringle)

On connaît le nombre de pièces restant en stock ainsi que le seuil de recommandation de chacune d'elles. Une pièce possède aussi un code, un descriptif et un matériau (soit inox, soit bois, soit mélaminé).

La vérification de la bonne composition du meuble (bon nombre de traverses, casiers, etc.) est vérifiée par programmation et non dans la base de données.

Le calcul du prix de chaque meuble est une somme des pièces qui le composent au moment où le meuble est créé. Les pièces sont sujettes à des fluctuations de prix.

Lorsqu'un nouveau meuble est composé et correspond à ce que souhaite le client, le vendeur sauvegarde le meuble dans la base de données.

On doit retenir quand le meuble a été créé, quel vendeur l'a créé, pour quel client et un code identifiant.

* Largement inspiré de KewLox (voir <http://www.kewlox.be>).

Exemple :

Meuble 1 :

Quantité	Code	Description
8	C52-H37	Planche coté MDF
4	PH-9Z	Tablette en MDF Naturel 7 mm
4	F90-H37	Planche fond MDF
8	T90	Traverse frontale
10	T37	Traverse latérale
1	T9	Tringle de 868mm
2	HSU-Z	Support tringle MDF naturel
2	AT	Attache pour tringle
4	SUM	Ferrure pour pann. Supp. Tring
1	TI-9ZC	Tablette intermédiaire blanche + support

On connaît le code, le nom, le prénom, l'adresse, le tel et l'email de chaque employé. Un employé est soit un vendeur soit un ouvrier. On connaît pour chaque ouvrier son nombre d'années d'ancienneté et pour chaque vendeur, son chiffre d'affaires annuel moyen.

On vous demande de :

1. Réaliser le diagramme de structures de données.
2. Ecrire la requête qui donne, pour chaque vendeur, le nombre total de meubles qu'il a sauvegardés. Classez-les par ordre décroissant du nombre total de meubles.
3. Ecrire la requête qui donne la liste des pièces qui n'entrent dans aucune composition de meuble.

6 Séance 6



6.1 Grantano



Une chaîne de magasins de chaussures souhaite informatiser son business afin de réaliser une application en ligne de vente de chaussures. Cette application a pour but de gérer les stocks de chaussures ainsi que les commandes.

Tout d'abord les chaussures ont été chacune catégorisées. Les catégories ont été établies mais peuvent encore évoluer. Ci contre, vous trouvez un premier tableau (résumé) des catégories. Chaque catégorie peut être sous-divisée en sous-catégories.

Dans la catégories *Hommes – Tongs* on trouve par exemple les paires de chaussures suivantes :

- TONG SUPERBE DE LE COQ SPORTIF 
- TONG SUPERBE DE GAASTRA 

Une paire de chaussures appartient à une seule catégorie.

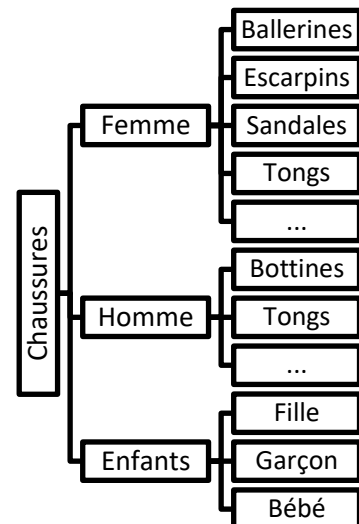
Une paire de chaussures possède un libellé comme celui mentionné ci-dessus *TONG SUPERBE DE LE COQ SPORTIF*. Il se peut que plusieurs paires de chaussures portent le même libellé.

Un descriptif permet, si nécessaire, d'en savoir plus sur la chaussure.

La paire de chaussures possède également un prix et une couleur. On enregistre si la paire de chaussures est noire N, brune BR, bleue BE, blanche BL ou colorée CO. Parfois ces prix changent, par exemple en été les bottines sont moins chères et les tongs plus chères.

Il s'agit de gérer des commandes en ligne de chaussures. Les commandes sont effectuées par des clients pour lesquels on ne retient que le nom (inutile de créer une table `clients`). On retient pour chaque commande la date à laquelle elle est passée ainsi qu'un numéro unique.

