

## Conception exercices

### Exercice 2 : Les invités de Caroline

Caroline a un cercle d'amis très étendu. Elle aime cuisiner pour ses amis. Elle les invite régulièrement à manger.

Au fil du temps, elle se pose plusieurs questions liées à sa mémoire, parfois défaillante ; des questions comme celles-ci :

- N'ai-je pas déjà réalisé cette recette pour Michel et Pascale ? A moins que ce ne soit pour Roland et Michèle ?
- Alain n'aime pas certains ingrédients mais je ne m'en souviens plus ; gingembre ou betterave rouge ?
- Pierre a été invité en même temps que Marie et Hélène ; il ne supporte pas la présence d'une des deux mais laquelle ?

Bref elle voudrait pouvoir prendre des notes structurées après chaque soirée :

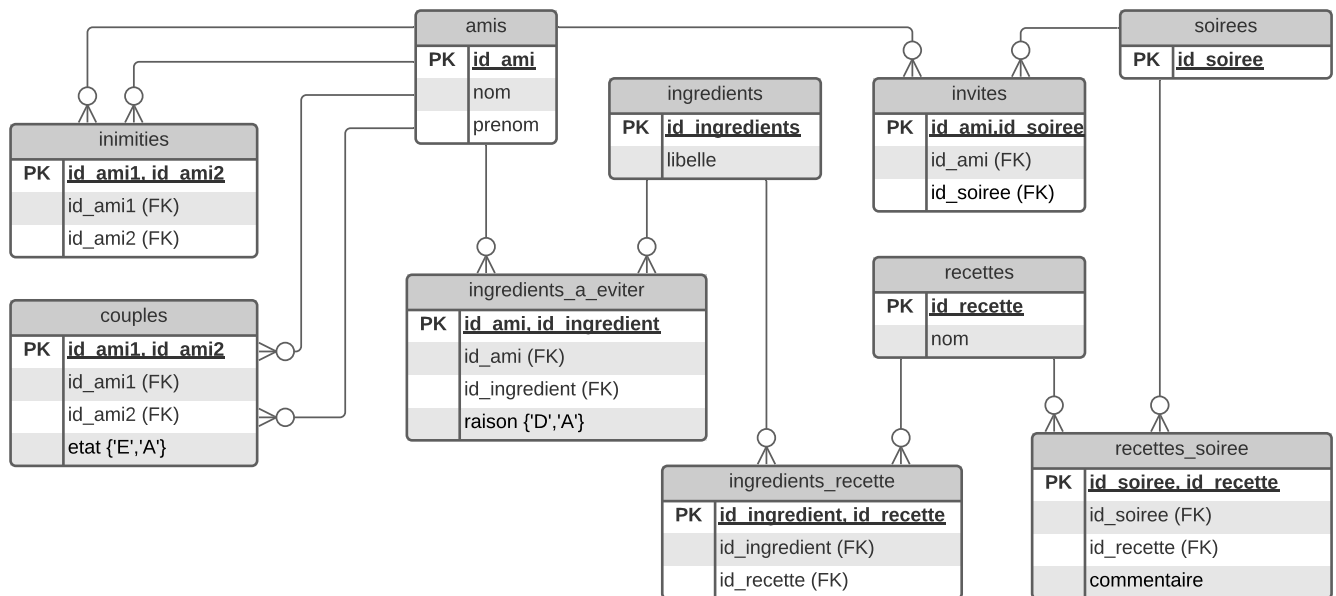
- Qui était présent ?
- Ce qu'elle a proposé comme recettes
- Un commentaire sur ces recettes (« wouaw » unanime ou un poli « c'était bon »)
- Les allergies ou carrément le dégoût de certains invités pour certains ingrédients
- Les couples (établis ou anciens, il vaut mieux éviter de « recomposer » ces derniers) et
- Les inimitiés déclarées.

On vous demande de

1. lister les entités et leurs **identifiants**; en donner une brève définition si nécessaire.
2. lister et **nommer** les associations ainsi que leurs cardinalités. Ceci peut être réalisé directement sur le DSD.
3. tracer le DSD correspondant.

Remarque importante : il n'y a pas ou peu de propriétés définies dans cet énoncé. Ajoutez simplement un nom et/ou une description quand cela est nécessaire.

