Saé S1-04 CONCEPTION – INTERROGATION D'UNE BASE DE DONNEES SAE-MIISION-3

1 - L'analyse des données :

- 1) KDou s'intéresse aux desserts et aux boissons dans sa réflexion en lien avec la santé.
- a) Combien y a-t-il des produits correspondants vendus 50€ ou plus ? On veut les connaître avec leur pays d'origine.

le nombre de produits des catégories 1 et 3 (Desserts et Boissons) dont le prix est de 50 euros ou plus, regroupés par pays d'origine

Il ya 2 Royaume Uni, Donc au RU il ya 4 Produits Boissons ou Desserts a 50 euros ou plus.



SELECT COUNT(Produit.CodeCateg) AS NbProd, Fournisseur.Pays

FROM Produit INNER JOIN Fournisseur ON Produit.NoFour = Fournisseur.NoFour

WHERE Produit.PrixUnit >= 50 AND (Produit.CodeCateg = 1 OR Produit.CodeCateg = 3)

GROUP BY Fournisseur. Pays;

Il est possible de corriger l'orthographe des pays qui sont les meme en faisant

UPDATE Fournisseur

SET Pays = "Royaume-Uni"

WHERE Pays IN ("Royaume Uni", "Royaume-Uni");

Mais on ne l'a pas fait.

b) Quels sont les pays de provenance de ces catégories?

Il y'a des doublons de pays avec les même orthographes.



SELECT DISTINCT Fournisseur. Pays

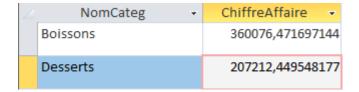
FROM Produit

INNER JOIN Fournisseur ON Produit.NoFour = Fournisseur.NoFour

WHERE Produit.CodeCateg = 1 OR Produit.CodeCateg = 3;

c) On veut le chiffre d'affaires généré par catégorie desserts ou catégorie.

Voici la premiere requete qui se concentre sur le chiffre d'affaires basé sur les commandes, car elle prend en compte les ventes :



SELECT Categorie. NomCateg,

SUM(DetailCommande.PrixUnitCli * DetailCommande.QteCom * (1 - DetailCommande.Remise / 100)) AS ChiffreAffaire

FROM (DetailCommande

INNER JOIN Produit ON DetailCommande.RefProd = Produit.RefProd)

INNER JOIN Categorie ON Produit.CodeCateg = Categorie.CodeCateg

WHERE Categorie.CodeCateg = 1 OR Categorie.CodeCateg = 3

GROUP BY Categorie. NomCateg;

Et voici la deuxième requete du chiffre d'affaires potentiel basé sur le stock et le prix unitaire :



SELECT SUM(PrixUnit * Produit.UniteStock) AS ChiffreAf, Produit.CodeCateg

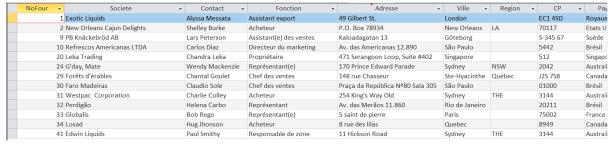
FROM Produit

WHERE Produit.CodeCateg = 1 OR Produit.CodeCateg = 3

GROUP BY Produit.CodeCateg;

- 2) KDou se demande ce qu'il faut changer dans son offre, voici quelques indicateurs?
- a) Donner des informations sur les fournisseurs qui proposent des produits qui ne sont pas commandés actuellement ?

Voici les informations des fournisseurs qui proposent des produits, qui ne sont pas commandés.



SELECT DISTINCT Fournisseur.*

FROM (Fournisseur INNER JOIN Produit ON Fournisseur.NoFour = Produit.NoFour) LEFT JOIN DetailCommande ON Produit.RefProd = DetailCommande.RefProd

WHERE DetailCommande.RefProd IS NULL;

b) Combien y –a-t-il de produits non commandés par catégorie?

On a pris le nombre de produit commandé en tout, et parmi eux on a fait le compte de ceux qui ne sont pas commandé selon leur catégorie.