Dans une commande, on peut retrouver plusieurs paires de chaussures. Par exemple, une paire de Tong pour Monsieur et une paire d'escarpin pour Madame.



	Pointure	Prix unitaire	Qté	Total	Supprimer l'article
	42 ▼	49.99	1 ▴ ▾	49.99	Supprimer l'article
	37 ▼	69.99	1 ▴ ▾	69.99	Supprimer l'article

Remarquez que lorsqu'un client passe commande, il doit mentionner les pointures de chaussures désirées. On vous demande de gérer le stock des paires de chaussures disponibles. Parfois, il arrive qu'une paire de chaussures soit indisponible dans une certaine pointure. Un code barre unique identifie chaque pointure d'une paire de chaussures.

Un client peut également décider d'acheter 2 paires d'*ESCARPIN LINEA LORESI* en pointure 38 :

	Pointure	Prix unitaire	Qté	Total	Supprimer l'article
	42 ▼	49.99	1 ▲▼	49.99	Supprimer l'article
	38 ▼	69.99	2 ▲▼	139.98	Supprimer l'article

L'application doit permettre d'avoir une vision claire du stock de chaussures !

On ne connaît pas à l'avance toutes les pointures. C'est-à-dire qu'on doit pouvoir ajouter des pointures par exemple 38,5 etc.

On vous demande de :

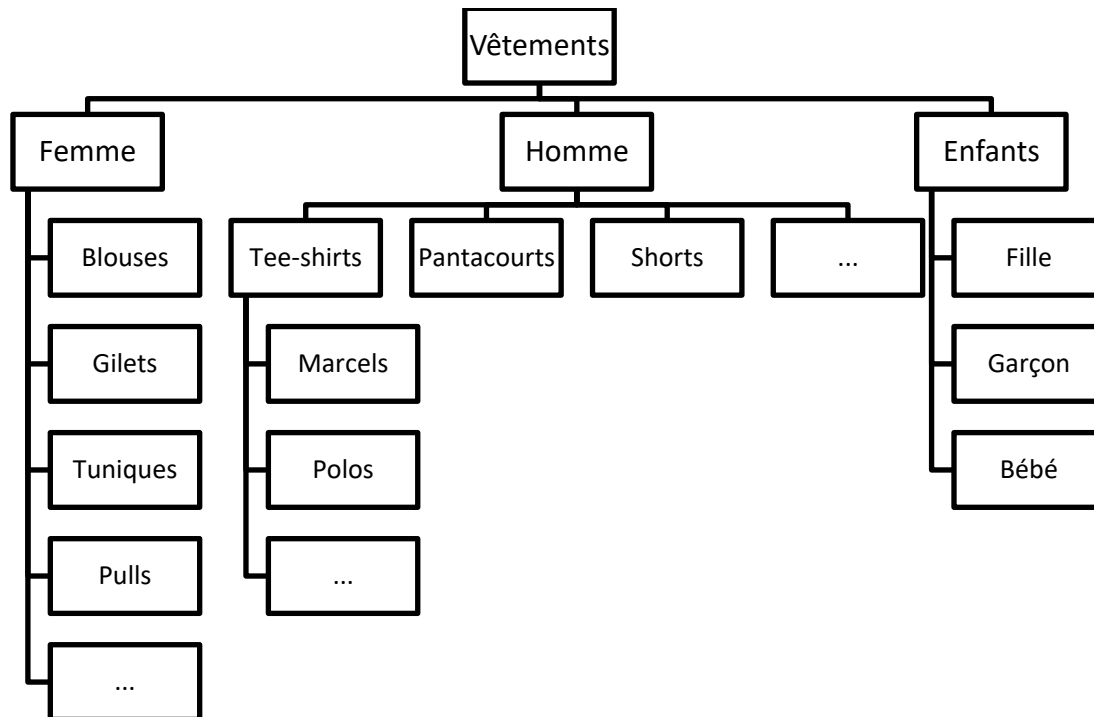
1. Réaliser le diagramme de structure de données.
2. Ecrire la requête qui permet de lister les chaussures (identifiant + libelle + descriptif) qui ne sont pas disponibles (remarque : cela doit prendre les chaussures qui ne sont pas repris dans stocks ainsi que les chaussures pour lesquelles les quantités en stock de chaque pointure sont à zéro).
3. Ecrire la requête qui donne, pour chaque couple de catégories ayant le même intitulé, leurs « identifiant » et le libellé.
4. Ecrire la requête qui permet de trouver la liste des chaussures qui ont déjà été vendues à un prix supérieur au prix actuel.