amis : <ul style="list-style-type: none"><li>▪ id_ami</li><li>▪ nom</li><li>▪ prenom</li></ul>	recettes : <ul style="list-style-type: none"><li>▪ id_recette</li><li>▪ nom</li></ul>
initimies : <ul style="list-style-type: none"><li>▪ id_ami1 (FK)</li><li>▪ id_ami2 (FK)</li></ul>	ingrédients : <ul style="list-style-type: none"><li>▪ id_ingredient</li><li>▪ libelle</li></ul>
couples : <ul style="list-style-type: none"><li>▪ id_ami1 (FK)</li><li>▪ id_ami2 (FK)</li><li>▪ etat {'E','A'}</li></ul>	ingrédients_a_eviter : <ul style="list-style-type: none"><li>▪ id_ami (FK)</li><li>▪ id_ingredient (FK)</li><li>▪ raison{'A','D'}</li></ul>
soirees : <ul style="list-style-type: none"><li>▪ id_soiree</li></ul>	ingrédients_recette : <ul style="list-style-type: none"><li>▪ id_ingredient (FK)</li><li>▪ id_recette (FK)</li></ul>
invites : <ul style="list-style-type: none"><li>▪ id_soiree (FK)</li><li>▪ id_ami (FK)</li></ul>	recette_soiree : <ul style="list-style-type: none"><li>▪ id_soiree (FK)</li><li>▪ id_recette (FK)</li><li>▪ commentaire</li></ul>



### Exercice 3 : La gestion des repas en 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.

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.

Lorsqu'ils ont fait les courses, ils indiquent le montant dépensé sur le tableau et le nom de la personne qui a déboursé ce montant. Il peut arriver que celui qui paye les courses ne soit pas un de ceux qui préparent le repas.

À la fin de la semaine, l'étudiant en charge de la comptabilité des repas, compte le montant dépensé par chacun des étudiants et met son solde à jour. Le solde est reporté comme solde de début de semaine de la semaine suivante.

Le **solde** d'un étudiant est le montant que la collectivité doit lui rembourser. Lorsque le solde est positif, cela veut dire que l'étudiant a trop payé par rapport aux dépenses effectuées et inversement lorsqu'il est négatif.

Les étudiants doivent essayer d'avoir un solde proche de zéro.

Vous trouverez, à la page suivante, un exemple du tableau rempli pour une semaine et quelques précisions pour vous aider à le comprendre.

Les étudiants souhaitent informatiser le système mis en place. Ils veulent également garder une trace de tous les menus qui seront réalisés.

Il faut permettre à chaque étudiant d'ajouter un commentaire sur le menu de chaque jour (excellent, trop salé...).

Solde de début de semaine	Semaine n° 4	Lundi 13 septembre	Mardi 14 septembre	Mercredi 15 septembre	Jeudi 16 septembre	Déboursé cette semaine	Mangé cette semaine	Solde de fin de semaine
	Préparent le repas :	Luc Martine	Isa Eric	Romain Marc	Chantal Annick			
	Une croix = je mange + nombre d'invités							
5,15 €	Luc	x +2	x	x	x	73,56 €	40,76 €	37,95 €
9,03 €	Isa	x	x		x		24,29 €	-15,26 €
-5,60 €	Martine	x		x +1	x		24,24 €	-29,84 €
7,75 €	Chantal		x		x		18,16 €	-10,41 €
56,89 €	Annick	x	x	x	x		28,50 €	28,39 €
-65,12 €	Romain	x +3		x+1	x	107,02 €	42,63 €	-0,73 €
5,75 €	Eric	x			x	42,35 €	15,82 €	32,28 €
1,12 €	Marc	x	x	x	x		28,50 €	-27,38 €
	Nbre total convives	12	5	7	8			
	Menu	Lasagne aux épinards	Omelette sans tête, pâtes, haricots verts	Spaghetti bolognese	Steak cordon-bleu ; riz ; salade composée			
	Dépenses	73,56 €	42,35 €	29,50 €	77,52 €	222,93 €	222,90 €	
	Payées par	Luc	Eric	Romain	Romain			
	Prix par convive	6,13 €	8,47 €	4,21 €	9,69 €			

Solde de début de semaine : Annick a beaucoup trop payé et Romain pas assez. Annick sera invitée à ne pas payer de courses tandis que Romain devrait déboursé cette semaine.

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

Luc fait les courses et dépense 73,56 € pour le repas.

Mardi, Isa et Éric prépareront le repas pour 5 convives. Ils vont cuisiner des oiseaux sans tête avec des pommes de terre et des haricots verts.

Éric fait les courses et dépense 42,35 € pour le menu. ...

En fin de semaine, l'étudiant-comptable calcule le prix par convive du repas. Par exemple,

- Lundi, le repas a coûté 73,56 € pour 12 convives.
- Le prix par convive est de 6,13 €.