NbParCateg • CodeCateg • 27 1 2 2 2 3 3 3 2 5 2 8

SELECT COUNT(Produit.CodeCateg) AS NbParCateg, Produit.CodeCateg

FROM Produit LEFT JOIN DetailCommande ON Produit.RefProd = DetailCommande.RefProd

WHERE DetailCommande.RefProd IS NULL

GROUP BY Produit.CodeCateg;

c) Quels sont les pays qui ont une offre importante de produit?

Voici le nombre d'offres de produits par pays. On a de nombreux doublons avec les pays qui sont les mêmes, mais avec une orthographe différente.

∠ Pays 🕶	NbProduits -
Royaume-Uni	18
Allemagne	11
Brésil	8
Canada	8
Etats-Unis	7
France	6
Australia	6
Japon	6
Italie	5
Royaume Uni	4
Etats Unis	4
Singapour	4
Australie	4
Suède	4
Norvège	3
Espagne	3
Suéde	3
Finlande	3
USA	
Pays-Bas	2 2
Danemark	2

SELECT Fournisseur. Pays, COUNT(Produit. RefProd) AS NbProduits

FROM Fournisseur

INNER JOIN Produit ON Fournisseur.NoFour = Produit.NoFour

GROUP BY Fournisseur. Pays

ORDER BY COUNT(Produit.RefProd) DESC;

3) KDou étudie les provenances de ses produits :

a) Pour l'Allemagne pays proche, quels sont les desserts ou boissons vendus à moins de 100 euros ?

Ce sont ces 4 Produits vendus a moins de 100 euros en Allemagne :

		Pays	w	RefProd -	NomProd	•	PrixUnit -	CodeCateg 🔻	
		Allemagne		25	NuNuCa Nuß-Nougat-Creme		11,67 €	3)
		Allemagne		26	Gumbär Gummibärchen		26,03 €	3	,
		Allemagne		27	Schoggi Schokolade		36,58€	3)
		Allemagne		75	Rhönbräu Klosterbier		86,46 €	1	
+	*								

SELECT Fournisseur.Pays, Produit.RefProd, Produit.NomProd, Produit.PrixUnit, Produit.CodeCateg

FROM Fournisseur INNER JOIN Produit ON Fournisseur.NoFour = Produit.NoFour

WHERE Fournisseur.Pays = "Allemagne" AND (Produit.CodeCateg = 1 OR Produit.CodeCateg = 3) AND Produit.PrixUnit < 100;

b) Quels fournisseurs vendent uniquement des boissons ou des desserts?

Pour cette requete on a procédé de cette manière :

- -Un fournisseur peut vendre uniquement des produits de la catégorie 1 (desserts) sans vendre de produits de la catégorie 3 (boissons) ou d'autres catégories.
- -Un fournisseur peut vendre uniquement des produits de la catégorie 3 (boissons) sans vendre de produits de la catégorie 1 (desserts) ou d'autres catégories.

Donc il ne doit rien vendre a part la categorie 1 ou categorie 2 et pas les deux.

4	NoFour	w	Societe ▼
		8	Specialty Biscuits, Ltd.
		10	Refrescos Americanas LTDA
11		11	Heli Süßwaren GmbH & Co. KG
18		18	Aux joyeux ecclésiastiques
22		22	Zaanse Snoepfabriek
		23	Karkki Oy
	;	30	Faro Madeiras
	;	31	Westpac Corporation
	;	32	Perdigão
		33	Globalis
		34	Losad

Pendant le tri on n'a pas réussi à filtrer ceux qui ont la catégorie 1 et 3 en même temps. Ce sont les fournisseurs 10 et 23.

SELECT Fournisseur.NoFour, Fournisseur.Societe

FROM Fournisseur

INNER JOIN Produit ON Fournisseur. No Four = Produit. No Four

GROUP BY Fournisseur.NoFour, Fournisseur.Societe

HAVING SUM(IIF(Produit.CodeCateg IN (1, 3), 1, 0)) > 0

AND SUM(IIF(Produit.CodeCateg NOT IN (1, 3), 1, 0)) = 0;

On a pensé aussi a une autre méthodes on l'on réalise les 2 requetes séparement.