6.2 Les 3 Cuisses



Une chaîne de magasins de prêt-à-porter souhaite informatiser son business afin de réaliser une application en ligne de vente. Cette application a pour but de gérer les stocks ainsi que les commandes.

Les vêtements sont répertoriés dans différentes catégories :



Les vêtements sont répartis dans des catégories qui ont déjà été établies mais qui peuvent encore évoluer. Chaque catégorie peut être sous-divisée en sous-catégories.

Dans la catégories *Femme – Blouses* on trouve par exemple :



Blouse Manches courtes – 24,99€



Blouse Manches courtes qui joue la transparence – 29,99€

Un vêtement appartient à une seule catégorie.



Un vêtement possède un titre comme celui mentionné ci-dessus *Blouse Manches courtes*. Il se peut que plusieurs vêtements portent le même titre. Par contre, un vêtement possède une référence unique de 7 chiffres, peu importe la taille.

Un descriptif permet d'en savoir plus sur le vêtement ainsi qu'une information sur la matière. Pour cette dernière, on se contente de dire si le vêtement est en coton CO, en viscose VI, en lin LI, en laine LA ou en synthétique SY.

On précisera également le prix de chaque vêtement. Ces prix fluctuent, par exemple en été les pulls en laine sont moins chers et les marcelles plus chers.



Il s'agit de gérer des commandes en ligne. Les commandes sont effectuées par des clients pour lesquels on ne retient que le nom (inutile de créer une table `clients`). On retient pour chaque commande la date à laquelle elle est passée ainsi qu'un numéro unique.

Dans une commande, on peut retrouver plusieurs vêtements. Par exemple, une blouse manches courtes et une autre qui joue la transparence.

Blouse manches courtes l		€ 24,99
	Réf.: 9675132 Coloris: BLANC Taille 48 Disponibile	Quantité : 1 PRIX ROUGE <input type="button" value="Modifier"/> <input type="button" value="Supprimer"/>
Blouse manches courtes l		€ 29,99
	Réf.: 8866490 Coloris: BICOLORE CORAIL/ROUGE Taille 38 Disponibile	Quantité : 1 PRIX ROUGE <input type="button" value="Modifier"/> <input type="button" value="Supprimer"/>

Remarquez que lorsqu'un client passe commande, il doit mentionner les tailles des vêtements qu'il achète. On vous demande de gérer le stock de vêtements disponibles. Parfois, il arrive qu'une blouse par exemple soit indisponible dans une certaine taille.

Un client peut également décider d'acheter 3 blouses manches courtes en taille 38 :

Blouse manches courtes		€ 24,99
	Réf.: 9675132 Coloris: BLANC Taille 48 Disponibile	Quantité : 1 PRIX ROUGE <input type="button" value="Modifier"/> <input type="button" value="Supprimer"/>
Blouse manches courtes l		€ 89,97
	Réf.: 8866490 Coloris: BICOLORE CORAIL/ROUGE Taille 38 Disponibile	Quantité : 3 PRIX ROUGE <input type="button" value="Modifier"/> <input type="button" value="Supprimer"/>

L'application doit permettre d'avoir une vision claire du stock de vêtements !

On ne connaît pas à l'avance toutes les tailles. C'est-à-dire qu'on doit pouvoir ajouter des tailles par exemple 41 etc.