Il calcule ensuite le solde de chacun:

Par exemple,

- Luc a déboursé 73,56 €
- Luc doit payer la nourriture pour 3 personnes lundi : soit  $6,13 \text{ €} \times 3 = 18,39 \text{ €}$
- Luc doit payer pour 1 personne (lui-même) mardi : 8,47 € ...
- Soit :

Lundi	6,13	3	18,39 €
Mardi	8,47	1	8,47 €
Mercredi	4,21	1	4,21 €
Jeudi	9,69	1	9,69 €
Total			40,76 €

- Il aura donc un solde positif en début de semaine prochaine de :  $73,56 \text{ €} - 40,76 \text{ €} + 5,15 \text{ €} = 37,95 \text{ €}$

On vous demande de réaliser le diagramme de la base de données correspondant à l'énoncé, en exploitant le tableau qui vous est fourni et en suivant les étapes ci-dessous :

1. lister les entités & en donner une brève définition si nécessaire.
2. lister leurs propriétés et les décrire, si nécessaire.
3. choisir un identifiant par entité.
4. lister et nommer les associations. Ceci peut être réalisé directement sur le DSD.
5. tracer le DSD correspondant.

etudiant : <ul style="list-style-type: none"><li>▪ id_etudiant</li><li>▪ nom</li></ul>	menus : <ul style="list-style-type: none"><li>▪ id_menu</li><li>▪ intitule</li></ul>
repas : <ul style="list-style-type: none"><li>▪ repas_id</li><li>▪ cuisinier1 (FK)</li><li>▪ cuisinier2 (FK)</li><li>▪ menu (FK) NULL</li><li>▪ depenses NULL</li><li>▪ semaine (FK)</li><li>▪ jour_semaine {'L','MA','ME','J'}</li><li>▪ date</li><li>▪ nbr_total_convive NULL</li><li>▪ prix_par_convive NULL</li><li>▪ etud_ayant-payé (FK) NULL</li></ul>	<div>solde_par_etud_par_sem :<ul style="list-style-type: none"><li>▪ id_etudiant (FK)</li><li>▪ id_semaine (FK)</li><li>▪ solde_debut_semaine</li><li>▪ solde_fin_semaine NULL</li><li>▪ montant_rebourse NULL</li></ul></div> <div>semaines :<ul style="list-style-type: none"><li>▪ id_semaine</li><li>▪ annee</li><li>▪ numero_semaine</li></ul></div>
commentaires : <ul style="list-style-type: none"><li>▪ id_etudiant (FK)</li><li>▪ id_repas (FK)</li><li>▪ commentaire</li></ul>	Inscriptions_etud_repas: <ul style="list-style-type: none"><li>▪ id_etudiant (FK)</li><li>▪ id_repas (FK)</li><li>▪ nbr_invite</li></ul>

## Exercice 5 : TicTaque, gestion du personnel

Tracez le DSD correspondant à l'énoncé suivant.

Je dois comprendre, à la seule lecture du DSD, les entités, leurs propriétés et les associations : donnez un nom aux associations et placez-le sur celles-ci.

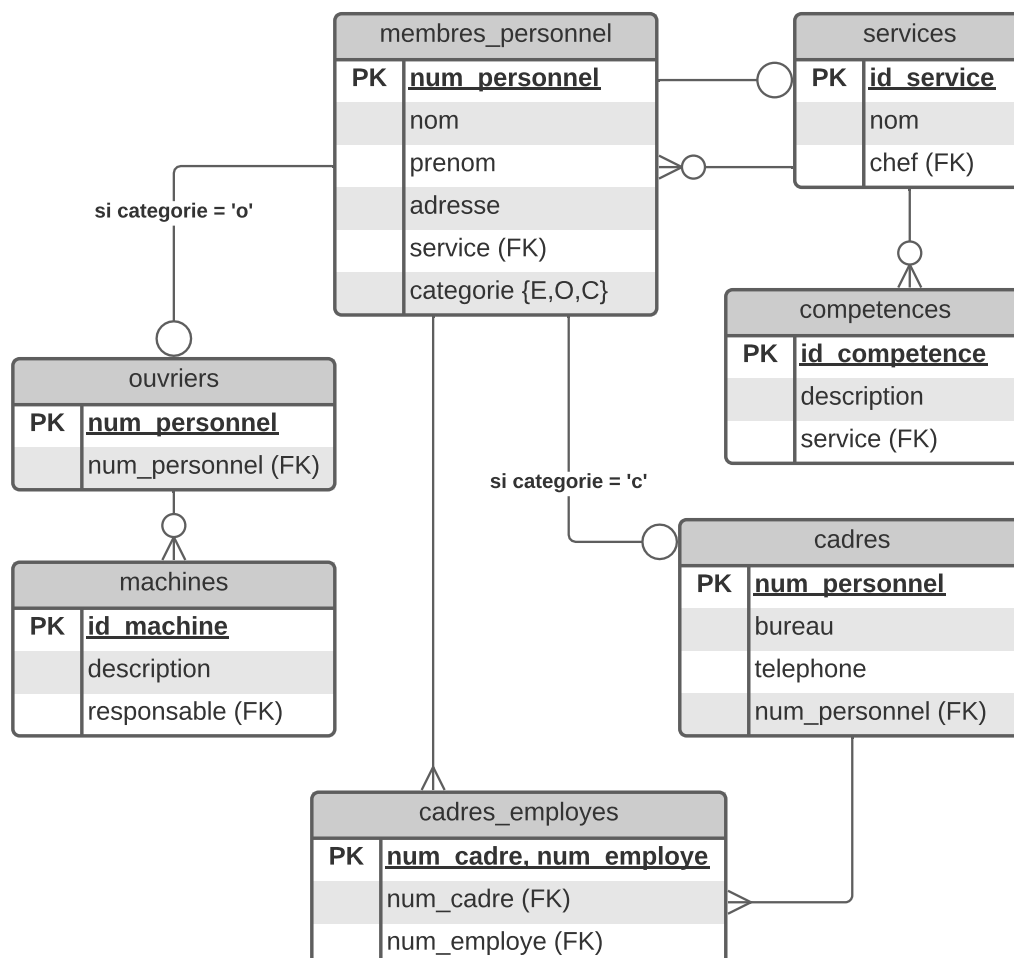
L'entreprise **TicTaque** souhaite informatiser la gestion de son personnel et vous demande de concevoir une base de données permettant de soutenir cette nouvelle application.