C'est a dire ceux qui sont uniquement de categorie 1 uniquement :

2	NoFour	¥	Societe	*
		18	Aux joyeux ecclésiastiques	
		30	Faro Madeiras	
		32	Perdigão	
		33	Globalis	
		34	Losad	

SELECT Fournisseur. No Four, Fournisseur. Societe

FROM Fournisseur

INNER JOIN Produit ON Fournisseur.NoFour = Produit.NoFour

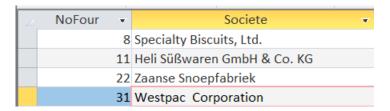
GROUP BY Fournisseur. No Fournisseur. Societe

HAVING

SUM(IIF(Produit.CodeCateg = 1, 1, 0)) > 0

AND SUM(IIF(Produit.CodeCateg <> 1, 1, 0)) = 0;

Ou ceux de categorie 3 uniquement :



SELECT Fournisseur. No Four, Fournisseur. Societe

FROM Fournisseur

INNER JOIN Produit ON Fournisseur.NoFour = Produit.NoFour

GROUP BY Fournisseur. No Four, Fournisseur. Societe

HAVING

SUM(IIF(Produit.CodeCateg = 3, 1, 0)) > 0

AND SUM(IIF(Produit.CodeCateg <> 3, 1, 0)) = 0;

Cela permet d'avoir les fournisseurs qui vendent seulements des boissons ou seulement des desserts.

c) Y en a-t-il qui vendent les deux boissons et desserts?

Voici les Fournisseurs qui vendent les boissons et les desserts.



SELECT Fournisseur. No Four, Fournisseur. Societe

FROM Fournisseur

INNER JOIN Produit ON Fournisseur.NoFour = Produit.NoFour

GROUP BY Fournisseur. No Four, Fournisseur. Societe

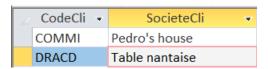
HAVING COUNT(IIF(Produit.CodeCateg = 1, 1, NULL)) > 0

AND COUNT(IIF(Produit.CodeCateg = 3, 1, NULL)) > 0;

4) KDou étudie ses clients et les remises accordées :

a) Un client nantais Le verre à soi souhaite savoir quels autres clients nantais commandent aussi des produits en Allemagne pour envisager de futures livraisons groupées.

Cette requête utilise de nombreuses jointures la table commande à la table Fournisseur.



SELECT Client.CodeCli, Client.SocieteCli

FROM (Client

INNER JOIN Commande ON Client.CodeCli = Commande.CodeCli)

INNER JOIN (DetailCommande

INNER JOIN (Produit

INNER JOIN Fournisseur ON Produit.NoFour = Fournisseur.NoFour)

ON DetailCommande.RefProd = Produit.RefProd)

ON Commande.NoCom = DetailCommande.NoCom

WHERE Client. VilleCli = "Nantes"

AND Fournisseur. Pays = "Allemagne"

AND Client.SocieteCli <> "Verre à soi"

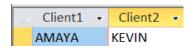
GROUP BY Client.CodeCli, Client.SocieteCli

•

b) Quels sont couples de clients de Besançon pourrait on créer?

Nous avons remarqué qu'il y'a 2 orthographes différents de la ville Besançon.

Malgré cette erreur nous avons trouvé le couple :



SELECT Client1.CodeCli AS Client1, Client2.CodeCli AS Client2

FROM Client AS Client1, Client AS Client2

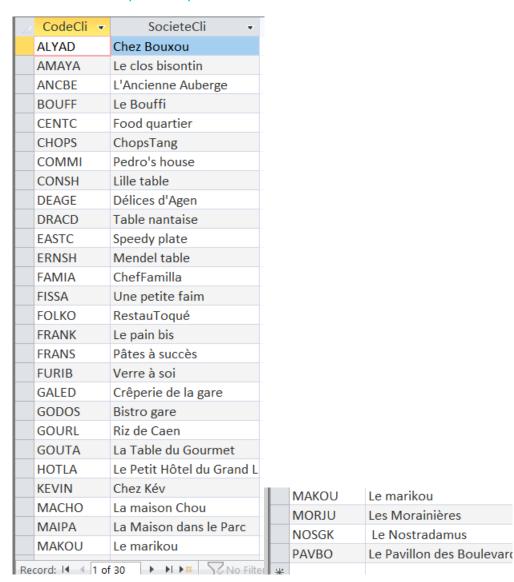
WHERE (Client1. VilleCli = 'Besançon' OR Client1. VilleCli = 'Besancon')

AND (Client2.VilleCli = 'Besançon' OR Client2.VilleCli = 'Besancon')

AND Client1.CodeCli < Client2.CodeCli;

c) Certains clients n'ont jamais eu la remise la plus importante de 25%. Qui sont-ils?