On vous demande de :

1. Réaliser le diagramme de structure de données.
2. Ecrire la requête qui permet de lister les vêtements (référence + titre + descriptif) qui ne sont pas disponibles (remarque : cela doit prendre les vêtements qui ne sont pas repris dans stocks ainsi que les vêtements pour lesquels les quantités en stock de chaque taille sont à zéro).
3. Ecrire la requête qui donne, pour chaque couple de catégories ayant le même libellé, leurs identifiants et le libellé.
4. Ecrire la requête qui permet de trouver la liste des vêtements qui ont déjà été vendus à un prix supérieur au prix actuel.

7 Séance 7

7.1 Courtier d'assurances

Un courtier d'assurances souhaite informatiser la gestion des contrats d'assurance-auto. Ce courtier offre quatre types de contrat d'assurance-auto : responsabilité civile, omnium partielle, omnium totale ou bris de vitre. (Il n'y a pas d'autre type possible).

Un véhicule peut être couvert par un ou plusieurs contrats. Mais un véhicule doit être couvert au minimum par un contrat. En effet, la loi belge impose que tout véhicule soit assuré par un contrat en responsabilité civile.

Le preneur d'assurances qui souhaite assurer un véhicule doit donc signer au minimum un contrat et peut en signer plusieurs pour ce même véhicule. Un contrat ne couvre qu'un seul véhicule.

Le courtier travaille avec presque toutes les compagnies d'assurance qui exercent sur le territoire belge. Pour chaque compagnie, il connaît son code en 4 caractères, son nom, son adresse, la personne de contact et le numéro de téléphone de celle-ci ainsi que le numéro de téléphone central.

Pour préparer un contrat, le courtier doit connaître le preneur d'assurance, le véhicule assuré, la compagnie d'assurances choisie, la durée du contrat, la date de prise d'effet du contrat. Le courtier demande au preneur d'assurances, ses nom, prénom, adresse, numéro de téléphone ou de GSM et e-mail. Pour le véhicule assuré, le courtier demande le numéro de châssis, la plaque d'immatriculation, la marque et le modèle ainsi que l'année de mise en circulation et le kilométrage.

Lorsque toutes ces informations sont connues, il calcule avec précision le montant du contrat en fonction de la périodicité de paiement définie par le client : par mois, par trimestre, par semestre ou par an. (Il n'y a pas d'autre périodicité possible).

Tout contrat de type omnium partielle, omnium totale ou bris de vitre fait référence au contrat en responsabilité civile du véhicule.

Lorsque le preneur d'assurance signe le contrat, on inscrit également la date de signature dans le contrat.

Lorsque le véhicule a un accident, le courtier enregistre le lieu, la date et l'heure de l'accident et le numéro de la déclaration d'accident. Il enregistre également s'il y a ou non des blessés et combien.

Ensuite, on évaluera si le véhicule peut être réparé ; si le véhicule peut être réparé, on évaluera les frais de réparation, sinon, le montant que l'assurance remboursera en compensation de la perte totale du véhicule. Le courtier évaluera enfin quel est ou quels sont les contrats qui peuvent couvrir le sinistre (= l'accident).

On vous demande de :

1. Réaliser le diagramme de structures de données.
2. Fournir un exemple de contenu des tables.
3. Ecrire les `CREATE TABLE` des tables contenant des clés étrangères.
4. Ecrire la requête qui donne le nombre d'accidents par preneur d'assurance.

7.2 Bouffe Kot

Un groupe de huit étudiants habitant ensemble dans un kot communautaire ont mis au point le système suivant pour gérer les repas pris en commun.

Ils mangent ensemble les quatre premiers jours de la semaine, du lundi au jeudi inclus, pendant les semaines de cours et à l'exception des jours fériés. Chaque soir, deux d'entre eux préparent le repas et font les courses pour celui-ci.

De retour au kot, souvent le dimanche soir, chaque étudiant s'inscrit pour un soir où il fera la cuisine. Il doit y avoir deux étudiants par soirée. Exceptionnellement, un étudiant peut cuisiner plusieurs fois sur la semaine ; si un autre est malade ou absent, par exemple.

Tous les étudiants indiquent également les jours où ils mangeront et le nombre de personnes qu'ils invitent ce jour-là. Ils doivent s'inscrire au plus tard avant 8h30 le jour du repas.