L'entreprise est organisée en services auxquels est affecté le personnel. Chaque service a un nom (ex : service comptabilité), un chef et une liste de compétences qui lui sont propres. Les compétences ne sont pas partagées entre services.

Le personnel est divisé en plusieurs catégories : les employés, les ouvriers et les cadres. Tous possèdent un numéro d'employé, un nom, un prénom, une adresse et sont rattachés à un service.

Chaque catégorie peut avoir des renseignements particuliers :

- Pour un ouvrier, la liste des machines dont il est responsable
- Pour un employé, le ou les cadres pour lesquels il travaille
- Pour un cadre, son bureau, son numéro de téléphone direct et le ou les employés qui lui sont rattachés.



## Exercice 7 : Commande d'achat – Novembre98

Tracez le DSD correspondant à l'énoncé suivant.

Je dois comprendre, à la seule lecture du DSD, les entités, leurs propriétés et les associations : donnez un nom aux associations et placez-le sur celles-ci.

Vous devez gérer les commandes d'achat et les réceptions d'articles de l'entreprise Novembre98 qui commercialise une cinquantaine. Les articles sont disponibles chez une vingtaine de fournisseurs différents. Maintenir un catalogue complet serait trop lourd : c'est lors de chaque commande que le service des achats interroge les fournisseurs pour connaître leurs prix.

Lorsque le stock d'un article tombe en-dessous d'un certain seuil, le service des achats crée une commande d'achat de cet article. Le seuil dont on parle est le stock actuel augmenté des commandes en cours ; il faut éviter de commander deux fois le même article en quantité importante parce que la première commande n'est pas encore livrée.

Lorsque la commande pour un article est créée, le service des achats appelle les différents fournisseurs pour connaître leurs prix. Il garde uniquement l'information concernant le meilleur prix et son fournisseur, et non pas les informations de tous les appels.

Les articles commandés sont regroupés par fournisseur pour former une commande d'achat envoyée à celui-ci.

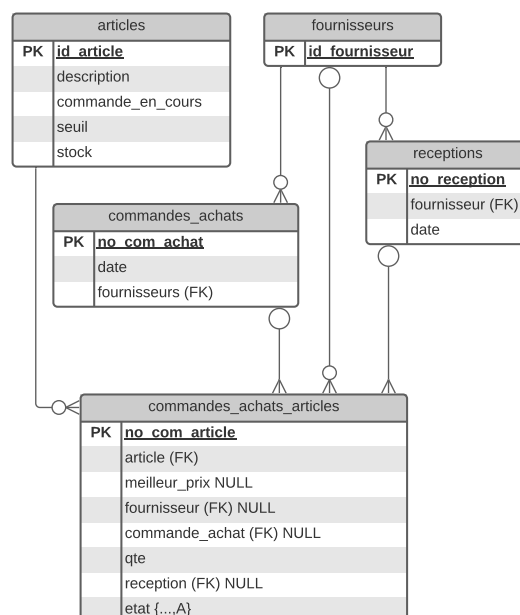
L'annulation d'une commande auprès du fournisseur provoque l'annulation de toutes les lignes de commandes sauf si une ou plusieurs de ces lignes ont déjà été livrées.

Les livraisons faites par le fournisseur ne correspondent pas nécessairement aux commandes d'achat: la réception des articles peut être faite en plusieurs fois et une réception peut également regrouper plusieurs commandes.

Une ligne de commande est réceptionnée en une seule fois.

*Si une ligne de commande ne peut être réceptionnée en une seule fois, on ajoutera une deuxième ligne de commande à l'originale. La première contiendra la ligne de commande avec la quantité réceptionnée et la seconde avec la quantité encore à livrer par le fournisseur.*

Lors de la réception des articles, le stock d'articles sera mis à jour.



## Exercice 11 : Magasin de vêtements

Une ancienne étudiante de l'IPL ouvre un magasin de vêtements d'un nouveau concept et demande votre aide pour l'aider à créer une base de données.