Voici les clients qui n'ont pas eu de remise a 25%



SELECT Client.CodeCli, Client.SocieteCli

FROM Client WHERE Client.CodeCli NOT IN (SELECT Commande.CodeCli FROM Commande INNER JOIN DetailCommande ON Commande.NoCom = DetailCommande.NoCom WHERE DetailCommande.Remise = 25);

d) Quels produits n'ont eu que des remises de 10% ou 20% ou pas de remise?

Voici les produits qui n'ont eu que des remises de 10% ou 20% ou aucune remise c'est à dire 0%.

RefProd ▼	NomProd
84	Limonadystel
86	Sugar bleu blanc
87	Bratislava Vodka
88	Beer native
91	linonade ctinonella
93	Cidre brut de saint Lô
96	Rhum martini Cubano
97	Calvanelle
99	Pimm' s jojo
102	Cachaça
105	Victoria Bitter
106	Limon Lux
107	Blue Wine
108	Guaraná em pô
111	Jus d'érable d'or
112	Arnott's Tim Tam Biscuits
120	CocaChina
121	Louisiana Coca
125	LimoCoca
126	Cachaça zuaviocado
127	Rouge sang
128	Jus Agava
129	Jus Kiwina
130	knäckebrod Kala
131	Dea Surströmming
132	Kanelbullar Stock
133	Guaraná Marvesa
134	Chocolado Aquim
135	Cachaça Mikala
136	Siropa Fraisea
137	CocaNet
138	Cachaça zuavio
139	LimoCoca Bio
140	Laksa Paste
141	Sambal Chilli Sauce
1.00	Tim Tams

SELECT p.RefProd, p.NomProd

FROM Produit AS p

WHERE p.RefProd NOT IN (

SELECT dc.RefProd

FROM DetailCommande AS dc

WHERE dc.Remise NOT IN (0, 10, 20)

);

2 - Proposition d'indicateurs (requêtes)

REQUETE JF

Requete en langage naturelle

Lister les produits qui n'ont jamais été commandés afin de repérer ceux qui n'attirent pas les clients et envisager leur suppression ou des promotions pour stimuler leur vente.

Interet de la requete

Cette requête aide KDou à optimiser son offre en identifiant les produits inactifs. Réduire les stocks de ces produits, les remplacer par d'autres ou proposer des promotions ciblées pourrait diminuer les coûts de stockage et améliorer la rotation du catalogue. Cela est également pertinent dans une optique de gestion environnementale.

NomProd Limonadystel Sugar bleu blanc Bratislava Vodka Beer native linonade ctinonella Cidre brut de saint Lô Rhum martini Cubano Calvanelle Pimm' s jojo Cachaça Victoria Bitter Limon Lux Blue Wine Guaraná em pô Jus d'érable d'or Arnott's Tim Tam Biscuits CocaChina Louisiana Coca LimoCoca Cachaça zuaviocado Rouge sang Jus Agava Jus Kiwina knäckebrod Kala Dea Surströmming Kanelbullar Stock Guaraná Marvesa

Extrait de la requete

Requete

SELECT Produit.NomProd

FROM Produit LEFT JOIN DetailCommande ON Produit.RefProd = DetailCommande.RefPro

WHERE DetailCommande.NoCom IS NULL;

Explication:

Jointure gauche (LEFT JOIN): On utilise un LEFT JOIN entre Produit et DetailCommande pour obtenir tous les produits, même ceux sans correspondance dans DetailCommande.

Filtre: La condition WHERE DetailCommande.NoCom IS NULL identifie les produits pour lesquels aucune ligne de commande n'existe, c'est-à-dire les produits jamais commandés.

Résultat : La requête renvoie la liste des noms de produits (NomProd) sans commandes, facilitant une analyse ciblée des produits sans demande.

Requête yanis

1. Requête en langage naturel : Identifier les produits importés qui ont une empreinte écologique élevée (par exemple, en raison de la distance d'importation) afin de proposer des alternatives plus locales.