Les étudiants en charge du repas du soir comptent le nombre de personnes inscrites et font les courses pour préparer un repas pour ce nombre de personnes. Ils inscrivent alors le menu et le montant total des courses effectuées sur le tableau.

Vous trouverez, à la page suivante, un exemple du tableau rempli pour une semaine.

Les étudiants souhaitent informatiser le système mis en place et garder une trace de tout ce qui se passera chaque semaine pour gérer leur compte.

Il faut permettre d'ajouter un commentaire sur le menu (excellent, trop salé...).

On vous demande de :

1. Réaliser le diagramme de structures de données.
2. Fournir un exemple de contenu des tables.
3. Ecrire les `CREATE TABLE`.
4. Ecrire la requête qui donne le nombre de convives par jour.

Semaine n° 4	Lundi 13 septembre	Mardi 14 septembre	Mercredi 15 septembre	Jeudi 16 septembre
Préparent le repas :	Luc Martine	Isa Éric	Romain Marc	Chantal Annick
Une croix = je mange + nombre d'invités				
Luc	x +2	x	x	x
Isa	x	x	x +2	x
Martine	x		x +1	x
Chantal		x		x
Annick	x	x	x	x
Romain	x +3		x	x
Eric	x	x		x
Marc	x		x	x
Nbre total de convives	12	5	9	8
Menu	Lasagne aux épinards	Oiseaux sans tête, pdt, haricots verts	Spaghetti bolo	Steak cordon-bleu ; riz ; salade composée

Lundi, Luc et Martine prépareront le repas ; il y aura 12 convives à table. Une lasagne aux épinards est au menu.

Mardi, Isa et Éric prépareront le repas pour 5 convives

Etc.

8 Séance 8

8.1 RecetteBook

L'objectif de cette application est de permettre le partage de recettes entre amis et la gestion de commentaires à propos de ces recettes. L'idée émane d'un groupe d'amis qui, à court d'idées certains soirs de la semaine, voudrait disposer d'un grand nombre de recettes dans lequel ils pourraient choisir celle qui leur conviendrait ce soir-là.

Pour chacun des amis, il serait nécessaire de connaître son nom, son prénom et son e-mail. Chaque e-mail doit être unique et il est nécessaire de pouvoir le changer facilement.

Pour chaque recette, le créateur, l'un des amis, introduirait le nom de celle-ci, une description de sa préparation, un temps de préparation et un temps de cuisson ainsi que le nombre de personnes pour lesquelles la recette est prévue. Cette dernière information est facultative (si non présente on considère que la recette est prévue pour 4 personnes).

Il est important de garder la trace du créateur de la recette.

Chaque recette comprend la liste des ingrédients nécessaires et leur quantité. De plus, une recette peut être composée de plusieurs autres recettes.

Par exemple, pour faire de la *sauce Béchamel*, il faut les ingrédients suivants :

- 50 grammes de beurre
- 50 grammes de farine
- 1/2 litre de lait tiède
- 1 pincée de sel
- ½ pincée de poivre
- Noix de muscade (facultative)

Pour chaque ingrédient composant la recette, il est important de noter la quantité et l'unité dans laquelle elle est exprimée.

Une recette peut également être composée de plusieurs autres recettes.

Par exemple, pour faire de la *Lasagne pour 8 personnes*, il faut pouvoir utiliser les recettes suivantes :

- Recette de la sauce Béchamel – 2x
- Recette de la sauce Bolognaise – 1x
- Recette de pâtes fraîches – 1kg
- 200 gr de gruyère râpé

Autre exemple de dessert pour 4 personnes :

- Recette de petits choux – 1x
- Recette de crème pralinée – 200gr
- Recette de craquelin croustillant – 1x

Pour chaque recette composant la recette principale, il faut noter la quantité nécessaire.

Enfin, chaque ami peut ajouter un (**et un seul**) commentaire à une recette.