Son idée est de permettre aux clients de visualiser sur de grands écrans toutes les tenues qui peuvent être proposées en accord avec un vêtement choisi. Une tenue peut être illustrée par plusieurs photos ou vidéos mais une photo est obligatoire. Une tenue a une date de validité.

Par exemple : une cliente a essayé un gilet vendu pendant la saison été de 2019. Elle souhaite l'acheter en couleur rouge et souhaite voir les tenues proposées avec cette couleur.

Plusieurs tenues sont proposées :

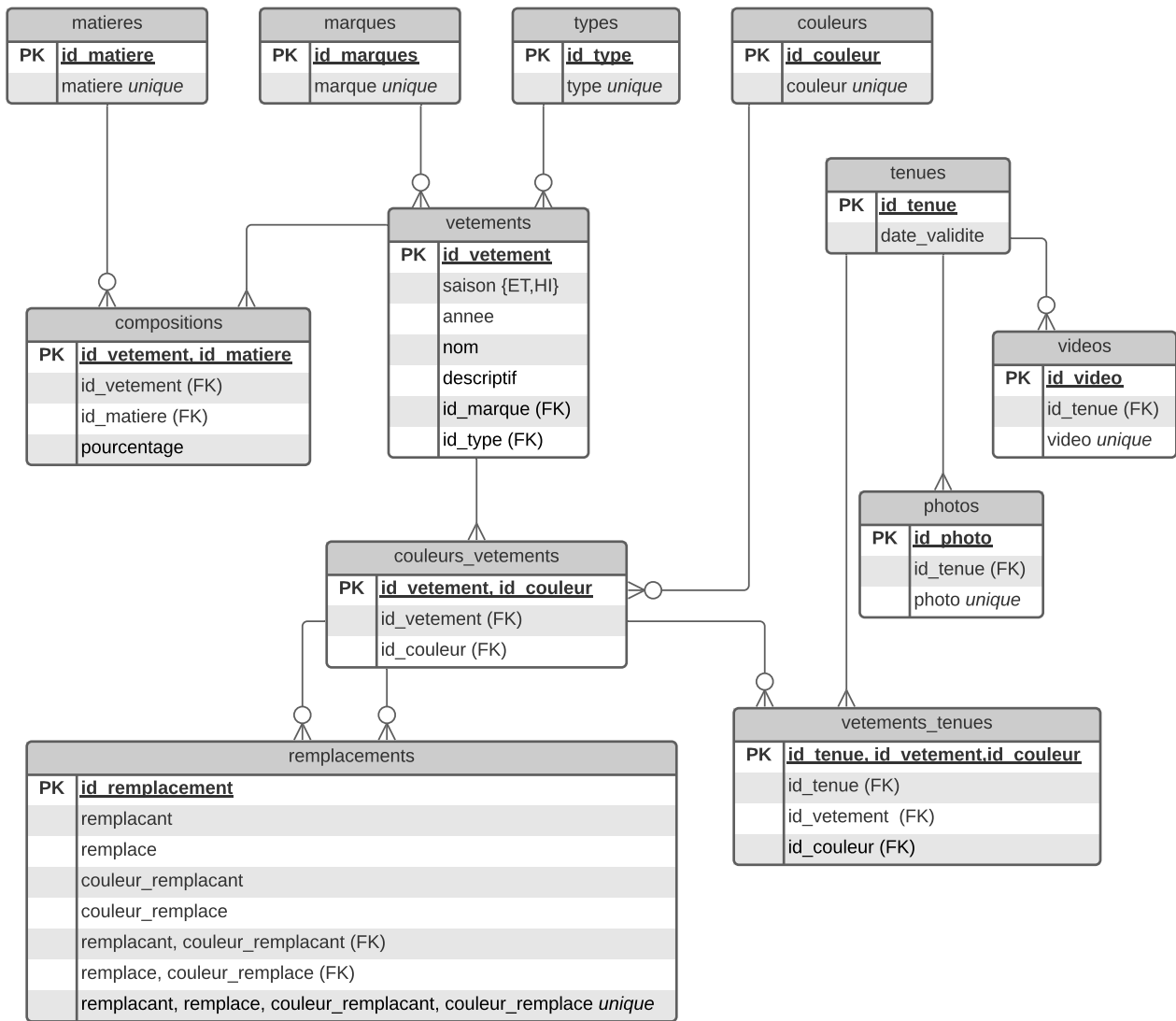
- Tenue 1 : le gilet rouge est associé avec un chemisier blanc, un foulard rouge et un pantalon de couleur taupe. Il y a trois photos : de face, de trois quarts et de profil.
- Tenue 2 : le gilet rouge est associé avec une robe d'été dans laquelle on retrouve une touche de même couleur. Une seule photo est disponible.
- Tenue 3 : le gilet rouge est associé avec une tunique longue blanche, déposée sur un leggings blanc. Une dizaine de photos et vidéos sont disponibles.
- ...