Objectif: L'objectif est d'aider KDou à respecter davantage l'environnement en réduisant les produits importés de pays lointains. En fournissant cette information, KDou pourrait envisager de privilégier des produits équivalents mais provenant de pays voisins pour limiter son empreinte carbone.

Intérêt de la requête : Cette requête permet de prendre des décisions plus écologiques dans le choix des produits proposés, alignant ainsi les valeurs de l'entreprise avec des pratiques respectueuses de l'environnement. Cela pourrait aussi améliorer l'image de l'entreprise auprès de ses clients, soucieux de l'impact écologique.

Résultat attendu: Un tableau indiquant le type de produit, le pays d'origine

SELECT

Produit.NomProd AS produit_nom,

Categorie. Nom Categ AS categorie,

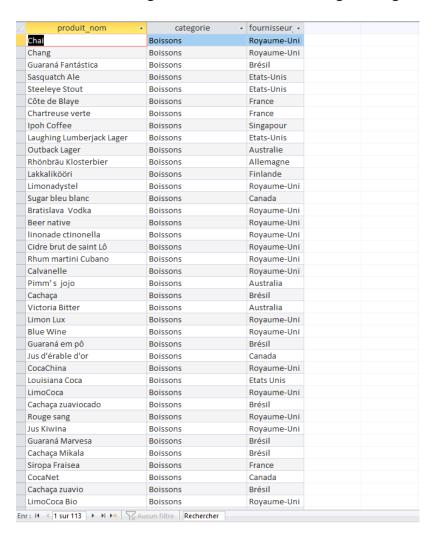
Fournisseur_pays AS fournisseur_pays

FROM

(Produit

INNER JOIN Fournisseur ON Produit.NoFour = Fournisseur.NoFour)

INNER JOIN Categorie ON Produit.CodeCateg = Categorie.CodeCateg;



Requête Apollinaire:

Objectif: Identifier les produits les plus vendus, classés par catégorie, afin de cibler les initiatives écologiques et optimiser l'offre de KDou en se concentrant sur les produits les plus populaires qui pourraient bénéficier d'un approvisionnement durable.

Intérêt de la requête : Cette requête permet de déterminer quels produits se vendent le mieux, ce qui aide à orienter les efforts vers des alternatives écologiques pour ces articles. Cela peut également soutenir des initiatives de marketing axées sur des

produits durables et renforcer la position de KDou en tant qu'entreprise responsable sur le plan environnemental.

Requête SQL:

 ${\tt SELECT\ CA.NomCateg,\ P.NomProd,\ Sum(D.QteCom)\ AS\ Quantite Vente}$

FROM DetailCommande AS D, Commande AS C, Produit AS P, Categorie AS CA

WHERE (((D.NoCom)=[C].[NoCom]) AND ((D.RefProd)=[P].[RefProd]) AND ((P.CodeCateg)=[CA].[CodeCateg]))

GROUP BY CA.NomCateg, P.NomProd

ORDER BY QuantiteVente DESC;

Structure de la requête :

- FROM : Cette requête utilise la syntaxe de jointure implicite, où les tables sont listées et les conditions de jointure sont spécifiées dans la clause WHERE.
- WHERE: Conditions de jointure pour relier les tables
 - D.NoCom = C.NoCom: Relie les détails de la commande aux commandes pour s'assurer que l'on examine les ventes liées à une commande spécifique.
 - D.RefProd = P.RefProd : Relie les détails de la commande aux produits pour obtenir des informations sur les produits vendus.
 - P.CodeCateg = CA.CodeCateg: Relie les produits à leurs catégories respectives.
- GROUP BY CA.NomCateg, P.NomProd: Regroupe les résultats par catégorie de produit et par produit. Cela permet de calculer la somme des quantités vendues pour chaque produit dans chaque catégorie.
- ORDER BY QuantiteVente DESC: Trie les résultats par quantité vendue en ordre décroissant. Cela aide à mettre en avant les produits les plus vendus, facilitant ainsi l'analyse pour KDou.