On vous demande de :

1. Réaliser le diagramme de structures de données.
2. Ecrire la requête qui donne la liste des recettes pour 4 personnes ayant un temps total (préparation + cuisson) inférieur à 60 minutes et utilisant des asperges comme ingrédient. Pour chaque recette, donnez son identifiant, son nom, la préparation et le temps total
3. Ecrire la requête qui donne la liste des amis qui ont publiés plusieurs recettes. Pour chaque ami, donnez son identifiant, son nom et son prénom ainsi que le nombre de recettes qu'il a publiées.
4. Ecrire la requête qui liste les recettes (identifiant, nom) qui n'ont aucune autre recette dans leur composition.

8.2 Lip dub mania

lipdub mania

Un lipdub est un clip vidéo réalisé en playback destiné à être diffusé sur internet. Véritable phénomène qui éclot avec cette vidéo de 2 étudiants de l'UQAM ([Université du Québec à Montréal](#)) ont créé. Ces vidéos cherchent généralement à témoigner de la créativité et de la bonne humeur qui règnent dans une équipe ou une entreprise. On réalise parfois aussi des lipdub pour un mariage, un anniversaire, ...



La société **lipdub mania** souhaite investir dans un système de gestion de ses clips vidéo. Elle requiert votre expertise d'analyste afin d'évaluer les données de son futur système.

Le système doit permettre de conserver les vidéos ainsi que le propriétaire de celle-ci. Chaque personne est identifiée par un email généré par la société **lipdub mania** qui permettra de les identifier. Une personne est également décrite par un nom, un prénom et possède un mot de passe pour se connecter au système.

Une vidéo possède un code identifiant unique appelé un ISANN (International Standard Audiovisual Number). On connaît également le type de vidéo (parmi AVI, MOV, MP4, DIVX, WMV), une description, une durée en minutes ainsi que l'url qui permet l'accès à la vidéo en question.

Une vidéo est souvent un montage d'autres vidéos qui elles-mêmes peuvent être des montages d'autres vidéos et ainsi de suite.

Par exemple, la vidéo « Le joyeux annif de Biquette » contient en fait 3 vidéos : « la naissance de Biquette », « les rires de Mr Bouc » et « le souffle des bougies ». La vidéo « les rires de Mr Bouc » contient elle-même 2 vidéos : « le Bouc émissaire » et « le Bouc 1 ».

Il est important de retenir le séquençement des vidéos qui contiennent d'autres vidéos. Pour la vidéo « Le joyeux annif de Biquette », il faut donc conserver l'ordre des vidéos incluses :

1. « la naissance de Biquette »
2. « les rires de Mr Bouc »
3. « le souffle des bougies »

Idem pour « les rires de Mr Bouc » qui contient :

1. « le Bouc émissaire »
2. « le Bouc 1 »

Une vidéo peut contenir plusieurs autres vidéos mais elle contient une seule fois chaque autre vidéo. Par exemple, il ne peut y avoir 2 fois « le Bouc 1 » dans la vidéo « les rires de Mr Bouc ». Mais « le Bouc 1 » peut se trouver dans plusieurs vidéos ; par exemple « les aventures du Bouc » qui contient :

1. « le Bouc 1 »
2. « le Bouc 2 »
3. « le Bouc 3 »

Ensuite, les personnes peuvent émettre des commentaires sur les vidéos ; une personne peut faire plusieurs commentaires sur une vidéo donnée. On retient la date ainsi que le texte du commentaire.

Finalement, une personne peut signifier si elle aime un commentaire (comme le « like » de Facebook). Une personne peut aimer une seule fois un commentaire. Il n'y a aucune information à retenir sur le like.

On vous demande de :

1. Réaliser le diagramme de structures de données.
2. Ecrire la requête qui donne la liste des personnes qui ont postés plusieurs vidéos. Pour chaque personne, donnez son email, son nom et son prénom ainsi que le nombre de vidéos qu'il a postées.
3. Ecrire la requête qui donne, pour chaque commentaire, le nom de la personne ayant fait le commentaire, l'ISANN et la description de la vidéo concernée, le texte et la date du commentaire. Classez le tout par ordre croissant d'ISANN et, en cas d'égalité, par ordre décroissant de date.
4. Ecrire la requête qui liste les vidéos (ISANN, description) qui ne contiennent aucune autre vidéo.