Les clientes peuvent cliquer sur un vêtement pour en connaître toutes les qualités :

- La saison qui peut prendre uniquement deux valeurs : été – hiver.
- L'année.
- Les couleurs dans lesquelles le vêtement est disponible. De nouvelles couleurs apparaissent chaque année, en fonction des idées des créateurs.
- Les matières. Par exemple, un pull peut être en 20% laine et 80% coton. De nouvelles matières peuvent apparaître régulièrement. Les compositions sont infinies et on ne garde la composition qu'au niveau du vêtement.
- Un descriptif.
- Un nom.
- Une marque. De nouvelles marques sont ajoutées régulièrement.
- Un type (pantalon, jupe, robe, tunique, t-shirt, top, sous-pull, pull, gilet...). De nouveaux types sont ajoutés régulièrement.

Un vêtement peut être remplacé par d'autres. Par exemple, notre cliente voudrait acheter la tenue 3 mais le leggings n'existe plus dans sa taille (les tailles ne sont pas gérées au niveau de l'affichage et nous n'en tenons pas compte ici). Un (ou plusieurs) autre(s) leggings(s) doit(doivent) donc lui être proposé(s).

Tracez le DSD correspondant à l'énoncé. Je dois comprendre, *à la seule lecture du DSD*, les entités, leurs propriétés et les associations : donnez un nom aux associations qui ne seraient pas nommées dans l'énoncé et placez-le sur celles-ci.





### Exercice 16 : La recherche des identifiants : agenda dentistes

Le schéma de la relation « Agenda » est le suivant :

**Agenda (Date\_Séance, Heure, No\_Patient, Code\_Dentiste, Montant\_dû, Montant\_payé)**

Date séance	Heure	No Patient	Code Dentiste	Montant dû	Montant payé
8/11/2012	8:30	80236	Dubois	60,00	60,00
8/11/2012	10:00	40152	Dubois	75,00	0,00
8/11/2012	8:30	60143	Moreau	135,00	75,00
8/11/2012	9:30	60143	Moreau	107,00	0,00
15/11/2012	8:30	80236	Dubois	75,00	75,00
15/11/2012	10:00	40178	Dubois	135,00	0,00
16/11/2012	8:30	80236	Moreau	75,00	75,00
...					

1. Recherche des superclés

Date\_Séance, Heure, No\_Patient, Code\_Dentiste, Montant\_dû, Montant\_payé

Date\_Séance, Heure, No\_Patient, Code\_Dentiste, ~~Montant\_dû, Montant\_payé~~

Date\_Séance, Heure, No\_Patient, ~~Code\_Dentiste~~

Date\_Séance, Heure, ~~No\_Patient~~, Code\_Dentiste

~~Date\_Séance, Heure, No\_Patient, Code\_Dentiste~~ IMPOSSIBLE

2. Recherche des clés candidates

Date\_Séance, Heure, No\_Patient

Date\_Séance, Heure, Code\_Dentiste

3. Clé primaire

Une des deux clés candidates en fonction du contexte -> ici Date\_Séance, Heure, Code\_Dentiste

4. Liste des attributs premiers/non-premiers

## Exercice 17 : Les dépendances fonctionnelles : commandes

Le schéma de la relation « Commande » est le suivant :

**Commande (Noclient, Nom, Adresse, Nocom, Date, Noarticle, Qte, Prixu)**

No client	Nom	Adresse	Nocom	Date	No article	Qte	Prixu
159	Dubois	Bruxelles	17	7/11/2013	7	12	7,59€
159	Dubois	Bruxelles	17	7/11/2013	14	6	6,68€
159	Dubois	Bruxelles	17	7/11/2013	21	3	4,56€
215	Durand	Liège	18	8/11/2013	7	5	7,59€
218	Dupont	Namur	19	8/11/2013	21	4	4,56€
159	Dubois	Bruxelles	20	9/11/2013	21	23	4,56€
159	Dubois	Bruxelles	20	9/11/2013	14	5	6,68€
175	Du Pont	Bruxelles	21	10/11/2013	14	7	6,68€
183	Dubois	Gand	22	10/11/2013	14	11	6,68€
183	Dubois	Gand	22	10/11/2013	8	7	7,87€

### 1) Recherche des DF

- DF avec déterminant « simple » (= une colonne déterminante)  
Noclient → Nom un client a un nom  
Noclient → Adresse un client a une adresse  
Nocom → Date une commande a une date  
Nocom → Noclient une commande est émise pour un client  
Noarticle → Prixu un produit a un prix unitaire
- Vérification que toutes les colonnes participent au moins à 1DF  
Qte ne participe à aucune DF
- DF avec déterminant « complexe » (= plusieurs colonnes déterminantes)  
Nocom, Noarticle → Qte dans une commande, il y a une quantité par article

### 2) Clé primaire de la relation

Avant de voir si la relation est normalisée, nous allons lui donner un identifiant. Pour cela, nous allons baser sur les DF.