_	NomCateg +	NomProd +	QuantiteVe -
	Desserts	Schoggi Schokolade	513
	Boissons	Chai	1289
	Condiments	Northwoods Cranberry Sauce	476
	Condiments	Original Frankfurter grüne Soße	980
	Condiments	Vegie-spread	505
	Desserts	Chocolade	168
	Desserts	Gumbär Gummibärchen	940
	Desserts	Maxilaku	590
	Condiments	Louisiana Hot Spiced Okra	263
	Desserts	Pavlova	1355
	Condiments	Louisiana Fiery Hot Pepper Sauc	855
	Desserts	Scottish Longbreads	985
	Desserts	Sir Rodney's Marmalade	356
	Desserts	Sir Rodney's Scones	1235
	Desserts	Sugar Pie	1297
	Desserts	Teatime Chocolate Biscuits	990
	Desserts	Valkoinen suklaa	325
	Desserts	Zaanse koeken	579
	Desserts	NuNuCa Nuß-Nougat-Creme	533
	Boissons	Sasquatch Ale	743
	Boissons	Chang	1387
	Boissons	Chartreuse verte	993
	Boissons	Côte de Blaye	822
	Boissons	Guaraná Fantástica	1610
	Boissons	Ipoh Coffee	862
	Boissons	Lakkalikööri	1216
	Boissons	Laughing Lumberjack Lager	280
	Condiments	Maple Syrup	680
	Boissons	Rhönbräu Klosterbier	1558
	Pâtes et céréales	Gustaf's Knäckebröd	400
	Boissons	Steeleye Stout	1145
	Condiments	Aniseed Syrup	428
	Condiments	Chef Anton's Cajun Seasoning	643
	Condiments	Chef Anton's Gumbo Mix	376
	Condiments	Genen Shouyu	185
	Condiments	Grandma's Boysenberry Spread	324
	Condiments	Gula Malacca	733

REQUETE MELIH

Requête en langage naturel :

Quels sont les pays non européens qui fournissent des produits à base de viande?

Objectif:

L'objectif est d'aider KDou à diminuer son empreinte carbone en analysant les fournisseurs non européens de viande. En fournissant cette information, KDou pourra envisager de privilégier les produits équivalents provenant de pays voisins, réduisant ainsi la dépendance aux importations lointaines. Cela permettra également de soutenir l'économie locale et de renforcer la durabilité de ses pratiques commerciales.

Intérêt:

Cette requête permet à KDou de mieux comprendre les sources de son approvisionnement en viande. En identifiant les pays non européens, l'entreprise pourra orienter ses choix vers des alternatives plus durables, tout en affirmant son engagement envers des pratiques responsables sur le plan environnemental.

Extrait du résultat :



Requête SQL:

SELECT DISTINCT Fournisseur. Pays

FROM Fournisseur, Produit, Categorie

WHERE Fournisseur.NoFour = Produit.NoFour AND Produit.CodeCateg = Categorie.CodeCateg AND Categorie.CodeCateg = 6 AND Fournisseur.Pays NOT IN (

'Danemark', 'Royaume-Uni', 'Royaume Uni', 'Finlande',

'Norvège', 'France', 'Pays-Bas', 'Italie', 'Suède',

'Suede', 'Espagne', 'Allemagne'

Structure de la requette

SELECT DISTINCT : Utilisé pour s'assurer que chaque pays n'apparaît qu'une seule fois dans les résultats.

FROM: Cette requête utilise la syntaxe de jointure implicite, où les tables Fournisseur, Produit et Categorie sont listées. Les conditions de jointure sont spécifiées dans la clause WHERE.

WHERE: Conditions de jointure pour relier les tables:

Fournisseur.NoFour = Produit.NoFour : Relie la table Fournisseur à la table Produit pour s'assurer que l'on examine les produits associés à un fournisseur spécifique.

Produit.CodeCateg = Categorie.CodeCateg : Relie la table Produit à la table Categorie pour filtrer les produits en fonction de leur catégorie.

AND Categorie.CodeCateg = 6 : Cette condition permet de ne conserver que les produits appartenant à la catégorie spécifiée (dans ce cas, la viande).

AND Fournisseur. Pays NOT IN: Exclut les pays européens mentionnés de la liste des résultats. Cela garantit que seuls les pays non européens apparaissent dans les résultats, ciblant ainsi les fournisseurs dont les produits pourraient contribuer à une empreinte carbone plus élevée.