Les déterminants sont donc :

- Nocom
- Noclient
- Nocom, Noarticle

On doit les concaténer ce qui donne la superclé. En simplifiant cette superclé, on obtient l'identifiant **Nocom, Noarticle** qui détermine donc l'entièreté des attributs de la relation.

### 3) Normalisation

Grâce à notre identifiant, on peut commencer à normaliser la relation commande.

/!\ L'identifiant est toujours souligné. /!\

Commande (Noclient, Nom, Adresse, Nocom, Date, Noarticle, Qte, Prixu)

- Relation en 1FN ?  
OUI car il n'y a que des valeurs atomiques
- Relation en 2FN ?  
NON
  - Parties strictes de la clé sont Nocom et Noarticle

ii) Il y a des DF à ces deux parties strictes donc il faut décomposer.

Commande (Noclient, Nom, Adresse, Nocom, Date, Noarticle, Qte, Prixu)



Commande1 (Nocom, Noclient, Nom, Adresse, Date)

Commande2 (Noarticle, Prixu)

Commande3 (Nocom, Noarticle, Qte)

iii) Les relations obtenues sont maintenant en 2FN.

c) Relations en 3FN ?

NON

i) Il y a une DF transitive dans Commande1 donc il faut décomposer

Commande1 (Nocom, Noclient, Nom, Adresse, Date)

Commande2 (Noarticle, Prixu)

Commande3 (Nocom, Noarticle, Qte)



Commande1a (Noclient, Nom, Adresse)

Commande1b (Nocom, Noclient, Date)

Commande2 (Noarticle, Prixu)

Commande3 (Nocom, Noarticle, Qte)

ii) Les relations obtenues sont maintenant en 3FN.

## Exercice 19 : Un courtier d'assurance

Un courtier d'assurances veut informatiser la gestion des contrats d'assurances-auto.

Ce courtier offre quatre types de contrat d'assurance-auto : responsabilité civile, omnium partielle, omnium totale ou bris de vitre.

Chaque contrat couvre des risques particuliers. Par exemple, le contrat d'omnium partielle couvre les risques d'incendie, de vol et de bris de glace.

Un véhicule peut être couvert par un ou plusieurs contrats.

Le preneur d'assurances qui souhaite assurer un véhicule peut donc signer plusieurs contrats pour un même véhicule.

Un contrat est signé avec une compagnie d'assurances. Le courtier travaille avec presque toutes les compagnies d'assurance qui exercent sur le territoire belge.

Tout ceci est mémorisé dans un grand fichier fourre-tout, dont vous trouverez quelques lignes à la page suivante.

Compagnie	Client	Nom	Prénom	Type	Prise_effet	Montant	Période	No-plaques	No-contrat
AXA	C204	Lemaitre	Pat	RC	01.02.2012	1.560,00	An	EKX-208 ; ERZ-908 ; DCR-405	LCI406
AXA	C204	Lemaitre	Pat	Omnium	01.02.2012	1.200,90	Mois	EKX-208 ; DCR-405	LCI4698
AXA	C1051	Leprof	Luc	RC	05.05.2008	200,30	Semestre	AZE-369	LCI56933
Mercator	C1051	Leprof	Luc	Omnium	05.08.2008	260,76	Semestre	AZE-369	ME1546
Winterthur	C2056	Leresto	Eric	Bris-de-vitre	06.07.2009	150,36	Semestre	PRT-623	WI1513468
Winterthur	C355	Lesaca	Plume	RC	05.11.2012	278,37	An	1-KFH-007	WI196494
Winterthur	C107	Apporte	Sam	OmniumP	07.12.2012	458,22	An	1-LKM-321	WI435684
AXA	C204	Lemaitre	Pat	RC	07.08.2012	460,00	An	ABC-405	LCI417
Winterthur	C9870	Apporte	Kim	OmniumP	01.12.2012	657,00	Semestre	PEZ-506	WI435712

Répondez aux questions suivantes :

- Quelles sont les DF ?  
client → nom, prénom  
no-contrat → compagnie, client, type, prise\_effet, montant, période, no-plaques  
no-plaques → client
- Quelles sont les clés-candidates possibles ? Choisissez-en une pour cette relation.  
Superclé = client, no-contrat, no-plaques  
Clé(s) minimale(s) : no-contrat détermine client et no\_plaques et no-plaques détermine client mais rien ne détermine no-contrat => clé primaire
- Quel est le niveau de normalisation de cette relation ?  
0FN
- Amenez cette relation en 3FN.
  - Transformation en 1FN : on met un no-plaques par tuple  
/!\ La PK n'est plus unique, on a donc perdu une dépendance fonctionnelle.  
no-contrat → compagnie, client, type, prise\_effet, montant, période, ~~no-plaques~~  
Suite à la perte de DF, changement dans la clé primaire :  
- Concaténation -> la clé primaire devient no-plaque, no-contrat  
- Alternative dans les clés minimales trouvées  
1FN ? OUI

2FN ? NON

2. Transformation en 2FN

contrats(no-contrat, compagnie, periode, prise\_effet, type, montant)

vehicules(no-plaque, client, nom, prenom)

contrat\_vehicule(no-contrat, no-plaque)

2FN ? OUI

3FN ?

### Exercice 23 : Cours- recherche clé et graphe des DF

On considère la relation R(cours, professeur, heure, salle, étudiant, note)

Un cours est dispensé par un professeur à une heure dans une salle et est suivi par un étudiant qui obtient une note.

On trouve les dépendances fonctionnelles suivantes :

- Cours  $\rightarrow$  professeur
- Heure, salle  $\rightarrow$  cours
- Heure, professeur  $\rightarrow$  salle
- Cours, étudiant  $\rightarrow$  note
- Heure, étudiant  $\rightarrow$  salle

Veillez répondre aux questions suivantes.

1. Donnez la clé primaire de cette relation.  
Détermination avec élimination des doublons :  
cours, heure, salle, professeur, étudiant  
~~cours, heure, salle, professeur, étudiant~~  
PK = heure, étudiant
2. Montrez qu'elle respecte toutes les dépendances fonctionnelles sur un graphe.  
PAS À L'EXAMEN
3. Y avait-il d'autres clés candidates ? Justifiez.  
Non car ...

## Exercice 27 : Offres & devis

Une société veut informatiser la gestion de ses offres.

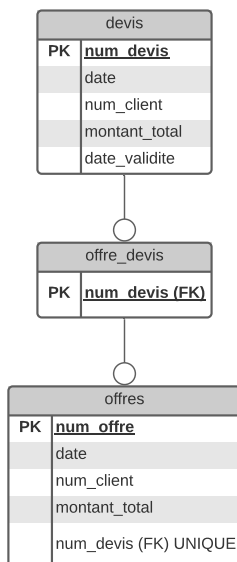
Elle avait donné l'information suivante dans son appel d'offre :

Un devis est un document provisoire rédigé par un employé. Une offre, dont on dit qu'elle est « **ferme et définitive** » est un devis qui a été accepté par la direction et est donc envoyé officiellement aux clients.

La société a reçu plusieurs cahiers des charges. Un seul permettait de respecter les contraintes-métiers mais de l'avis des étudiants de l'IPL, n'était pas suffisamment bon.

La société fait donc appel à ces mêmes étudiants pour pouvoir modifier le DSD.

1. Citez les contraintes-métiers à prendre en compte.
  - une offre a une propriété « fermée »
  - un devis n'entraîne pas forcément une offre
  - une offre correspond absolument à un devis
  - un devis a été rédigé par un employé
2. Modifier le DSD pour que le SGBD gère au mieux ces contraintes. Vous ne pouvez absolument pas modifier le nombre de tables ni leur utilité fonctionnelle.



## Exercice Janvier 2020

Tracez le DSD correspondant à l'énoncé suivant. La dénomination de chaque association doit pouvoir être clairement identifiée.

La société Au Jardinier Vert vous demande de concevoir une base de données permettant de stocker toutes les informations nécessaires à son activité.

Le patron, un passionné, a des fiches décrivant toutes les plantes d'Europe. Pour chaque plante, il a listé son nom commun, ses genres et espèces ainsi qu'une description permettant aux particuliers de choisir des plantes en accord avec la typologie de leur jardin. Pour certaines plantes, il a déjà décrit des conseils de plantation.

Pour les plantes qu'il rentre en magasin, on dispose du prix de vente, de la taille du jeune plant, de la quantité stockée et du fournisseur qui livre cette plante actuellement. En effet, un seul fournisseur livre une plante à un moment donné. Cependant, ce fournisseur peut varier dans le temps et il faut garder la trace de tous les fournisseurs ayant livré une plante ainsi que du prix d'achat de la plante auprès du fournisseur. Il y a des moments où aucun fournisseur ne peut livrer une plante.

Par exemple, le hêtre commun est une plante qui est toujours en stock. Elle a été livrée par plusieurs pépiniéristes :

Fournisseur	Date début	Date fin	Prix d'achat
Ferme des Pilifs	01.02.2016	30.06.2017	2,50€
Comme une fleur	01.07.2017	31.01.2018	2,35€
Hamblenne	01.03.2018	31.12.2019	2,40€
Ferme des Pilifs	01.01.2019		2,55€

Le fournisseur actuel est la Ferme des Pilifs.

Afin de pouvoir servir ses clients au mieux, le patron a défini la règle suivante : chaque plante en stock peut être remplacée par une autre plante mais peut en remplacer plusieurs.

